

# 000 001 002 003 004 005 006 007 008 009 010 011 012 013 014 015 016 017 018 019 020 021 022 023 024 025 026 027 028 029 030 031 032 033 034 035 036 037 038 039 040 041 042 043 044 045 046 047 048 049 050 051 052 053 TASK-AWARE DATA SELECTION VIA PROXY-LABEL ENHANCED DISTRIBUTION MATCHING FOR LLM FINETUNING

**Anonymous authors**

Paper under double-blind review

## ABSTRACT

Task-specific fine-tuning of foundation models is critically dependent on the quality and relevance of the instruction data. While prevailing data selection methods rely exclusively on instruction instances  $X$  to approximate the target distribution, we argue that selection should align with the joint distribution of instructions and task-specific labels ( $X, Y$ ). However, task-specific labels  $Y$  are typically unavailable in practice. To address this, we reformulate the task-specific data selection problem and present a novel pipeline that leverages the reasoning capabilities of large language models (LLMs) to infer proxy labels, thereby facilitating joint distribution alignment. Our approach begins by propagating proxy labels from a small target set to a large, unlabeled source corpus. A two-stage filtering process then removes instances with label noise and refines the subset through distribution alignment. This strategy produces more semantically meaningful and task-aware selections than conventional similarity measures based on  $X$  alone. Experimental results show that fine-tuning on a subset of only 10K samples—selected from a pool of 300K—achieves performance competitive or superior to state-of-the-art methods.

## 1 INTRODUCTION

Large language models (LLMs) have demonstrated remarkable capabilities across a wide range of natural language processing tasks, owing to their extensive pretraining on diverse corpora. (Touvron et al., 2023; Achiam et al., 2023; Guo et al., 2025). To adapt these models to specific downstream applications, fine-tuning has become a standard practice. However, the success of fine-tuning critically depends on the quality and relevance of the instruction data, as low-quality or irrelevant data can lead to degraded generalization (Chen et al., 2023; Bukharin & Zhao, 2023). Therefore, selecting high-quality, task-appropriate instruction data is essential for effective model adaptation.

In the field of data selection for LLMs, two primary settings have emerged. The first is task-unspecific data selection (Liu et al., 2023; Chen et al., 2023), which is motivated by the observation that instruction-tuning datasets often contain low-quality or false instructions, which can degrade model performance if used directly in fine-tuning. Moreover, fine-tuning on very large datasets is computationally expensive. Hence, efficient data selection strategies are essential. In contrast, task-specific data selection (Xia et al., 2024; Liu et al., 2024) leverages a small task (*target*) dataset to retrieve the most relevant instruction samples from a *source* pool, aiming at maximizing model performance on particular downstream tasks.

Compared to general task-unspecific data selection, task-specific data selection has drawn growing interest, particularly as practitioners seek to specialize LLMs for specific tasks where limited task (target) examples are available. Most approaches in this area rely exclusively on aligning input features ( $X$ ), typically by measuring embedding similarity or gradient similarity between source and target samples (Xie et al., 2023; Xia et al., 2024; Liu et al., 2024). However, such  $X$ -only alignment strategies suffer from inherent limitations. For example, consider a target set from the legal domain containing “marital disputes” and “construction contracts.” A source sample related to an “cell phone contract” might be selected based on embedding similarity due to semantic overlap with “construction contract.” Similarly, if gradient similarity is used—where the model assigns high influence to shared concepts like “contract”—the same irrelevant sample may still be matched. Nevertheless, its actual

054 domain label (“telecom services”) does not align with the target label distribution. This discrepancy  
 055 underscores that input similarity alone is insufficient to ensure task relevance.

056 To address this, we propose leveraging the joint distribution  $P(X, Y)$  rather than marginal distribution  
 057  $P(X)$  for alignment. Specifically, we harness the reasoning capability of LLMs to infer proxy-labels  
 058 ( $Y$ ) for the target task, enabling a more semantically meaningful and task-aware selection process.  
 059 Nevertheless, aligning two random variables is more challenging than aligning only one, and the use  
 060 of proxy-labels inevitably introduces label noise. In this work, we reformulate the task-specific data  
 061 selection problem and introduce a novel framework that effectively tackles these issues.

062 The idea of incorporating auxiliary labels/taxonomy to improve task performance is evident in LLM  
 063 research. For example, instruction finetuning with the taxonomy of human knowledge has been shown  
 064 to enhance model generalization (Li et al., 2024). Similarly, in mathematical reasoning, generating  
 065 intermediate steps (as pseudo-labels) has proven essential for learning complex problem-solving skills  
 066 (Didolkar et al., 2024). These successes motivate us to leverage proxy-labels to enhance distribution  
 067 matching for task-specific data selection. Our contributions are threefold:

- 069 • We reformulate task-specific data selection as a joint distribution alignment problem and  
 070 propose a framework that uses LLM-generated proxy-labels to enable alignment.
- 071 • We design a two-stage filtering process that first removes out-of-distribution instances prone  
 072 to open-set label noise, and then refines the subset via distribution alignment, ensuring  
 073 higher semantic fidelity.
- 074 • Experiments show that fine-tuning on a subset of only 10K samples selected from a pool  
 075 of 300K instruction instances achieves competitive performance compared to some state-  
 076 of-the-art methods, and even outperforms models trained on the full dataset across several  
 077 benchmarks.

## 079 2 RELATED WORKS

080 **Task-Unspecific Data Selection for Instruction Tuning.** Existing approaches on task-unspecific  
 081 data selection can be broadly categorized into two types: LLM-free and LLM-based methods. LLM-  
 082 free approaches (Cao et al., 2023; He et al., 2024) typically leverage handcrafted heuristics or proxy  
 083 metrics—such as k-NN embedding distances in a feature space, sentence or token-level length or  
 084 Shapley values to approximate the utility or difficulty of individual samples without invoking large  
 085 models. In contrast, LLM-based strategies (Chen et al., 2023; Liu et al., 2023; Pang et al., 2024) treat  
 086 powerful pre-trained language models as automated evaluators to score and filter candidate instruction.  
 087 Notably, (Lu et al., 2023) propose InsTag, which uses LLM to generate pseudo-labels and refines  
 088 them for sample selection based on tag statistics. However, InsTag operates in a task-unspecific  
 089 manner and lacks alignment with a target task distribution. In comparison, our framework is more  
 090 comprehensive and explicitly incorporates task-aware alignment.

091 **Task-Specific Data Selection for Instruction Tuning.** Task-Specific Data Selection for Instruction  
 092 Tuning aims to identify and retrieve the most relevant samples from a large, general-purpose corpus  
 093 to improve performance on a specific target task, given only a small subset from that target domain.  
 094 Some methods rely on feature-space similarity: (Yao et al., 2022) select data by ranking candidates  
 095 based on their similarity to the target set, while (Xie et al., 2023) estimate importance weights in a  
 096 reduced feature space for resampling. Other methods leverage gradient information: (Xia et al., 2024)  
 097 score candidates by their maximum gradient similarity to any example in the target set. A more recent  
 098 approach by (Liu et al., 2024) formulates the problem through the lens of optimal transport, seeking  
 099 a minimal-cost mapping to align the distributions of the candidate pool and the target set. This  
 100 method further incorporates diversity-aware regularization to mitigate sample redundancy. However,  
 101 a common limitation across these methods is that they perform alignment or ranking using only  
 102 input features, without incorporating the target labels from the target set. This omission may fail to  
 103 fully capture task-specific relevance, as the semantic alignment between a candidate’s output and the  
 104 desired target outputs remains unverified.

105 **Domain Adaptation and Learning with Noisy Labels.** Our approach is further motivated by  
 106 principles from Domain Adaptation (Wang & Deng, 2018; Zhang et al., 2013) and Learning with

108 Noisy Labels (Natarajan et al., 2013; Song et al., 2022). Specifically, the process of propagating  
 109 proxy-labels from the target set to the source pool inevitably introduces label noise, a challenge  
 110 central to learning with noisy labels. Furthermore, even after filtering this noise, a domain shift may  
 111 persist between the refined source dataset and the target domain, which is a core problem addressed  
 112 by domain adaptation. Our pipeline is designed to tackle these challenges: we explicitly incorporate  
 113 mechanisms to identify and filter OOD samples with noisy labels, then align the distributions between  
 114 the source and target domains to ensure task relevance.

115  
 116 **Our Position.** Our work challenges the prevailing paradigm in task-specific data selection for  
 117 instruction tuning (e.g., LESS (Xia et al., 2024), TSDS (Liu et al., 2024)), which primarily focuses  
 118 on aligning marginal input distributions. We argue that true task relevance is defined by the joint  
 119 distribution of data and labels. By reformulating the selection criteria to incorporate target labels, we  
 120 unlock a new class of solutions for this problem. The pipeline we introduce is a concrete instantiation  
 121 of this principle, designed to select data that are not just superficially similar, but semantically  
 122 congruent with the target task.  
 123

### 3 PROBLEM FORMULATION

124 Define the following concepts:

125 • **Target Dataset:** Let  $\mathcal{T}$  denote a target dataset, where each instance is sampled from a joint  
 126 distribution  $P_{\text{target}}(X, Y_t)$ . Here,  $X$  represents text instruction instance (typically comprising an  
 127 instruction, an input, and a response), and  $Y_t$  denotes the corresponding domain-specific labels (e.g.,  
 128 for the legal domain,  $\mathcal{Y}_t = \{\text{family\_law}, \text{contract\_law}, \dots\}$ ).<sup>1</sup>

129 • **Source Dataset:** Let  $\mathcal{D} = \bigcup_{k=1}^K \mathcal{D}_k$  represent a multi-source dataset comprising  $K$  source  
 130 domains. Each  $\mathcal{D}_k$  is sampled from a distribution  $P_k(X, Y_k)$ . We assume that the support of the  
 131 target distribution is contained within the union of the supports of the source distributions, that is,  
 132

$$\text{supp}(P_{\text{target}}) \subseteq \bigcup_{k=1}^K \text{supp}(P_k),$$

133 which implies that the source domains are relevant to target domains.<sup>2</sup>  
 134

135 **Objective:** The goal is to select a fixed-size subset  $\mathcal{S} \subset \mathcal{D}$  where  $|\mathcal{S}| = N$  such that the empirical  
 136 distribution  $P_{\mathcal{S}}$  approximates  $P_{\text{target}}$  as closely as possible. The text instances from  $\mathcal{S}$  will  
 137 then be used to fine-tune a large language model (LLM) with the aim of maximizing performance on  
 138 the target task.

