Enhancing LLM-as-a-Judge through Active-Sampling-based Prompt Optimization

Cheng Zhen[1][2][*], **Ervine Zheng**[2], **Geoffrey Tso**[2], **Jilong Kuang**[2],
[1]Oregon State University, [2]Samsung Research America,
Correspondence: ervine.zheng@samsung.com

Abstract

We introduce an active-sampling-based framework for automatic prompt optimization, designed to enhance the performance of Large Language Model (LLM)-as-a-judge systems, which use LLMs to evaluate the quality of generated contents in label-scarce settings. Unlike existing approaches that rely on extensive annotations, our method starts with no labeled data and iteratively selects and labels a small, diverse, and informative subset of samples to guide prompt refinement. At each iteration, our method evaluates the current prompt based on selected data and automatically updates the prompt, enabling efficient prompt optimization with minimal supervision. Moreover, we formulate sample selection as a convex optimization problem that balances uncertainty and diversity, maximizing the utility of limited labeling budgets. We validate our framework across popular LLMs and real-world datasets, including one from a deployed industry product. Results show that our optimized prompts consistently outperform baselines, achieving significant gains in evaluation quality and robustness while substantially reducing labeling costs.

1 Introduction

Large Language Models (LLMs) are increasingly used as automated evaluators, often referred to as *LLM-as-a-judge*, for tasks such as evaluating text generation quality and chatbot performance. While leveraging LLMs as evaluators can substantially reduce human labeling costs, their effectiveness heavily depends on the quality of the prompts. Suboptimal prompts can introduce biases (e.g., verbosity or positional biases), inconsistencies, and unreliable evaluations. These issues, as highlighted by recent studies, pose significant challenges to the reliability and robustness of LLM-based evaluation systems (Shinn et al., 2023; Yan et al., 2024).

Recent automatic prompt optimization (APO) methods have shown promise in enhancing prompt quality through techniques such as paraphrasing, LLM-based candidate generation, and feedback-driven refinement (Prasad et al., 2022; Xu et al., 2022; Zhou et al., 2022; Pryzant et al., 2023; He et al., 2024). However, these approaches often rely on ground-truth labels for the entire dataset to guide prompt refinement, restricting their use in real-world applications. Labeling data for LLM-as-a-judge systems—especially for open-ended tasks like summarization or dialogue evaluation—can be exceedingly costly and time-intensive, frequently requiring domain expertise or detailed annotations. As a result, large-scale supervision becomes impractical, with only a small fraction of data typically labeled within budget constraints.

While some prior methods address this issue by sampling data using simple heuristics (Chen et al., 2024), those strategies may miss some informative and diverse examples needed for effective prompt updates. Active learning provides a promising solution to the above challenge, which aims to efficiently train models by labeling only the most informative and diverse samples (Settles, 2009). Our work extends active learning to target prompt optimization for LLM-as-a-judge systems in evaluation tasks.

In this paper, we propose a novel approach that does not require any labeled data to start with. Our method iteratively refines the evaluation prompt through selective labeling and feedback-driven updates. At each iteration, our method actively selects a small subset of unlabeled data samples that are both diverse in content and uncertain in their prediction of the evaluation score. These samples are then labeled by human annotators and used to evaluate the performance of the current prompt. Based on the discrepancies between the prompt's outputs

[*]Work done during an internship at Samsung Research America

Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 6: Industry Track), pages 960–970
July 28-30, 2025 ©2025 Association for Computational Linguistics

and the labeled ground truth, a reflection process generates insights to guide prompt refinement. This iterative process continues until the labeling budget is exhausted, progressively improving the evaluation quality. Central to our approach is a principled sample selection mechanism, formulated as a convex optimization problem that balances uncertainty and diversity to maximize the value of each labeled sample. By focusing labeling efforts on the most informative data, our framework ensures the efficient use of limited supervision while enhancing the performance of LLM-as-a-judge systems. Our **major contributions** are summarized below:

- We introduce an automatic prompt optimization method designed specifically for LLM-as-a-judge systems in label-scarce settings, an underexplored research area
- Compared with prior works, our approach significantly enhances the efficiency of automated prompt optimization by incorporating an innovative active sampling strategy to select the most informative and diverse data
- Our active sampling strategy is framed as a subset selection problem, incorporating carefully designed constraints and convex optimization to ensure a tractable solution.
- We validate our method across multiple real-world datasets, including one from a deployed product, demonstrating that it consistently outperforms baseline methods in accuracy and labeling efficiency.

Additional Background The proposed method addresses our challenge of developing an efficient and scalable evaluation system for a conversational agent that provides personalized health coaching to users. The conversational agent uses data from wearable devices to provide actionable health insights and recommendations, in order to empower users to improve their health outcomes. Evaluating such an agent at scale presents significant difficulties due to the reliance on domain experts with health coaching backgrounds, which incurs high costs and limits scalability. To overcome these limitations, we explore LLM-based evaluation, which relies on prompt optimization with iterative refinement on annotated data. Based on the proposed approach, we designed and deployed an automated system to refine prompts via active sampling and feedback, reducing manual annotation needs. It balances trade-offs in cost, accuracy, and automation, overcoming deployment challenges and enabling scalable, cost-effective evaluation.

