GRAIT: Gradient-Driven Refusal-Aware Instruction Tuning for Effective Hallucination Mitigation

Runchuan Zhu[2*], **Zinco Jiang**[2*], **Jiang Wu**[1*†], **Zhipeng Ma**[3],
Jiahe Song[2], **Fengshuo Bai**[4], **Dahua Lin**[1,5], **Lijun Wu**[1], **Conghui He**[1‡]

[1]Shanghai Artificial Intelligence Laboratory, [2]Peking University,
[3]Southwest Jiaotong University, [4]Shanghai Jiaotong University,
[5]Chinese University of Hong Kong

Correspondence: heconghui@pjlab.org.cn

Abstract

Refusal-Aware Instruction Tuning (RAIT) aims to enhance Large Language Models (LLMs) by improving their ability to refuse responses to questions beyond their knowledge, thereby reducing hallucinations and improving reliability. Effective RAIT must address two key challenges: firstly, effectively reject unknown questions to minimize hallucinations; secondly, avoid over-refusal to ensure questions that can be correctly answered are not rejected, thereby maintain the helpfulness of LLM outputs. In this paper, we address the two challenges by deriving insightful observations from the gradient-based perspective, and proposing the **G**radient-driven **R**efusal-**A**ware **I**nstruction **T**uning Framework (**GRAIT**): GRAIT (1) employs gradient-driven sample selection to effectively minimize hallucinations and (2) introduces an adaptive weighting mechanism during fine-tuning to reduce the over-refusal. Experiments on open-ended and multiple-choice question answering tasks demonstrate that GRAIT significantly outperforms existing RAIT methods in the overall performance. The source code and data will be available at https://github.com/opendatalab/GRAIT.

1 Introduction

Large Language Models (LLMs), including notable turbos like GPTs (OpenAI, 2022, 2023) and LLaMA (Touvron et al., 2023; Dubey et al., 2024), have achieved remarkable advances, demonstrating exceptional capabilities across a diverse range of downstream tasks (Kaplan et al., 2020; Vu et al., 2024; Achiam et al., 2023; Bai et al., 2024a; Jiang et al., 2024b,a). Despite this success, critical challenges persist, particularly in the generation of *hallucinations*—the models generate incorrect or fabricated information when confronted with unfamiliar or ambiguous queries (Ji et al., 2023; Feng et al.,
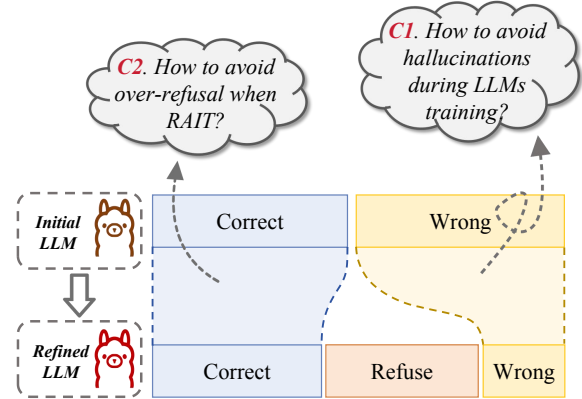

Figure 1: Descriptions of *C1 & C2*. After RAIT, the initial LLM model will largely reject unknown questions to avoid errors. However, the overly conservative nature of RAIT also led to a decrease in accuracy.

2024; Kang et al., 2024), which ultimately limits the reliability and usefulness of LLMs.

Ideally, the responsible LLM should decline to answer questions beyond its knowledge to minimize hallucinations (Wen et al.; Li et al., 2024). Recent studies (Yang et al., 2023; Zhang et al., 2024a; Xu et al., 2024a; Cheng et al., 2024; Bai et al., 2024b; Zhu et al., 2024) have developed Refusal-Aware Instruction Tuning (RAIT), which constructs the refusal-aware dataset and employs Supervised Fine-Tuning (SFT) (Dong et al., 2024; Ouyang et al.; Luo et al., 2024) to teach models to appropriately decline responses. Typically, refusal-aware datasets (Zhang et al., 2024a,b) categorize training samples into ik (correct) and idk (incorrect) groups based on response correctness. Samples with incorrect responses (idk) are treated as unknown knowledge and the answers are replaced with refusal responses like "I don't know", while the correct (ik) remain unchanged. Despite RAIT's success in reducing hallucinations (Zhang et al., 2024a; Yang et al., 2023; Zhang et al., 2025; Wan et al., 2024), studies like (Varshney et al., 2023) and (Cheng et al., 2024) highlight that models can

become overly cautious, leading to over-refusals. Therefore, as shown in Figure 1 and Figure 2, the RAIT should address the following two challenges *simultaneously*: *C1*. **How to effectively reduce the hallucinations by refusing the unknown questions?** *C2*. **How to avoid over-refusal to ensure questions that can be correctly answered are not rejected?**

To address these challenges, we introduce the **G**radient-based **R**efusal-**A**ware Instruction **T**uning Framework (**GRAIT**), which has several advantages over previous RAIT methods: (1) Unlike prior methods that rely solely on the outputs of LLMs, GRAIT utilizes gradients to achieve a more accurate representation of LLMs' internal knowledge states, ensuring better alignment between the constructed training samples and the LLMs' knowledge. (2) Our gradient-based sample selection process is more efficient, achieving comparable training results with fewer samples (Xia et al., 2024; Xu et al., 2024b). (3) By incorporating gradient information, we account for influences among training samples (Ren and Sutherland, 2024), effectively minimizing sample conflicts (Zhu et al., 2024) and better addressing the aforementioned two challenges.

In practical implementation, we first derive two theoretical observations (O_1 & O_2) by progressively addressing reduced inaccuracies (*C1*) and mitigating over-refusal (*C2*), which form the basis for designing GRAIT. Then in the framework designing: ❶ For *C1*, we leverage the Refusal Influence formula within the idk set to select a small subset, enabling the LLMs to learn the refusal paradigm while filtering out inefficient samples. ❷ For *C2*, we implement an adaptive weighting method derived from the Stable Influence formula between the correct and incorrect sample sets, assigning varying sample weights during the RAIT phase to alleviate the issue of over-refusal. In summary, our contributions are as follows:

- To the best of our knowledge, we are the first to conduct a theoretical analysis of the causes underlying the over-refusal phenomenon in LLMs.
- The GRAIT framework establishes a comprehensive workflow encompassing data construction and fine-tuning, utilizing two gradient-driven observations to enhance the model's refusal capability while effectively mitigating over-refusal.
- Through extensive experimental evaluation on both open-ended question answering and

Figure 2: Case of mitigating hallucination and avoiding over-refusal.

multiple-choice tasks, we demonstrate that GRAIT surpasses existing baselines by significantly reducing hallucination rates and enhancing overall performance.

The paper is organized as follows. In Introduction (c.f. Section 1), we present our research question and its corresponding two challenges (*C1* & *C2*). The Related Work (c.f. Section 2) provides a literature analysis concerning each challenge. In the Preliminary section (c.f. Section 3), we define the symbols and tasks relevant to GRAIT. Following this, the Theoretical Analysis section (Section 4) derives two observations, O_1 & O_2, in a progressive manner corresponding to *C1* & *C2*. The Method section (Section 5) aligns O_1 with data construction and O_2 with the RAIT phase. Section 6 introduces the experiment settings and the experimental results. Finally, Section 7 concludes this paper and discusses future research directions.

2 Related Work

2.1 Refusal-Aware Instruction Tuning

RAIT is a supervised technique designed to enhance the ability of LLMs to handle unanswerable or uncertain questions by training them to respond directly with "I don't know" (Zhang et al., 2024a; Yang et al., 2023; Zhu et al., 2024) through super-

vised fine-tuning (SFT). In Wan et al. (2024), a knowledge-based verification mechanism is proposed to ensure that the model's knowledge remains consistent with external trusted sources to prevent the spread of misinformation. Moreover, CoKE (Chen et al., 2024) probes LLMs' knowledge boundaries via internal confidence given a set of questions and then informs the LLM's decision on whether to respond with "I don't know" based on the knowledge boundaries. Additionally, (Zhu et al., 2024) and (Wang et al., 2024) refine data filtering and modification by leveraging both response certainty and correctness. Recent works have incorporated Low-Rank Adaptation (LoRA) (Hu et al.) and AdaLoRA (Wolfe et al., 2024) into RAIT to achieve further improvements. However, those RAIT methods tend to make LLMs more conservative, leading to incorrect and even **over-refusals** (Cheng et al., 2024).

2.2 Gradient Effect on LLMs' Learning

Gradient-based methods are central to recent advances in data selection. For instance, (Zhao et al., 2021; Xia et al., 2024; Yang et al., 2024) propose *Dataset Condensation*, where synthetic data is created by matching the gradients or the learning trajectory of a deep model trained on a small synthetic set to those of a model trained on the full dataset. Similarly, (Killamsetty et al., 2021; Killamsetty and Iyer, 2021) extend this concept with their *Grad-Match* framework, which selects subsets of data that closely align with the gradient information from the full dataset, allowing for efficient training with minimal performance degradation. More recently, (Ren and Sutherland, 2024; Zhao and Bilen, 2020; Qiao et al., 2024; Bai et al., 2023) emphasize that gradients can dynamically influence the learning process of LLMs during fine-tuning. (Liu et al., 2020; Xiao et al., 2024) study the influence of gradient signal-to-noise ratio's result on the test set. While gradient-based selection techniques are widely studied, applying these methods to RAIT to mitigate issues such as hallucinations and over-refusal remains largely unexplored, which presents an opportunity for further research in RAIT.