139 **Challenges:** The true Labels  $Y$  are unobservable in both  $\mathcal{T}$  and  $\mathcal{D}$ .

### 4 METHODOLOGY

140 This section is organized as follows. Section 4.1 describes the use of an LLM for target dataset  
 141 annotation. Section 4.2 covers label clustering and propagation to the source data. Section 4.3 details  
 142 the filtering of samples with open-set label noise. Section 4.4 outlines our sampling approach to  
 143 mitigate domain shift, and Section 4.5 provides an information-theoretic interpretation. An overview  
 144 of the pipeline is presented in Figure 1.

#### 4.1 GENERATING PROXY-LABELS BY LLM

145 We design a structured proxy-label space comprising four key fields: Task, Topic, Style, and  
 146 Audience. These dimensions are designed to capture distinct yet complementary aspects of each

147  
 148 <sup>1</sup>In general,  $\mathcal{T}$  may comprise instances from multiple domains (Isobe et al., 2021), in which case  $Y_t$  would  
 149 represent the set of tags across all constituent domains.  
 150

151 <sup>2</sup>This assumption aligns with standard domain adaptation theory (Ben-David et al., 2010) and the common  
 152 premise of data selection for instruction tuning, where a large source corpus is available. However, our pipeline  
 153 is also applicable when the source and target domains only partially overlap, such as when source data is limited.  
 154 See Appendix C for details.

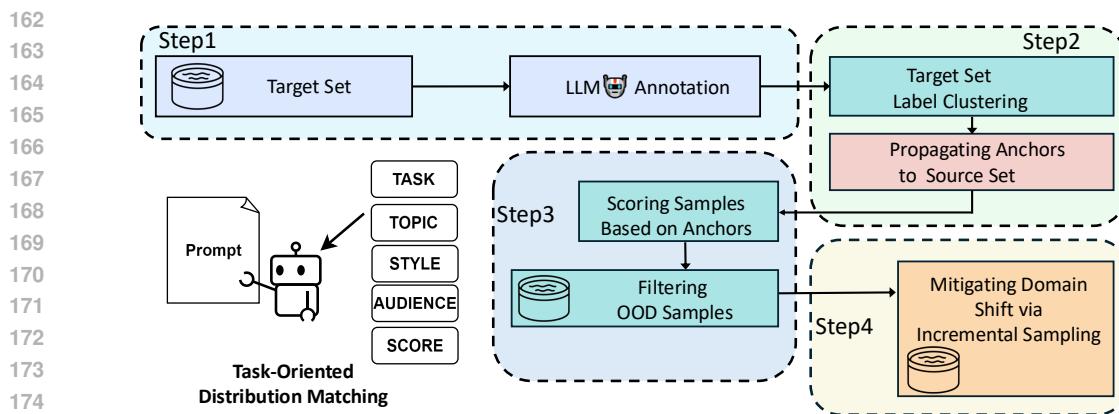


Figure 1: An overview of our 4-step data selection pipeline. We start by using prompt-based LLM annotation to generate domain-specific labels, which are then clustered and propagated to source dataset. Next, an LLM-generated quality score filters out OOD samples with open-set label noise. Finally, incremental sampling mitigates domain shifts between the source and target datasets.

instruction instance. Specifically, *Task* refers to the core functional intent, *Topic* reflects the subject, *Style* describes the rhetorical manner, and *Audience* indicates the intended user group. *Task* is modeled as a single-label field since most instructions have a dominant purpose, whereas *Topic*, *Style*, and *Audience* are multi-label fields to account for multifaceted instruction characteristics. We then prompt a pre-trained LLM with a structured summarization template (see Appendix B) to annotate each instruction with tags in four semantic fields—*Task*, *Topic*, *Style*, and *Audience*.

This semantic decomposition results in modular and interpretable representations that enable structured downstream processing, including clustering, alignment, and semantic similarity estimation. The use of a single-label format for *Task* provides a well-defined functional anchor for each instance, while multi-label modeling for *Topic*, *Style*, and *Audience* captures the multifaceted nature of instructional content. Free-form, descriptive phrase tags allow the representation space to remain open-ended and continuous, avoiding constraints imposed by fixed label taxonomies and facilitating compatibility with embedding-based semantic matching.

It is worth noting that our defined four **general** label domains (*Task*, *Topic*, *Style*, *Audience*) are used to demonstrate our framework’s flexibility. In practice, **general** label domains are applied when target knowledge is scarce, for instance, when only a set of target samples is available without clear target details. Conversely, when prior target knowledge exists, more **specific** label domains can be adopted. For example, using a domain-specific label such as "mathematical problem-solving skills" for math reasoning tasks has been shown to improve exemplar quality (Didolkar et al., 2024), indicating that more precise domain definitions generally lead to better data selection. In this paper, we use general label domains in experiments to emphasize our pipeline’s efficiency.

Note that LLM-generated labels may carry noise. We provide an analysis of their accuracy and consistency in Appendix D.2.

#### 4.2 PROXY-LABEL CLUSTERING AND PROPAGATION

To bridge the semantic gap between source domains and our LLM-generated labels, we employ a robust methodology centered on clustering and semantic propagation. First, we encode each label into a dense vector space using a sentence-level embedding model, processing each field (e.g., *Topic*, *Style*) independently to preserve their unique semantic characteristics.

**Proxy Label clustering** We apply *k*-means clustering (with  $k = 100$ ) to these embeddings, which effectively groups semantically similar tags into distinct clusters (Figure 3 demonstrates the robustness to  $k$ ). This process is fundamental because it accomplishes two key objectives: it abstracts away the significant lexical variation inherent in LLM-originated phrases and condenses their broad diversity into a compact set of semantic centroids. These centroids function as semantic anchors, providing a stable and noise-reduced representation of the entire target domain’s conceptual landscape.

**Propagation** We project samples from the source domains into the same embedding space and compute their cosine similarity against all semantic anchors. This allows for a soft, interpretable matching process. We then assign the most relevant anchors—selecting the top-1 for Task and the top-3 for Topic, Style, and Audience, to each source sample. This selective assignment is not a hard label transfer but a test-guided semantic projection, enabling a nuanced and measurable integration of the target domain’s characteristics into the source.

In Appendix D.4, we demonstrate that clustering enhances the robustness of label propagation from the target set to the source set, particularly against label noise.

### 4.3 FILTERING OOD SAMPLES WITH LABEL NOISE

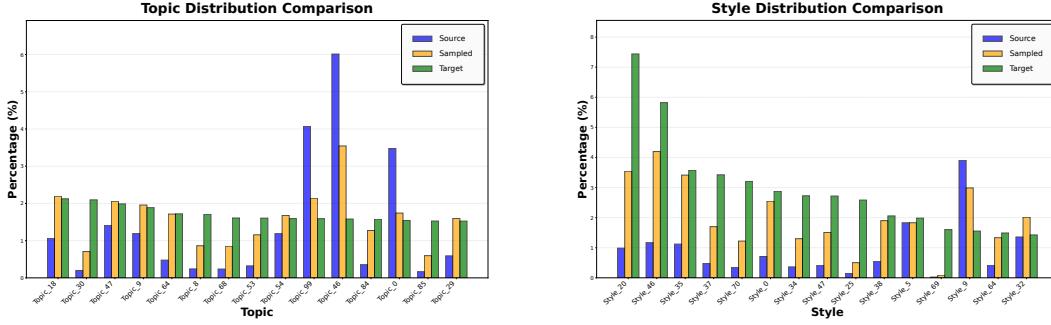


Figure 2: These figures visualize the results of our incremental sampling method. The left and right panels show the matching results for the *Topic* and *Style*, respectively. The anchors are ranked from highest to lowest count of target set (left to right) for clarity, and the top 15 anchors are shown to enhance readability.

The propagation of target domain anchors to source domains inevitably introduces open-set label noise where irrelevant instructions are assigned to an target label. To ensure data quality, filtering these mismatched samples becomes essential. In learning with noisy labels, a common detection strategy leverages the memorization effect of deep neural networks (DNNs), wherein DNNs learn clean patterns before memorizing noisy ones (Arpit et al., 2017). By training a DNN for a few epochs and ranking samples by loss values, high-loss instances are typically identified as noisy—an approach effective for both closed-set and open-set noise (Jiang et al., 2017; Han et al., 2018; Li et al., 2020; Sachdeva et al., 2021).

However, this method presents limitations in our multi-field setting: training separate DNNs is computationally expensive and lacks generalization capability. Each new task or label space change would necessitate retraining, creating scalability bottlenecks. To overcome these constraints, we leverage the robust reasoning capabilities of large language models (LLMs). Specifically, we employ LLMs to score each sample based on its semantic alignment with assigned anchors, evaluating the quality and relevance of the assigned anchors (see Appendix B for prompt details). Samples falling below a predefined threshold (e.g., score < 6) are identified as out-of-distribution (OOD)—indicating open-set label noise—and subsequently removed from the source corpus (Appendix C has further explanations).

### 4.4 INCREMENTAL SAMPLING FOR MITIGATING DOMAIN SHIFTS

In practice, after removing OOD samples, a simple approach is to randomly sample a fixed number of instances (e.g., 10,000) from the source domain for fine-tuning. However, even after OOD filtering, a distribution shift may persist between the remaining source data and the target set. This section presents a method to mitigate such domain shifts more effectively.

Leading works in domain adaptation (Zhang et al., 2013; Zhao et al., 2019) theoretically and empirically demonstrate that target shift (or prior probability shift) is a common and critical type of distribution discrepancy in real-world scenarios and suggest that aligning for target shift is often more advantageous for robustness and effectiveness than aligning for covariate shift. Consequently,

270 our pipeline prioritizes mitigating target shift between the source and target domains. The implicit  
 271 assumption of target shift is that the conditional distribution  $P(X|Y)$  remains stable across the target  
 272 and cleaned source domains, while the label distribution  $P(Y)$  varies. Note that the label space here  
 273 corresponds to the anchors (clusters) defined in Section 4.2, not the labels/tags generated by the  
 274 LLM in Section 4.1.

