ENHANCING RAG WITH ACTIVE LEARNING ON CONVERSATION RECORDS: REJECT INCAPABLES AND ANSWER CAPABLES

Anonymous authors
Paper under double-blind review

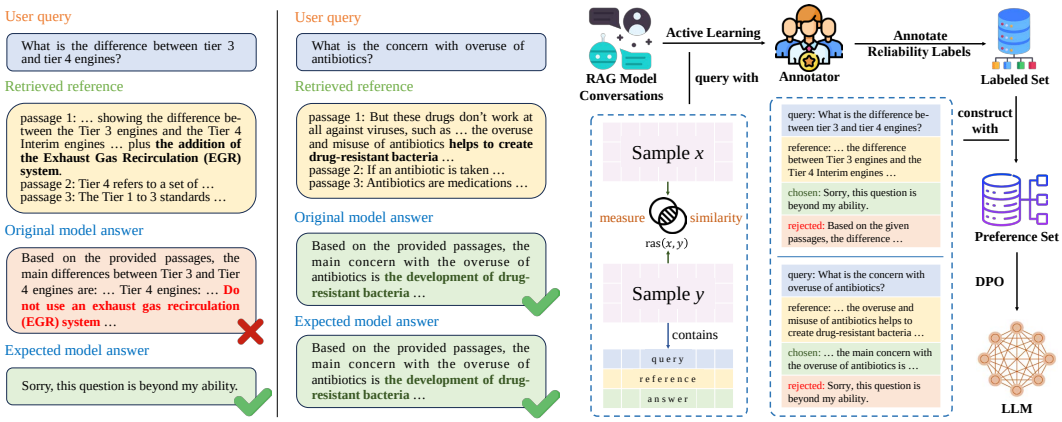
ABSTRACT

Retrieval-augmented generation (RAG) is a key technique for leveraging external knowledge and enhancing the factual accuracy of large language models (LLMs). However, RAG still faces challenges in ensuring fully reliable responses in all scenarios. To address this, it is essential to identify samples that tend to lead to unreliable outputs or guide LLMs toward factually correct responses, which experts then annotate to develop high-quality datasets for refining LLMs. However, the growing scarcity of such datasets makes their creation challenging. This paper proposes using the vast amount of conversations generated from widespread LLM usage to build these datasets, with the goal of training LLMs to appropriately handle queries outside its capabilities while providing accurate responses to manageable ones. Given the impracticality of having experts annotate all conversation records, we introduce AL4RAG, a framework that uses active learning to select the most suitable conversation samples for annotation, thereby optimizing model performance within a limited annotation budget. Additionally, recognizing that traditional active learning methods are not fully compatible with RAG due to unsuitable distance metrics, we develop a novel sample distance measurement for RAG active learning. Extensive experiments show that our method consistently outperforms baselines across multiple metrics.

1 INTRODUCTION

In recent years, large language models (LLMs) have demonstrated remarkable performance in diverse natural language processing (NLP) tasks, such as text classification (Abburi et al., 2023), summarization (Jin et al., 2024), and question answering (Zhuang et al., 2023). However, they frequently encounter the issue of output deviations (Huang et al., 2025), which undermines the reliability of their responses. Retrieval-Augmented Generation (RAG) (Lewis et al., 2020), integrated into leading models like GPT-4o (Hurst et al., 2024), Deepseek-v3 (Liu et al., 2024) and Claude-3.5 (Anthropic, 2024), aims to tackle this problem. RAG combines a retriever to fetch relevant documents and a generator to formulate answers, enhancing the model's reliability by leveraging external knowledge. Nevertheless, RAG cannot completely eliminate such deviations (Chen et al., 2024; Wood & Forbes, 2024), posing a persistent challenge in ensuring the quality of model-generated content.

The challenge of addressing output deviations highlights the need for systems that can effectively identify and manage situations prone to unreliable outputs. Our research aims to train models to reject queries outside its capability while ensuring stable and accurate responses to manageable ones, as illustrated in Figure 1(a). Achieving this goal requires high-quality model conversation datasets specifically designed to teach the model when to refuse to answer, while also including queries that the model can answer correctly in order to maintain its capabilities. However, relevant datasets are exceedingly scarce, and the creation of such datasets is highly challenging due to the need for extensive manual annotation, while the majority of existing model conversation data remains unlabeled. To address this, we propose leveraging active learning (AL) (Settles, 1995) to screen model conversation records. AL systematically identifies the most informative samples from extensive collections of unlabeled data, enabling the creation of a small but high-quality human-annotated dataset through focused manual annotation.



(a) An example related to our tasks (b) Overall framework

Figure 1: (a) An example regarding our tasks. In the scenario on the left, the original model provides an incorrect response to the user; in this case, we expect the model to decline answering the question, whereas in the scenario on the right, where the original model delivers a correct response, we aim for it to generate accurate responses more consistently. (b) Overall framework of our approach. The example of preference set construction is the same as (a).

Unfortunately, existing AL methods perform poorly in the RAG scenario. These methods can be categorized into two categories: uncertainty-based approaches that select samples with the model's prediction uncertainty, and diversity-based approaches that prioritize samples distinct from annotated ones to avoid redundancy. Traditional uncertainty-based methods demonstrate instability when applied to RAG-related tasks (Tsvigun et al., 2022; Snijders et al., 2023), while existing diversity-based methods (Maekawa et al., 2022; Xie et al., 2023) fail to consider the unique three-segment structure of RAG conversation records (e.g., a query, retrieved documents, and a model-generated response), resulting in insufficient diversity in the collected data. With the advent of LLMs, some researchers have employed LLMs to measure sample uncertainty (Li et al., 2024), but this approach requires LLMs to evaluate the uncertainty of all unlabeled samples sequentially, leading to excessive time and computational costs when applied in practical RAG scenarios. Therefore, developing AL methods tailored specifically for RAG systems becomes crucial.

To address this gap, this paper proposes **AL4RAG**, a novel AL strategy specifically designed for RAG models, as illustrated in Figure 1(b). We improve upon existing diversity-based methods (Tsvigun et al., 2022; Margatina et al., 2023) by independently considering the various fields of RAG data, which effectively accommodates the unique nature of RAG data. This approach selectively identifies the most diverse samples from unlabeled model conversation records. By annotating these strategically selected samples, we can construct a high-quality human-annotated preference dataset. However, to construct a preference dataset, it is essential to introduce a refusal option. Yet, in existing RAG model conversation datasets, such as RAGTruth (Niu et al., 2024), each question corresponds to only a single answer, which allows for the identification of unreliable outputs but lacks the option to mark when a model should refuse to answer. To address this challenge, we ask annotators to assign labels related to output reliability to the selected model conversations. Each sample is then labeled based on whether its output is reliable, indicating whether the model should prefer to answer or refuse to respond. This approach enables us to build a dedicated preference dataset. Using this dataset for model optimization can significantly improve the performance of RAG models, thereby achieving our intended objectives. In summary, our main contributions are:

- To the best of our knowledge, we are the first to propose an AL framework for RAG contexts. Our approach introduces an efficient selection strategy designed to address the distinctive data patterns inherent in RAG-based datasets.
- We proposed retrieval-augmented similarity (***ras***) for measuring the similarity between samples within the unique data patterns of RAG contexts, enabling more accurate measurement of sample distances.

- We proposed a novel annotation method for constructing preference datasets. With that, we expanded the RAGTruth dataset (Niu et al., 2024) with responses derived from Llama-3-8B-Instruct (Grattafiori et al., 2024) and Qwen2.5-7B-Instruct (Team, 2024), and created the first human preference dataset tailored to the RAG scenario for handling both problematic queries and answerable ones.
- Extensive AL-driven model optimization was conducted on the constructed dataset, with results demonstrating the effectiveness of our approach.

2 Related Work

2.1 Active Learning

Active learning (AL) is a widely adopted technique for optimizing the trade-off between annotation costs and model performance by selecting the most informative samples from large unlabeled datasets. Central to AL are three components: a labeling oracle, an unlabeled data pool, and a query strategy. Common strategies include uncertainty-based methods, which prioritize difficult samples based on prediction uncertainty (Beluch et al., 2018; Liu et al., 2021; Schröder et al., 2021; Maekawa et al., 2022; Rouzegar & Makrehchi, 2024), and diversity-based methods, which focus on selecting diverse samples to enrich datasets (Hasan et al., 2018; Sinha et al., 2019; Agarwal et al., 2020; Maekawa et al., 2022; Xie et al., 2023).