2 Related Work

Prompt Optimization. APO aims to refine prompts for LLMs without modifying the parameters. Early methods leverage paraphrasing, including phrase editing (Prasad et al., 2022) and back translation (Xu et al., 2022), to generate diverse candidate prompts. Subsequent advancements leveraged LLMs for prompt generation and evaluation. Notably, Automatic Prompt Engineering (Zhou et al., 2022) introduced iterative prompt generation guided by LLM feedback. Similarly, error-reflection-driven approaches (Pryzant et al., 2023; He et al., 2024) refined prompts by analyzing incorrect predictions. Other techniques have incorporated historical prompt performance data (Yang et al., 2023), expert-level planning (Wang et al., 2023), evolutionary algorithms (Fernando et al., 2023), and heuristic-driven prompt selection (Wen et al., 2025; Cui et al., 2025). Recently, heuristic-based sampling methods (Chen et al., 2024) have prioritized promising prompts informed by human feedback. Despite the advancements, most approaches heavily rely on extensive labeled data, posing challenges for low-resource scenarios.

Active Learning. Active learning is a machine learning paradigm where models selectively query the most informative samples for labeling to enhance performance while minimizing supervision (Settles, 2009). Common strategies prioritize samples based on uncertainty, diversity, or representativeness (Ren et al., 2021). Although active learning has been extensively applied to classification and regression tasks, its potential integration into prompt optimization—particularly within the context of LLM-as-a-judge—remains unexplored.

LLM-as-a-Judge and Efficient Evaluation. Recent works have explored LLMs as evaluators (i.e., judges) for ranking and scoring language model outputs. MT-Bench and Chatbot Arena (Zheng et al., 2023), JuStRank (Song et al., 2024), and Re-Evaluating LLM Judges (Liu et al., 2024) have evaluated the consistency and reliability of LLM-as-a-judge setups. In parallel, efforts in efficient benchmarking aim to reduce the annotation cost for evaluation tasks, such as by selecting fewer yet informative test examples (Li et al., 2023; Fu et al., 2024). Our work builds on these insights and focuses on the automatic optimization of LLM judges under limited labeling budgets.

3 Methodology

3.1 Problem Formulation

We consider an LLM-as-a-judge scenario, where LLM serves as a judge for evaluation tasks. The evaluation is performed through an LLM prompt p designed for a specific task T. Formally, given an initial prompt p_0, an unlabeled dataset $D = \{x_n|_{n=1}^{N}\}$ related to task T, and a labeler $L(x_n) \rightarrow y_n$ capable of providing ground-truth labels within a limited labeling budget B (i.e., the labeler can label at most B samples), our goal is to optimize the initial prompt p_0 into an improved prompt p^* to maximize the LLM's performance on the evaluation task T.

3.2 Overview of the Approach

Our framework begins with an initial prompt p_0 for the LLM-as-a-judge and an unlabeled dataset, and iteratively optimizes the prompt through active sampling and reflection-driven updates. As illustrated in Figure 1, three LLM agents collaborate in this process: the **Judge**, the **Reflector**, and the **Updater**, all implemented using the same underlying LLM but serving distinct roles.

At each iteration i, an **active sampling module**, formulated as a numerical optimization problem (Section 3.3), selects a small subset of unlabeled samples that are most informative and diverse. These selected samples are labeled by human annotators (labeler), and then passed to the **Judge**, which uses the current prompt p_i to generate evaluation predictions for the labeled samples. Next, the **Reflector** compares the Judge's predictions with the ground-truth labels and generates reflections that identify weaknesses or improvement opportunities in the current prompt. These reflections are then used by the **Updater**, which synthesizes them into a refined prompt p_{i+1} for the **Judge**. We provide example prompts for Judge, Reflector and Updater in the Appendix.

This process continues iteratively, refining the prompt at each step until the labeling budget is exhausted. At the end of the process, the finalized prompt is returned. This design enables efficient use of limited labels by ensuring that only the most impactful samples are used for prompt improvement. Algorithm 1 outlines the detailed procedure for our active prompt optimization framework.

In Algorithm 2, we provide more details on the active sampling process. The method ensures that the selected subset consists of the most uncertain

Algorithm 1 Proposed active prompt optimization

```
$k \leftarrow$ batch size
$B \leftarrow$ labeling budget
$i_{max} \leftarrow \frac{B}{k}$                ▷ max number of iterations
$p \leftarrow$ initial prompt
$D_{\text{unlabeled}} \leftarrow D_{full}$      ▷ initialize with full data
$D_{\text{labeled}} \leftarrow \varnothing$     ▷ initialize with an empty set
$i \leftarrow 1$
while $i \leq i_{max}$ do
    $D_{\text{select}} \leftarrow ActSamp(D_{\text{unlabeled}}, p)$
    $D_{\text{select}} \leftarrow Judge(D_{\text{select}})$
    $D_{\text{labeled}} \leftarrow D_{\text{labeled}} + D_{\text{select}}$
    $D_{\text{unlabeled}} \leftarrow D_{\text{unlabeled}} - D_{\text{select}}$
    reflection $\leftarrow Reflector(p, D_{\text{labeled}})$
    $p \leftarrow Updater(p, \text{reflection})$
    $i \leftarrow i + 1$
end while
return $p$
```