275 **Sampling Objective:** Let  $P^*(Y)$  be the empirical anchor distribution of the target dataset and  $\hat{P}(Y)$   
 276 be the empirical anchor distribution of the sampled source dataset. To mitigate label (anchor) shift,  
 277 our objective is to minimize  $\|P^*(Y) - \hat{P}(Y)\|_1$ .  
 278

279 Given a sampling budget  $N$ , we propose an incremental sampling method outlined in Algorithm 1.  
 280 At each step, we compute the label gap and select a training instance associated with the label with  
 281 the largest positive gap (Lines 4-5, Algorithm 1). Because each instance contains a fixed number  
 282 of labels per field (e.g., one for *Task*, three for *Topic*), all candidates contribute equally to the label  
 283 counts. The process iterates until the budget is exhausted or no viable candidates remain, producing a  
 284 subset whose empirical distribution  $\hat{P}(Y)$  closely approximates the target distribution  $P^*(Y)$ .  
 285

Algorithm 1 is applied independently to each field (e.g., *Task* or *Topic*). For each selected sample,  
 we update the counts for all associated labels. This introduces a natural balancing effect across the  
 label space, accelerating convergence toward  $P^*(Y)$ , especially in dense semantic fields where tags  
 frequently co-occur. Figure 2 presents the matching results, demonstrating that incremental sampling  
 produces a source subset whose anchor distribution aligns with the target distribution. Although  
 perfect matching is rarely achievable due to limited sample sizes for certain classes in the source  
 dataset and the sampling budget, our incremental sampling method effectively reduces the distribution  
 gap between the source and target domains. A key implication of this method is its flexibility. Since  
 alignment is performed per field, we can choose to align by *Task*, *Topic*, or other fields based on the  
 specific requirements of the downstream application.  
 294

---

### 295 Algorithm 1 Distribution Matching via Incremental Sampling

---

296 **Require:** source pool  $\mathcal{D}$ , target set  $\mathcal{T}$ , sampling budget  $N$   
 297 1: Compute target label distribution  $P^*(Y)$  from  $\mathcal{T}$   
 298 2: Initialize empirical distribution  $\hat{P}(Y) \leftarrow \mathbf{0}$  and selection set  $\mathcal{S} \leftarrow \emptyset$   
 299 3: **while**  $|\mathcal{S}| < N$  **do**  
 300 4:   Compute label-wise gap:  $g(y) = P^*(y) - \hat{P}(y)$   
 301 5:   Identify label  $y^* = \arg \max_y g(y)$  such that unused candidates with tag  $y$  exist  
 302 6:   **if** no valid  $y^*$  found **then**  
 303     **break**  
 304   **end if**  
 305 9:   Select candidate  $x^*$  with tag  $y^*$   
 306 10:    $\mathcal{S} \leftarrow \mathcal{S} \cup \{x^*\}$ ; update  $\hat{P}(Y)$  using labels from  $x^*$   
 307 11: **end while**  
 308 12: **return** Sample set  $\mathcal{S}$

---

310 4.5 INFORMATION-THEORETIC EXPLANATION

311 We explain the benefits of incorporating label information in the dataset selection process from an  
 312 information-theoretic perspective. Let  $T$  be a random variable representing the **target domain**, which  
 313 is characterized by  $P_{\text{target}}(X, Y_t)$ . The outcome of the selection algorithm is a **subset**  $\mathcal{S} \subset \mathcal{D}$ . The  
 314 instances and their associated labels of  $\mathcal{S}$  are described by the random variables  $X_{\mathcal{S}}$  and  $Y_{\mathcal{S}}$ .  
 315

316 The mutual information between the task and the selected data can be decomposed as:  
 317

$$I(T; (X_{\mathcal{S}}, Y_{\mathcal{S}})) = I(T; X_{\mathcal{S}}) + I(T; Y_{\mathcal{S}} | X_{\mathcal{S}})$$

318 This decomposition reveals two distinct sources of information:  
 319

- 320   •  $I(T; X_{\mathcal{S}})$  quantifies the information about the task  $T$  contained in the *input features*  $X_{\mathcal{S}}$  of  
 321   the selected subset.
- 322   •  $I(T; Y_{\mathcal{S}} | X_{\mathcal{S}})$  captures the *additional information* about  $T$  provided by the *labels*  $Y_{\mathcal{S}}$  of  
 323   the subset, given the inputs  $X_{\mathcal{S}}$ .

Conventional task-specific data selection methods in LLM (Xia et al., 2024; Liu et al., 2024), which rely exclusively on the similarity between input features (i.e.,  $X_S$ ), primarily aim to maximize the first term,  $I(T; X_S)$ . However, they inherently ignore the second term,  $I(T; Y_S | X_S)$ . This constitutes a limitation, since instances with semantically similar  $X$  may be associated with labels  $Y$  that are irrelevant or even contradictory to the target task  $T$ . In contrast, our proposed method matches the joint distribution between the selected data and the target domain, which implicitly maximize the complete objective  $I(T; (X_S, Y_S))$ .

## 5 EXPERIMENTS

### 5.1 EXPERIMENTS SETUP

We utilize three pretrained LLMs across different stages of our pipeline: **Qwen2.5-7B-Instruct** (Yang et al., 2024) is used for annotation and scoring, **BGE-M3** (Chen et al., 2024) for embedding and clustering, and **LLaMA-3.1-8B** (Grattafiori et al., 2024) for supervised instruction tuning on the selected subsets using **LoRA** (Hu et al., 2022).

Table 1: Instruction tuning data pool used for supervised fine-tuning

Dataset	Flan V2	Open-Assistant 1	WizardLM	Dolly	Stanford Alpaca	Total
Training Size	100K	33K	100K	15K	52K	300K

Table 2: Evaluation benchmarks test set sizes and associated model capabilities

Benchmark	MMLU	TruthfulQA	GSM8K	BBH	TyDiQA
Test Size	14,042	790	1,319	6,511	5,077
Capability	Factuality	Truthfulness	Reasoning	Reasoning	Multilinguality

**Data Pool** Our data pool comprises two components: a source corpus collected from five widely-used instruction tuning datasets—**Flan V2** (Longpre et al., 2023), **Open-Assistant 1** (Köpf et al., 2023), **WizardLM** (Xu et al., 2023), **Dolly** (Databricks, 2023), and **Stanford Alpaca** (Taori et al., 2023)—and a set of evaluation benchmarks. Detailed training set statistics are provided in Table 1. Evaluation is conducted on five standard alignment benchmarks, each targeting a distinct model capability: **MMLU** (Hendrycks et al., 2020), **TruthfulQA** (Lin et al., 2021), **GSM8K** (Cobbe et al., 2021) and **BBH** (Suzgun et al., 2022), and **TyDiQA** (Clark et al., 2020). The sizes of these evaluation sets are summarized in Table 2. These benchmarks also serve as our alignment targets: for each dataset, we sample 20% of the test set as target set to guide our subset selection. The remaining 80% is held out for final evaluation.

**Baselines:** We compare our method against a set of representative baselines commonly used in instruction tuning and data selection:

- **Vanilla Base Model:** The base model evaluated in a zero-shot setting, without any supervised fine-tuning. This serves as a reference point to measure the impact of instruction tuning.
- **Completion Length:** Following (Zhao et al., 2024), we rank all samples by the total number of tokens in the prompt-response pair. The top 10K longest samples are selected under the assumption that longer responses carry higher information density.
- **k-NN:** Each sample is scored by its average distance to the  $k$  nearest neighbors in a sentence embedding space (Reimers & Gurevych, 2019). Higher distances indicate semantic uniqueness, and the 10K most isolated samples are selected.
- **BM25:** A retrieval metric which considers term frequency and inverse document frequency.
- **RDS+ (Ivison et al., 2025):** A representation-based retrieval method which uses weighted mean pooling of pretrained LLM hidden states.
- **Random:** A simple baseline that randomly selects 10K examples from the data pool.

378  
 379 Table 3: Performance comparison across alignment benchmarks. All the methods select 10K samples  
 380 from the source except the method of Full (300K). *min score* denotes the score threshold described in  
 381 Section 4.3. For each benchmark, top-performing score is shown in bold, while second-best score is  
 382 underlined.

<b>Methods</b>	<b>MMLU</b>	<b>TruthfulQA</b>	<b>GSM</b>	<b>BBH</b>	<b>TyDiQA</b>
Vanilla Base Model	64.3	32.8	51.0	54.8	22.7
Completion Length	63.3	6.3	52.5	61.8	61.8
k-NN-10	61.4	41.9	51.5	61.9	61.3
Random	64.0	33.5	52.5	59.8	60.9
BM25	63.2	28.6	51.5	59.1	61.6
RDS+	63.6	3.5	54.0	59.1	60.2
LESS	63.2	33.5	56.5	60.4	<b>64.5</b>
TSDS	63.7	44.9	48.5	<u>62.3</u>	<u>64.1</u>
Full (300k)	63.5	42.0	<b>61.0</b>	59.1	62.8
<i>Align-based selection, min score <math>\geq 7</math></i>					
Align_topic	64.6	<u>46.4</u>	57.0	60.0	59.3
Align_style	64.1	<b>47.3</b>	57.0	59.2	58.7
Align_task	64.2	44.6	<u>58.0</u>	58.2	58.2
Align_audience	64.0	45.6	54.0	56.4	58.8
<i>Align-based selection, min score <math>\geq 6</math></i>					
Align_topic	65.0	24.7	55.5	61.9	57.0
Align_style	<b>65.1</b>	32.1	55.0	<b>62.5</b>	58.9
Align_task	65.0	38.0	54.0	59.8	58.1
Align_audience	64.9	42.6	55.0	59.7	59.1
<i>Align-based selection, min score <math>\geq 5</math></i>					
Align_topic	<u>65.1</u>	36.2	52.5	61.4	57.6
Align_style	64.9	31.2	55.0	60.4	59.4
Align_task	64.9	35.6	54.5	60.0	57.7
Align_audience	65.1	39.2	55.0	60.6	58.0

- 410  
 411
  - **LESS** (Xia et al., 2024): A gradient-based data selection framework that estimates the influence of training samples using gradient similarity to few-shot target examples.
  - **TSDS** (Liu et al., 2024): A task-specific data selection framework that aligns the source distribution with a small representative set from the target task. TSDS formulates data selection as an optimal transport problem with a diversity-promoting regularizer based on kernel density estimation. We use the target set to estimate the task distribution and select 10K training samples accordingly, ensuring parity with our proposed method.
  - **Full (300K)**: A strong baseline where the entire 300K training set is used for instruction tuning.