3 Preliminary

(Definition 1. RAIT Dataset)   The RAIT process can be described as follows: the initial LLM is prompted to answer all questions in the training set D_{src}. Based on the correctness of the responses, the

samples are categorized into two groups. ❶ Samples with correct responses are considered known knowledge of the LLM. These answers will remain unchanged and are referred to as ik samples, denoted as $D_{\text{ik}} = \{(x_{\text{ik}}, y_{\text{ik}})\}$ (where 'ik' stands for 'I know', x_{ik} is the known question and y_{ik} is the ground-truth label). ❷ Conversely, samples with incorrect responses are treated as unknown knowledge. Their original answers are replaced with refusal responses such as "I don't know" forming $D_{\text{idk}} = \{(x_{\text{idk}}, y_{\text{idk}})\}$ (where 'idk' stands for 'I don't know', x_{ik} is the unknown question and y_{ik} is modified refusal response such as "I don't know"). The constructed RAIT dataset, $D_{\text{rait}} = D_{\text{ik}} \cup D_{\text{idk}}$, is used to fine-tune the initial LLM, parameterized by θ, to improve its ability to refuse to answer questions beyond its knowledge.

(Definition 2. Influence Formulation)   To estimate the influence of a training datapoint on a validation sample, we use the first-order Taylor expansion of the loss function (Pruthi et al., 2020)[1]. Specifically, for a model θ_t at step t, the loss on unobservant validation sample x^u can be approximated as: $\mathcal{L}(x^u, y^u; \theta_{t+1}) \approx \mathcal{L}(x^u, y^u; \theta_t) + \langle \nabla \mathcal{L}(x^u, y^u; \theta_t), \theta_{t+1} - \theta_t \rangle$. If the model is trained using Stochastic Gradient Descent (SGD) with batch size 1 and learning rate η_t, for the observant training sample x^o, the SGD update is written as: $\theta_{t+1} - \theta_t = -\eta_t \nabla \mathcal{L}(x^o, y^o; \theta_t)$. At this point, we can define the influence formula of (x^o, y^o):

$$\mathcal{I}(x^o, y^o, x^u, y^u; \theta_t) \triangleq \eta_t \langle \nabla \mathcal{L}(x^o, y^o; \theta_t), \nabla \mathcal{L}(x^u, y^u; \theta_t) \rangle. \tag{1}$$

(Task Definition)   The objective of this task is to leverage D_{rait} to fine-tune a model and minimize the loss on two distinct types of test samples. Specifically, for samples that were previously incorrect, we aim for the model to output answers like "I don't know", while for correct samples, the predicted label should be as close as possible to the ground-truth label y_{ik} . The task can be formalized as minimizing the following loss:

$$\min \{ \mathbb{E}_{x_{\text{idk}}^u \sim D_{\text{idk}}} [\Delta \mathcal{L}(x_{\text{idk}}^u, y_{\text{idk}}^u; \theta)] + \mathbb{E}_{x_{\text{ik}}^u \sim D_{\text{ik}}} [\Delta \mathcal{L}(x_{\text{ik}}^u, y_{\text{ik}}^u; \theta)] \}, \tag{2}$$

In addition to minimizing it, a key objective of this task is to select the most suitable subset $\widetilde{D}_{\text{rait}} \subseteq D_{\text{rait}}$ for fine-tuning (c.f. Section 4 for proof).

---

[1]The reasons for using the influence formula are outlined in the appendix A.2

4008

By selecting optimal data from the RAIT dataset, we aim to improve the model's ability to refuse answers to unknown questions while minimizing over-refusal.

4 Theoretical Analysis

This part is organized as $\mathbf{O}_1 \rightarrow \mathbf{O}_2$. Before obtaining formal observation results, we first propose two assumptions:

Assumptions 1. Distribution Assumption *We assume that the distributions of* ik *or* idk *from train and test sets are identically distributed, formally expressed as:* $\Pi_{D_{\mathrm{idk}}^o} \sim \Pi_{D_{\mathrm{idk}}^u}, \Pi_{D_{\mathrm{ik}}^o} \sim \Pi_{D_{\mathrm{ik}}^u}$.

Assumptions 2. Orthogonality of Means *We further assume that the means of the gradient distributions for* idk *and* ik *are orthogonal as verified in Appendix A.3, and we have:*

$$\left\langle \mathbb{E}_{x_*}\left[\nabla\mathcal{L}(x_*, y_{\mathrm{idk}}; \theta)\right], \mathbb{E}_{x_*}\left[\nabla\mathcal{L}(x_*, y_{\mathrm{ik}}; \theta)\right]\right\rangle \approx 0, \quad (3)$$

where the $*$ denotes the symbol of either idk or ik.

4.1 Reducing Incorrectness ($\mathbf{O}_1$)

We begin by focusing on minimizing the loss to improve the rejection rate, specifically aiming to minimize the first term of (2) $\mathbb{E}_{x_{\mathrm{idk}}^u \sim D_{\mathrm{idk}}}\left[\Delta\mathcal{L}(x_{\mathrm{idk}}^u, y_{\mathrm{idk}}^u; \theta)\right]$. Then, combining equation (1), we can express this as:

$$\begin{aligned} \mathbb{E}_{x_{\mathrm{idk}}^u \sim D_{\mathrm{idk}}}\left[\Delta\mathcal{L}(x_{\mathrm{idk}}^u, y_{\mathrm{idk}}^u; \theta)\right] &\approx -\mathbb{E}_{(x_{\mathrm{idk}}^u, x_{\mathrm{idk}}^o) \sim D_{\mathrm{idk}}} \\ &\left[\mathcal{I}(x_{\mathrm{idk}}^o, y_{\mathrm{idk}}^o, x_{\mathrm{idk}}^u, y_{\mathrm{idk}}^u; \theta)\right] \end{aligned} \quad (4)$$

and the full proof is detailed in Appendix A.1.1.

Thus, samples with gradients similar to the average gradient direction of D_{idk} are the most effective in reducing the model's hallucination rate.

4.2 Alleviating Over-Refusal ($\mathbf{O}_2$)

However, we observed that if we optimize the model merely depends on RAIT, it leads to the issue of **over-refusal** (i.e., ik samples also tend to output "I don't know"). Therefore, we delved deeper into the whole target in (2) and derived the following(the full proof is detailed in Appendix A.1.2):

$$\begin{aligned} &\mathbb{E}_{x_{\mathrm{idk}}^u \sim D_{\mathrm{idk}}}\left[\Delta\mathcal{L}(x_{\mathrm{idk}}^u, y_{\mathrm{idk}}^u; \theta)\right] + \mathbb{E}_{x_{\mathrm{ik}}^u \sim D_{\mathrm{ik}}}\left[\Delta\mathcal{L}(x_{\mathrm{ik}}^u, y_{\mathrm{ik}}^u; \theta)\right] \\ &\approx -\Big\{\mathbb{E}_{x_{idk}^u, x_{idk}^o \sim D_{idk}}\left[\mathcal{I}(x_{\mathrm{idk}}^o, y_{\mathrm{idk}}^o, x_{\mathrm{idk}}^u, y_{\mathrm{idk}}^u; \theta)\right] - \\ &\mathbb{E}_{x_{\mathrm{ik}}^u \sim D_{\mathrm{ik}}, x_{\mathrm{idk}}^o \sim D_{\mathrm{idk}}}\left[\mathcal{I}(x_{\mathrm{idk}}^o, y_{\mathrm{idk}}^o, x_{\mathrm{ik}}^u, y_{\mathrm{ik}}^u; \theta)\right]\Big\} \end{aligned}$$
$$(5)$$

The first expectation term in equation (5) captures the reduction in the model's error rate, while the second term reflects the occurrence of over-refusal. Training samples where the difference between these two terms is smaller tend to exacerbate over-refusal, though they may also contribute to stronger overall model performance.

5 Methodology

In this Section, we follow the two observations ($\mathbf{O}_1 \& \mathbf{O}_2$) in Section 4 and split GRAIT into three stages:

- **Stage 1: Construct $\mathbf{D}_{\mathbf{ik}}$ & $\mathbf{D}_{\mathbf{idk}}$**, which obtain $\mathbf{D}_{\mathrm{ik}}$ & $\mathbf{D}_{\mathrm{idk}}$ by querying the internal state of LLMs and modifying the label of the incorrect set to 'I don't know'.
- **Stage 2: Dataset Construction based on $\mathbf{O}_1$**, select idk samples from the first observation.
- **Stage 3: Influence-directed Refusal-aware Instruction Tuning based on $\mathbf{O}_2$**, which allocates different weight when Refusal-aware Instruction Tuning from the second observation.

Figure 3 presents a detailed overview of our proposed GRAIT framework, Algorithm 1 details the overall process with subsequent subsections detailing each component.

5.1 Stage 1: Construct $\mathbf{D}_{\mathbf{ik}}$ & $\mathbf{D}_{\mathbf{idk}}$

In the first stage, we compute the accuracy $\mathcal{C}(x)$ for each sample x. Our research focuses on two distinct tasks: Multiple-Choice Question Answering (MCQA) and Open-Ended Question Answering (OEQA). For the MCQA task, we use the token probability of the ground truth to evaluate the correctness of each sample. In the OEQA task, the model generates answers N times (with $N = 10$) for each question, and we calculate accuracy based on the generated responses.

Next, we construct the dataset $\mathbf{D}_{\mathrm{ik}}$ using samples with $\mathcal{C}(x)$ of the correctness threshold $\mathcal{T}_{\mathcal{C}}$ or higher. For samples with $\mathcal{C}(x)$ below $\mathcal{T}_{\mathcal{C}}$, we modify their output to "I don't know" and construct these samples into a separate dataset, $\mathbf{D}_{\mathrm{idk}}$.