Algorithm 2 Active sampling

```
Require: Dataset with $n$ unlabeled samples, maximum selection size $k$, number of clusters $c$,
    hyperparameter $\lambda$
Ensure: Subset of $k$ selected samples for labeling
Initialize: Load data; extract texts and summaries
Generate $n$ uncertainty scores randomly
Encode texts into embedding space; apply K-means clustering ($c$ clusters) and assign each
    sample a cluster label
Define selection variable $\mathbf{w} \in \mathbb{R}^n$ where $w_i \in [0, 1]$
Define Objective: Maximize informativeness and diversity
    Compute entropy-based diversity scores $H(S)$ from:
      - Sample representation across text groups
      - Sample distribution across cluster groups
Optimization Problem:
Maximize: $\lambda \sum_i w_i U_i + (1 - \lambda) H(S)$
    Subject to:
      - $\sum w_i \leq k$ (selection budget)
      - Category coverage constraints for text and cluster diversity
      - $w_i \in [0, 1]$ (feasibility constraint)
Solve using a convex solver
Select top-$k$ samples with highest optimization scores; return selected subset for labeling
```

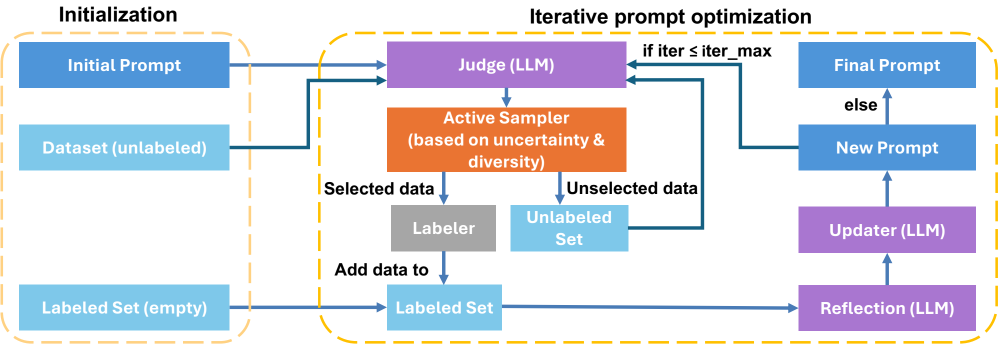


Figure 1: Flowchart of our active-sampling-based automatic prompt optimization. The process iteratively selects informative and diverse samples, refines the prompt, and stops when the labeling budget is exhausted.

and diverse samples, leading to a more efficient labeling process.

3.3 Proposed Active Sampling

Traditional reflection-based APO methods primarily select samples whose predictions (based on the current prompt) disagree with existing ground-truth labels, facilitating reflection-driven updates (Pryzant et al., 2023). However, they are not applicable in label-scarce scenarios, as they rely on full access to labeled data to identify discrepancies.

While one could randomly sample from the pool of unlabeled data, not all samples are equally helpful—some are more informative or diverse than others and thus contribute more effectively to prompt improvement. Consequently, we propose an active sampling strategy that identifies and selects samples without labels by explicitly maximizing uncertainty and diversity. This enables efficient utilization of limited labeling budgets and enhances the optimization process when initial labeled data is unavailable.

Subset Selection with Maximal Diversity and Uncertainty: Given an unlabeled dataset $D_{\text{unlabeled}}$ with N samples, our objective is to select an optimal subset $D_{\text{select}} \subseteq D_{\text{unlabeled}}$ of size k that maximizes an affine combination of uncertainty and diversity scores. To facilitate optimization in continuous space, we represent sample selection using a weight vector $\mathbf{w} \in \mathbb{R}^N$, where each element $w_x \in [0, 1]$ indicates whether sample x is selected or not (0 for unselected, 1 for fully selected). w_x is relaxed to a continuous number instead of a discrete number, which will be discussed later.

For each sample $x \in D_{\text{unlabeled}}$, an uncertainty score $U(x)$ is computed as the sum of two variances:

$$U(x) = \text{Var}_{\text{temp}}(x) + \text{Var}_{\text{rephrase}}(x), \qquad (1)$$

where $\text{Var}_{\text{temp}}(x)$ is the variance of predictions from the LLM with different sampling temperatures, reflecting uncertainty in probabilistic token generation, and $\text{Var}_{\text{rephrase}}(x)$ is the variance across predictions from paraphrased inputs, capturing sensitivity to small input perturbations. Together, these variances quantify the model's uncertainty about the sample.

Diversity is quantified using categories assigned to each sample. Formally, the diversity score $H(S)$ for subset S is defined as the weighted entropy over multiple category dimensions:

$$H(S) = -\sum_{d=1}^{D} \eta_d \sum_{c_d} P_{c_d}(S) \log P_{c_d}(S), \qquad (2)$$

$$P_{c_d}(S) = \frac{\sum_{x \in D_{\text{unlabeled}}, C_d(x) = c_d} w_x}{\sum_{x' \in \mathcal{U}} w_{x'}} \qquad (3)$$

where d denotes the index of dimension, $P_{c_d}(S)$ represents the proportion of selected samples belonging to category c_d, η_d is the weighting factor. Higher entropy indicates a more balanced representation across categories, thereby encouraging the selection of samples that cover diverse content. Researchers can select an appropriate clustering method (e.g., K-means clustering) to generate category assignments and use them as diversity dimensions. When such categorical dimensions are already part of the data (e.g., topic or user intent labels), they can be directly used instead of clustering. This flexibility allows the framework to generalize across datasets with or without predefined sub-category labels.