## 5.2 OPENLLM LEADERBOARD EVALUATION RESULTS

424 Following the OpenLLM Leaderboard evaluation protocol, we adopt *exact match* (EM) as the scoring  
 425 metric for MMLU, TruthfulQA, GSM8K, and BBH. For TyDiQA, we use the 1-shot F1 score as  
 426 reported by the leaderboard.

427 **Results Analyses:** As shown in Table 3, our method achieves competitive or superior performance  
 428 compared to SOTA methods (LESS, TSDS) on task-specific data selections for the MMLU, Truth-  
 429 fulQA, GSM8K, and BBH benchmarks. However, our approach underperforms on TyDiQA. We  
 430 hypothesize that this is because TyDiQA is a multilingual dataset, and the label fields we designed  
 431 (Task, Topic, Style, Audience) may not adequately capture its inherent characteristics. Consequently,  
 samples related to TyDiQA in the source pool were likely filtered out by our selection algorithm.

Table 4: Performance comparison across alignment benchmarks with 1K/5K selection size.

Methods	MMLU	TruthfulQA	GSM	BBH	TyDiQA
Random (1K)	64.0	3.8	49.0	54.2	56.4
Align_style (1K)	64.2	6.8	50.5	57.7	54.5
Random (5K)	63.7	3.5	52.0	59.4	58.7
Align_style (5K)	63.9	32.4	52.0	61.0	59.7

The *min score* threshold is an important parameter in our pipeline. Analogous to the problem of learning with noisy labels (Jiang et al., 2017; Han et al., 2018), a higher threshold risks retaining too few clean (relevant) samples, while a lower threshold risks polluting the source set with too many irrelevant samples. Empirically, we find that  $\text{min score} \geq 7$  yields stable results across all datasets, while  $\text{min score} \geq 6$  also performs robustly, with the exception of TruthfulQA.

It is also evident that *different semantic fields lead to varying alignment effectiveness*. For instance, the Align-based selection results show that the choice of label field has a minimal impact on performance for the MMLU dataset but significantly affects results on the TruthfulQA dataset, as indicated by its high deviation. Furthermore, the optimal label field differs across datasets: Align-Style yields the best performance on TruthfulQA, while Align-Task yields the best performance for GSM. It is important to note that our intention is not to suggest empirically testing all label domains for each new task. The primary purpose of presenting four domains is to illustrate the inherent flexibility of our pipeline. The results in Table 3 demonstrate that with a  $\text{min score} \geq 7$ , selecting any of these domains yields strong performance on MMLU, TruthfulQA, and GSM benchmarks compared to baseline methods. For tasks with more specific information, a custom domain can also be designed, as discussed in Section 4.1. Finally, our approach offers significant cost savings over task-specific selection methods like LESS (Xia et al., 2024) and TSDS (Liu et al., 2024), as detailed in our cost analysis (Appendix D.9). We further evaluate our pipeline with limited data. As shown in Table 4, our method achieves good performance even with smaller selection sizes (1K and 5K), demonstrating its effectiveness.

### 5.3 ABLATION STUDIES

Table 5 presents the results of our ablation study on the contributions of OOD sample filtering and incremental sampling. We compare three configurations:

- **Incremental Sampling only:** This baseline applies incremental sampling to the entire 300K dataset after label propagation, without any OOD filtering.
- **OOD Filtering only:** This variant filters the dataset by retaining only samples with an LLM-assigned  $\text{min score} \geq 6$ , followed by random sampling.
- **Complete Method:** This is our full pipeline, which first applies OOD filtering ( $\text{min score} \geq 6$ ) and then performs incremental sampling on the filtered subset.

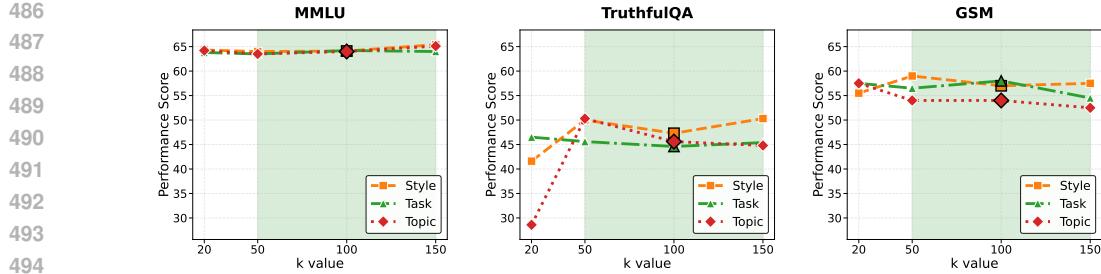
For each benchmark, results are reported under the best-performing label field. The result clearly indicates that combining both techniques is most effective and leads to superior performance.

Table 5: Evaluating individual contributions of OOD sample filtering and incremental sampling.

Methods		MMLU	TruthfulQA	GSM	BBH	TyDiQA
OOD Filtering	Incremental Sampling					
×	✓	64.3	3.5	55.0	49.1	60.5
✓	✗	64.8	42.1	55.0	61.9	56.4
✓	✓	65.1	42.6	55.5	62.5	59.1

To further verify the necessity of joint alignment on  $P(X, Y)$  over input-only alignment on  $P(X)$ , and to demonstrate the advantage of our design, we conduct the following experiments:

- **Adding semantic information to an existing selection pipeline.** After LLM annotation, we insert the proxy label information into each instruction and apply the standard RDS+ selection pipeline.

Figure 3: Performance of our pipeline with respect to the number of clusters  $k$  in K-means clustering.

• **Applying our pipeline directly to inputs  $X$ .** We remove the LLM-annotation step entirely. Target instructions are clustered into anchors, source samples are propagated according to anchor–source embedding similarity, and we then perform selection and incremental sampling.

• **Replacing LLM-based OOD filtering with embedding-similarity filtering.** Instead of using LLM-assigned scores, we compute cosine similarity between source and anchor embeddings, filter out low-similarity samples, and then apply our selection and incremental sampling procedure.

As shown in Table 6: (1) The direct integration of semantic information (RDS+ with semantic info) fails to yield consistent improvements over RDS+, indicating that explicit distribution alignment is necessary to utilize semantic-field information effectively; (2) Our pipeline outperforms its direct application to  $X$ , confirming that the performance gains stem from joint alignment on  $P(X, Y)$  rather than from clustering and sampling alone; (3) Compared to similarity-based OOD filtering, our method demonstrates that LLM-based judgments provide more reliable filtering.

Table 6: Comparison of alternative designs for our joint alignment.

Methods	MMLU	TruthfulQA	GSM	BBH	TyDiQA
RDS+	63.6	3.5	54.0	59.1	60.2
RDS+ (with semantic information)	63.7	5.4	53.5	60.7	59.7
Applying our pipeline directly to $X$	63.4	40.8	54.0	59.9	58.2
Our pipeline with similarity-based OOD filtering	63.0	28.3	51.0	57.4	61.6
Our full pipeline (align topic, min score:7)	64.6	46.4	57.0	60.0	59.3

We also evaluate the impact of the K-means cluster count  $k$  on our pipeline’s performance in Figure 3. The results show consistent performance for  $k$  between 50 and 150 across datasets and label domains. However, a smaller  $k$  can lead to higher variance, as seen on TruthfulQA. For robustness and reproducibility, we fix  $k = 100$  in all experiments.

The Appendix includes additional experiments and analyses. For the experimental part, we examine the consistency between LLM annotation and LLM scoring, evaluate the robustness of our pipeline to variations in label quality, and assess its selection performance when fine-tuning other foundation models, etc. For the analytical part, we provide a deeper investigation into the connection between our pipeline and the fields of Domain Adaptation (DA) and Learning with Noisy Labels (LNL).

## 6 CONCLUSION

In this paper, we revisit task-aware data selection by formulating it as a joint alignment of both input features ( $X$ ) and task-specific labels ( $Y$ ). This paradigm shift moves beyond input-only alignment to naturally address real-world challenges like label noise and domain shift. By integrating techniques from noise-robust learning and domain adaptation, we propose a pipeline that effectively mitigates these issues. Our experiments on multiple benchmarks demonstrate the efficacy of our pipeline. The primary contribution, however, is not merely the specific pipeline but the reconceptualization of the problem itself. Framing it as joint distribution alignment unlocks a broader solution space for future research. Our work serves as an initial instantiation of this perspective. Future efforts can be made to explore alternative efficient methods for handling label noise and domain shift.

540  
**Ethics Statement.** Our work focus on task-specific data selection for fine-tuning large language  
 541 models (LLMs). All datasets and models used are publicly available; therefore, no intellectual  
 542 property or personal privacy concerns are involved.  
 543

544 **Reproducibility Statement.** We have released our code anonymously at <https://anonymous.4open.science/r/TADS-B14D/README.md>, along with detailed guidelines  
 545 in the README file. Additionally, we provide a comprehensive description of our experimental  
 546 setup in Appendix E.  
 547

548 **REFERENCES**  
 549

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

553 Woosuk Kwon anda Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu,  
 554 Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language  
 555 model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on*  
 556 *Operating Systems Principles*, 2023.

557 Devansh Arpit, Stanisław Jastrzębski, Nicolas Ballas, David Krueger, Emmanuel Bengio, Maxinder S  
 558 Kanwal, Tegan Maharaj, Asja Fischer, Aaron Courville, Yoshua Bengio, et al. A closer look at  
 559 memorization in deep networks. In *International conference on machine learning*, pp. 233–242.  
 560 PMLR, 2017.

561 Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman  
 562 Vaughan. A theory of learning from different domains. *Machine learning*, 79(1):151–175, 2010.

563 Alexander Bukharin and Tuo Zhao. Data diversity matters for robust instruction tuning. *arXiv*  
 564 *preprint arXiv:2311.14736*, 2023.

565 Yihan Cao, Yanbin Kang, Chi Wang, and Lichao Sun. Instruction mining: Instruction data selection  
 566 for tuning large language models. *arXiv preprint arXiv:2307.06290*, 2023.

567 Jianlv Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. Bge m3-embedding:  
 568 Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge  
 569 distillation, 2024.

570 Lichang Chen, Shiyang Li, Jun Yan, Hai Wang, Kalpa Gunaratna, Vikas Yadav, Zheng Tang, Vijay  
 571 Srinivasan, Tianyi Zhou, Heng Huang, et al. Alpagasus: Training a better alpaca with fewer data.  
 572 *arXiv preprint arXiv:2307.08701*, 2023.