5.2 Stage 2: Dataset Construction based on $\mathbf{O}_1$

In this module, as outlined in Algorithm1, we select the most efficient data to train the model. **First, for the idk set $\mathbf{D}_{\mathbf{idk}}$**, we identify the most impactful idk samples that significantly enhance the model's ability to refuse unknown questions based on the conclusion of $\mathbf{O}_1$. Specifically, we approximate
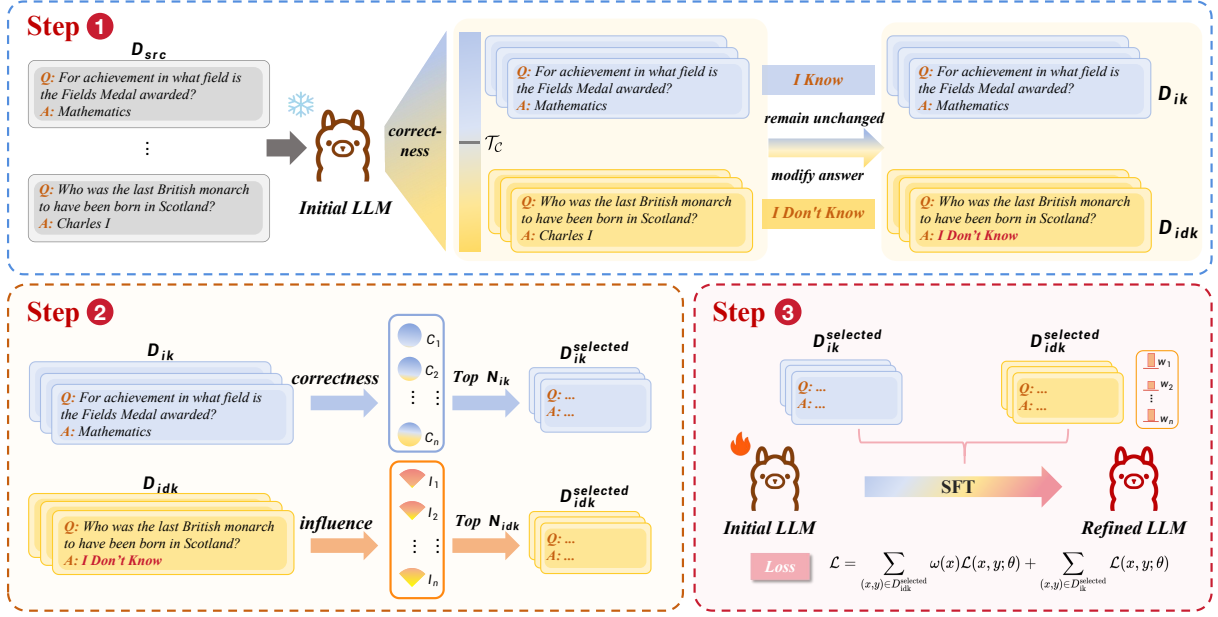
Figure 3: **Overview of our framework.** GRAIT contains three stages: **(1)** Constructing datasets $\mathbf{D}_{\text{ik}}$ and $\mathbf{D}_{\text{idk}}$ by querying the internal state of LLMs. **(2)** Distilling the datasets to select idk samples based on the first observation $\mathbf{O}_1$. **(3)** Performing Influence-directed Refusal-aware Instruction Tuning using the second observation $\mathbf{O}_2$.

(4) and define **Refusal Influence** of each training sample on the model's loss as:

$$\mathcal{I}^{\text{ref}}(x) = \left\langle \nabla\mathcal{L}(x;\theta), \mathbb{E}_{x^o \sim D_{\text{idk}}}\left[\nabla\mathcal{L}(x^o_{\text{idk}}, y^o_{\text{idk}};\theta)\right]\right\rangle, \quad (6)$$

where x denotes the training sample, $\mathcal{L}$ is the loss function, and θ represents the model parameters. This influence metric allows us to assess the contribution of each idk sample towards minimizing the incorrectness. Building upon the influence scores, we employ a ranking strategy to select the most influential idk samples for fine-tuning:

$$\mathbf{D}^{\text{selected}}_{\text{idk}} = \{x \in \mathbf{D}_{\text{idk}} \mid \mathsf{Rank}\big(\mathcal{I}^{\text{ref}}(x)\big) \le N_{\text{idk}}\}. \quad (7)$$

The selected idk samples, which exhibit the highest refusal influence, will form a distilled dataset used for targeted training. Furthermore, to ensure computational efficiency, we apply two techniques following (Xia et al., 2024) to construct valuable low-dimensional gradient features: parameter efficient fine-tuning via LoRA (Hu et al., 2021) and random projections (Johnson and Lindenstrauss, 1984).

Secondly, for the dataset $\mathbf{D}_{\text{ik}}$, we select the subset of data with the highest accuracy. Previous work (Ren et al., 2024b) has demonstrated that using this subset for fine-tuning does not negatively impact the model's overall performance. The selection of this data can be represented as:

$$\mathbf{D}^{\text{selected}}_{\text{ik}} = \{x \in \mathbf{D}_{\text{ik}} \mid \mathsf{Rank}\big(\mathcal{C}(x)\big) \le N_{\text{ik}}\}. \quad (8)$$

5.3 Stage 3: Influence-directed Refusal-aware Instruction Tuning based on $\mathbf{O}_2$

In this stage, we introduce an Influence-directed Refusal-aware Instruction Tuning based on the conclusions from O_2, which mentions that the issue of Over-Refusal is closely related to the difference in influence between training samples. The larger this difference, the more stable the accuracy of the model remains when learning the ability to reject. Therefore, we propose the concept of **Stable Influence** and assign each sample a weight ω, which is calculated as follows:

$$\mathcal{I}^{\text{sta}}(x) = \Big\langle \nabla\mathcal{L}(x;\theta), \mathbb{E}_{x^o \sim D_{\text{idk}}}\left[\nabla\mathcal{L}(x^o_{\text{idk}}, y^o_{\text{idk}};\theta)\right]$$
$$- \mathbb{E}_{x^o \sim D_{\text{ik}}}\left[\nabla\mathcal{L}(x^o_{\text{ik}}, y^o_{\text{idk}};\theta)\right]\Big\rangle, \quad (9)$$

$$\omega(x^o) = \frac{e^{\mathcal{I}^{\text{sta}}(x^o)/\tau}}{\mathbb{E}_{x^o \sim D^{\text{selected}}_{\text{idk}}}\left[e^{\mathcal{I}^{\text{sta}}(x^o)/\tau}\right]}, \quad (10)$$

where x^o represents a training sample, and τ refers to the temperature parameter. As τ approaches 0, the differences in weight distribution become more pronounced, while as τ approaches 1, the changes in weights become minimal. The constraint $\sum \omega(x^o) = 1$ ensures that the model maintains its ability to reduce error rates.

Finally, during training, we apply a weight to the loss of the idk samples to mitigate over-refusal. The SFT loss is calculated as follows:

Algorithm 1 GRAIT Process

Input: $D_{\text{src}} = \{x_0, x_1, ..., x_N\}$, $\mathcal{T}_{\mathcal{C}}$, τ N_{ik}, N_{idk}
Output: D_{rait}

1: $D_{\text{ik}} = \{x \mid x \in D_{\text{src}}, \mathcal{C}(x) \geq \mathcal{T}_{\mathcal{C}}\}$
2: $D_{\text{idk}} = \{x \mid x \in D_{\text{src}}, \mathcal{C}(x) < \mathcal{T}_{\mathcal{C}}\}$
3: $\overline{g}_{\text{idk}}(D_{\text{idk}}) = \frac{1}{|D_{\text{idk}}|} \sum_{x \in D_{\text{idk}}} x.g_{\text{idk}}$
4: $\overline{g}_{\text{idk}}(D_{\text{ik}}) = \frac{1}{|D_{\text{ik}}|} \sum_{x \in D_{\text{ik}}} x.g_{\text{idk}}$
5: **for** x_i in D_{idk} **do**
6:      $x_i.\mathcal{I}^{\text{ref}} = x_i.g_{\text{idk}} \cdot \overline{g}_{\text{idk}}(D_{\text{idk}})$
7:      $x_i.\mathcal{I}^{\text{sta}} = x_i.g_{\text{idk}} \cdot [\overline{g}_{\text{idk}}(D_{\text{idk}}) - \overline{g}_{\text{idk}}(D_{\text{ik}})]$
8: **end for**
9: $D_{\text{ik}} = \text{sort}(D_{\text{ik}}, \text{key} = \mathcal{C}, \text{order=descend})$
10: $D_{\text{idk}} = \text{sort}(D_{\text{idk}}, \text{key} = \mathcal{I}^{\text{ref}}, \text{order=descend})$
11: $D_{\text{ik}}^{\text{selected}} = \text{TopK}(D_{\text{ik}}, N_{\text{ik}})$
12: $D_{\text{idk}}^{\text{selected}} = \text{TopK}(D_{\text{idk}}, N_{\text{idk}})$
13: Initialize: $Z = 0$
14: **for** x_i in $D_{\text{idk}}^{\text{selected}}$ **do**
15:      $Z = Z + e^{x_i.\mathcal{I}^{\text{sta}}/\tau}$
16: **end for**
17: **for** x_i in $D_{\text{idk}}^{\text{selected}}$ **do**
18:      $x_i.\omega_i = \frac{e^{x_i.\mathcal{I}^{\text{sta}}/\tau}}{Z/|D_{\text{idk}}^{\text{selected}}|}$
19: **end for**
20: $D_{\text{rait}} = D_{\text{ik}} \cup D_{\text{idk}}$
21: **return** D_{rait}

$$\mathcal{L}_{SFT} = \sum_{(x^o, y^o) \in D_{\text{idk}}^{\text{selected}}} \omega(x^o)\mathcal{L}(x, y^o; \theta)$$
$$+ \sum_{(x^o, y^o) \in D_{\text{ik}}^{\text{selected}}} \mathcal{L}(x^o, y^o; \theta). \quad (11)$$

6 Experiment

In this section, we provide detailed information on experimental setup, and further analysis to validate the performance and rationality of GRAIT.

