InfiAlign: A Scalable and Sample-Efficient Framework for Aligning LLMs to Enhance Reasoning Capabilities

Shuo Cai[1], **Su Lu**[2], **Qi Zhou**[2], **Kejing Yang**[1], **Zhijie Sang**[1], **Congkai Xie**[1], **Hongxia Yang**[*12]

[1]InfiX.ai [2]The Hong Kong Polytechnic University
hongxia.yang@polyu.edu.hk

Abstract

Large language models (LLMs) have exhibited impressive reasoning abilities on a wide range of complex tasks. However, enhancing these capabilities through post-training remains resource intensive, particularly in terms of data and computational cost. Although recent efforts have sought to improve sample efficiency through selective data curation, existing methods often rely on heuristic or task-specific strategies that hinder scalability. In this work, we introduce **InfiAlign**, a scalable and sample-efficient post-training framework that integrates supervised fine-tuning (SFT) with Direct Preference Optimization (DPO) to align LLMs for enhanced reasoning. At the core of InfiAlign is a robust data selection pipeline that automatically curates high-quality alignment data from open-source reasoning datasets using multidimensional quality metrics. This pipeline enables significant performance gains while drastically reducing data requirements and remains extensible to new data sources. When applied to the Qwen2.5-Math-7B-Base model, our SFT model achieves performance on par with DeepSeek-R1-Distill-Qwen-7B, while using only approximately 12% of the training data, and demonstrates strong generalization across diverse reasoning tasks. Additional improvements are obtained through the application of DPO, with particularly notable gains in mathematical reasoning tasks. The model achieves an average improvement of 3.89% on AIME 24/25 benchmarks. Our results highlight the effectiveness of combining principled data selection with full-stage post-training, offering a practical solution for aligning large reasoning models in a scalable and data-efficient manner. The model checkpoints are available at https://huggingface.co/InfiX-ai/InfiAlign-Qwen-7B-SFT.

1 Introduction

Large language models (LLMs) have demonstrated strong performance across a wide range of reasoning tasks, including mathematics, science, and programming. Post-training methods such as supervised fine-tuning (SFT) and reinforcement learning (RL)—often referred to as the alignment stage in LLM development—can further enhance reasoning capabilities, but they remain computationally expensive and data-intensive. These challenges are especially pronounced in chain-of-thought (CoT) reasoning (Wei et al. 2022), which requires high-quality, domain-specific instruction data that are costly to curate and difficult to scale.

To address this, recent research has explored improving sample efficiency through selective data curation. Approaches such as model-based scoring (Chen et al. 2024; Ge et al. 2024), gradient-driven clustering (Zhang et al. 2025; Xia et al. 2024; Pan et al. 2024), and embedding-based filtering (Bukharin et al. 2024; Wu et al. 2023) have shown promising results. For domain-specific reasoning, multi-criteria selection methods like LIMO (Ye et al. 2025) and s1 (Muennighoff et al. 2025) demonstrate that carefully curated, small-scale datasets—guided by factors such as difficulty, diversity, and generality—can yield substantial performance gains. Additionally, entropy-based compression techniques (Yin et al. 2024) aim to retain data diversity while reducing redundancy. However, many existing pipelines still suffer from critical limitations: they often rely on handcrafted heuristics (e.g., keyword filters or fixed scoring rules) or rigid teacher-student distillation schemes that lack generalization across tasks and domains (Li et al. 2024). Moreover, these frameworks frequently require extensive manual effort or are tailored to specific domains, making them difficult to scale or adapt to new data sources. Such issues hinder the development of unified, automated, and broadly applicable alignment strategies for reasoning tasks.

In this work, we introduce **InfiAlign**, a unified and scalable post-training framework for aligning LLMs on reasoning tasks with high sample efficiency. InfiAlign integrates SFT and Direct Preference Optimization (DPO) (Rafailov et al. 2023), built upon a robust data selection pipeline that automatically identifies high-quality alignment data from large open-source corpora using multi-dimensional metrics—capturing diversity, difficulty, and quality. Applied to the Qwen2.5-Math-7B-Base model, InfiAlign matches the performance of DeepSeek-R1-Distill-Qwen-7B while using only 20% of the training data. Additional improvements are obtained through the application of DPO, with particularly notable gains in mathematical reasoning tasks. The model achieves an average improvement of 3.89% on the AIME 2024 and AIME 2025 benchmarks. These results underscore the effectiveness of principled data selection and multi-stage alignment in enhancing LLM reasoning capabilities efficiently.