573 Jonathan H Clark, Eunsol Choi, Michael Collins, Dan Garrette, Tom Kwiatkowski, Vitaly Nikolaev,  
 574 and Jennimaria Palomaki. Tydi qa: A benchmark for information-seeking question answering in ty  
 575 logically di verse languages. *Transactions of the Association for Computational Linguistics*, 8:  
 576 454–470, 2020.

577 Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser,  
 578 Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve  
 579 math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

580 Databricks. Introducing dolly: The world’s first truly open instruction-  
 581 tuned llm. [https://www.databricks.com/blog/2023/04/12/  
 582 dolly-first-open-commercially-viable-instruction-tuned-l1m](https://www.databricks.com/blog/2023/04/12/dolly-first-open-commercially-viable-instruction-tuned-l1m), 2023.  
 583 Accessed: 2024-05-15.

584 Aniket Didolkar, Anirudh Goyal, Nan Rosemary Ke, Siyuan Guo, Michal Valko, Timothy Lillicrap,  
 585 Danilo Jimenez Rezende, Yoshua Bengio, Michael C Mozer, and Sanjeev Arora. Metacognitive  
 586 capabilities of llms: An exploration in mathematical problem solving. *Advances in Neural*  
 587 *Information Processing Systems*, 37:19783–19812, 2024.

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

- 594 Sylvain Gugger, Lysandre Debut, Thomas Wolf, Philipp Schmid, Zachary Mueller, Sourab Man-  
 595 grulkar, Marc Sun, and Benjamin Bossan. Accelerate: Training and inference at scale made simple,  
 596 efficient and adaptable. <https://github.com/huggingface/accelerate>, 2022.
- 597
- 598 Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu,  
 599 Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms  
 600 via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.
- 601 Bo Han, Quanming Yao, Xingrui Yu, Gang Niu, Miao Xu, Weihua Hu, Ivor Tsang, and Masashi  
 602 Sugiyama. Co-teaching: Robust training of deep neural networks with extremely noisy labels.  
 603 *Advances in neural information processing systems*, 31, 2018.
- 604
- 605 Yexiao He, Ziyao Wang, Zheyu Shen, Guoheng Sun, Yucong Dai, Yongkai Wu, Hongyi Wang, and  
 606 Ang Li. Shed: Shapley-based automated dataset refinement for instruction fine-tuning. *arXiv  
 607 preprint arXiv:2405.00705*, 2024.
- 608 Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution  
 609 examples in neural networks. *arXiv preprint arXiv:1610.02136*, 2016.
- 610
- 611 Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and  
 612 Jacob Steinhardt. Measuring massive multitask language understanding. *arXiv preprint  
 613 arXiv:2009.03300*, 2020.
- 614 Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang,  
 615 Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022.
- 616
- 617 Takashi Isobe, Xu Jia, Shuaijun Chen, Jianzhong He, Yongjie Shi, Jianzhuang Liu, Huchuan Lu,  
 618 and Shengjin Wang. Multi-target domain adaptation with collaborative consistency learning.  
 619 In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp.  
 620 8187–8196, 2021.
- 621 Hamish Ivison, Muru Zhang, Faeze Brahman, Pang Wei Koh, and Pradeep Dasigi. Large-scale data  
 622 selection for instruction tuning. *arXiv preprint arXiv:2503.01807*, 2025.
- 623
- 624 Lu Jiang, Zhengyuan Zhou, Thomas Leung, Li-Jia Li, and Li Fei-Fei. Mentornet: Learning  
 625 data-driven curriculum for very deep neural networks on corrupted labels. *arXiv preprint  
 626 arXiv:1712.05055*, 2017.
- 627
- 628 Andreas Köpf, Yannic Kilcher, Dimitri Von Rütte, Sotiris Anagnostidis, Zhi Rui Tam, Keith  
 629 Stevens, Abdullah Barhoum, Duc Nguyen, Oliver Stanley, Richárd Nagyfi, et al. Openassistant  
 630 conversations-democratizing large language model alignment. *Advances in Neural Information  
 631 Processing Systems*, 36:47669–47681, 2023.
- 632
- Haoran Li, Qingxiu Dong, Zhengyang Tang, Chaojun Wang, Xingxing Zhang, Haoyang Huang, Shao-  
 633 han Huang, Xiaolong Huang, Zeqiang Huang, Dongdong Zhang, et al. Synthetic data (almost) from  
 634 scratch: Generalized instruction tuning for language models. *arXiv preprint arXiv:2402.13064*,  
 635 2024.
- 636
- Junnan Li, Richard Socher, and Steven CH Hoi. Dividemix: Learning with noisy labels as semi-  
 637 supervised learning. *arXiv preprint arXiv:2002.07394*, 2020.
- 638
- Stephanie Lin, Jacob Hilton, and Owain Evans. Truthfulqa: Measuring how models mimic human  
 639 falsehoods. *arXiv preprint arXiv:2109.07958*, 2021.
- 640
- Hong Liu, Zhangjie Cao, Mingsheng Long, Jianmin Wang, and Qiang Yang. Separate to adapt: Open  
 641 set domain adaptation via progressive separation. In *Proceedings of the IEEE/CVF conference on  
 642 computer visionMetacognitive Capabilities of LLMs: An Exploration in Mathematical Problem  
 643 Solving and pattern recognition*, pp. 2927–2936, 2019.
- 644
- Wei Liu, Weihao Zeng, Keqing He, Yong Jiang, and Junxian He. What makes good data for  
 645 alignment? a comprehensive study of automatic data selection in instruction tuning. *arXiv preprint  
 646 arXiv:2312.15685*, 2023.

- 648 Zifan Liu, Amin Karbasi, and Theodoros Rekatsinas. Tsds: Data selection for task-specific model  
 649 finetuning. *arXiv preprint arXiv:2410.11303*, 2024.
- 650
- 651 Shayne Longpre, Le Hou, Tu Vu, Albert Webson, Hyung Won Chung, Yi Tay, Denny Zhou, Quoc V  
 652 Le, Barret Zoph, Jason Wei, et al. The flan collection: Designing data and methods for effective  
 653 instruction tuning. In *International Conference on Machine Learning*, pp. 22631–22648. PMLR,  
 654 2023.
- 655 Keming Lu, Hongyi Yuan, Zheng Yuan, Runji Lin, Junyang Lin, Chuanqi Tan, Chang Zhou, and  
 656 Jingren Zhou. # instag: Instruction tagging for analyzing supervised fine-tuning of large language  
 657 models. *arXiv preprint arXiv:2308.07074*, 2023.
- 658
- 659 Nagarajan Natarajan, Inderjit S Dhillon, Pradeep K Ravikumar, and Ambuj Tewari. Learning with  
 660 noisy labels. *Advances in neural information processing systems*, 26, 2013.
- 661
- 662 Pau Panareda Busto and Juergen Gall. Open set domain adaptation. In *Proceedings of the IEEE  
 663 international conference on computer vision*, pp. 754–763, 2017.
- 664
- 665 Jinlong Pang, Jiaheng Wei, Ankit Parag Shah, Zhaowei Zhu, Yaxuan Wang, Chen Qian, Yang Liu,  
 666 Yujia Bao, and Wei Wei. Improving data efficiency via curating llm-driven rating systems. *arXiv  
 667 preprint arXiv:2410.10877*, 2024.
- 668
- 669 Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks.  
*arXiv preprint arXiv:1908.10084*, 2019.
- 670
- 671 Ragav Sachdeva, Filipe R Cordeiro, Vasileios Belagiannis, Ian Reid, and Gustavo Carneiro. Evi-  
 672 dentialmix: Learning with combined open-set and closed-set noisy labels. In *Proceedings of the  
 673 IEEE/CVF winter conference on applications of computer vision*, pp. 3607–3615, 2021.
- 674
- 675 Hwanjun Song, Minseok Kim, Dongmin Park, Yooju Shin, and Jae-Gil Lee. Learning from noisy  
 676 labels with deep neural networks: A survey. *IEEE transactions on neural networks and learning  
 677 systems*, 34(11):8135–8153, 2022.
- 678
- 679 Mirac Suzgun, Nathan Scales, Nathanael Schärlí, Sebastian Gehrmann, Yi Tay, Hyung Won Chung,  
 680 Akanksha Chowdhery, Quoc V Le, Ed H Chi, Denny Zhou, et al. Challenging big-bench tasks  
 681 and whether chain-of-thought can solve them. *arXiv preprint arXiv:2210.09261*, 2022.
- 682
- 683 Rohun Taori, Ishaan Gulrajani, Ting Zhang, Yann Dubois, Xiaodan Li, Carlos Guestrin, Percy Liang,  
 684 and Tatsunori B. Hashimoto. Stanford alpaca: An instruction-following llama model. [https://github.com/tatsu-lab/stanford\\_alpaca](https://github.com/tatsu-lab/stanford_alpaca), 2023. GitHub repository. Accessed:  
 685 2024-05-15.
- 686
- 687 Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée  
 688 Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and  
 689 efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.
- 690
- 691 Mei Wang and Weihong Deng. Deep visual domain adaptation: A survey. *Neurocomputing*, 312:  
 692 135–153, 2018.
- 693
- 694 Mengzhou Xia, Sadhika Malladi, Suchin Gururangan, Sanjeev Arora, and Danqi Chen. Less:  
 695 Selecting influential data for targeted instruction tuning. *arXiv preprint arXiv:2402.04333*, 2024.
- 696
- 697 Sang Michael Xie, Shibani Santurkar, Tengyu Ma, and Percy S Liang. Data selection for language  
 698 models via importance resampling. *Advances in Neural Information Processing Systems*, 36:  
 699 34201–34227, 2023.
- 700
- 701 Can Xu, Qingfeng Sun, Kai Zheng, Xiubo Geng, Pu Zhao, Jiazhan Feng, Chongyang Tao, and Dixin  
 702 Jiang. Wizardlm: Empowering large language models to follow complex instructions. *arXiv  
 703 preprint arXiv:2304.12244*, 2023.
- 704
- An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li,  
 Dayiheng Liu, Fei Huang, Haoran Wei, et al. Qwen2. 5 technical report. *arXiv preprint  
 705 arXiv:2412.15115*, 2024.

702 Xingcheng Yao, Yanan Zheng, Xiaocong Yang, and Zhilin Yang. Nlp from scratch without large-scale  
703 pretraining: A simple and efficient framework. In *International Conference on Machine Learning*,  
704 pp. 25438–25451. PMLR, 2022.