6.1 Experiment Setup

Datasets. In this study, we assess the efficacy of GRAIT in handling two distinct types of Question and Answering tasks: the knowledge-based Multiple Choice Question Answering (MCQA) and Open-ended Question Answering (OEQA). For the MCQA task, the test split of MMLU (Hendrycks et al., 2020) is adopted as the training dataset, while the validation split of the same serves as the In-Domain (ID) test set, and the ARC-c (Clark et al., 2018) test split is utilized as the Out-Of-Domain (OOD) test set. In the context of the OEQA task, we use the training split of TriviaQA (Joshi et al., 2017) for training purposes, the development split
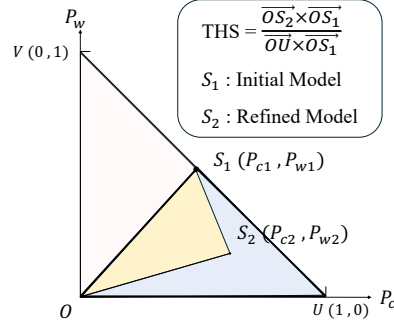

Figure 4: Illustration of Truthful Helpfulness Score.

of TriviaQA as the ID test set, and the validation split of NQ (Kwiatkowski et al., 2019) as the OOD test set. Additional information is provided in Table 1.

Table 1: Datasets Details.

	MCQA	**OEQA**
Train	MMLU test (14,079)	TriviaQA train (87,622)
ID Eval	MMLU val (1,540)	TriviaQA dev (11,313)
OOD Eval	ARC-c dev (1,172)	NQ dev (3,610)

Baselines. To evaluate the performance of GRAIT, we conducted comparisons with several existing approaches: **Init-Basic**: Employs the initial LLM setup, utilizing standard question-answering prompts to guide the model in generating answers. **Init-Refuse**: Builds on Init-Basic by incorporating instructions such as "*If you do not know the answer, please respond with 'I don't know.'*" to promote safer responses (Bianchi et al., 2024; Zhang et al., 2024b). **Van-Tuning**: Randomly selects $N_{\text{ik}} + N_{\text{idk}}$ samples from D_{src} for straightforward instruct-tuning, without any sample modification. **R-Tuning**: Follows the settings from (Zhang et al., 2024a), where samples in the RAIT dataset are modified based on the correctness of the model's replies. **CRaFT**: This method is implemented according to (Zhu et al., 2024), addressing both static and dynamic conflicts within the RAIT dataset to provide a thorough evaluation of potential issues.

Evaluation Metrics. We utilize the Truthful Helpfulness Score (THS) as detailed by (Zhu et al., 2024) to assess the performance of LLMs after RAIT. Accuracy (P_c), error rate (P_w), THS, etc. are key metrics for evaluating the performance of models after RAIT. Among these, P_c and P_w form a competing pair, where optimizing for P_c often leads to a decline in P_w. Focusing on only one of

these metrics is insufficient to evaluate the model's overall capability. Thus, a singular and comprehensive metric is required to simplify the assessment process and eliminate the complexity of balancing multiple trade-off metrics.

For each test sample, we classify the response as correct, incorrect, or refused. From these categories, we calculate the accuracy (P_c), error rate (P_w), and refusal rate (P_r). We then set up a Cartesian coordinate system with P_c and P_w on the axes. The point S_1 represents the coordinates of the baseline LLM, and S_2 corresponds to the refined model. If S_2 is positioned below the line from the origin O to S_1 (denoted as OS_1), then a larger area of the triangle $\triangle OS_1S_2$ signifies an improvement in the model. If, however, S_2 is above OS_1, it indicates a reduction in performance. As shown in Figure 4, THS is defined as the ratio of the cross product of vectors $\overrightarrow{OS_1}$ and $\overrightarrow{OS_2}$ to the maximum possible value of this cross product:

$$\text{THS} = (\overrightarrow{OS_2} \times \overrightarrow{OS_1})/(\overrightarrow{OU} \times \overrightarrow{OS_1}). \qquad (12)$$

Implementation Details. In our studies, we utilized LLaMA2-7B-Chat and LLaMA3-8B-Instruct as the initial LLMs θ_0. For the MCQA task, we selected 5,000 samples from the MMLU dataset for training purposes, and for the OEQA task, 10,000 samples from TriviaQA were used. With the exception of the Van-Tuning setting, where all samples were kept unchanged, other RAIT settings used a 1:4 ratio of `ik` samples to `idk` samples. In the MCQA and OEQA tasks, correctness is obtained using 5-shot and 3-shot setups[2], respectively. More implementation details are listed in Appendix A.4. In contrast to (Zhu et al., 2024), to ensure the fairness of the experiments, we employ LoRA for training across both MCQA and OEQA tasks.

During both training and testing phases, XTuner [3] was employed for RAIT experiments, which were conducted over 3 epochs with a maximum context length set to 2048. The LoRA (Hu et al.) was implemented with the parameters: $r = 64$, $\alpha = 16$, dropout rate of 0.1, and a learning rate of 2×10^{-4}. For evaluations, the 0-shot approach with greedy decoding was adopted. OpenCompass [4] is used for all evaluations and correctness calculations. In the GRAIT method, we as-

signed the hyperparameter $\mathcal{T}_C$ a value of 0.5 and τ a value of 0.05. All experiments were executed on eight NVIDIA A100-80GB GPUs.

6.2 Experiment Results

We present the main experimental results, along with an ablation study of GRAIT across various models in Table 2. A summary of the key findings is provided below.

6.2.1 Main Results

We assess the effectiveness of GRAIT by addressing the challenges *C1* and *C2*, with the corresponding experimental results presented in Table 2.

Comparison based on *C1*: *C1* relates to the metric P_w, where a lower P_w indicates better avoidance of hallucinations by the model. As shown in the results, our proposed GRAIT achieves a significantly lower P_w compared to other baselines, demonstrating its effectiveness in reducing hallucination rates.

Comparison based on *C2*: *C2* focuses on minimizing hallucinations while maintaining accuracy, addressing the challenge of over-refusal. GRAIT surpasses existing methods in THS score with an average of 3.66. Specifically, the THS results clearly show that our method significantly outperforms other baselines on both in-domain (ID) and out-of-domain (OOD) settings. For instance, on MMLU dataset, the LLaMA2-7B-Chat model achieves a THS score of 19.3, whereas the best-performing baseline, CRaFT, only reaches 12.5. Moreover, our approach consistently demonstrates superior performance on OOD datasets as well.

6.2.2 Ablation Study

We conduct ablation studies to evaluate the contribution of each component in GRAIT, as presented in Table 2, using two variants: (1) GRAIT without Refusal Influence, which follows the R-Tuning approach during the dataset distillation phase (denoted as w/o O_1), and (2) GRAIT without Stable Influence, where no weight adjustment is applied to emphasize the importance of `idk` samples (denoted as w/o O_2). The results indicate that each component contributes positively to the overall performance of GRAIT and the removal of any component leads to a noticeable decline in effectiveness. Specifically, replacing Refusal Influence-based dataset distillation with other baselines results in a significant increase in hallucination rate, underscoring the importance of Refusal Influence in addressing *C1*. Additionally, the use of Stable

---

Table 2: Performance comparisons on MMLU, ARC-c, TriviaQA and NQ. The best performance is highlighted in **boldface**, while the second-best performance is underlined.