Active learning (AL) has been effectively applied across various NLP tasks, including text classification (Yan et al., 2020; Schröder et al., 2021), text summarization (Gidiotis & Tsoumakas, 2022; Tsvigun et al., 2022), and question answering (Karamcheti et al., 2021; Padmakumar & Mooney, 2021), achieving significant cost reductions and performance improvements. These approaches have demonstrated strong potential in optimizing model training efficiency and enhancing overall system performance. Despite their successes, existing methods often neglect the influence of inherent sample properties on diversity. Addressing this gap, our work introduces a novel approach for evaluating sample diversity in the RAG context by comparing similarities across different data fields.

2.2 Active Learning Meets LLMs

As large language models (LLMs) continue to advance, their integration with AL has become a focal point for addressing high annotation costs (Tan et al., 2024) and challenges in effective knowledge utilization (Xu et al., 2024). Currently, the integration of AL with LLMs primarily involves three approaches: employing traditional active learning methods to select samples for the downstream processes of LLMs (e.g., fine-tuning, in-context learning, evaluation) (Xie et al., 2023; Margatina et al., 2023; Bayer & Reuter, 2024), utilizing LLMs to assess sample quality (e.g., uncertainty estimation) (Li et al., 2024), and leveraging LLMs to replace human annotators (Xiao et al., 2023; Kholodna et al., 2024). For instance, Margatina et al. (2023) demonstrated the effectiveness of similarity sampling for classification, framing in-context learning's example selection as a single-round AL task. Li et al. (2024) proposed LDCAL for text summarization, while Rouzegar & Makrehchi (2024) balanced cost and accuracy in text classification. Other studies addressed noisy data filtering (Taneja & Goel, 2024) and explored the use of LLMs as annotators (Zhang et al., 2023), highlighting both strengths and limitations.

Our research integrates AL into the RAG framework, leveraging its capabilities to address the unique challenges of fine-tuning LLMs. Specifically, we focus on selecting high-impact samples that enhance model performance while considering diversity within the RAG setting. To the best of our knowledge, this is the first study to explore AL-driven optimization for LLMs in the RAG context.

3 Problem Definition & Preliminaries

3.1 Problem Definition

To enhance a model's performance under RAG setting, we employ preference optimization based on the model's own conversational history. The goal is twofold: to strengthen the model's ability to reject queries it cannot answer accurately and consistently, and to improve response stability for

in-capability queries. Each unlabeled sample consists of a user query q, a reference document r, and a model-generated answer a. We identify the most informative samples and label them by assessing consistency between a and r, then use these curated samples for preference optimization, thereby **refining the model under limited annotation resources**.

Inspired by prior works (Tsvigun et al., 2022; Margatina et al., 2023), we adopt a diversity-based approach, evaluating diversity by computing distances among samples to prioritize varied and informative data. Experimental results in Section 6 and Appendix A also prove its effectiveness.

3.2 Diversity-based Active Learning

Given an unlabeled dataset $U = \{x_1, x_2, \ldots, x_n\}$, feature extraction algorithms (e.g., TF-IDF) transform each sample x_i into a feature vector $\mathbf{x_i}$. Initially, k samples are randomly selected to form the initial selected set $S = \{s_1, s_2, \ldots, s_k\}$, and these samples are removed from U.

In subsequent rounds, the average distance $\mathcal{D}$ between each sample x_i in U and all samples in S is measured as:

$$\mathcal{D}(x_i) = \frac{1}{|S|} \sum_{j=1}^{|S|} (1 - \frac{\mathbf{x_i} \cdot \mathbf{s_j}}{\|\mathbf{x_i}\| \|\mathbf{s_j}\|}) \tag{1}$$

The top-k samples ranked by $\mathcal{D}$ are added to S and subsequently removed from U. This process repeats until the annotation budget is reached.

Unlike methods that label samples during each selection round, we label all samples in S collectively at the end, reducing annotators' waiting time and improving efficiency.

3.3 RAG Framework

In the domain of NLP, RAG is a crucial methodology. Suppose a user inputs a natural language query q, there is a pre-established knowledge repository housing a collection of text chunks r_i. The system computes a set of relevance scores $f(q, r_i)$ to screen relevant text chunks from this repository against q. Next, based on a predefined threshold τ or a predefined number k, the text chunks that meet the condition $f(q, r_i) > \tau$ or the top-k relevant chunks are retrieved and ranked, generating a set $R = \{r_1, r_2, \ldots, r_k\}$. These retrieved chunks are combined with q into a prompt $P = [q; R]$, which is the input to an LLM $\mathcal{M}$ to generate an answer $a = \mathcal{M}(P)$. By leveraging the retrieved text, the generated answer attains enhanced contextual accuracy.

Evidently, in contrast to other scenarios, within the RAG framework, each conversation of the model encompasses multiple attributes, specifically the user-posed query, the retrieved references, and the answer generated by the model. In this setting, the measurement of distances between samples becomes more complicated. The direct application of AL can lead to inaccurate measurement of distances among samples. Hence, it is imperative to develop a method for measuring sample distances tailored to the RAG scenario.

4 AL4RAG

To train models to address both out-of-capability and in-capability queries while working within a limited annotation budget, we need to select a subset of the most informative samples from the model's conversation records. Next, we will annotate these samples and construct a preference dataset. Finally, we use these annotated samples for DPO training. The specific process is as follows:

4.1 Active Learning Process

We employ AL to select informative samples from the RAG model's historical conversation records. Specifically, the process entails the following steps:

(1) Random selection of initial samples for labeling. Initially, a small subset of samples is selected randomly from the entire dataset as the selected set.

(2) Measurement of similarity between remaining and selected samples. The second step involves measuring the similarity between unselected samples and the selected ones. In this step, we use the

user input (e.g., user query, question) q as the unit for measuring sample similarity.

$$\text{sim}(x, y) = \frac{\mathbf{x_q} \cdot \mathbf{y_q}}{\|\mathbf{x_q}\|\|\mathbf{y_q}\|} \tag{2}$$

where $\mathbf{x_q}$, $\mathbf{y_q}$ refer to queries of sample x and y.

(3) Measurement of similarity among remaining samples. The same method is used to measure the similarity among the remaining samples in the unselected set.

(4) Scoring of remaining samples. An IDDS score (Tsvigun et al., 2022) is assigned to each of the remaining samples based on the equation as follows:

$$\text{IDDS}(x) = \lambda \frac{\sum_{j=1}^{|U|} \text{sim}(\mathbf{x}, \mathbf{x_j})}{|U|} - (1-\lambda) \frac{\sum_{i=1}^{|S|} \text{sim}(\mathbf{x}, \mathbf{x_i})}{|S|} \tag{3}$$

where λ is a hyper-parameter, U refers to the unselected set and S refers to the selected set.

(5) Expansion of the selected set: Move the top %k samples, based on their scores, to the selected set. Then, proceed back to step (2).

Our goal is to pick samples that are different from the ones already selected but similar to those not yet chosen, thereby excluding outliers. This process iterates until the target sample count is achieved. The detailed pseudo code is shown in Appendix D.

4.2 Data Annotation & Preference Dataset Construction

After obtaining an informative subset through AL, we need to annotate these samples to construct a preference dataset. In general, the construction of a preference dataset involves the annotator choosing the better one among two candidate answers (a_1, a_2) to an input prompt p. Then, the better one is set as a_w, and the other one is set as a_l. However, in practical applications of the RAG model, users typically request only a single response from the model for a given query. As a result, we are unable to obtain a pair of answers to a query. For our task, we propose a novel method for dataset construction. Specifically, we ask annotators to assess the reliability of the model's response in each RAG conversation and then generate a label h, where $h = 0$ indicates that the response has no unreliable elements, and $h = 1$ indicates that the response contains unreliable element. Here, "unreliable element" refer to content in the model's response that is **unsupported by or contradict with the reference**, which is also known as faithfulness hallucination in Huang et al. (2025). Next, we examine the h label of each sample. For samples where $h = 0$, we designate the model's original response as a_w and an explicit rejection response as a_l. Conversely, for samples where $h = 1$, we assign the explicit rejection response as a_w and the model's original response as a_l. By modifying the feedback strategy, we successfully construct a preference dataset tailored for scenarios involving only a single model response. The detailed pseudo code can be found in Appendix D.

4.3 Fine-tuning Process

Inspired by previous work (Khaki et al., 2024), we adopt Direct Preference Optimization (DPO) (Rafailov et al., 2024) to achieve the aforementioned objectives. DPO fine-tunes LLMs to align their outputs with human preferences, simplifying optimization by eliminating the need for complex reward function estimation (Ouyang et al., 2022; Bai et al., 2022).