705  
706 Kun Zhang, Bernhard Schölkopf, Krikamol Muandet, and Zhiqun Wang. Domain adaptation under  
707 target and conditional shift. In *International conference on machine learning*, pp. 819–827. Pmlr,  
708 2013.

709 Han Zhao, Remi Tachet Des Combes, Kun Zhang, and Geoffrey Gordon. On learning invariant  
710 representations for domain adaptation. In *International conference on machine learning*, pp.  
711 7523–7532. PMLR, 2019.

712 Hao Zhao, Maksym Andriushchenko, Francesco Croce, and Nicolas Flammarion. Long is more  
713 for alignment: A simple but tough-to-beat baseline for instruction fine-tuning. *arXiv preprint*  
714 *arXiv:2402.04833*, 2024.

715

716

717

718

719

720

721

722

723

724

725

726

727

728

729

730

731

732

733

734

735

736

737

738

739

740

741

742

743

744

745

746

747

748

749

750

751

752

753

754

755

756 APPENDIX ARRANGEMENT  
757758 The Appendix is arranged as follows:  
759

- 760 • **Section A:** Statement of the Use of Large Language Models (LLMs) in writing.  
 761 • **Section B:** Prompt templates employed in our proposed pipeline.  
 762 • **Section C:** Deeper analysis of the connections between our pipeline and the fields of Domain  
 763 Adaptation (DA) and Learning with Noisy Labels (LNL).  
 764 • **Section D:** Additional experimental results and visualizations.  
 765 • **Section E:** Detailed experimental settings.  
 766

768  
769 A STATEMENT OF THE USE OF LARGE LANGUAGE MODELS  
770771 In this paper, DeepSeek was used solely to check grammar and polish writing style. No scientific  
 772 material or core research content was generated by any Large Language Model (LLM).  
 773774 B LLM PROMPT TEMPLATE  
775776 The following is the prompt template we used in the task-specific label generation in Section 4.1.  
 777778 Prompt Template for LLM Annotation  
779

780 <System Prompt>: You are a helpful assistant.  
 781 <User Prompt>: [Background Information]  
 782 Instruction: [Instruction]  
 783 Input: [Input]  
 784 Response: [Response]  
 785 Please strictly output in JSON format and generate rich, high-quality summaries suitable for LLM.  
 786 Summarize the entire conversation \*\*as a single summary\*\*, not per message or per role.

787 Requirements:  
 788

1. Task: Describe the single most appropriate task type from the background as a short sentence or detailed multi-phrase descriptor (e.g., "Answering open-domain factual questions based on user input").
2. Style: List the top 3 most relevant styles or tones as detailed phrases or brief sentences (e.g., "Uses a formal and professional tone suitable for academic writing").
3. Topic: List 3 key topics covered in the background.
4. Audience: List the top 3 intended audiences.

793 Only return a single JSON object as shown below:  
 794

```
{  

  "Task": "<short sentence or descriptive phrase>",  

  "Style": [  

    "<descriptive phrase>","  

    "<descriptive phrase>","  

    "<descriptive phrase>"  

  ],  

  "Topic": [  

    "<short sentence>","  

    "<short sentence>","  

    "<short sentence>"  

  ],  

  "Audience": [  

    "<descriptive phrase>","  

    "<descriptive phrase>","  

    "<descriptive phrase>"  

  ],  

}
```

810  
811 The following is the prompt template we used in Section 4.3 to score the source sample based on its  
812 assigned anchors.

813     **Prompt Template for Scoring Source Samples**

814

815         **<System Prompt>**: You are a helpful assistant.  
 816         **<User Prompt>**: You are an expert evaluator. Please evaluate the following sample based on  
 817         these criteria:  
 818             – Completeness (1-10): How complete is the response?  
 819             – Information Richness (1-10): How much useful information does it contain?  
 820             – Rarity (1-10): How unique or rare is this type of content?  
 821             – Complexity (1-10): How complex is the task/content?  
 822         Sample to Evaluate: [sample\_format]  
 823         Associated Tags:  
 824             – Task: [Assigned Task anchor]  
 825             – Style tags: [Assigned Style anchor]  
 826             – Topic tags: [Assigned Topic anchor]  
 827             – Audience tags: [Assigned Audience anchor]  
 828         Use these tags as reference points for your scoring decisions. Consider ALL the tags in each  
 829         category when evaluating. Please respond with ONLY a JSON object in this exact format:  
 830         {  
 831             "Completeness": "<1-10>",  
 832             "Information Richness": "<1-10>",  
 833             "Rarity": "<1-10>",  
 834             "Complexity": "<1-10>",  
 835             "Overall Score": "<1-10>"  
 836         }

837     Note that each anchor may contain many domain-specific tags, as the anchors are generated by  
 838     clustering. Therefore, the anchor description in the prompt template above is summarized by LLM,  
 839     which extracts the 20 most representative keywords or key phrases for each anchor.

## 841 C DEEPER ANALYSES OF OUR PIPELINE

844     In this section, we provide answers and explanations for the following two questions:

- 845     1. Why our approach in Section 4.3 can be regarded as a proxy for filtering label noise?  
 846     2. Can our pipeline be applied to scenarios where the target and source domains only partially  
 847         overlap?

849     For the first question, our approach in Section 4.3 can be viewed as a proxy for filtering label noise  
 850     because it mirrors a established technique in learning with noisy labels. In traditional classification,  
 851     models tend to be less confident about mislabeled samples compared to clean samples, resulting in  
 852     a low maximum softmax probability (the "confidence score"). This score is a common signal for  
 853     identifying potential label noise (Hendrycks & Gimpel, 2016). Instead of training a model to obtain a  
 854     confidence score, we leverage the reasoning capability of an LLM. The LLM assesses a sample's  
 855     relevance to its most similar anchors and assigns a score, which functions as a semantics-aware  
 856     confidence score. Thus, our method provides a direct analog to confidence-based filtering without the  
 857     need for model training.

858     For the second question, we use Figure 4 to demonstrate our pipeline's applicability to scenarios  
 859     where the target and source domains only partially overlap.

- 860
- 861         • In Figure 4 (a), the support of the target domain is a subset of the source domain. Our  
 862         OOD filtering step first produces a filtered source dataset with a support that approximately  
 863         matches the target. Subsequent incremental sampling then aligns the distribution of the  
 864         selected dataset with the target distribution.

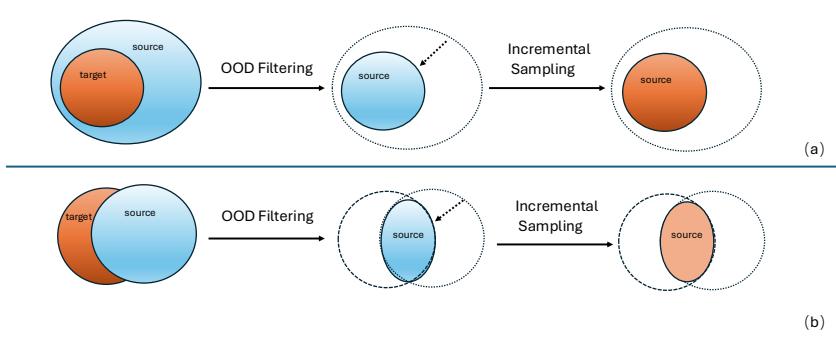


Figure 4: Illustration of how our pipeline aligns the target distribution by sampling from the source. The upper figure (a) shows the case where the support of the target distribution is a subset of the source support. The lower figure (b) shows the case of partial overlap between the target and source distributions.

Table 7: Performance comparison of different data selection approaches for fine-tuning the Mistral-7B-v0.3 model. Each approach selects 10K samples. For each benchmark, top-performing score is shown in bold, while second-best score is underlined.

Methods	MMLU	TruthfulQA	GSM	BBH	TyDiQA
Vanilla Base Model	59.7	30.0	38.0	49.1	54.4
LESS	59.6	32.0	38.5	51.9	<b>59.2</b>
TSDS	59.4	<b>37.0</b>	41.5	<b>57.4</b>	<u>56.4</u>
Ours (Align-topic)	<u>60.3</u>	35.4	<u>44.0</u>	52.8	53.9
Ours (Align-task)	<b>60.4</b>	<u>36.6</u>	<b>46.0</b>	<u>53.0</u>	54.2
Ours (Align-style)	60.0	34.3	40.5	51.0	55.3
Ours (Align-audience)	60.3	35.1	43.5	51.5	55.1

- In Figure 4 (b), after OOD filtering, the support of the filtered source dataset becomes a subset of the target domain. Incremental sampling then matches the distribution of the selected data to the distribution of the partial target (i.e., the shared portion), confirming that our pipeline successfully selects task-relevant samples.

Notably, the scenario in (b) corresponds to the open-set Domain Adaptation (DA) setting. Similar to traditional open-set DA methods (Panareda Busto & Gall, 2017; Liu et al., 2019), which focus on aligning shared classes while rejecting unknown ones, our pipeline inherently handles this case by filtering and aligning only the overlapping regions.

## D ADDITIONAL EXPERIMENTAL RESULTS AND EXAMPLES.

### D.1 FINETUNING ON OTHER FOUNDATION MODELS

In the main paper, we fine-tune the Llama-3.1-8B model on our selected data. In this section, we instead fine-tune Mistral-7B-v0.3 to demonstrate the generalizability of our selection pipeline. The results in Table 7 demonstrate that our method achieves competitive performance on the Mistral model. Specifically, Align-task achieves top-1 or top-2 performance across all benchmarks except TyDiQA.

### D.2 EXAMINING THE LLM ANNOTATION QUALITY

We conduct a thorough evaluation of the quality of proxy labels generated by the LLM, assessing both precision (whether a label correctly describes its corresponding instruction) and consistency (whether the same label is applied to semantically similar instructions). Our examining methodology follows the framework established by Lu et al. (2023).

- 918  
 919  
 920  
 921  
 922  
 923  
 924  
 925
- **For precision evaluation:** We randomly sampled 500 instruction-label pairs. GPT-4 was asked to assess if the label adequately describes the instruction. From these, the first 50 pairs were selected for human evaluation by three independent annotators.
  - **For consistency evaluation:** We randomly sampled 500 labels, ensuring each label was associated with at least two distinct instructions. GPT-4 was asked to determine if the instructions for a given label were semantically consistent. Similarly, the first 50 sets of instructions were assessed by three human annotators.