| LLMs | QA Type | MCQA | | | | | | OEQA | | | | | |
	Dataset	MMLU (ID)			ARC-c (OOD)			TriviaQA (ID)			NQ (OOD)		
	Metric	P_c	P_w ↓	THS↑	P_c	P_w ↓	THS↑	P_c	P_w ↓	THS↑	P_c	P_w ↓	THS↑
Llama2-7B Chat	Init-Basic	45.6	52.8	00.0	53.9	46.0	00.0	54.0	46.0	00.0	28.9	71.1	00.0
	Init-Refuse	36.4	38.9	03.9	44.4	35.7	02.6	37.1	21.7	11.5	19.8	**34.8**	**05.6**
Baselines	Van-Tuning	46.9	53.0	01.2	54.5	45.5	01.2	55.5	44.5	03.2	23.2	76.8	-0.80
	R-Tuning	44.5	39.6	11.3	55.8	38.1	11.1	52.2	35.9	10.0	22.6	60.9	-0.22
	CRaFT	43.9	36.4	12.5	54.7	35.9	12.6	47.8	28.1	14.8	26.7	62.0	01.5
Ours	GRAIT	43.5	27.1	**20.1**	55.2	**26.5**	**24.2**	43.6	18.4	**22.0**	20.8	49.7	00.0
Ablations	w/o O$_1$	44.7	39.8	10.3	55.4	37.9	11.0	52.4	36.5	09.6	23.9	63.5	-01.9
	w/o O$_2$	42.8	**26.5**	20.0	54.1	26.7	22.8	41.9	18.1	20.6	20.1	48.3	00.5
Llama3-8B Instruct	Init-Basic	66.8	33.1	00.0	80.6	19.5	00.0	66.8	33.2	00.0	40.3	59.7	00.0
	Init-Refuse	50.0	17.0	15.7	65.3	14.4	05.6	53.9	20.8	12.0	31.1	**38.6**	**05.0**
Baselines	Van-Tuning	69.5	30.5	08.0	80.3	19.7	-01.3	60.0	40.0	-19.0	21.0	48.5	-11.7
	R-Tuning	63.9	21.6	20.4	79.4	16.2	12.2	56.6	28.3	-00.5	25.1	74.9	-25.6
	CRaFT	53.3	09.6	34.0	74.1	12.7	21.4	57.8	27.7	02.0	27.0	57.6	-12.0
Ours	GRAIT	50.4	**06.9**	**36.4**	70.2	08.7	**34.3**	55.3	18.3	18.5	21.9	38.8	-04.4
Ablations	w/o O$_1$	64.1	21.4	20.9	79.3	16.4	11.5	57.5	28.7	-00.2	25.6	75.0	-25.0
	w/o O$_2$	49.6	07.0	35.5	69.1	08.6	33.6	54.3	18.3	17.4	21.6	39.1	-04.8
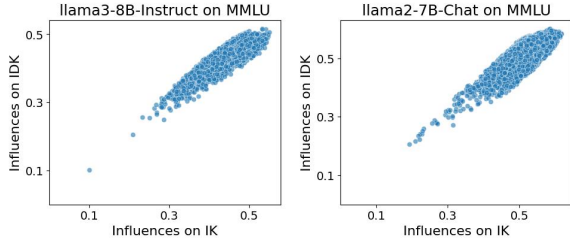
Figure 5: Relationship between $\mathcal{I}^{\text{ref}}$ and $\mathcal{I}^{\text{over}}$ in MMLU performance on LLaMA2-7B-Chat and LLaMA3-8B-Instruct.

Table 3: Performance comparisons on MMLU and ARC-c for different `ik` selection methods on LLaMA3-8B-Instruct.

Dataset	MMLU (ID)			ARC-c (OOD)		
Metric	P_c	P_w ↓	THS↑	P_c	P_w ↓	THS↑
`ik-top`	50.4	06.9	36.4	70.2	08.7	34.3
`ik-random`	61.4	20.5	20.0	78.7	15.6	14.2
`ik-bottom`	64.0	25.3	12.9	79.7	19.0	-00.2

Influence helps reduce over-refusal while maintaining a stable hallucination rate, effectively addressing the challenges posed by **C2**. In addition, we conducted sensitivity experiments, the details of which can be found in the appendix A.6.

6.3 Analysis

The selection of `ik` samples is crucial. Our analysis and experiments primarily focus on optimizing the selection of `idk` samples. However, the selection of `ik` samples is also crucial. We employed three different strategies: `ik-random`, where data is randomly selected from D_{ik}; `ik-bottom`, where the data with the lowest correctness from D_{ik} is selected; and `ik-top`, the method used in GRAIT, where the data with the highest correctness from D_{ik} is chosen. We used the MMLU (ID) and ARC-c (OOD) datasets and conducted experiments with the LLaMA3-8B-Instruct model. The results are shown in Table 3. When using either the

`ik-bottom` or `ik-random` methods, the model's hallucination reduction does not improve, and the refusal rate remains low. We believe the potential reason for this is that the `ik` samples selected by these methods may share similar characteristics with the `idk` samples, but different supervision signals were applied during the SFT process. This weakens the model's ability to learn effective refusals. In contrast, the `ik-top` strategy used in GRAIT helps to distinctly separate the features of the two types of samples, addressing the static conflict mentioned in (Zhu et al., 2024).

Over-Refusal can only be alleviated, but not completely eliminated. During the RAIT process, we observed and analyzed the `idk` influence (corresponding to O_2) of `idk` samples on D_{ik} and D_{idk} using the LLaMA2-7B-Chat and LLaMA3-8B-Instruct model on the MMLU dataset. As shown in Figure 5, we identified a strong correlation between the two, with a Pearson Correlation Coefficient of 0.886. This correlation may be a contributing factor to the occurrence of Over-Refusal. While our proposed method, as indicated in Table 2, cannot fully eliminate Over-Refusal due to certain

limitations, it significantly mitigates the issue.

7 Conclusion and Future Work

In this paper, we present GRAIT, a Gradient-based Refusal-Aware Instruction Tuning Framework, which addresses the critical challenge of over-refusal in existing RAIT approaches. By leveraging insights derived from a gradient perspective, GRAIT effectively distills refusal-aware datasets and incorporates an adaptive weighting mechanism during fine-tuning. Our experimental results demonstrate that GRAIT not only mitigates hallucinations but also enhances the reliability and accuracy of LLM outputs. Looking ahead, we aim to further investigate two key avenues of research. First, we plan to explore the dynamic trajectory influence of gradients throughout the RAIT process, which could provide deeper insights into how various training samples impact LLM refusal behavior. Second, we intend to examine the role of GRAIT in detecting knowledge boundaries within LLMs, focusing on its potential contributions to enhancing LLM safety.

Limitations

While our work has yielded promising results, it is important to recognize several limitations. First, the GRAIT framework currently treats the training process as static, rather than incorporating the dynamic influence of gradient trajectories throughout the RAIT process. Additionally, the `idk` and `ik` sets are divided through a straightforward query of the LLMs; future work could explore ways to leverage GRAIT for more nuanced identification of knowledge boundaries within LLMs for splitting. Finally, although GRAIT has demonstrated strong generalizability across various evaluation datasets, expanding the dataset range to include a more diverse set of high-quality resources could enhance the robustness and versatility of the framework.

Acknowledgments

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Fengshuo Bai, Runze Liu, Yali Du, Ying Wen, and Yaodong Yang. 2024a. Rat: Adversarial attacks on deep reinforcement agents for targeted behaviors. *arXiv preprint arXiv:2412.10713*.

Fengshuo Bai, Mingzhi Wang, Zhaowei Zhang, Boyuan Chen, Yinda Xu, Ying Wen, and Yaodong Yang. 2024b. Efficient model-agnostic alignment via bayesian persuasion. *arXiv preprint arXiv:2405.18718*.

Fengshuo Bai, Hongming Zhang, Tianyang Tao, Zhiheng Wu, Yanna Wang, and Bo Xu. 2023. Picor: Multi-task deep reinforcement learning with policy correction. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(6):6728–6736.

Federico Bianchi, Mirac Suzgun, Giuseppe Attanasio, Paul Röttger, Dan Jurafsky, Tatsunori Hashimoto, and James Zou. 2024. Safety-tuned llamas: Lessons from improving the safety of large language models that follow instructions. *Preprint*, arXiv:2309.07875.

Lida Chen, Zujie Liang, Xintao Wang, Jiaqing Liang, Yanghua Xiao, Feng Wei, Jinglei Chen, Zhenghong Hao, Bing Han, and Wei Wang. 2024. Teaching large language models to express knowledge boundary from their own signals. *Preprint*, arXiv:2406.10881.

Qinyuan Cheng, Tianxiang Sun, Xiangyang Liu, Wenwei Zhang, Zhangyue Yin, Shimin Li, Linyang Li, Kai Chen, and Xipeng Qiu. 2024. Can ai assistants know what they don't know? *arXiv preprint arXiv:2401.13275*.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. 2022. Knowledge neurons in pretrained transformers. *Preprint*, arXiv:2104.08696.

Guanting Dong, Hongyi Yuan, Keming Lu, Chengpeng Li, Mingfeng Xue, Dayiheng Liu, Wei Wang, Zheng Yuan, Chang Zhou, and Jingren Zhou. 2024. How abilities in large language models are affected by supervised fine-tuning data composition. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 177–198, Bangkok, Thailand. Association for Computational Linguistics.

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

Shangbin Feng, Weijia Shi, Yike Wang, Wenxuan Ding, Vidhisha Balachandran, and Yulia Tsvetkov. 2024. Don't hallucinate, abstain: Identifying llm knowledge gaps via multi-llm collaboration. *Preprint*, arXiv:2402.00367.

Rhys Gould, Euan Ong, George Ogden, and Arthur Conmy. 2023. Successor heads: Recurring, interpretable attention heads in the wild. *Preprint*, arXiv:2312.09230.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.

E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen. 2021. LoRA: Low-rank adaptation of large language models.

Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38.

Xinke Jiang, Yue Fang, Rihong Qiu, Haoyu Zhang, Yongxin Xu, Hao Chen, Wentao Zhang, Ruizhe Zhang, Yuchen Fang, Xu Chu, et al. 2024a. Tc-rag: Turing-complete rag's case study on medical llm systems. *arXiv preprint arXiv:2408.09199*.

Xinke Jiang, Ruizhe Zhang, Yongxin Xu, Rihong Qiu, Yue Fang, Zhiyuan Wang, Jinyi Tang, Hongxin Ding, Xu Chu, Junfeng Zhao, et al. 2024b. Hykge: A hypothesis knowledge graph enhanced framework for accurate and reliable medical llms responses. *arXiv preprint arXiv:2312.15883*.

William B. Johnson and Joram Lindenstrauss. 1984. Extensions of lipschitz mappings into a hilbert space.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. 2017. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. *arXiv preprint arXiv:1705.03551*.

Katie Kang, Eric Wallace, Claire Tomlin, Aviral Kumar, and Sergey Levine. 2024. Unfamiliar finetuning examples control how language models hallucinate. *arXiv preprint arXiv:2403.05612*.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *Preprint*, arXiv:2001.08361.

Krishnateja Killamsetty and Rishabh Iyer. 2021. Subset selection with gradient matching. *arXiv preprint arXiv:2103.00123*.