With the constructed preference dataset, we define two policies: π_θ, the model being optimized, and π_o, the original model used as a baseline. The optimization phase involves minimizing a loss function based on human preferences:

$$\mathcal{L}_{\text{DPO}}(\theta) = -\log \sigma \beta \left(\log \frac{\pi_\theta(a_w|p)}{\pi_o(a_w|p)} - \log \frac{\pi_\theta(a_l|p)}{\pi_o(a_l|p)} \right) \tag{4}$$

where σ is the sigmoid function, and β adjusts alignment speed with human preferences.

By fine-tuning the model with DPO, we can achieve the previously outlined optimization objectives, thereby mitigating model trust issues caused by unreliable responses.

5 AL4RAG$_{\text{RAS}}$

5.1 OBSERVATION

Conventional sample similarity measurement based on user input can lead to inaccurate judgment of sample distances. Classical Active Learning (AL) has achieved significant success in both Computer Vision (CV) (Budd et al., 2021; Tuia et al., 2021; Wang et al., 2023) and Natural Language Processing (NLP) (Zhang et al., 2022), with numerous methods leveraging sample diversity to enhance learning efficiency (Shi et al., 2021; Kim et al., 2021; Li et al., 2022; Jin et al., 2022). In both CV and NLP tasks such as text classification (Tan et al., 2021), text summarization (Tsvigun et al., 2022), semantic parsing (Li et al., 2023), and information extraction (Duan, 2024), samples typically consist of a single attribute (e.g., input images or text), allowing straightforward similarity measurements. In contrast, the RAG scenario involves conversation records with three distinct attributes: user query, retrieved references, and LLM-generated response. Measuring similarity solely based on user input is inadequate, as unreliability in RAG applications often stems from misunderstanding references (Chen et al., 2024). Combining multiple attributes or using prompt-based similarity also introduces inaccuracies, as length variations (e.g., long references or template text dominating short queries) artificially inflate similarity between otherwise distinct samples. To address this, our approach evaluates each attribute independently, enabling more accurate similarity measurements and clearer sample differentiation, thereby better capturing the critical role of the query. As shown in Figure 2, our method consistently outperforms input-based and prompt-based similarity across varying data ratios and tasks, demonstrating its robustness and adaptability to real-world RAG applications. For detailed experimental results, please refer to Section 6.2 and Appendix A.

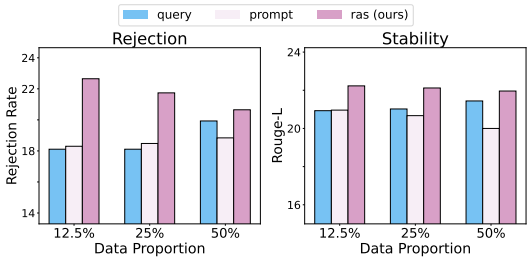


Figure 2: Part of the experimental results of query similarity, prompt similarity and *ras*, which shows the impact of different similarity measurement on model performance. The left graph shows the performance of handling out-of-capability queries, and the graph on the right shows the performance of handling model-answerable queries.

5.2 AL4RAG WITH RETRIEVAL-AUGMENTED SIMILARITY

Similar to the method mentioned previously, we carry out the same series of steps to screen samples. However, in step (2) and (3), to address the observed issues, we propose **retrieval-augmented similarity** (*ras*), a novel metric explicitly designed for RAG-structured data. It redefines sample comparison by holistically modeling the multi-faceted nature of RAG samples, comprising a user query q, retrieved reference documents r, and a generated answer a, where the prompt $p = [q; r]$ serves as the generator's input.

Unlike typical AL algorithms, which often treat the input as a single entity (Margatina et al., 2023; Taneja & Goel, 2024), *ras* adopts a structured, attribute-wise similarity mechanism. It first computes semantic similarities for the query q and reference r independently, then aggregates them into an intermediate value. Crucially, *ras* applies a minimum operation between this aggregate and the full-prompt similarity, effectively emphasizing the most discriminative attributes and reducing bias from imbalanced text lengths. This dual strategy ensures robustness in scenarios where lengthy references overshadow short queries or templated prompts distort the true relationships between samples. Simultaneously, it guarantees accurate measurement when queries are similar but references are not.

Formally, given two samples x and y, the *ras* metric is defined as:

$$\text{ras}(x, y) = \min\left(\frac{\mathbf{x_p} \cdot \mathbf{y_p}}{\|\mathbf{x_p}\|\|\mathbf{y_p}\|}, \frac{1}{2}\left(\frac{\mathbf{x_q} \cdot \mathbf{y_q}}{\|\mathbf{x_q}\|\|\mathbf{y_q}\|} + \frac{\mathbf{x_r} \cdot \mathbf{y_r}}{\|\mathbf{x_r}\|\|\mathbf{y_r}\|}\right)\right) \tag{5}$$

where $\mathbf{x_p}$, $\mathbf{y_p}$, $\mathbf{x_q}$, $\mathbf{y_q}$ and $\mathbf{x_r}$, $\mathbf{y_r}$ refer to p, q and r embeddings of x and y. Detailed pseudo code is presented in Appendix D. In the main experiments of this paper, we utilized TF-IDF vectorization. In the subsequent experiments, we compared the impacts of different vectorization methods on the final performance of the model.

6 EXPERIMENTS

6.1 EXPERIMENTAL SETTINGS

Tasks & Dataset

We evaluated our method on Question Answering, Summary, and Data-to-text tasks using RAGTruth (Niu et al., 2024), a high-quality manually annotated dataset for RAG faithfulness, containing queries, references, model responses, and reliability labels. Since RAGTruth lacks responses from newer models, we deployed Llama-3-8B-Instruct and Qwen2.5-7B-Instruct to generate these, then used Deepseek-v3-0324 for annotation (replacing manual work due to budget constraints). For quality control (Figure 3), each entry was annotated three times, with final labels determined by majority vote; conflicting cases (e.g., format-noncompliant outputs) were manually verified. To build the preference dataset, we selected responses from the chosen models, identified their reliability labels, and designated preferred/non-preferred answers. The resulting dataset (around 3,000 entries per model) was split into training/validation/test sets at 8:1:1, with each sample containing a query, reference, chosen answer, and rejected answer. AL strategies were applied solely to the training set.

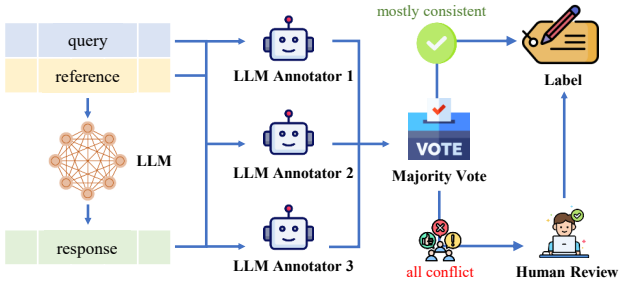


Figure 3: Quality control of LLM-assisted annotation.

Model We selected Llama-3-8B-Instruct, Llama-2-7B-chat and Qwen2.5-7B-Instruct as our base models. However, the standard DPO process relies on an SFT model, but none exists for our task. Thus, we fine-tuned each model on its corresponding training set. This empowered the model to better evaluate query complexity and handle rejection-required scenarios.

Baselines To verify the effectiveness of our method, we selected several baselines for comparison, including Random, Entropy (Wang & Shang, 2014), Coreset (Sener & Savarese, 2017), BLEUVar (Xiao et al., 2020), Naive IDDS (Tsvigun et al., 2022), LDCAL (Li et al., 2024), $\text{IDDS}_{\text{prompt}}$ and $\text{IDDS}_{\text{q,r combined}}$. For baseline introductions, please refer to Appendix A.1.

Evaluation Metrics For the performance of refusing to answer out-of-capability queries, we use **Rejection Rate (RR)** and **F1** score for evaluation. For the performance of correctly answering queries, we measure it by the text similarity between the model's responses and the correct reference answers, including **Rouge-L** and **BERTScore**. Furthermore, we employ Spark X1 to evaluate the **Faithfulness** of model-generated answers by quantifying their consistency with reference materials. Specifically, we present the proportion of answers that demonstrate alignment with the references.

Implementation Details For the AL process, we set the number of iteration rounds for both AL4RAG and IDDS to 5. In the fine-tuning processes, we employ LoRA (Hu et al., 2021) with a learning rate of 1×10^{-5} and train for one epoch. During the generation phase, we set the temperature of the LLM to 0.7 and report the mean performance and MSE obtained from five independent generation runs. Note that MSE for Rouge-L and BERTScore are not provided. This is because their MSE values are exceedingly small, making it impossible to discern differences among the results.