926 We calculated the agreement between raters using Cohen’s Kappa (for pairwise agreement between  
 927 human and GPT-4) and Fleiss’ Kappa (for inter-annotator agreement among humans). The results are  
 928 presented in Table 8.

929  
 930 Table 8: Annotation Quality Comparison

931  
 932  
 933  
 934

Metric	GPT-4 Annotation	Human Annotation	Agreement	
			Human-Human	Human-GPT
Tag Precision	0.958	0.94	0.4823	0.7899
Tag Consistency	0.856	0.9	0.73	0.736

935 As shown in Table 8, both GPT-4 and human evaluations indicate high precision and consistency  
 936 for the proxy labels. The Human-GPT agreement scores (exceeding 0.7) indicate solid alignment  
 937 between human and LLM judgments. We have uploaded the GPT and human evaluation scores, along  
 938 with sampled instructions and labels, to our anonymized code repository: <https://anonymous.4open.science/r/TADS-B14D/README.md>.

### 941 D.3 LLM SCORE CONSISTENCY

943 To assess the reliability of LLM-based OOD filtering, we evaluate score consistency in two ways:

- 945  
 946  
 947  
 948  
 949  
 950
- **Intra-consistency:** We instruct Llama-3-8B-Instruct to score the same set of source samples twice, using different random seeds. The consistency between the two scoring rounds is measured using Spearman’s rank correlation (range: -1 to 1).
  - **Inter-consistency:** We compare scores generated independently by Llama-3-8B-Instruct and Qwen-2.5-7B-Instruct on the same samples, again using Spearman’s correlation to quantify agreement.

951 After calculation, the Intra-consistency is 0.72, and the Inter-consistency is 0.65, indicating LLM-  
 952 based OOD scoring is reasonably stable across repeated runs (intra-consistency) and aligned between  
 953 different models (inter-consistency).

### 955 D.4 ROBUSTNESS TO LABEL NOISE

957 While Table 8 confirms the high quality of our proxy labels, we further evaluate the robustness of our  
 958 pipeline to label noise through a controlled experiment. Inspired by techniques in *learning with noisy*  
 959 *labels*, we inject synthetic noise by replacing 20% of the original LLM annotated tags in the MMLU,  
 960 TruthfulQA, BBH, and TyDiQA datasets with randomly selected tags from the GSM dataset. This  
 961 allows precise control over the noise level while preserving the label distribution’s structure.

962 We focus our evaluation on MMLU, BBH, and TyDiQA due to their larger test sets, which provide  
 963 statistically reliable performance measurements on noisy labels. As shown in Table 9, introducing  
 964 20% label noise leads to only marginal performance drops, demonstrating the pipeline’s robustness.  
 965 We attribute this resilience to our anchor-based propagation design: rather than propagating labels  
 966 directly, we first cluster proxy labels to form stable anchors. This clustering step effectively averages  
 967 out the impact of individual noisy labels, making the propagation process inherently more robust.

### 968 D.5 IMPACT OF ANNOTATION MODEL CHOICE

969 To evaluate the sensitivity of our pipeline to the choice of annotation model, we conduct an experiment  
 970 where we vary the LLM used for generating proxy labels while holding all other hyper-parameters

972 Table 9: Performance of Selected Data Under Label Noise (Averaged Across Label Domains)  
973

Label noise ratio	MMLU	BBH	TyDiQA
0% label noise	64.3	60.9	59.0
20% label noise	64.1	60.7	58.5

978  
979 constant. Specifically, we compare Llama-3-8B-Instruct and Qwen-2-7B-Instruct as annotators,  
980 finetuning the same target model (Llama-3-8B) in both cases.  
981982 As shown in Table 10, the performance across most datasets (MMLU, TruthfulQA, GSM, and BBH)  
983 remains stable ( $< \pm 1.5$  points difference) when switching annotators. The sole exception is TyDiQA,  
984 where performance drops by 6.6 points with Qwen-2.5 annotations. This suggests that: (1) Our  
985 pipeline is largely robust to the annotator model family. (2) Language-specific biases (e.g., Qwen-2's  
986 Chinese pretraining) may impact performance on certain tasks like TyDiQA (multilingual QA).  
987

988 Table 10: Performance with Different Annotation Models (Finetuned on Llama-3.1-8B)

Methods	MMLU	TruthfulQA	GSM	BBH	TyDiQA
Annotation: Llama 3.1-8B instruct	63.5	38.3	54.5	61.9	64.6
Finetuning: Llama3.1-8B					
Annotation: Qwen2.5-7B instruct	65.1	39.2	55.0	60.6	58.0
Finetuning: Llama3.1-8B					

## 995 D.6 PERFORMANCE IN LOW-DATA REGIMES

996 To evaluate our pipeline's effectiveness when source data is scarce, we simulate a low-resource  
997 scenario by subsampling the original 300K source pool to 5K samples, and selecting 1K samples  
998 from this subset for fine-tuning.  
9991000 As shown in Table 11, our method achieves consistent improvements over random selection, with  
1001 particularly striking gains on TruthfulQA. This aligns with our analysis in Figure 4, demonstrating  
1002 that our pipeline succeeds even when source-target only partial overlaps.  
1003

1004 Table 11: Performance with 5K Source Samples (Selected: 1K)

Methods	MMLU	TruthfulQA	GSM	BBH	TyDiQA
Random selection	64.3	3.5	49.5	56.7	58.9
Full-data finetuning	64.9	3.7	56.5	61.7	59.3
Our pipeline	64.5	21.4	50.5	58.2	59.0

## 1012 D.7 MULTI-DOMAIN LABEL MATCHING

1013 While our main pipeline matches samples based on a single label domain (e.g., topic or task), we  
1014 propose an extension to jointly optimize across multiple domains. The procedure is as follows:  
1015

- 1016
- **Domain-Specific Sampling:** For each label domain (topic, style, task), independently  
1017 sample 10K candidate samples using incremental sampling.
  - **Vote Aggregation:** Count how many times each sample appears across domains (each  
1018 domain's selection counts as one vote).
  - **Ranked Selection:** Sort samples by vote count and select the top 10K highest-voted samples  
1019 as the final subset.

1020 Table 12 shows that joint matching achieves surprisingly high performance for BBH and TyDiQA,  
1021 suggesting its potential. Since we give equal weights to the samples selected from each label domain,  
1022 altering the weights or optimizing them may yield better performance.  
1023

1026

Table 12: Performance of Single-Domain vs. Joint Multi-Domain Matching

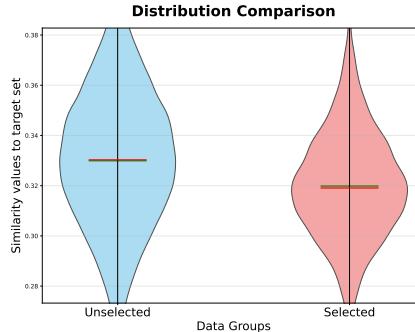
1027

1028

Methods	MMLU	TruthfulQA	GSM	BBH	TyDiQA
match topic	64.6	46.4	57.0	60.0	59.3
match style	64.1	47.3	57.0	59.2	58.7
match task	64.2	44.6	58.0	58.2	58.2
joint-match	63.2	40.0	55.0	62.8	64.0

1033

1034



1035

1036

1037

1038

1039

1040

1041

1042

1043

1044

1045

1046

1047

Figure 5: Distribution of cosine similarities to the target set for selected vs. unselected samples. Lower similarity indicates less representative samples.

1048

1049

1050

#### D.8 DISTRIBUTION ANALYSIS: SELECTED VS. UNSELECTED SAMPLES

1051

1052

1053

1054

To characterize how our pipeline distinguishes between selected and unselected samples, we compare their embedding distributions relative to the target set. For each sample in the source, we compute the average cosine similarity between its embedding and the target set embedding. Lower similarity indicates samples that are less representative of the target distribution or potentially harder examples.

1055

1056

1057

1058

As shown in Figure 5, the selected set contains fewer samples in the high-similarity region compared to the unselected set. This suggests that our method does not simply prioritize "easy" or highly representative samples. Instead, it likely balances representativeness with diversity or hardness, potentially selecting challenging examples that promote robust learning.

1059

1060

1061

#### D.9 COMPUTATIONAL EFFICIENCY ANALYSIS

1062

1063

1064

Although our pipeline involves multiple stages, it is computationally efficient compared to gradient-based baselines. We benchmark the running time on an A100 GPU against two prominent methods: LESS (Xia et al., 2024) and TSDS (Liu et al., 2024). Results are summarized in Table 13.

1065

1066

Table 13: Running Time Comparison

1067

1068

1069

1070

1071

1072

1073

Methods	LESS	TSDS	Our pipeline
Stage-1	Lora-training (6h)	Lora-training (6h)	Target annotation(1h)
Stage-2	Gradient computation (51h)	Gradient computation (51h)	Clustering and propagation(4h)
Stage-3	Data-selection (1 min)	KNN-KDE Data-selection (1h)	OOD filtering (11h)
Stage-4			incremental-sampling (3min)
Finetuning time	Llama-3.1-8B (3h)	Llama-3.1-8B (3h)	Llama-3.1-8B (3h)
Total running time	60 h	61h	19h

1074

1075

1076

1077

1078

LESS is bottlenecked by gradient computation, as it requires model-specific gradients from the same architecture used for fine-tuning, and our reported runtime aligns with the original LESS paper. Similarly, TSDS inherits this substantial computational overhead by reusing the gradients computed by LESS, as acknowledged in Section 5.1 of the TSDS paper.

1079

In contrast, our method avoids expensive backward passes through large LLMs. The core of our efficiency comes from using the lightweight BGE-M3 model to extract embeddings for the entire

1080 source set. This step, which is essential for enabling the subsequent label propagation, takes only 4  
 1081 hours. Target set annotation is fast due to its small scale. The overall pipeline, including clustering (4  
 1082 minutes), propagation (4 hours), OOD filtering (11 hours), and incremental sampling (3 minutes),  
 1083 results in a total runtime of just 19 hours. Furthermore, by decoupling embedding extraction from the  
 1084 fine-tuning model, our approach remains model-agnostic, offering greater flexibility and scalability.  
 1085  
 1086  
 1087  
 1088  
 1089

#### 1090 D.10 EXAMPLES FOR PROXY-LABEL GENERATION BY LLM

1091  
 1092 In the following, we provide examples for the proxy-label generation by LLM.  
 1093  
 1094  
 1095  
 1096  
 1097  
 1098

##### 1099 Target Samples labeling

1100  
 1101 **Instruction:** "What would happen if you were struck by a penny dropped from the top of the Empire  
 1102 State Building?"