Krishnateja Killamsetty, Durga Sivasubramanian, Ganesh Ramakrishnan, Abir De, and Rishabh Iyer. 2021. Grad-match: Gradient matching based data subset selection for efficient deep model training. In

Proceedings of the 38th International Conference on Machine Learning (ICML).

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466.

Siheng Li, Cheng Yang, Taiqiang Wu, Chufan Shi, Yuji Zhang, Xinyu Zhu, Zesen Cheng, Deng Cai, Mo Yu, Lemao Liu, et al. 2024. A survey on the honesty of large language models. *arXiv preprint arXiv:2409.18786*.

Jinlong Liu, Guoqing Jiang, Yunzhi Bai, Ting Chen, and Huayan Wang. 2020. Understanding why neural networks generalize well through gsnr of parameters. *Learning,Learning*.

Jiayuan Luo, Songhua Yang, Xiaoling Qiu, Panyu Chen, Yufei Nai, Wenxuan Zeng, Wentao Zhang, and Xinke Jiang. 2024. Kuaiji: the first chinese accounting large language model. *arXiv preprint arXiv:2402.13866*.

OpenAI. 2022. Introducing chatgpt. https://openai.com/blog/chatgpt.

OpenAI. 2023. Gpt-4 technical report. *ArXiv*, abs/2303.08774.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback.

Garima Pruthi, Frederick Liu, Satyen Kale, and Mukund Sundararajan. 2020. Estimating training data influence by tracing gradient descent. *Neural Information Processing Systems,Neural Information Processing Systems*.

Jingyang Qiao, Zhizhong Zhang, Xin Tan, Chengwei Chen, Yanyun Qu, Yong Peng, and Yuan Xie. 2024. Prompt gradient projection for continual learning. In *International Conference on Learning Representations*.

Jie Ren, Qipeng Guo, Hang Yan, Dongrui Liu, Quanshi Zhang, Xipeng Qiu, and Dahua Lin. 2024a. Identifying semantic induction heads to understand in-context learning. *Preprint*, arXiv:2402.13055.

Mengjie Ren, Boxi Cao, Hongyu Lin, Liu Cao, Xianpei Han, Ke Zeng, Guanglu Wan, Xunliang Cai, and Le Sun. 2024b. Learning or self-aligning? rethinking instruction fine-tuning. *arXiv preprint arXiv:2402.18243*.

Yi Ren and Danica J. Sutherland. 2024. Learning dynamics of llm finetuning. *arXiv preprint arXiv:2407.10490*.

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

Neeraj Varshney, Pavel Dolin, Agastya Seth, and Chitta Baral. 2023. The art of defending: A systematic evaluation and analysis of llm defense strategies on safety and over-defensiveness. *arXiv preprint arXiv:2401.00287*.

Minh Duc Vu, Han Wang, Zhuang Li, Jieshan Chen, Shengdong Zhao, Zhenchang Xing, and Chunyang Chen. 2024. Gptvoicetasker: Llm-powered virtual assistant for smartphone. *Preprint*, arXiv:2401.14268.

Fanqi Wan, Xinting Huang, Leyang Cui, Xiaojun Quan, Wei Bi, and Shuming Shi. 2024. Mitigating hallucinations of large language models via knowledge consistent alignment. *arXiv preprint arXiv:2401.10768*.

Yikun Wang, Rui Zheng, Liang Ding, Qi Zhang, Dahua Lin, and Dacheng Tao. 2024. Uncertainty aware learning for language model alignment. *Preprint*, arXiv:2406.04854.

Bingbing Wen, Jihan Yao, Shangbin Feng, Chenjun Xu, Yulia Tsvetkov, Bill Howe, and Lucy Lu Wang. Know your limits: A survey of abstention in large language models.

Chris Wendler, Veniamin Veselovsky, Giovanni Monea, and Robert West. 2024. Do llamas work in english? on the latent language of multilingual transformers. *Preprint*, arXiv:2402.10588.

Robert Wolfe, Isaac Slaughter, Bin Han, Bingbing Wen, Yiwei Yang, Lucas Rosenblatt, Bernease Herman, Eva Brown, Zening Qu, Nic Weber, and Bill Howe. 2024. Laboratory-scale ai: Open-weight models are competitive with chatgpt even in low-resource settings. In *The 2024 ACM Conference on Fairness, Accountability, and Transparency*, volume 35 of *FAccT '24*, page 1199–1210. ACM.

Mengzhou Xia, Sadhika Malladi, Suchin Gururangan, Sanjeev Arora, and Danqi Chen. 2024. Less: Selecting influential data for targeted instruction tuning. *arXiv preprint arXiv:2402.04333*.

Yang Xiao, Zijie Zhang, Yuchen Fang, Da Yan, Yang Zhou, Wei-Shinn Ku, and Bo Hui. 2024. Advancing certified robustness of explanation via gradient quantization. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management (CIKM 2024)*. (Accepted).

Hongshen Xu, Zichen Zhu, Da Ma, Situo Zhang, Shuai Fan, Lu Chen, and Kai Yu. 2024a. Rejection improves reliability: Training llms to refuse unknown

questions using rl from knowledge feedback. *arXiv preprint arXiv:2403.18349*.

Yongxin Xu, Ruizhe Zhang, Xinke Jiang, Yujie Feng, Yuzhen Xiao, Xinyu Ma, Runchuan Zhu, Xu Chu, Junfeng Zhao, and Yasha Wang. 2024b. Parenting: Optimizing knowledge selection of retrieval-augmented language models with parameter decoupling and tailored tuning. *arXiv preprint arXiv:2410.10360*.

Yu Yang, Siddhartha Mishra, Jeffrey N Chiang, and Baharan Mirzasoleiman. 2024. Smalltolarge (s2l): Scalable data selection for fine-tuning large language models by summarizing training trajectories of small models. *Preprint*, arXiv:2403.07384.

Yuqing Yang, Ethan Chern, Xipeng Qiu, Graham Neubig, and Pengfei Liu. 2023. Alignment for honesty. *arXiv preprint arXiv:2312.07000*.

Qingchen Yu, Zifan Zheng, Shichao Song, Zhiyu Li, Feiyu Xiong, Bo Tang, and Ding Chen. 2024. xfinder: Robust and pinpoint answer extraction for large language models. *Preprint*, arXiv:2405.11874.

Zeping Yu and Sophia Ananiadou. 2024. Neuron-level knowledge attribution in large language models. *Preprint*, arXiv:2312.12141.

Hanning Zhang, Shizhe Diao, Yong Lin, Yi Fung, Qing Lian, Xingyao Wang, Yangyi Chen, Heng Ji, and Tong Zhang. 2024a. R-tuning: Instructing large language models to say 'i don't know'. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 7106–7132.

Zhaowei Zhang, Fengshuo Bai, Qizhi Chen, Chengdong Ma, Mingzhi Wang, Haoran Sun, Zilong Zheng, and Yaodong Yang. 2025. Amulet: Realignment during test time for personalized preference adaptation of LLMs. In *International Conference on Learning Representations (ICLR)*.

Zhexin Zhang, Junxiao Yang, Pei Ke, Fei Mi, Hongning Wang, and Minlie Huang. 2024b. Defending large language models against jailbreaking attacks through goal prioritization. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8865–8887, Bangkok, Thailand. Association for Computational Linguistics.

Bo Zhao and Hakan Bilen. 2020. Dataset condensation. *arXiv preprint arXiv:2006.05929*.

Bo Zhao, Konda Reddy Mopuri, and Hakan Bilen. 2021. Dataset condensation with gradient matching. In *International Conference on Learning Representations (ICLR)*.

Runchuan Zhu, Zhipeng Ma, Jiang Wu, Junyuan Gao, Jiaqi Wang, Dahua Lin, and Conghui He. 2024. Utilize the flow before stepping into the

same river twice: Certainty represented knowledge flow for refusal-aware instruction tuning. *Preprint*, arXiv:2410.06913.

A   Appendix

A.1   Theoretical Analysis Details

A.1.1   More Proof on O_1

$$\mathbb{E}_{x_{\mathrm{idk}}^u \sim D_{\mathrm{idk}}} \left[\Delta\mathcal{L}(x_{\mathrm{idk}}^u, y_{\mathrm{idk}}^u; \theta) \right]$$

$$\equiv -\eta_t \left\langle \mathbb{E}_{x_{\mathrm{idk}}^u \sim D_{\mathrm{idk}}} \left[\nabla\mathcal{L}(x_{\mathrm{idk}}^u, y_{\mathrm{idk}}^u; \theta) \right], \right.$$

$$\left. \mathbb{E}_{x^o \sim D} \left[\nabla\mathcal{L}(x^o, y^o; \theta) \right] \right\rangle + \mathcal{O}(\eta^2)$$

$$= -\eta_t \left\langle \mathbb{E}_{x_{\mathrm{idk}}^u \sim D_{\mathrm{idk}}} \left[\nabla\mathcal{L}(x_{\mathrm{idk}}^u, y_{\mathrm{idk}}^u; \theta) \right], \right.$$

$$\left(\mathbb{E}_{x_{\mathrm{ik}}^o \sim D_{\mathrm{ik}}} \left[\nabla\mathcal{L}(x_{\mathrm{ik}}^o, y_{\mathrm{ik}}^o; \theta) \right] + \right.$$

$$\left. \left. \mathbb{E}_{x_{\mathrm{idk}}^o \sim D_{\mathrm{idk}}} \left[\nabla\mathcal{L}(x_{\mathrm{idk}}^o, y_{\mathrm{idk}}^o; \theta) \right] \right) \right\rangle + \mathcal{O}(\eta^2) \tag{13}$$

$$\approx -\eta_t \mathbb{E}_{x_{\mathrm{idk}}^u \sim D_{\mathrm{idk}}} \left[\nabla\mathcal{L}(x_{\mathrm{idk}}^u, y_{\mathrm{idk}}^u; \theta) \right] \cdot$$

$$\mathbb{E}_{x_{\mathrm{idk}}^o \sim D_{\mathrm{idk}}} \left[\nabla\mathcal{L}(x_{\mathrm{idk}}^o, y_{\mathrm{idk}}^o; \theta) \right]$$

$$= -\eta_t \mathbb{E}_{(x_{\mathrm{idk}}^u, x_{\mathrm{idk}}^o) \sim D_{\mathrm{idk}}} \left[\nabla\mathcal{L}(x_{\mathrm{idk}}^u, y_{\mathrm{idk}}^u; \theta) \cdot \right.$$

$$\left. \nabla\mathcal{L}(x_{\mathrm{idk}}^o, y_{\mathrm{idk}}^o; \theta) \right] .$$

$$= -\mathbb{E}_{(x_{idk}^u, x_{idk}^o) \sim D_{idk}} \left[\mathcal{I}(x_{idk}^o, y_{idk}^o, x_{idk}^u, y_{idk}^u; \theta) \right]$$