6.2 MAIN RESULTS

In the main context, we report the results of Llama-3-8B-Instruct. For results of Llama-2-7B-chat and Qwen2.5-7B-Instruct, please refer to Appendix A.2 and A.3.

6.2.1 REJECTION RESULTS

Table 1 compares the performance of models trained on data selected by different active learning methods in handling out-of-capability queries, evaluated at data usage proportions of 12.5%, 25%, and 50% with Llama-3-8B-Instruct as the base model. Our method consistently outperforms all baselines across all data ratios for both models. Interestingly, as data proportion increases, our

Table 1: Overall rejection performance of different algorithms under different data proportions when applied to Llama-3. The best results are **bolded**, and the second-best results are underlined.

Algorithms	12.5%		25%		50%	
	RR	F1	RR	F1	RR	F1
Random	23.04±0.03	28.25±0.06	23.04±0.01	27.60±0.05	24.02±0.24	28.13±0.42
Entropy	26.96±0.01	30.47±0.00	23.53±0.01	27.78±0.04	23.53±0.27	28.27±0.47
Coreset	26.96±0.13	30.19±0.09	25.98±0.21	31.06±0.21	22.06±0.01	25.55±0.00
BLEUVar	25.00±0.10	28.50±0.16	26.96±0.08	30.85±0.06	25.49±0.06	31.09±0.12
Naive IDDS	25.00±0.13	30.32±0.16	22.55±0.02	26.58±0.05	23.04±0.06	26.87±0.09
LDCAL	23.04±0.12	27.12±0.06	25.98±0.03	31.24±0.00	28.43±0.06	32.14±0.08
$\text{IDDS}_{\text{prompt}}$	25.49±0.21	29.73±0.10	23.53±0.00	26.38±0.00	24.02±0.08	27.77±0.05
$\text{IDDS}_{\text{q,r combined}}$	25.00±0.10	29.43±0.11	23.04±0.06	27.42±0.04	19.61±0.15	23.61±0.14
AL4RAG	27.45±0.09	32.29±0.04	25.98±0.03	30.52±0.00	23.53±0.04	28.81±0.04
$\text{AL4RAG}_{\text{ras}}$	**31.37**±0.13	**35.37**±0.18	**30.39**±0.21	**34.20**±0.13	**28.43**±0.00	**33.76**±0.02

Table 2: Overall stability performance of various algorithms across different data proportions when applied to Llama-3. The best results are **bolded**, and the second-best results are underlined.

Algorithms	12.5%			25%			50%		
	Rouge-L	BERTScore	Faithfulness	Rouge-L	BERTScore	Faithfulness	Rouge-L	BERTScore	Faithfulness
Random	33.83	50.97	74.53±0.01	33.19	50.13	71.97±0.00	33.31	50.48	74.96±0.04
Entropy	33.62	51.09	69.05±0.00	33.23	50.56	74.24±0.03	33.19	50.51	72.45±0.06
Coreset	33.59	50.82	74.78±0.03	33.59	50.57	71.03±0.05	33.25	50.08	74.82±0.01
BLEUVar	33.38	50.25	74.16±0.10	33.28	50.27	72.37±0.03	33.63	50.99	74.82±0.01
Naive IDDS	33.74	50.74	73.14±0.00	32.92	49.71	74.74±0.03	32.64	50.01	73.62±0.06
LDCAL	33.26	50.60	74.54±0.05	33.67	50.44	74.75±0.01	33.71	50.56	74.79±0.07
$\text{IDDS}_{\text{prompt}}$	33.37	50.44	73.87±0.04	32.59	49.33	72.30±0.01	33.05	50.04	69.93±0.11
$\text{IDDS}_{\text{q,r combined}}$	33.61	50.66	73.47±0.02	33.23	50.35	72.88±0.02	**34.10**	**51.37**	68.12±0.06
AL4RAG	33.65	50.56	70.56±0.02	33.60	50.96	73.65±0.00	33.71	50.97	73.76±0.07
$\text{AL4RAG}_{\text{ras}}$	**34.15**	**51.13**	**75.11**±0.05	**33.73**	**50.98**	**75.33**±0.01	33.40	50.36	**75.25**±0.02

method's performance slightly declines while baselines improve, suggesting our method effectively selects impactful samples under smaller proportions, reducing noise. Additionally, all AL algorithms perform well at 12.5% and outperform random selection of 50% data, indicating AL's effectiveness with small proportions by capturing useful samples. As proportion increases, AL's advantage over random selection diminishes; some are even surpassed at 50%. However, our method maintains an edge, demonstrating its viability at high proportions.

6.2.2 Stability Results

Table 2 demonstrates the performance of models trained on data selected by different active learning methods, in ensuring accurate responses to questions that fall within the model's capacity. It can be seen that our method comprehensively outperforms the baseline methods when the data ratios are 12.5% and 25%, regardless of whether it is the traditional similarity metrics or the faithfulness evaluated by the LLM. When the data ratio is 50%, our method still maintains the lead in terms of faithfulness, and also demonstrates a strong performance in traditional metrics. Notably, random selection performs remarkably well across all data ratios, especially in terms of faithfulness, outperforming most AL methods. However, our method consistently surpasses random selection, which further highlights the advancement of our method. Additionally, it can be observed that there is no obvious correlation between faithfulness and traditional similarity metrics. This further highlights the need for new text quality evaluation metrics in the era of LLMs.

6.3 Ablation Study

In the ablation experiments, we modify the terms involved in the weighted-average part (i.e., $\frac{\mathbf{x_q}\cdot\mathbf{y_q}}{\|\mathbf{x_q}\|\|\mathbf{y_q}\|}$ and $\frac{\mathbf{x_r}\cdot\mathbf{y_r}}{\|\mathbf{x_r}\|\|\mathbf{y_r}\|}$) of equation 5. Specifically, we investigated the individual and combined effects of the components (i.e., question, reference, answer). We conducted experiments under a quarter of the Llama-3-8B-Instruct training set, with the results presented in Table 3. It can be seen that including query and reference yields optimal model performance in both tasks. Specifically,

Table 3: Effect of different components on model performance. The best results are **bolded**, and the second-best results are underlined.

Components			Rejection		Stability		
query	*reference*	*answer*	RR	F1	Rouge-L	BERTScore	Faithfulness
✓	×	×	27.94±0.01	31.96±0.01	32.81	49.53	72.78±0.04
×	✓	×	26.96±0.18	31.33±0.16	32.87	49.97	74.67±0.02
×	×	✓	21.57±0.21	25.08±0.15	33.18	50.07	72.12±0.02
✓	×	✓	20.10±0.00	24.21±0.02	33.66	50.64	73.28±0.06
×	✓	✓	24.02±0.24	28.50±0.24	32.91	49.95	74.13±0.04
✓	✓	✓	23.04±0.15	27.44±0.23	33.59	50.57	75.11±0.09
✓	✓	×	**30.39±0.21**	**34.20±0.13**	**33.73**	**50.98**	**75.33±0.01**

Table 4: The impact of different vectorization methods on model performance under a quarter of the data volume. The best results are **bolded**.

Methods	Rejection		Stability		
	RR	F1	Rouge-L	BERTScore	Faithfulness
TF-IDF	**30.39±0.21**	**34.20±0.13**	**33.73**	**50.98**	**75.33±0.01**
Sentence-BERT	23.53±0.19	27.59±0.17	32.81	49.93	72.85±0.03
stella	25.49±0.06	28.97±0.07	33.29	50.48	73.51±0.04

with other components identical, models using reference info consistently outperform those without in rejection tasks. Though the reference-only model does not outperform the query-only one, this does not weaken the reference's contribution to rejection performance. Additionally, query and reference both significantly impact stability, highlighting their key roles. Notably, incorporating the answer usually worsens performance, indicating interference. Thus, we advocate combining query and reference. We further explored the impact of diverse data ratios and training steps on model performance, and the results are presented in Appendix A.4 and A.5.

6.4 Impact of different vectorization methods

To explore the influence of different vectorization methods on the model performance, we selected two pre-trained text embedding models for comparison. One is *Sentence-BERT* (Reimers, 2019), which is widely used in AL research. The other one is *stella*[1], a 1.5B model ranked highly on the MTEB (Muennighoff et al., 2022) leaderboard[2]. Table 4 shows the performance of TF-IDF and these two text embedding models with a quarter of the data on Llama-3-8B-Instruct. It can be seen that TF-IDF significantly outperforms the two pre-trained models in both tasks. This discrepancy likely stems from fundamental differences in their approach: pre-trained models focus on capturing deep semantic relationships, while TF-IDF emphasizes surface-level text form and structure. This distinction allows TF-IDF to better differentiate samples with similar meanings but varying expressions and labels, making it particularly effective at identifying fine-grained variations. Moreover, the performance of *stella* is better than that of *Sentence-BERT* in both tasks, which indicates that more advanced text embedding models can better distinguish samples, in line with intuitive expectations.