To actively select samples, we formulate the following optimization problem:

$$\max_{\mathbf{w}} \quad \lambda \sum_{x \in \mathcal{U}} w_x U(x) + (1 - \lambda) H(S) \qquad (4)$$

subject to the following constraints:

Example: Active-sampling-based Prompt Optimization in Hate Speech Detection
Task Description: Detect whether a given speech contains hate speech by assigning a score between 0 and 1, where 0 indicates no hate speech and 1 indicates explicit hate speech.
Training Set (Unlabeled Samples):

1. XXX immigrants are criminals who come here to steal and destroy our way of life.
2. Men/women like XXX often struggle in leadership because they think differently.
3. People with disabilities like XXX are just burdens on society.
4. People who follow XXX ideology are easily manipulated and dangerous to society.

Testing Set (Labeled Samples):

1. It's unfortunate, but XXX just aren't as productive in the workplace. (Score: **0.75**)
2. Fat/ugly people like XXX disgust me—they have no self-respect and should stay out of sight. (Score: **0.89**)

Initial Prompt: *"Given a text, assign a score between 0 and 1 to determine the likelihood that it contains hate speech."*
Issue: This prompt yields inaccurate results in the testing set, failing to guide the model to identify nuanced hate speech.
Active Sampling Step: Our approach selects the most uncertain and diverse samples for labeling based on the optimization problem as discussed before. Samples **(2)** and **(4)** are selected as they receive high uncertainty score and high diversity score based on the proposed metric. These samples are labeled by human annotators as follows:

- **(2)** Men/women like XXX often struggle in leadership because they think differently. → **0.70**
- **(4)** People who follow XXX ideology are easily manipulated and dangerous to society. → **0.75**

Reflection generated by analyzing labeled samples: *"Hate speech is not always direct or explicit. Some statements present themselves as factual observations but contain implicit biases and stereotypes targeting specific groups. Such speech, even if framed as an opinion or observation, should be considered hate speech and assigned a higher score."*
Updated prompt based on the reflection and the original prompt: *"Given a text, assign a hate speech score between 0 and 1. Consider that hate speech can be explicit (e.g., direct slurs) or implicit (e.g., statements that reinforce stereotypes or promote bias under the guise of opinion or fact). Ensure that biased and stereotype-driven speech is also identified and scored appropriately."*
Result: By labeling fewer samples, we refined the prompt to better capture implicit hate speech, significantly reducing MSE.

Figure 2: Demonstration of active sampling in hate speech detection. Our approach selects the most uncertain and diverse samples for labeling, generates reflections, and updates the prompt to improve detection performance.

$$\text{s.t.} \sum_{x \in D_{\text{unlabeled}}} w_x \le k \tag{5}$$

$$\sum_{x \in D_{\text{unlabeled}}, C_d(x)=c_d} w_x \ge \alpha |c_d|, \quad \forall d, \forall c_d \tag{6}$$

$$\sum_{x \in D_{\text{unlabeled}}} w_x U(x) \ge \beta |D_{unlabeled}| \tag{7}$$

$$0 \le w_x \le 1, \quad \forall x \tag{8}$$

Eq.4 defines our objective: we aim to select a subset of samples such that the weighted sum of their uncertainty and diversity scores is maximized. Intuitively, this helps prioritize samples that are both uncertain (the model is less confident) and diverse (spanning different content categories or topics). Maximizing this objective ensures that each selected batch of samples contributes meaningful new information to the prompt optimization process.

The constraints further shape the selection strategy to ensure efficient use of the labeling budget. Inequalities 5 and 8 help enforce sparsity by limiting the selection of k samples (details will be discussed later). Inequalities 6 and 7 impose lower bounds on diversity and uncertainty from selected samples, respectively, where α and β are constants with pre-set values, $|c_d|$ is the size of cluster c_d.

In the context of APO, there is a critical challenge to be resolved: without ground-truth labels, we cannot rely on traditional disagreement-based selection strategies that compare predictions to known labels. To address this, we estimate uncertainty based on the LLM's own predictive variability (i.e., the degree of disagreement across multiple outputs given the same input). By targeting samples that exhibit high model uncertainty and broad diversity, we increase the likelihood that the selected and labeled samples will yield meaningful reflections for prompt updates, accelerating optimization while minimizing redundancy.

The optimization problem in Eq. 4 is convex: the objective combines a linear term (uncertainty) and a concave entropy term (diversity), and the feasible region defined by Constraints is convex, consisting of linear inequalities (5-7) and a box constraint (8).

Ideally, we prefer an $\mathcal{L}_0$ pseudo-norm where w_x strictly equals 0 or 1 to enforce binary selection of data samples. However, due to computational complexity, we apply an $\mathcal{L}_1$ relaxation to facilitate efficient convex optimization (Ramirez et al., 2013). After solving this relaxed optimization problem, we perform a top-k data sample selection based on the optimized weights $\mathbf{w}^*$, determining the final subset of samples for labeling at each iteration.