1103 **Answer:** "You would feel a light impact if you were struck by a penny dropped from the Empire State  
 1104 Building"

1105 **Labeled Tags:**

- 1106 • **Task:** Calculating the risk and outcome of being hit by a falling coin
- 1107 • **Topic:** Impact force calculation; Physics principles related to free-falling objects; Urban legends and  
 real-world consequences
- 1108 • **Style:** Provides clear and concise information without unnecessary elaboration; Maintains a scientific  
 and educational tone to ensure accuracy; Addresses common misconceptions about physics in a  
 straightforward manner
- 1109 • **Audience:** General public curious about science; Students learning basic physics concepts; People  
 who enjoy reading about urban myths and their debunkings

1110  
 1111 **Instruction:** According to the Bible, what forbidden fruit did Adam and Eve eat in the Garden of Eden?  
 1112 **Answer:** The Bible doesn't specify what kind of fruit Adam and Eve ate

1113 **Labeled Tags:**

- 1114 • **Task:** Answering open-domain factual questions based on user input
- 1115 • **Topic:** Biblical accounts and religious texts; Interpretations of sacred narratives; Faith-based inquiries  
 and discussions
- 1116 • **Style:** Provides clear and concise answers without unnecessary embellishment; Adopts a neutral and  
 informative tone to ensure accuracy; Includes specific references where applicable
- 1117 • **Audience:** Religious scholars and theologians; Laypeople curious about biblical stories; Students  
 studying comparative religion or theology

1118  
 1119 **Instruction:** How long should you wait between eating a meal and going swimming?

1120 **Answer:** "You do not need to wait between eating and swimming."

1121 **Labeled Tags:**

- 1122 • **Task:** Providing simple health-related advice based on common knowledge"
- 1123 • **Topic:** Eating habits and their effects on physical activities; Safety guidelines during recreational  
 water sports; Common misconceptions about exercise and nutrition timing
- 1124 • **Style:** Factual and concise; Direct and informative; User-friendly
- 1125 • **Audience:** General public including individuals who enjoy swimming; Parents considering safety  
 measures for children around pools; Health-conscious adults looking to maintain fitness while  
 engaging in leisure activities

1134  
1135

## Target Samples labeling

1136  
1137**Instruction:** "What color is the sun when viewed from space?"**Answer:** "The sun is white when viewed from space"**Labeled Tags:**

- **Task:** Explaining basic astronomical facts
- **Topic:** Properties of celestial bodies like the Sun; Observations made from space; Basic astrophysical knowledge accessible to general audiences
- **Style:** Provides clear and concise answers to simple queries; Maintains a neutral yet informative tone; Offers straightforward explanations without unnecessary elaboration
- **Audience:** General public with minimal scientific background; Students at middle school or early high school levels; Individuals curious about astronomy but lacking specialized knowledge

1146

**Instruction:** "What is a Wendigo?"**Answer:** "A mythical man-eating monster or evil spirit native to the northern forests of the Atlantic Coast and Great Lakes Region of the United States and Canada."**Labeled Tags:**

- **Task:** Describing a mythological creature from Algonquin folklore
- **Topic:** Mythical creatures in Native American folklore; Cultural beliefs regarding supernatural beings; Historical and social associations of wendigos
- **Style:** Provides concise definitions supported by contextual information; Maintains an informative yet engaging narrative flow; Includes clear and precise descriptions
- **Audience:** People interested in world mythology; Students studying North American indigenous cultures; Readers curious about folkloric traditions

1157

**Instruction:** "I have a car, a fridge, an oven, a toaster, a couch, three lamps, five chairs, a stove, and a microwave. How many objects do I have?"**Answer:** "15"**Labeled Tags:**

- **Task:** Counting items within a given list
- **Topic:** Enumerative counting and classification of physical objects; Identification of different categories such as fruits and vegetables; Object enumeration in various contexts like household items or food
- **Style:** Simple and straightforward dialogue between a knowledgeable model and a curious user; Casual and engaging interaction aimed at providing clear answers; Accurate and concise responses focused on numerical information
- **Audience:** General users seeking quick and accurate answers about item counts; Students learning basic categorization skills; Anyone needing assistance in identifying categories within lists

1160

**Instruction:** "Defendant was on trial for robbery. Defendant's direct examination continued until late in the day when it finally concluded. At this time, the trial court judge adjourned the proceedings for the evening. The judge then instructed Defendant not to speak with anyone during the night and scheduled cross-examination to begin in the morning. The judge's instruction to Defendant was A. proper, because a judge has broad discretion to instruct witnesses in such a manner. B. proper, because it would have the same effect as permitting cross-examination to continue after direct was concluded. C. improper, because it violates the defendant's Sixth Amendment right to counsel. D. improper, because it violates the attorney-client privilege."**Answer:** "C"**Labeled Tags:**

- **Task:** Evaluating legal instructions given by judges regarding witness behavior
- **Topic:** Judicial discretion in courtroom procedures; Protection of defendants' rights under criminal procedure laws; Sixth Amendment protections against self-incrimination
- **Style:** Formal and precise for accurate legal analysis; Clear and concise for easy understanding; Objective and neutral without bias
- **Audience:** Legal professionals familiar with courtroom practices; Law students studying procedural law; Defendants and their attorneys navigating criminal trials

1184  
1185  
1186  
1187

1188 Since we assign the most similar anchors to each source sample, we provide an example below to  
 1189 verify that our anchors indeed align well with the source samples. This illustrates that our designed  
 1190 label field carries meaningful semantics for the source samples, where we show the most similar tag  
 1191 from the anchor.

1192

1193

### Example of Label Propogation

1194

1195

1196

1197

1198

1199

1200

1201

1202

1203

1204

1205

1206

1207

1208

1209

1210

1211

1212

1213

1214

1215

1216

1217

1218

1219

1220

1221

1222

1223

1224

1225

1226

1227

1228

**Instruction:** "How will advances in technology and the evolution of human consciousness shape the future of psychotherapy, and how can therapists adapt to these changes to provide optimal care for their clients?"

**Answer:** "Advances in technology are already transforming the field of psychotherapy, and their impact is likely to increase in the future. One significant trend is the increasing use of online therapy, which allows clients to access therapy from the comfort of their own homes and over long distances. This has the potential to increase access to therapy for people who might not otherwise have the opportunity to receive it. Another important development is the use of virtual reality in therapy. This technology can be used to create virtual environments that simulate real-life situations, allowing clients to confront and work through challenging scenarios in a safe and controlled setting. This technology has already been used to treat anxiety disorders and phobias, and it is likely to be used for a range of other mental health conditions in the future. Advances in neuroscience are also likely to have a significant impact on the field of psychotherapy. As we gain a better understanding of how the brain works, therapists will be better equipped to develop interventions that target specific neural pathways and processes. This could lead to more effective and efficient therapies that produce better outcomes for clients. Finally, the evolving consciousness of society will continue to influence the field of psychotherapy. As people become more aware of the importance of mental health and seek out therapy as a means of personal growth and self-discovery, therapists will need to adapt to meet their clients' changing needs. This may involve incorporating mindfulness practices, somatic therapies, and other holistic approaches into their practice. To provide optimal care for their clients in the face of these changes, therapists will need to be open to new ideas and approaches, and willing to embrace technology as a tool for delivering therapy. They will also need to stay up-to-date with the latest research and developments in the field, and be willing to adapt their practice as new evidence emerges. Above all, they will need to remain committed to providing the best possible care for their clients, using all available resources to help them achieve their goals."

#### Propagated Tags:

- *Task*: Explaining psychological concepts
- *Topic*: Therapeutic techniques in psychology
- *Style*: Formal and concise communication suitable for educational purposes
- *Audience*: Students and professionals in psychology and related fields

## E DETAILED EXPERIMENTAL SETTINGS

The hyperparameter settings are detailed in Table 14. All instruction tuning experiments are conducted on a 2-GPU node equipped with **NVIDIA H20** GPUs (96GB memory each, NVLink interconnect), running CUDA 12.7 and driver version 565.57.01. The host machine is configured with a **Xeon Platinum 8457C** 40-core CPU, **400GB** of RAM. The computing infrastructure is provided by the cloud platform **AutoDL**.

Our method includes several components, among which LLM-based annotation, embedding and clustering, and fine-tuning with evaluation—account for the majority of runtime cost.

*LLM Annotation* is conducted on the same hardware as training using the **VLLM** (anda Zhuohan Li et al., 2023) inference engine with a batch size of 512. The annotation process takes approximately 4.5 hours. In practice, LLM annotation may occasionally fail for some samples; thus, we apply a post-pass collection and re-annotation step until all examples are successfully labeled.

*Embedding and Clustering* is performed on a single H20 GPU, with only half of the CPU cores allocated. We use a batch size of 256 and actively clear GPU cache using `empty-cache` to manage

Table 14: Hyperparameters used in SFT.

Hyperparameter	Value
Precision	bf16
LoRA Rank	64
LoRA Alpha	16
LoRA Dropout	0.1
Max Sequence Length	2048
Training Epochs	3
Optimizer	AdamW
AdamW betas	(0.9, 0.999)
AdamW eps	1e-8
Learning Rate	$1 \times 10^{-4}$
LR Scheduler	Linear
Warmup Ratio	0.03
Weight Decay	0
Batch Size per GPU	1
Total Batch Size	128
Gradient Accumulation Steps	128
Random Seed	42
Preprocessing Workers	16

memory. The peak memory usage ranges from 60GB to 70GB, and the total embedding process takes about 4 hours. Both the annotation and embedding steps are one-time preprocessing costs that can be reused via intermediate file caching.

*Training and Evaluation* is conducted using the accelerate (Gugger et al., 2022) framework with LoRA (Hu et al., 2022). Each GPU consumes approximately 70GB of memory during training. On a dataset with 10K samples, the fine-tuning step typically completes in around 2 hours using 2 GPUs. Evaluation is most time-consuming on the MMLU benchmark, which takes roughly 20 minutes. To optimize utilization, we allocate one GPU exclusively for MMLU evaluation and run the remaining benchmarks sequentially on the second GPU, completing all evaluations within 20 minutes.