7 Conclusion

In this work, we introduce AL4RAG, the first AL framework for RAG, proposing an effective selection strategy tailored to the unique data patterns of RAG. To improve sample differentiation, we develop retrieval-augmented similarity (*ras*), enabling more accurate measurement of sample distances. Additionally, with our proposed annotation method, we expand the RAGTruth dataset and construct the first human preference dataset for RAG, allowing models to handle both problematic and answerable queries effectively. Extensive AL-driven optimization on the constructed dataset demonstrates that our approach consistently outperforms baselines, enhancing both response stability and the model's ability to reject unreliable queries. These contributions provide a strong foundation for advancing RAG-based learning and optimizing LLMs with external knowledge.

[1] https://huggingface.co/dunzhang/stella_en_1.5B_v5

[2] https://huggingface.co/spaces/mteb/leaderboard

REFERENCES

Harika Abburi, Michael Suesserman, Nirmala Pudota, Balaji Veeramani, Edward Bowen, and Sanmitra Bhattacharya. Generative ai text classification using ensemble llm approaches. *arXiv preprint arXiv:2309.07755*, 2023.

Sharat Agarwal, Himanshu Arora, Saket Anand, and Chetan Arora. Contextual diversity for active learning. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVI 16*, pp. 137–153. Springer, 2020.

AI Anthropic. Claude 3.5 sonnet model card addendum. *Claude-3.5 Model Card*, 2, 2024.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022.

Markus Bayer and Christian Reuter. Activellm: Large language model-based active learning for textual few-shot scenarios. *arXiv preprint arXiv:2405.10808*, 2024.

William H Beluch, Tim Genewein, Andreas Nürnberger, and Jan M Köhler. The power of ensembles for active learning in image classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 9368–9377, 2018.

Samuel Budd, Emma C Robinson, and Bernhard Kainz. A survey on active learning and human-in-the-loop deep learning for medical image analysis. *Medical image analysis*, 71:102062, 2021.

Jiawei Chen, Hongyu Lin, Xianpei Han, and Le Sun. Benchmarking large language models in retrieval-augmented generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 17754–17762, 2024.

Lianzhai Duan. Research on relation extraction method based on active learning. *International Journal of Computer and Information System (IJCIS)*, 5(2):92–101, 2024.

Alexios Gidiotis and Grigorios Tsoumakas. Should we trust this summary? bayesian abstractive summarization to the rescue. In *Findings of the Association for Computational Linguistics: ACL 2022*, pp. 4119–4131, 2022.

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

Mahmudul Hasan, Sujoy Paul, Anastasios I Mourikis, and Amit K Roy-Chowdhury. Context-aware query selection for active learning in event recognition. *IEEE transactions on pattern analysis and machine intelligence*, 42(3):554–567, 2018.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, et al. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*, 43(2):1–55, 2025.

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.

Hanlei Jin, Yang Zhang, Dan Meng, Jun Wang, and Jinghua Tan. A comprehensive survey on process-oriented automatic text summarization with exploration of llm-based methods. *arXiv preprint arXiv:2403.02901*, 2024.

Qiuye Jin, Mingzhi Yuan, Qin Qiao, and Zhijian Song. One-shot active learning for image segmentation via contrastive learning and diversity-based sampling. *Knowledge-Based Systems*, 241: 108278, 2022.

Siddharth Karamcheti, Ranjay Krishna, Li Fei-Fei, and Christopher D Manning. Mind your outliers! investigating the negative impact of outliers on active learning for visual question answering. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 7265–7281, 2021.

Saeed Khaki, JinJin Li, Lan Ma, Liu Yang, and Prathap Ramachandra. Rs-dpo: A hybrid rejection sampling and direct preference optimization method for alignment of large language models. *arXiv preprint arXiv:2402.10038*, 2024.

Nataliia Kholodna, Sahib Julka, Mohammad Khodadadi, Muhammed Nurullah Gumus, and Michael Granitzer. Llms in the loop: Leveraging large language model annotations for active learning in low-resource languages. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 397–412. Springer, 2024.

Kwanyoung Kim, Dongwon Park, Kwang In Kim, and Se Young Chun. Task-aware variational adversarial active learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8166–8175, 2021.

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

Dongyuan Li, Ying Zhang, Zhen Wang, Shiyin Tan, Satoshi Kosugi, and Manabu Okumura. Active learning for abstractive text summarization via llm-determined curriculum and certainty gain maximization. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 8959–8971, 2024.

Shibo Li, Jeff M Phillips, Xin Yu, Robert Kirby, and Shandian Zhe. Batch multi-fidelity active learning with budget constraints. *Advances in Neural Information Processing Systems*, 35:995–1007, 2022.

Zhuang Li, Lizhen Qu, Philip R Cohen, Raj Tumuluri, and Gholamreza Haffari. The best of both worlds: Combining human and machine translations for multilingual semantic parsing with active learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 9511–9528, 2023.

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*, 2024.

Zhuoming Liu, Hao Ding, Huaping Zhong, Weijia Li, Jifeng Dai, and Conghui He. Influence selection for active learning. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 9274–9283, 2021.

Seiji Maekawa, Dan Zhang, Hannah Kim, Sajjadur Rahman, and Estevam Hruschka. Low-resource interactive active labeling for fine-tuning language models. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pp. 3230–3242, 2022.

Katerina Margatina, Timo Schick, Nikolaos Aletras, and Jane Dwivedi-Yu. Active learning principles for in-context learning with large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 5011–5034, 2023.

Niklas Muennighoff, Nouamane Tazi, Loïc Magne, and Nils Reimers. Mteb: Massive text embedding benchmark. *arXiv preprint arXiv:2210.07316*, 2022.

Cheng Niu, Yuanhao Wu, Juno Zhu, Siliang Xu, Kashun Shum, Randy Zhong, Juntong Song, and Tong Zhang. Ragtruth: A hallucination corpus for developing trustworthy retrieval-augmented language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 10862–10878, 2024.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35: 27730–27744, 2022.

Aishwarya Padmakumar and Raymond J Mooney. Dialog policy learning for joint clarification and active learning queries. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 13604–13612, 2021.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36, 2024.

N Reimers. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*, 2019.

Hamidreza Rouzegar and Masoud Makrehchi. Enhancing text classification through llm-driven active learning and human annotation. In *Proceedings of The 18th Linguistic Annotation Workshop (LAW-XVIII)*, pp. 98–111, 2024.

Christopher Schröder, Andreas Niekler, and Martin Potthast. Revisiting uncertainty-based query strategies for active learning with transformers. *arXiv preprint arXiv:2107.05687*, 2021.

Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. *arXiv preprint arXiv:1708.00489*, 2017.

Burr Settles. Active learning literature survey. *Science*, 10(3):237–304, 1995.

Tianze Shi, Adrian Benton, Igor Malioutov, and Ozan İrsoy. Diversity-aware batch active learning for dependency parsing. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 2616–2626, 2021.

Samarth Sinha, Sayna Ebrahimi, and Trevor Darrell. Variational adversarial active learning. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 5972–5981, 2019.

Ard Snijders, Douwe Kiela, and Katerina Margatina. Investigating multi-source active learning for natural language inference. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 2187–2209, 2023.

Wei Tan, Lan Du, and Wray Buntine. Diversity enhanced active learning with strictly proper scoring rules. *Advances in Neural Information Processing Systems*, 34:10906–10918, 2021.

Zhen Tan, Dawei Li, Song Wang, Alimohammad Beigi, Bohan Jiang, Amrita Bhattacharjee, Mansooreh Karami, Jundong Li, Lu Cheng, and Huan Liu. Large language models for data annotation: A survey. *arXiv preprint arXiv:2402.13446*, 2024.

Karan Taneja and Ashok Goel. Can active label correction improve llm-based modular ai systems? In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 9019–9031, 2024.

Qwen Team. Qwen2.5: A party of foundation models, September 2024. URL `https://qwenlm.github.io/blog/qwen2.5/`.