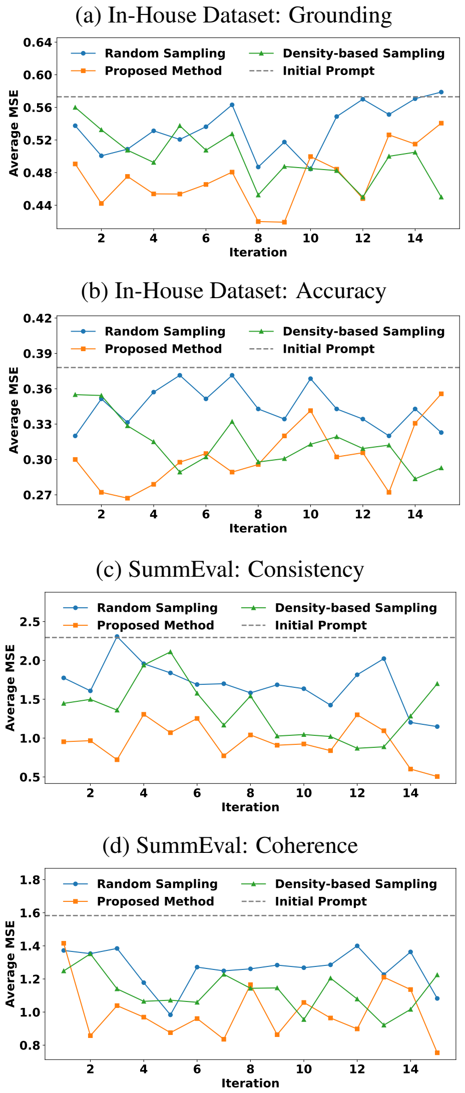
4 Experiments

Datasets We experiment on the following datasets: 1. *SummEval:* This dataset consists of original texts paired with machine-generated summaries. Each summary is evaluated by scores (1-5) in four dimensions: coherence, consistency, fluency, and relevance (Fabbri et al., 2020). 2. *In-House Health Coaching Datasets:* The dataset used in product development for an AI-based conversational agent for health coaching. In this dataset, each conversation data sample consists user health data and additional user profile information, user message, response from the conversational agent, and conversation history. The evaluation task is to judge the quality of the response. Each data sample is associated with evaluation scores (0–3) in three dimensions: accuracy (whether the response is consistent with domain knowledge), grounding (whether the response is relevant to user's personal information and history), and safety (the extent to which the response avoids harmful or inappropriate content).

Implementation Details The dataset is randomly split into training and testing sets with a 60/40 ratio. We assume all testing samples have ground-truth labels, allowing us to report MSE on the test set. The total number of iterations is set to 15. The maximum total labeling budget (B) is set to 50% of the number of samples in the training set, with each iteration using a labeling budget of $B/15$. For each experiment, we ran five times and recorded the average MSE to reduce the randomness. However, it should be noted that we may not use the entire budget to achieve sufficiently good performance. Empirically, we can apply early stopping at around Iteration 5 based on empirical alignment between human annotation and auto evaluation scores, using only around $B/3$ budget. We will provide more details on the discussions later.

For evaluation, we employ the Mean Squared Error (MSE) metric, which quantifies the average squared difference between the ground-truth scores and the scores assigned by the LLM-as-a-judge based on the current prompt.

LLMs and Baselines We conduct our experiments using four models: Gemini-1.5-Pro (Team et al., 2024), Mistral Large 2, Llama3-70b-instruct (Grattafiori et al., 2024), and Claude-3.5-sonnet (Anthropic, 2024). For each set of experiments, the same LLM is used across all three core modules of our framework: the **Judge**, the **Reflector**, and the **Updater**. As baselines, we compare our active sampling strategy with two relevant sample selection methods: 1) random selection (Ghojogh et al., 2020), and 2) density-based core-set selection (Phillips, 2017).


(a) In-House Dataset: Grounding

(b) In-House Dataset: Accuracy

(c) SummEval: Consistency

(d) SummEval: Coherence

Figure 3: MSE trends over prompt updates using different sampling strategies. (a–b) coaching quality evaluations, (c–d) text summarization evaluations. The proposed method achieves significant MSE reduction within 3-4 iterations, and we empirically apply early-stopping to achieve the best results

Uncertainty and Diversity Computation To guide active sample selection, we compute both uncertainty and diversity scores at each iteration.

The uncertainty score consists of two components: (1) the variance of predicted scores from the LLM using the current prompt at different temperature settings $\{0, 0.25, 0.5, 1\}$, and (2) the variance of predicted scores from the LLM using the current prompt on five different rephrasings of the input, generated by a pre-trained paraphrasing model (ChatGPT Paraphraser (Vladimir Vorobev, 2023)).

Diversity is computed by encoding input texts using a transformer-based encoder, MiniLM (Wang et al., 2020), followed by K-means clustering. For SummEval, we apply K-means on embeddings of summarization texts, while for the in-house health coaching dataset, clustering is performed on embeddings of user messages. The resulting cluster assignments serve as categorical labels for diversity computation, with the number of clusters set to $k = 5$ based on empirical grid search.

5 Results

Figure 3 presents Mean Squared Error (MSE) trends over 15 iterations for four strategies: *Active Sampling* (proposed), *Random Sampling*, *Density-Based Sampling*, and *Initial Judging Prompt* (no updates). Due to space limitations, we report results with *Gemini* on two evaluation tasks from each of the datasets. Results with three additional LLMs (Llama, Claude, and Mistral) and the complete set of evaluation tasks show similar trends and are provided in the **Appendix** A.1.

5.1 Performance and Efficiency Analysis

All prompt optimization strategies, including *Active Sampling*, *Random Sampling*, and *Density-Based Sampling*, reduce MSE compared to the *Initial Judging Prompt*, validating the effectiveness of iterative prompt refinement. Notably, *Active Sampling* consistently achieves the lowest MSE across most iterations, with the most substantial gains observed early in the optimization process.

By prioritizing uncertain and diverse samples, *Active Sampling* achieves significant MSE reduction within the first five iterations (labeling 16% of training samples), while alternative strategies require labeling 32–50% of samples to reach similar accuracy. As the optimization progresses, performance gains diminish and MSE curves converge, indicating limited value from remaining unlabeled samples. These trends highlight the efficiency of our method in improving evaluation quality under strict labeling budgets, particularly in early iterations when sample selection plays a critical role.