A.1.2   More Proof on O_2

RAIT can lead to the phenomenon of over-refusal, where the model refuses to answer questions it is inherently capable of addressing, thereby resulting in a decrease in accuracy. Accordingly, for an unlabeled input query x_{ik}^u, the output responses will shift from y_{ik} to y_{idk}. Assuming the use of a symmetric loss function, the difference in the loss function values for the same input x_{ik} with target labels y_{ik} and y_{idk} is approximately opposite in sign: $\Delta L(x_{ik}, y_{ik}) \approx -\Delta L(x_{ik}, y_{idk})$. Therefore,

the proof is as follows:

$$\mathbb{E}_{x_{\mathrm{idk}}^u \sim D_{\mathrm{idk}}} \left[\Delta\mathcal{L}(x_{\mathrm{idk}}^u, y_{\mathrm{idk}}^u; \theta) \right] + \mathbb{E}_{x_{\mathrm{ik}}^u \sim D_{\mathrm{ik}}} \left[\Delta\mathcal{L}(x_{\mathrm{ik}}^u, y_{\mathrm{ik}}^u; \theta) \right]$$

$$\approx \mathbb{E}_{x_{\mathrm{idk}}^u \sim D_{\mathrm{idk}}} \left[\Delta\mathcal{L}(x_{\mathrm{idk}}^u, y_{\mathrm{idk}}^u; \theta) \right] - \mathbb{E}_{x_{\mathrm{ik}}^u \sim D_{\mathrm{ik}}} \left[\Delta\mathcal{L}(x_{\mathrm{idk}}^u, y_{\mathrm{idk}}^u; \theta) \right]$$

$$\equiv -\eta_t \left\langle \mathbb{E}_{x_{\mathrm{idk}}^u \sim D_{\mathrm{idk}}} \left[\nabla\mathcal{L}(x_{\mathrm{idk}}^u, y_{\mathrm{idk}}^u; \theta) \right], \right.$$

$$\mathbb{E}_{x^o \sim D} \left[\nabla\mathcal{L}(x^o, y^o; \theta) \right] \right\rangle + \eta_t \left\langle \mathbb{E}_{x_{\mathrm{ik}}^u \sim D_{\mathrm{ik}}} \left[\nabla\mathcal{L}(x_{\mathrm{ik}}^u, y_{\mathrm{ik}}^u; \theta) \right], \right.$$

$$\left. \mathbb{E}_{x^o \sim D} \left[\nabla\mathcal{L}(x^o, y^o; \theta) \right] \right\rangle + \mathcal{O}(\eta^2)$$

$$= -\eta_t \left\langle \left\{ \mathbb{E}_{x_{\mathrm{idk}}^u \sim D_{\mathrm{idk}}} \left[\nabla\mathcal{L}(x_{\mathrm{idk}}^u, y_{\mathrm{idk}}^u; \theta) \right] \right. \right.$$

$$\left. - \mathbb{E}_{x_{\mathrm{ik}}^u \sim D_{\mathrm{ik}}} \left[\nabla\mathcal{L}(x_{\mathrm{ik}}^u, y_{\mathrm{ik}}^u; \theta) \right] \right\},$$

$$\left\{ \mathbb{E}_{x_{\mathrm{idk}}^o \sim D_{\mathrm{idk}}} \left[\nabla\mathcal{L}(x_{\mathrm{idk}}^o, y_{\mathrm{idk}}^o; \theta) \right] + \mathbb{E}_{x_{\mathrm{ik}}^o \sim D_{\mathrm{ik}}} \left[\nabla\mathcal{L}(x_{\mathrm{ik}}^o, y_{\mathrm{ik}}^o; \theta) \right] \right\} \right\rangle$$

$$+ \mathcal{O}(\eta^2)$$

$$\approx -\eta_t \left\langle \left\{ \mathbb{E}_{x_{\mathrm{idk}}^u \sim D_{\mathrm{idk}}} \left[\nabla\mathcal{L}(x_{\mathrm{idk}}^u, y_{\mathrm{idk}}^u; \theta) \right] \right. \right.$$

$$\left. -\mathbb{E}_{x_{\mathrm{ik}}^u \sim D_{\mathrm{ik}}} \left[\nabla\mathcal{L}(x_{\mathrm{ik}}^u, y_{\mathrm{ik}}^u; \theta) \right] \right\}, \mathbb{E}_{x_{\mathrm{idk}}^o \sim D_{\mathrm{idk}}} \left[\nabla\mathcal{L}(x_{\mathrm{idk}}^o, y_{\mathrm{idk}}^o; \theta) \right] \right\rangle$$

$$= -\eta_t \left\{ \mathbb{E}_{(x_{\mathrm{idk}}^u, x_{\mathrm{idk}}^o) \sim D_{\mathrm{idk}}} \left[\nabla\mathcal{L}(x_{\mathrm{idk}}^u, y_{\mathrm{idk}}^u; \theta) \cdot \nabla\mathcal{L}(x_{\mathrm{idk}}^o, y_{\mathrm{idk}}^o; \theta) \right] \right.$$

$$\left. - \mathbb{E}_{x_{\mathrm{ik}}^u \sim D_{\mathrm{ik}}, x_{\mathrm{idk}}^o \sim D_{\mathrm{idk}}} \left[\nabla\mathcal{L}(x_{\mathrm{ik}}^u, y_{\mathrm{ik}}^u; \theta) \cdot \nabla\mathcal{L}(x_{\mathrm{idk}}^o, y_{\mathrm{idk}}^o; \theta) \right] \right\}$$

$$= -\left\{ \mathbb{E}_{x_{\mathrm{idk}}^u, x_{\mathrm{idk}}^o \sim D_{\mathrm{idk}}} \left[\mathcal{I}(x_{\mathrm{idk}}^o, y_{\mathrm{idk}}^o, x_{\mathrm{idk}}^u, y_{\mathrm{idk}}^u; \theta) \right] \right.$$

$$\left. - \mathbb{E}_{x_{\mathrm{ik}}^u \sim D_{\mathrm{ik}}, x_{\mathrm{idk}}^o \sim D_{\mathrm{idk}}} \left[\mathcal{I}(x_{\mathrm{idk}}^o, y_{\mathrm{idk}}^o, x_{\mathrm{ik}}^u, y_{\mathrm{ik}}^u; \theta) \right] \right\} \tag{14}$$

Here, the first approximation is transformed as follows:

- RAIT can lead to the phenomenon of over-refusal, where the model refuses to answer questions it is inherently capable of addressing, thereby resulting in a decrease in accuracy. Accordingly, for an unlabeled input query x_{ik}^u, the output responses will shift from y_{ik} to y_{idk}.

- Assuming the use of a symmetric loss function, the difference in the loss function values for the same input x_{ik} with target labels y_{ik} and y_{idk} is approximately opposite in sign:

$$\Delta\mathcal{L}(x_{ik}, y_{ik}) \approx -\Delta\mathcal{L}(x_{ik}, y_{idk}).$$

A.2   Reasons for Choosing Influence Formula

The reasons for choosing the influence formula of Equations (4) and (5) instead of the optimizing of Equation (2):

- The left-hand term, $\Delta\mathcal{L}$, in Equations (4) and (5) represents our optimization objective on the test set. Our goal is to identify the training samples that minimize this $\Delta\mathcal{L}$.

- **Computational complexity without approximation is** $O(m \cdot n)$: Directly computing the contribution of each training sample to $\Delta\mathcal{L}$ is computationally expensive. Assuming the training

set size is m and the test set size is n, we need to compute the loss change for **each test sample** after training on **each individual training sample**, with time complexity of $O(m \cdot n)$.

- **Computational complexity is reduced to $O(m + n)$ after approximation**: Inspired by (Pruthi et al., 2020; Xia et al., 2024), we approximate $\Delta\mathcal{L}$ using the influence function. Using the influence function to approximate $\Delta\mathcal{L}$ only requires computing the gradients of each training sample and test sample without repeated calculations. Its time complexity is $O(m+n)$, which significantly reduces computational overhead. This approximation arises from omitting the higher-order terms in the Taylor expansion during the derivation of the influence function.