Akim Tsvigun, Ivan Lysenko, Danila Sedashov, Ivan Lazichny, Eldar Damirov, Vladimir Karlov, Artemy Belousov, Leonid Sanochkin, Maxim Panov, Alexander Panchenko, et al. Active learning for abstractive text summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pp. 5128–5152, 2022.

Devis Tuia, Michele Volpi, Loris Copa, Mikhail Kanevski, and Jordi Munoz-Mari. A survey of active learning algorithms for supervised remote sensing image classification. *arXiv preprint arXiv:2104.07784*, 2021.

Dan Wang and Yi Shang. A new active labeling method for deep learning. In *2014 International joint conference on neural networks (IJCNN)*, pp. 112–119. IEEE, 2014.

Haoran Wang, Qiuye Jin, Shiman Li, Siyu Liu, Manning Wang, and Zhijian Song. A comprehensive survey on deep active learning and its applications in medical image analysis. *arXiv preprint arXiv:2310.14230*, 2023.

Michael C Wood and Adam A Forbes. 100% hallucination elimination using acurai. *arXiv preprint arXiv:2412.05223*, 2024.

Ruixuan Xiao, Yiwen Dong, Junbo Zhao, Runze Wu, Minmin Lin, Gang Chen, and Haobo Wang. Freeal: Towards human-free active learning in the era of large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 14520–14535, 2023.

Tim Z Xiao, Aidan N Gomez, and Yarin Gal. Wat zei je? detecting out-of-distribution translations with variational transformers. *arXiv preprint arXiv:2006.08344*, 2020.

Yichen Xie, Han Lu, Junchi Yan, Xiaokang Yang, Masayoshi Tomizuka, and Wei Zhan. Active finetuning: Exploiting annotation budget in the pretraining-finetuning paradigm. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 23715–23724, 2023.

Zhipeng Xu, Zhenghao Liu, Yibin Liu, Chenyan Xiong, Yukun Yan, Shuo Wang, Shi Yu, Zhiyuan Liu, and Ge Yu. Activerag: Revealing the treasures of knowledge via active learning. *arXiv preprint arXiv:2402.13547*, 2024.

Yi-Fan Yan, Sheng-Jun Huang, Shaoyi Chen, Meng Liao, and Jin Xu. Active learning with query generation for cost-effective text classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 6583–6590, 2020.

Ruoyu Zhang, Yanzeng Li, Yongliang Ma, Ming Zhou, and Lei Zou. Llmaaa: Making large language models as active annotators. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 13088–13103, 2023.

Zhisong Zhang, Emma Strubell, and Eduard Hovy. A survey of active learning for natural language processing. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 6166–6190, 2022.

Yuchen Zhuang, Yue Yu, Kuan Wang, Haotian Sun, and Chao Zhang. Toolqa: A dataset for llm question answering with external tools. *Advances in Neural Information Processing Systems*, 36: 50117–50143, 2023.

A Additional Experimental Details

Experiments of Llama-2-7B-chat and Qwen2.5-7B-Instruct are carried out on eight NVIDIA RTX 3090 GPUs, and experiments of Llama-3-8B-Instruct are conducted on four NVIDIA RTX A6000 GPUs. In terms of stability performance, we present the results of the ROUGE family and BERTScore for Llama-2-7B-chat and present the same metric as Llama-3-8B-Intruct with faithfulness evaluated by Deepseek-v3 for Qwen2.5-7B-Instruct.

A.1 Introduction of the compared baselines

We present a brief introduction of the compared baselines below. Among these baselines, Entropy and BLEUVar are uncertainty-based, Coreset and IDDS are diversity-based, and LDCAL combines both uncertainty and diversity.

Table 5: Overall rejection performance when base model is Llama-2-7B-chat. The best results are **bolded**, and the second-best results are underlined.

Algorithms	12.5%		25%		50%	
	RR	F1	RR	F1	RR	F1
Random	18.52±0.02	29.29±0.06	18.87±0.03	29.90±0.05	20.46±0.04	31.88±0.07
Entropy	19.40±0.01	30.38±0.02	20.99±0.08	32.52±0.13	20.99±0.05	32.40±0.10
Coreset	19.05±0.04	29.80±0.07	18.17±0.00	29.30±0.00	21.19±0.02	32.90±0.03
BLEUVar	19.05±0.18	29.50±0.34	20.28±0.00	31.63±0.01	20.99±0.02	32.49±0.02
Naive IDDS	20.11±0.01	31.74±0.03	20.99±0.08	32.67±0.14	20.99±0.05	32.29±0.08
LDCAL	16.51±0.02	27.11±0.05	17.96±0.00	28.36±0.01	16.32±0.01	27.98±0.02
$\text{IDDS}_{\text{prompt}}$	19.40±0.07	30.61±0.12	18.69±0.09	29.80±0.15	20.11±0.09	31.27±0.16
$\text{IDDS}_{\text{q,r combined}}$	19.75±0.00	31.11±0.01	20.46±0.01	32.27±0.03	18.34±0.00	29.21±0.00
AL4RAG	19.05±0.07	30.03±0.13	18.87±0.06	29.43±0.12	20.63±0.02	32.38±0.04
$\text{AL4RAG}_{\text{ras}}$	**23.46±0.04**	**35.43±0.06**	**22.93±0.06**	**35.53±0.14**	**21.69±0.01**	**33.36±0.02**

Table 6: Overall stability performance when base model is Llama-2-7B-chat. The best results are **bolded**, and the second-best results are underlined.

Algorithms	12.5%				25%				50%			
	Rouge-L	Rouge-1	Rouge-2	BERTScore	Rouge-L	Rouge-1	Rouge-2	BERTScore	Rouge-L	Rouge-1	Rouge-2	BERTScore
Random	20.68	29.02	14.41	28.34	20.07	28.38	13.86	27.66	20.52	29.14	14.50	28.36
Entropy	20.60	29.28	14.40	29.09	21.59	30.15	15.25	30.00	20.37	29.08	14.08	28.10
Coreset	20.30	28.92	14.18	28.49	20.78	29.01	14.50	28.67	20.69	29.27	14.63	28.69
BLEUVar	19.75	28.48	13.65	27.38	21.62	30.46	15.33	29.89	20.46	28.99	14.13	28.21
Naive IDDS	20.49	29.07	14.47	28.75	20.65	29.06	14.24	29.05	19.80	27.83	13.85	27.67
LDCAL	14.82	20.65	9.54	18.65	15.96	22.11	10.81	20.57	14.45	19.82	9.21	18.20
$\text{IDDS}_{\text{prompt}}$	20.96	29.69	14.88	28.90	20.67	29.15	14.44	28.64	20.00	28.79	13.95	28.02
$\text{IDDS}_{\text{q,r combined}}$	20.70	29.18	14.38	28.47	19.92	28.10	13.66	27.52	20.69	28.95	14.35	28.18
AL4RAG	20.93	29.74	14.57	29.60	21.02	29.57	14.61	29.37	21.44	30.64	14.81	29.64
$\text{AL4RAG}_{\text{ras}}$	**22.23**	**31.17**	**15.63**	**30.90**	**22.12**	**31.08**	**15.81**	**31.01**	**21.96**	**31.45**	**15.28**	**30.82**

- **Random**: The samples for annotation are randomly selected all at once.
- **Entropy** (Wang & Shang, 2014): An AL strategy that selects samples with the highest prediction uncertainty, measured by maximum entropy, to improve model performance.
- **Coreset** (Sener & Savarese, 2017): Select a small, representative subset of data that approximates the entire dataset's distribution for efficient learning.
- **BLEUVar** (Xiao et al., 2020): Use BLEU variance to prioritize samples with high uncertainty by treating documents as points in a high-dimensional space.
- **Naive IDDS** (Tsvigun et al., 2022): Select samples dissimilar to labeled ones but similar to unlabeled ones, based on document embeddings.
- **LDCAL** (Li et al., 2024): Employ LLMs (originally GPT-3.5, we use Deepseek-v3 instead) to partition the data and select samples with a strategy that computes the average certainty gain.
- $\textbf{IDDS}_{\text{prompt}}$: Use prompt similarity, incorporating queries, references, and template text, as the sample similarity measure, with IDDS as the query strategy.
- $\textbf{IDDS}_{\text{q,r combined}}$: Similar to the above method, but only concatenate the query and the reference, and use the similarity of this part as the sample similarity.