5.2 Overfitting & Early Stopping

In some experiments, we observed an increase in MSE during later iterations across all strategies. This phenomenon, consistent with prior findings (Pryzant et al., 2023), is attributed to overfitting. As prompts are updated using a fixed set of labeled data, they may become overly tailored to those examples, reducing generalization to unseen data. We apply early stopping to address this issue.

5.3 Cost Analysis

We provide a simplified cost analysis for prompt optimization in LLM-as-a-Judge scenarios. For automated prompt optimization, the total cost may include human annotation f_{anno} and LLM-related cost $f_{LLM} = f_{judge} + f_{reflector} + f_{updater}$. We denote the total number of data samples as N. For the proposed method, annotation cost is estimated as $f_{anno}(arN)$ where r is the rounds of annotation (usually set to 3 with early stopping) and a is the percentage of data samples to be annotated per round (usually set to 1/30). Similarly, the cost of Judge and Reflector is related to the total number of tokens, which depends on the number of data samples. The cost can be estimated as $f_{judge}((t + p)rN)$ and $f_{reflector}(arN)$ where t is the number of temperature values and p is the number of paraphrased text per sample to estimate uncertainty. The Updater does not analyze data samples and its cost is $f_{updator}(r)$. For the baseline method of prompt optimization without active sampling, annotation cost could be higher $f_{anno}(kN)$ where k is the preset percentage of samples to be labeled. The LLM-related cost is $f_{judge}(rkN)$, $f_{reflector}(rkN)$ and $f_{updator}(r)$. Empirically, $f_{anno} \gg f_{LLM}$ and $a < k$, making the proposed method cost-efficient.

6 Conclusion

We propose an active-sampling-based APO method to enhance the reliability of LLM-as-a-judge in label-scarce scenarios. Our approach strategically selects diverse and informative samples for labeling, enabling more effective prompt refinement with minimal human annotation. A theoretical contribution of our work is the formulation of the active sampling problem as a convex optimization problem to identify the most diverse and informative subset of samples. Experimental results across multiple datasets demonstrated that the proposed prompt optimization method achieves lower MSE compared to prior works, especially during early iterations. Consequently, our method significantly reduces the annotation budget while facilitating efficient prompt tuning. These findings underscore the critical role of active sampling strategies in improving the effectiveness of APO for LLM in evaluation tasks.

Ethical Considerations This research utilizes synthetic health data and does not involve data collection from human participants. Therefore, Institutional Review Board (IRB) approval is not required. To the best of our knowledge, the research and experiments presented in this paper do not raise ethical concerns. However, it is important to note that evaluations based on large language models (LLMs) may exhibit bias, particularly when applied to human-related data. Should the proposed algorithm be employed in future studies involving human-related data, a systematic evaluation of ethical implications and potential risks will be essential.

Acknoledgement We thank the anonymous reviewers for reviewing the manuscript.

References

AI Anthropic. 2024. The claude 3 model family: Opus, sonnet, haiku. *Claude-3 Model Card*, 1:1.

Yongchao Chen, Jacob Arkin, Yilun Hao, Yang Zhang, Nicholas Roy, and Chuchu Fan. 2024. Prompt optimization in multi-step tasks (promst): Integrating human feedback and heuristic-based sampling. *arXiv preprint arXiv:2402.08702*.

Wendi Cui, Jiaxin Zhang, Zhuohang Li, Hao Sun, Damien Lopez, Kamalika Das, Bradley A Malin, and Sricharan Kumar. 2025. Automatic prompt optimization via heuristic search: A survey. *arXiv preprint arXiv:2502.18746*.

Alexander R Fabbri, Wojciech Kryściński, Bryan McCann, Caiming Xiong, Richard Socher, and Dragomir Radev. 2020. Summeval: Re-evaluating summarization evaluation. *arXiv preprint arXiv:2007.12626*.

Chrisantha Fernando, Dylan Banarse, Henryk Michalewski, Simon Osindero, and Tim Rocktäschel. 2023. Promptbreeder: Self-referential self-improvement via prompt evolution. *arXiv preprint arXiv:2309.16797*.

Hao Fu et al. 2024. tinybenchmarks: Evaluating llms with fewer examples. *arXiv preprint arXiv:2402.14992*.

Benyamin Ghojogh, Hadi Nekoei, Aydin Ghojogh, Fakhri Karray, and Mark Crowley. 2020. Sampling algorithms, from survey sampling to monte carlo methods: Tutorial and literature review. *arXiv preprint arXiv:2011.00901*.

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

Han He, Qianchu Liu, Lei Xu, Chaitanya Shivade, Yi Zhang, Sundararajan Srinivasan, and Katrin Kirchhoff. 2024. Crispo: Multi-aspect critique-suggestion-guided automatic prompt optimization for text generation. *arXiv preprint arXiv:2410.02748*.

Long Li et al. 2023. Efficient benchmarking of language models. *arXiv preprint arXiv:2308.11696*.

Xiao Liu et al. 2024. Re-evaluating automatic llm system ranking for alignment with human preference. *arXiv preprint arXiv:2501.00560*.

Jeff M Phillips. 2017. Coresets and sketches. In *Handbook of discrete and computational geometry*, pages 1269–1288. Chapman and Hall/CRC.

Archiki Prasad, Peter Hase, Xiang Zhou, and Mohit Bansal. 2022. Grips: Gradient-free, edit-based instruction search for prompting large language models. *arXiv preprint arXiv:2203.07281*.