A.3 Orthogonal Experiment

In our experiments, we observed that the gradient distributions for idk and ik are nearly orthogonal, as illustrated in Figure 6. Through experiments conducted on the Llama2-7B-Chat model and the MMLU dataset, we found that the inner product distribution between idk gradients and ik gradients is centered around zero, with the computed value being $\left\langle \mathbb{E}_x\left[\nabla\mathcal{L}(x, y_{\text{idk}}; \theta)\right], \mathbb{E}_x\left[\nabla\mathcal{L}(x, y_{\text{ik}}; \theta)\right]\right\rangle = 0.008$. In contrast, the inner product of *ik* with itself is significantly larger, $\left\langle \mathbb{E}_x\left[\nabla\mathcal{L}(x, y_{\text{ik}}; \theta)\right], \mathbb{E}_x\left[\nabla\mathcal{L}(x, y_{\text{ik}}; \theta)\right]\right\rangle = 0.103$, and the inner product of idk with itself is even greater, $\left\langle \mathbb{E}_x\left[\nabla\mathcal{L}(x, y_{\text{idk}}; \theta)\right], \mathbb{E}_x\left[\nabla\mathcal{L}(x, y_{\text{idk}}; \theta)\right]\right\rangle = 0.513$.

Our explanation for this phenomenon includes the following points:

- ik samples and idk samples train different capabilities of the LLM: From the perspective of the internal knowledge of the LLM, ik samples help the LLM "**learn knowledge**", while idk samples grant the LLM the ability to "**reflect on self-knowledge**", that is, to predict the boundaries of its own knowledge.

- **The different capabilities of the LLM are usually associated with different regions or activation patterns of the transformer**: For example, (Dai et al., 2022; Yu and Ananiadou, 2024) indicates that knowledge is primarily stored in specific neurons of the LLM, while some studies show that different attention heads perform different functions, such as the Successor Head (Gould et al., 2023) and the Induction Head (Ren et al., 2024a). Additionally, the multilingual capabilities of the LLM mainly depend on the layers near the input and output ends of the transformer, rather than the middle layers (Wendler et al., 2024).

- The abilities to "learn knowledge" (ik samples) and "reflect on self-knowledge" (idk samples) correspond to different regions or activation patterns of the LLM transformer. Therefore, during the training process, the gradients of ik samples and idk samples act on different regions or activation patterns of the transformer respectively. This helps explain why the gradient distributions of ik samples and idk samples exhibit a state of near orthogonality.

A.4 Prompts in GRAIT

A.4.1 Prompts for Getting Correctness.

Prompts for getting correctness on MMLU and TriviaQA datasets are shown in Table 4 and Table 5. They use 5-shot and 3-shot settings respectively.

A.4.2 Prompts for training.

For the *Van-Tuning*, we use the *basic* prompt as shown in Table 4 and Table 5 without in-context example. All other experiments use the *refuse* prompt as shown in Table 6 and Table 7. Loss is only computed on the target answer {answer$_{\text{rait}}$}.

A.4.3 Prompts for evaluation.

The *Init-Basic* method uses the original question format for evaluation, without any prior instructions. For the other methods, the evaluation prompts are shown in Tables 8 and 9.

A.5 Reasons for Choosing Few-Shot Setting

We use few-shot prompting to calculate the correctness of LLMs, aiming to ensure that **LLMs strictly follow instructions to achieve the following objectives**:

- **Easy-to-parse response**: We extract answers from the response using rules and then compare them with the gt to calculate correctness. Both research (Yu et al., 2024) and our practice show that accurately extracting answers is challenging. Therefore, we provide examples of easy-to-parse answers through few-shot examples to ensure compliance with instructions.
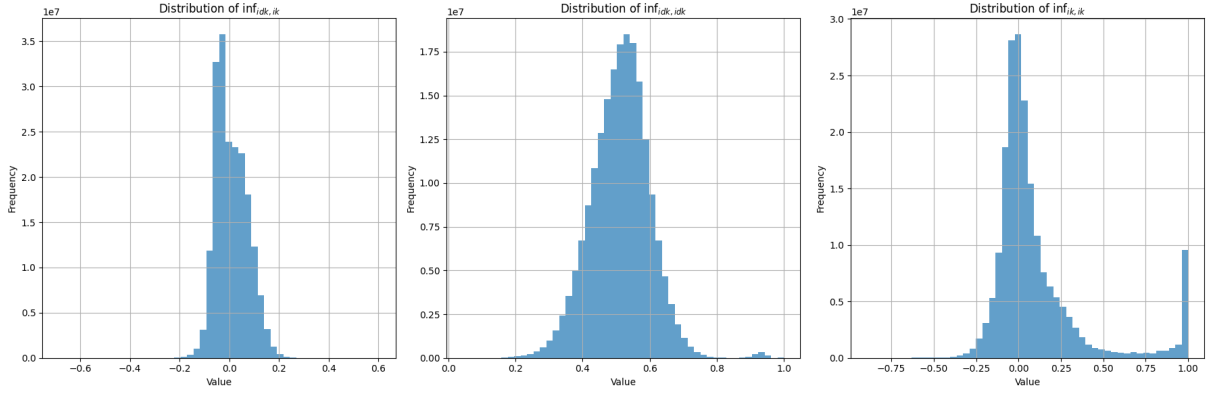
Figure 6: The overview of our proposed GRAIT.

- **Clear answers**: Existing chat LLMs often have a certain level of self-awareness and may choose not to answer or give vague responses like "I'm not sure" or "for reference only" when facing uncertain questions. However, we expect the LLM to answer the query directly, regardless of whether the answer is correct.

- **Concise answers to reduce inference costs**: In a 0-shot scenario, LLMs tend to produce longer responses (chain of thought processes or detailed descriptions). Using few-shot prompts helps obtain concise answers.

Although descriptive instructions can be added to 0-shot prompts to require LLMs to meet the above standards, LLMs generally do not strictly follow them. Furthermore, we believe that **the risk of introducing knowledge by few-shot prompting is minimal**. When choosing few-shot samples, we select from **different dataset splits**: for MCQA, test samples are from the MMLU test split and few-shot samples from the val split; for OEQA, test samples are from the triviaqa train split, with few-shot samples from the dev split.

A.6 Sensitive Experiment

Effect of temprature on `idk` samples' weight. We analyzed the Stable Influence of the samples discussed in Stage 3 of GRAIT and found that their values were relatively small. As a result, it was necessary to adjust the temperature to better normalize the weight assigned to each sample. To explore the specific effects of these adjustments, we conducted experiments using the MMLU dataset and the LLaMA3-8B-Instruct model. As shown in Table 10, the weight of the `idk` samples only becomes effective when τ is within a reasonable

range. When τ is set to 1, the weights of all samples remain almost unchanged, being equal to 1.

Effect of correctness threshold. In our experiments, we set the correctness threshold $\mathcal{T}_{\mathcal{C}}$ to 0.5. We conducted detailed experiments to determine this threshold, and as shown in Table 11, the model's performance is not highly sensitive to the choice of $\mathcal{T}_{\mathcal{C}}$. Within a reasonable range, our method consistently delivers strong results. All experiments were performed on the MMLU dataset using the LLaMA3-8B-Instruct model.

Table 4: The Prompt Template for Knowledge State
Query on MMLU. The Italic {*text*} in Curly Braces
Represents Variables That Need To be Replaced.

In-Context Examples

Answer the following question as simple as
possible.
Question: {*Question1*}
Answer: {*Answer1*}

Answer the following question as simple as
possible.
Question: {*Question2*}
Answer: {*Answer2*}

Answer the following question as simple as
possible.
Question: {*Question3*}
Answer: {*Answer3*}

Instruction

Answer the following question as simple as
possible.
Question: {*Question*}
Answer:

Table 5: The Prompt Template for Knowledge State
Query on TriviaQA. The Italic {*text*} in Curly Braces
Represents Variables That Need To be Replaced.

Instruction

There is a single choice question about
{*Task*}. If you know the answer, please di-
rectly respond with the correct answer A,
B, C, or D. If you do not know the answer,
please respond with "I don't know.".
Question:{*Question*}
A. {*Content_of_A*}
B. {*Content_of_B*}
C. {*Content_of_C*}
D. {*Content_of_D*}
Answer: {$Answer_{\text{rait}}$}

Table 6: The **REFUSE** Prompt Template for **Training**
on MMLU. The The Italic {*text*} in Curly Braces Rep-
resents Variables That Need To be Replaced.

Instruction
Answer the following question, and if you don't know the answer, only reply with "I don't know":{*Question*}
{$Answer_{\text{rait}}$}

Table 7: The **REFUSE** Prompt Template for **Training** on TriviaQA. The Italic {*text*} in Curly Braces Represents Variables That Need To be Replaced.

τ	0.01	0.05	0.1	0.2	0.5	1.0
THS	36.6	36.4	36.5	35.9	35.5	35.5

Table 10: Effect of temperature τ on `idk` samples' weights.

Instruction
There is a single choice question about {*Task*}. If you know the answer, please directly respond with the correct answer A, B, C, or D. If you do not know the answer, please respond with "I don't know.".
Question:{*Question*}
A. {*Content_of_A*}
B. {*Content_of_B*}
C. {*Content_of_C*}
D. {*Content_of_D*}
Answer:

Table 8: The **REFUSE** Prompt Template for **Evaluation** on MMLU. The Italic {*text*} in Curly Braces Represents Variables That Need To be Replaced.

$\mathcal{T_C}$	0.3	0.4	0.5	0.6	0.7
THS	37.4	36.7	36.4	36.4	36.3

Table 11: Effect of correctness threshold.

Instruction
Answer the following question, and if you don't know the answer, only reply with "I don't know": {*Question*}

Table 9: The **REFUSE** Prompt Template for **Evaluation** on TriviaQA. The Italic {*text*} in Curly Braces Represents Variables That Need To be Replaced.