A.2 Main Results for Llama-2-7B-chat

To evaluate the effectiveness of our method on less capable base models, we applied it to Llama-2-7B-chat. Table 5 and Table 6 present the performance of models trained on data selected by different active learning methods. For the ability of rejecting queries outside the model's capability, our method consistently outperforms all baselines across various data proportions. Notably, similar to Llama-3, our method achieve the best performance when the data ratio is 12.5%, again demonstrating its superiority. It should be noted that the performance of LDCAL on Llama-2 is not as good as that on Llama-3. We believe that this is because Llama-2 itself has relatively weaker capabilities,

Table 7: Overall rejection performance when base model is Qwen2.5-7B-Instruct. The best results are **bolded**, and the second-best results are underlined.

Algorithms	12.5%		25%	
	RR	F1	RR	F1
Random	23.08±0.00	32.73±0.00	23.08±0.00	32.73±0.00
Entropy	23.08±0.00	32.54±0.03	24.36±0.03	35.19±0.07
Coreset	25.64±0.03	35.39±0.05	23.08±0.10	32.95±0.14
BLEUVar	23.85±0.02	33.45±0.06	24.62±0.05	34.39±0.10
Naive IDDS	24.36±0.03	35.50±0.06	24.36±0.03	34.20±0.04
LDCAL	25.64±0.01	36.87±0.05	24.62±0.02	34.57±0.05
AL4RAG$_{ras}$	**26.92±0.03**	**37.83±0.06**	**26.92±0.03**	**38.18±0.07**

Table 8: Overall stability performance when base model is Qwen2.5-7B-Instruct. The best results are **bolded**, and the second-best results are underlined.

Algorithms	12.5%			25%		
	Rouge-L	BERTScore	Faithfulness	Rouge-L	BERTScore	Faithfulness
Random	41.82	60.38	57.56±0.04	41.78	60.53	56.83±0.03
Entropy	41.92	60.43	58.18±0.03	41.73	60.42	60.40±0.02
Coreset	41.44	60.03	58.92±0.01	41.49	60.13	61.01±0.01
BLEUVar	41.55	60.14	58.55±0.03	41.56	60.27	59.66±0.05
Naive IDDS	41.79	**60.77**	58.79±0.05	41.85	60.52	60.15±0.00
LDCAL	41.95	60.40	59.90±0.04	**42.04**	60.72	61.13±0.02
AL4RAG$_{ras}$	**42.01**	60.65	**61.01±0.03**	41.97	**60.81**	**61.62±0.06**

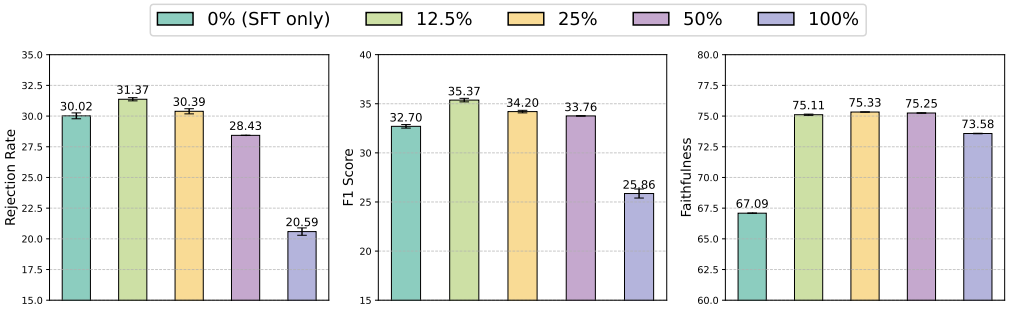
and the more difficult samples selected by LDCAL will confuse the model, resulting in a deterioration of its performance. As for the ability of ensuring accurate responses, our method consistently outperforms all baselines across various data proportions. Notably, similar to the rejection task, it achieves peak performance at a data ratio of 12.5%, a trend also observed in both IDDS-variant methods. This suggests that these methods can effectively identify informative samples even with a limited amount of data, ensuring efficient model optimization. Among them, our approach exhibits the most significant advantage, further demonstrating its effectiveness in stability performance.

A.3 Main Results for Qwen2.5-7B-Instruct

To verify the effectiveness of our method on other model series, we conducted experiments on Qwen2.5-7B-Instruct with data proportions to be 12.5% and 25%. Table 7 and Table 8 present the performance of models trained on data selected by different active learning methods. Our method still maintain its superiority in both rejection performance and stability performance, demonstrating its effectiveness and generalizability across different model series.

A.4 Ablation study on data ratios

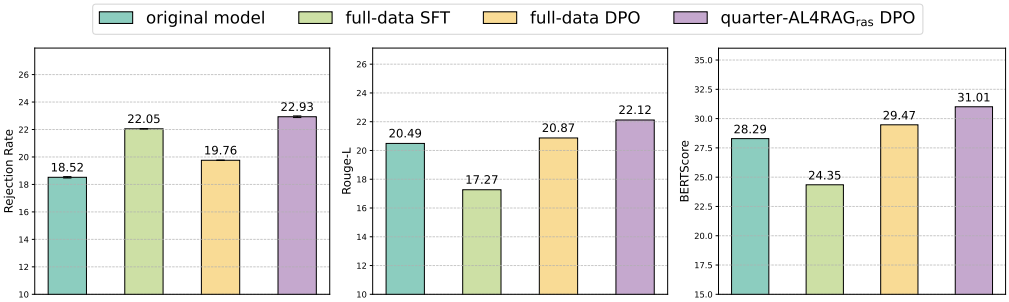
Figure 4 shows the performances of models trained via DPO with varying data ratios. Our method is used for data selection. It can be observed that training with a small amount of data using DPO can not only strengthen the rejection performance of the SFT model but also significantly improve its accuracy in answering questions within its ability. As the amount of training data increases, the model's ability to learn to refuse to answer questions during the fine-tuning process will first strengthen and then start to decline. When the proportion of training data reaches 100%, the rejection performance of the model is already much worse than that of the SFT-only model. Regarding the performance of answering questions within the model's ability, as the proportion of data increases, the performance of the model also shows a trend of first increasing and then decreasing. This is because the data selected by our method has almost no noise when the data proportion is small, but when the data proportion increases, the noise will also increase, leading to a decline in the model's performance. Moreover, the model trained with all the data using DPO still significantly outperforms the SFT-only model, which demonstrates the excellent effectiveness of DPO in this aspect.



(a) Rejection Performance (b) Rejection Performance (c) Stability Performance

Figure 4: Ablation study on data ratios. We compared the performances of models trained via DPO with varying data ratios. (a) Rejection rate of rejection performance; (b) F1 score of rejection performance; (c) Faithfulness of stability performance.

A.5 Ablation study on training steps



(a) Rejection Performance (b) Stability Performance (c) Stability Performance

Figure 5: Ablation study on training steps. We compared the performances of the original Llama-2-7B-chat model, the full-data SFT model, the full-data DPO-trained model, and the model trained via DPO with 25% data selected by our method. (a) Rejection rate of rejection performance; (b) Rouge-L of stability performance; (c) BERTScore of stability performance.

Figure 5(a) compares rejection performance across Llama-2-based models: the original model, a fine-tuned model (100% training data), a DPO-trained model (100% training data), and a DPO-trained model (25% training data, selected by our method). The model trained via DPO with data selected by our method achieves the best refusal performance, and the SFT model performs second-best, while the model trained via DPO with 100% data performs worst after the original model. This highlights that fine-tuning equips models with the ability to reject answering out-of-capability queries. Moreover, DPO training with a small amount of high-quality data further improves this capability, while training with a noisy dataset greatly weakens it. Compared to baselines, our method is the only one that consistently outperforms SFT in all data proportions, highlighting its superiority.

Figures 5(b) and 5(c) compare the stability performance of different training approaches based on Llama-2, including the original model, the SFT model trained on 100% of the training data, a DPO-trained model using 100% of the training data, and a DPO-trained model using only 25% of the data selected by our method. While fine-tuning achieves strong rejection capabilities, it significantly reduces stability compared to the original model, indicating a trade-off between the two objectives. In contrast, DPO training with 100% of the data reduces rejection performance compared to the SFT model but significantly improves stability, suggesting that optimizing for preference learning helps maintain correct answers. Furthermore, DPO training with 25% data selected by our method improves stability without compromising rejection performance, demonstrating that our approach effectively balances both goals and enhances the model's overall robustness.

B IMPACT STATEMENT

Our work proposes a method for efficiently selecting data samples for model training, which can make more effective use of computing resources and further contribute to sustainable development. However, there may be potential social consequences of our work. In the context of data privacy, the use of model conversation records is a critical concern. As we rely on these records to train and optimize the model, there is a potential risk of exposing sensitive user information. If proper security measures are not in place, this could lead to privacy breaches, where personal details, opinions, or interactions captured in the conversations might be accessed without authorization.