Reid Pryzant, Dan Iter, Jerry Li, Yin Tat Lee, Chenguang Zhu, and Michael Zeng. 2023. Automatic prompt optimization with" gradient descent" and beam search. *arXiv preprint arXiv:2305.03495*.

Carlos Ramirez, Vladik Kreinovich, and Miguel Argaez. 2013. Why l1 is a good approximation to l0: A geometric explanation.

Pengzhen Ren, Yun Xiao, Xiaojun Chang, Po-Yao Huang, Zhihui Li, Brij B Gupta, Xiaojiang Chen, and Xin Wang. 2021. A survey of deep active learning. *ACM computing surveys (CSUR)*, 54(9):1–40.

Burr Settles. 2009. Active learning literature survey.

Noah Shinn, Federico Cassano, Beck Labash, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2023. Reflexion: Language agents with verbal reinforcement learning, 2023. *URL https://arxiv.org/abs/2303.11366*.

Yixin Song et al. 2024. Justrank: Benchmarking llm judges for system ranking. *arXiv preprint arXiv:2412.09569*.

Gemini Team, Petko Georgiev, Ving Ian Lei, Ryan Burnell, Libin Bai, Anmol Gulati, Garrett Tanzer, Damien Vincent, Zhufeng Pan, Shibo Wang, et al. 2024. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*.

Maxim Kuznetsov Vladimir Vorobev. 2023. A paraphrasing model based on chatgpt paraphrases.

Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. 2020. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *Advances in neural information processing systems*, 33:5776–5788.

Xinyuan Wang, Chenxi Li, Zhen Wang, Fan Bai, Haotian Luo, Jiayou Zhang, Nebojsa Jojic, Eric P Xing, and Zhiting Hu. 2023. Promptagent: Strategic planning with language models enables expert-level prompt optimization. *arXiv preprint arXiv:2310.16427*.

Bosi Wen, Pei Ke, Yufei Sun, Cunxiang Wang, Xiaotao Gu, Jinfeng Zhou, Jie Tang, Hongning Wang, and Minlie Huang. 2025. Hpss: Heuristic prompting strategy search for llm evaluators. *arXiv preprint arXiv:2502.13031*.

Hanwei Xu, Yujun Chen, Yulun Du, Nan Shao, Yanggang Wang, Haiyu Li, and Zhilin Yang. 2022. Gps: Genetic prompt search for efficient few-shot learning. *arXiv preprint arXiv:2210.17041*.

Hanqi Yan, Qinglin Zhu, Xinyu Wang, Lin Gui, and Yulan He. 2024. Mirror: A multiple-perspective self-reflection method for knowledge-rich reasoning. *arXiv preprint arXiv:2402.14963*.

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

Tianyi Zheng, Percy Liang, Tianyi Zhang, et al. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *arXiv preprint arXiv:2306.05685*.

Yongchao Zhou, Andrei Ioan Muresanu, Ziwen Han, Keiran Paster, Silviu Pitis, Harris Chan, and Jimmy Ba. 2022. Large language models are human-level prompt engineers. In *The Eleventh International Conference on Learning Representations*.

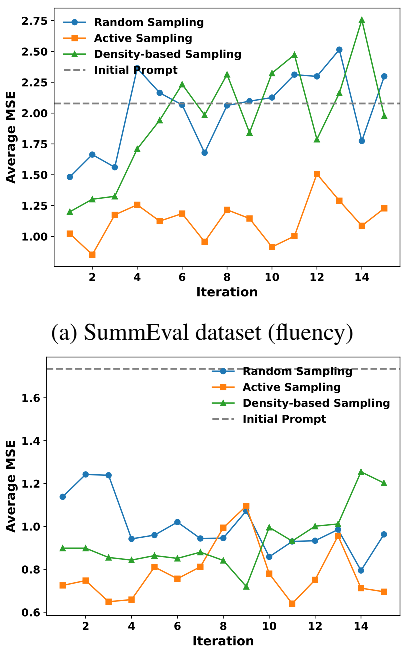
A Appendix

In the appendix, we report additional experiment results, provide examples of prompts and discuss limitations of the paper.

A.1 Additional Experimental Results

A.1.1 Gemini

Figures 4–5 present additional results from the Gemini model that are not included in the main content.



(a) SummEval dataset (fluency)

(b) SummEval dataset (relevance)

Figure 4: MSE over 15 iterations of prompt updates for SummEval across different aspects of text summarization quality.

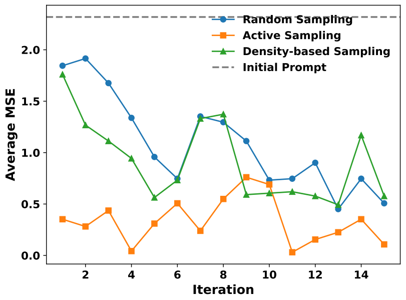
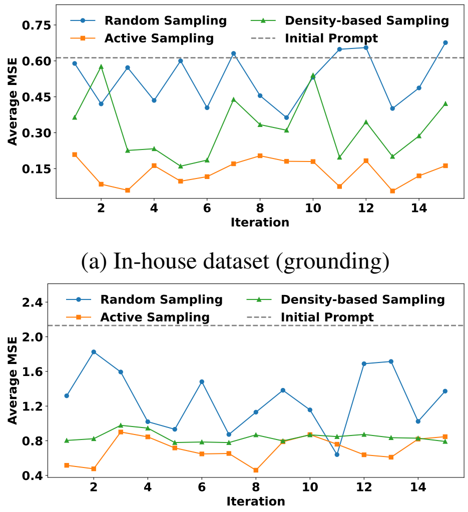


Figure 5: MSE over 15 iterations of prompt updates for the in-house dataset (safety).