C PROMPTS

Prompt for Annotating Faithfulness

This is the "question", "reference" and "generated_answer" fields in my data. "generated_answer" is the content that the model answers the "question" according to the "reference". Your task is to check this content and determine whether "generated_answer" correctly answers the "question" with reference to the "reference". If so, output "1"; if not, output "0". Do not output any other content.
question: {question}
reference: {reference}
generated_answer: {answer}
Output:

Prompts for Answer Generation

Question Answering

Below is a question:
{question}
Below are related passages:
{reference}
Your task is to answer the question strictly based on the related passages.
In case the passages do not contain the necessary information to answer the question, please reply with: "Unable to answer based on given passages."
If you cannot answer the question precisely, please reply with: "Sorry, this question is beyond my ability."
Output:

Summarization

Below are some news:
{reference}
Your task is to write a summary of the news.
If you cannot summarize the news precisely, please reply with: "Sorry, this question is beyond my ability."
Output:

Data-to-text Writing

Your task is to write an objective overview about the following local business based only on the provided structured data in the JSON format.
You should include details and cover the information mentioned in the customers' review. The overview should be 100 - 200 words. Don't make up information.
If you cannot summarize the data precisely, please reply with: "Sorry, this question is beyond my ability."
Below are the structured data:
{reference}
Output:

D ALGORITHMS OF AL4RAG

Algorithm 1 AL4RAG: Sample Selection Algorithm for RAG Context

Require: The entire dataset D, user input q, hyper-parameter λ, the proportion k of samples to be expanded each time, the expected number of samples in the selected set N

Ensure: The selected sample set S after step-by-step expansion

Initialize the selected sample set $S = \varnothing$
Randomly select a small subset S' from the dataset D as the initial selected sample set
$S = S'$
$D = D - S'$ ▷ Remove the selected samples from the dataset
while D is not empty and $|S| < N$ **do** ▷ Step (2): Measure the similarity between unselected samples and selected samples
    for each unselected sample x in D **do**
        for each selected sample y in S **do**
            $\mathbf{x}_q \leftarrow$ Obtain the query representation of sample x
            $\mathbf{y}_q \leftarrow$ Obtain the query representation of sample y
            $\text{sim}(x, y) \leftarrow \frac{\mathbf{x}_q \cdot \mathbf{y}_q}{\|\mathbf{x}_q\| \|\mathbf{y}_q\|}$
        end for
    end for ▷ Step (3): Measure the similarity among unselected samples
    for each unselected sample x in D **do**
        for each unselected sample z in D where $z \neq x$ **do**
            $\mathbf{x}_q \leftarrow$ Obtain the query representation of sample x
            $\mathbf{z}_q \leftarrow$ Obtain the query representation of sample z
            $\text{sim}(x, z) \leftarrow \frac{\mathbf{x}_q \cdot \mathbf{z}_q}{\|\mathbf{x}_q\| \|\mathbf{z}_q\|}$
        end for
    end for ▷ Step (4): Score the remaining unselected samples
    for each unselected sample x in D **do**
        $\text{sum\_sim}_U \leftarrow 0$
        for each unselected sample x_j in D **do**
            $\text{sum\_sim}_U \leftarrow \text{sum\_sim}_U + \text{sim}(x, x_j)$
        end for
        $\text{sim}_U \leftarrow \frac{\text{sum\_sim}_U}{|D|}$
        $\text{sum\_sim}_S \leftarrow 0$
        for each selected sample x_i in S **do**
            $\text{sum\_sim}_S \leftarrow \text{sum\_sim}_S + \text{sim}(x, x_i)$
        end for
        $\text{sim}_S \leftarrow \frac{\text{sum\_sim}_S}{|S|}$
        $\text{IDDS}(x) \leftarrow \lambda \cdot \text{sim}_U - (1 - \lambda) \cdot \text{sim}_S$
    end for ▷ Step (5): Expand the selected sample set
    Sort the samples in D in descending order according to their IDDS scores
    Select the top $k\%$ of the samples, denoted as set E
    $S \leftarrow S \cup E$
    $D \leftarrow D - E$
end while
return S

Algorithm 2 Preference Dataset Construction for Single-Response RAG Model

Require: A set of samples obtained through Active Learning (AL), a RAG model, annotators
Ensure: A preference dataset P

Initialize the preference dataset $P = \varnothing$
for each sample s in the set of samples obtained through AL **do**
  $p \leftarrow s.prompt$ ▷ Get the input prompt p from sample s
  $r \leftarrow s.response$ ▷ Get the model's response r from sample s
  $h \leftarrow$ annotators.assessReliability(r) ▷ Assess Reliability labels
  $a_{reject} \leftarrow$ generateExplicitRejectionResponse() ▷ Generate an explicit rejection response
  if $h = 0$ **then**
    $a_w \leftarrow r$, $a_l \leftarrow a_{reject}$
  else
    $a_w \leftarrow a_{reject}$, $a_l \leftarrow r$
  end if
  $P \leftarrow P \cup \{(p, a_w, a_l)\}$ ▷ Add the pair (p, a_w, a_l) to the preference dataset
end for
return P

Algorithm 3 Sample Screening with Retrieval-Augmented Similarity (RAS)

Require: The entire dataset D, hyper-parameter λ, the proportion k of samples to be expanded each time, the expected number of samples in the selected set N
Ensure: The selected sample set S after step-by-step expansion

Initialization ▷ The initialization steps are the same as those of Algorithm 1.
while D is not empty and $|S| < N$ **do** ▷ Step (2): Measure the retrieval-augmented similarity between unselected samples and selected samples
  for each unselected sample x in D **do**
    $\mathbf{x_q} \leftarrow$ Obtain the query representation of sample x
    $\mathbf{x_r} \leftarrow$ Obtain the reference representation of sample x
    $\mathbf{x_p} \leftarrow$ Combine $\mathbf{x_q}$ and $\mathbf{x_r}$ to form the prompt representation of sample x
    for each selected sample y in S **do**
      $\mathbf{y_q} \leftarrow$ Obtain the query representation of sample y
      $\mathbf{y_r} \leftarrow$ Obtain the reference representation of sample y
      $\mathbf{y_p} \leftarrow$ Combine $\mathbf{y_q}$ and $\mathbf{y_r}$ to form the prompt representation of sample y
      $\text{sim}_q \leftarrow \frac{\mathbf{x_q} \cdot \mathbf{y_q}}{\|\mathbf{x_q}\| \|\mathbf{y_q}\|}$
      $\text{sim}_r \leftarrow \frac{\mathbf{x_r} \cdot \mathbf{y_r}}{\|\mathbf{x_r}\| \|\mathbf{y_r}\|}$
      $\text{sim}_p \leftarrow \frac{\mathbf{x_p} \cdot \mathbf{y_p}}{\|\mathbf{x_p}\| \|\mathbf{y_p}\|}$
      $\text{ras}(x, y) \leftarrow \min(\text{sim}_p, \frac{1}{2}(\text{sim}_q + \text{sim}_r))$
    end for
  end for ▷ Step (3): Measure the retrieval - augmented similarity among unselected samples
  for each unselected sample x in D **do**
    Obtain $\mathbf{x_q}, \mathbf{x_r}, \mathbf{x_p}$ ▷ The same representation obtainment for sample x as Step (2).
    for each unselected sample z in D where $z \neq x$ **do**
      $\mathbf{z_q} \leftarrow$ Obtain the query representation of sample z
      $\mathbf{z_r} \leftarrow$ Obtain the reference representation of sample z
      $\mathbf{z_p} \leftarrow$ Combine $\mathbf{z_q}$ and $\mathbf{z_r}$ to form the prompt representation of sample z
      $\text{sim}_q \leftarrow \frac{\mathbf{x_q} \cdot \mathbf{z_q}}{\|\mathbf{x_q}\| \|\mathbf{z_q}\|}$
      $\text{sim}_r \leftarrow \frac{\mathbf{x_r} \cdot \mathbf{z_r}}{\|\mathbf{x_r}\| \|\mathbf{z_r}\|}$
      $\text{sim}_p \leftarrow \frac{\mathbf{x_p} \cdot \mathbf{z_p}}{\|\mathbf{x_p}\| \|\mathbf{z_p}\|}$
      $\text{ras}(x, z) \leftarrow \min(\text{sim}_p, \frac{1}{2}(\text{sim}_q + \text{sim}_r))$
    end for
  end for
  Subsequent steps ▷ The subsequent steps are the same as those of Algorithm 1, including scoring and expanding the set.
end while
return S