A.1.2 Mistral, Llama, and Claude

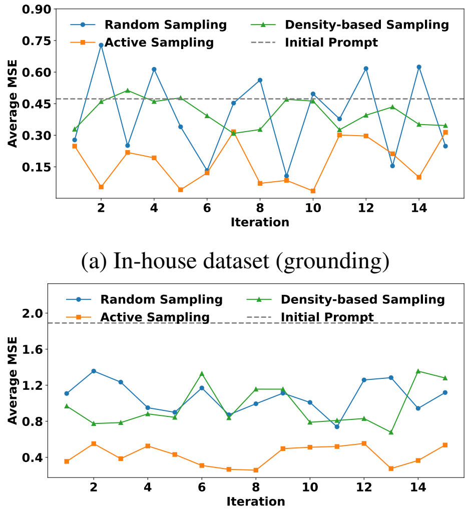
We report results for three open-sourced LLMs on one evaluation task per dataset, as other tasks exhibit similar performance trends (Figures 6–8). The same observation holds across other open-sourced LLMs evaluated in our study, where active sampling consistently outperforms baseline strategies in both prompt optimization efficiency and evaluation accuracy.



(a) In-house dataset (grounding)

(b) SummEval dataset (consistency)

Figure 6: MSE over 15 iterations of prompt updates for LLM-as-a-judge using **Mistral Large 2**, evaluated on one task per dataset.



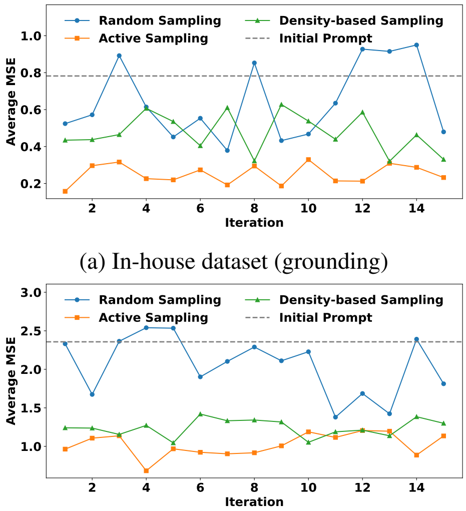
(a) In-house dataset (grounding)

(b) SummEval dataset (consistency)

Figure 7: MSE over 15 iterations of prompt updates for LLM-as-a-judge using **Llama3-70b-instruct**, evaluated on one task per dataset.

A.2 Prompt Templates

In this section, we share prompt templates for different agents as discussed in the main paper. We use the SummEval dataset as an example.



(a) In-house dataset (grounding)

(b) SummEval dataset (consistency)

Figure 8: MSE over 15 iterations of prompt updates for LLM-as-a-judge using **Claude-3.5-sonnet**, evaluated on one task per dataset.

Prompt for Judge

```
<ORIGINAL_TEXT>
{original_text}
</ORIGINAL_TEXT>

<SUMMARIZATION>
{summarization}
</SUMMARIZATION>

ORIGINAL_TEXT contains a piece of
long text, and SUMMARIZATION is another
piece of text summarizing ORIGINAL_TEXT.

Generate a score to characterize the
consistency of the SUMMARIZATION with
respect to ORIGINAL_TEXT.
Output a score between 0 and 5 ONLY
(strictly follow this rule).
```

Prompt for Reflector

```
I'm trying to write a prompt to ask LLM
provide a {metric} score for a piece
of text summarization of its original
text.
My current prompt is:
"{prompt}"

I got some results from running this
prompt with LLM on some examples. The
result for each example follows the
key-value pair format:
"'original text: xxx', 'text
summarization: xxx', '{metric} score
output with the prompt: xxx',
'ground-truth {metric} score from human
annotators: xxx'".
Below are the results for examples:
{labeled example}
```

```
For EACH example, carefully analyze the
differences between score output with
the prompt and ground-truth score from
human annotators.
Identify specific area where the prompt
could be improved to better align with
the ground-truth scores.
Provide constructive feedback by
highlighting:
1. Any patterns in the differences...
2. Suggested refinements...
3. If the current prompt is already
performing well, provide positive
feedback...
```

Prompt for Updater

```
I’m trying to write a prompt to ask LLM
to provide a {metric} score for a piece
of text summarization of its original
text.

My current prompt is:
{prompt}

I got some results from running this
prompt with LLM on some examples. The
result for each example follows the
key-value pair format:
"’original text: xxx’, ’text
summarization: xxx’, ’{metric} score
output with the prompt: xxx’,
’ground-truth {metric} score from human
annotators: xxx’".
Below are the results for examples:
{labeled example}

Based on these results, the feedback on
the current prompt is: {feedback}

Based on the above information, please
refine the prompt in ways that you
believe will genuinely enhance the
accuracy of score output.
Your major goal is to make the new prompt
better achieving alignment between score
output with the prompt and ground-truth
score from human annotators.
```

A.3 Limitations

While our approach effectively optimizes prompts with limited labeled data, we observe an issue: overfitting of APO happens in later iterations. Currently, we use early stopping to address this issue. We leave more advanced techniques (e.g., adaptive regularization, meta-learning strategies) for future work.