Our main contributions are as follows:

[*]Corresponding author.

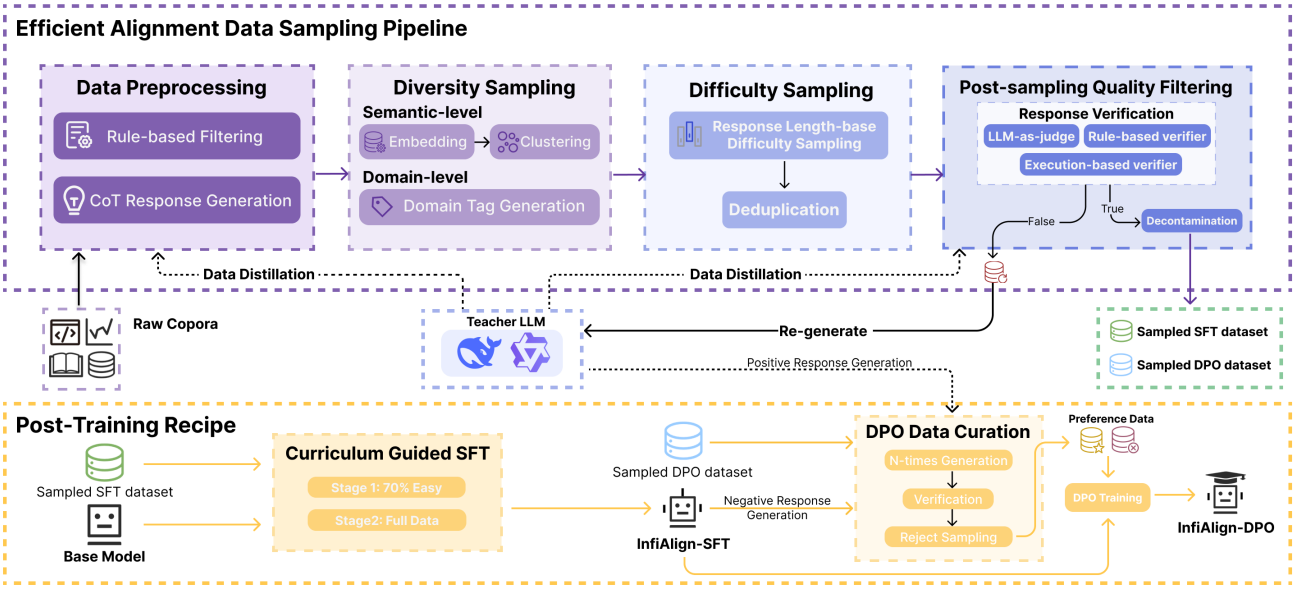

Figure 1: **Overview of the InfiAlign Framework** The InfiAlign framework combines an efficient data sampling pipeline with a modular post-training strategy. The pipeline includes rule-based filtering, CoT distillation, diversity-aware sampling, and difficulty control via response length. The post-sampling quality filtering module applies both rule-based and LLM-based scoring. Post-training consists of a curriculum-guided SFT phase followed by a preference-based DPO stage. This framework enables scalable and automated generation of high-quality, domain-diverse alignment data.

- **Data-Efficient Alignment via Multi-Dimensional Filtering.** We design an automated pipeline that selects high-quality instruction data from open-source corpora using diversity, difficulty, and quality metrics, achieving strong performance with only ∼20% of the data used by distilled baselines.
- **Modular and Scalable Framework.** InfiAlign enables seamless integration of new data sources and tasks via its modular design, allowing flexible and low-overhead adaptation across domains.
- **Enhanced Reasoning through Multi-Stage Training.** We adopt a multi-stage training regimen that balances data mixing, curriculum-guided SFT, and DPO to boost reasoning across various benchmarks.

2 Related Work

Recent advances in post-training have largely relied on data-intensive strategies such as SFT and RL to align LLMs for complex reasoning tasks. Many efforts construct reasoning datasets via distillation from stronger teacher models (e.g., QwQ (Qwen-Team 2025), DeepSeek-R1 (DeepSeek-AI 2025)), yielding models like DeepSeek-R1-Distill-Qwen and Light-R1 (Wen et al. 2025) that demonstrate strong downstream performance. However, these approaches often depend on heuristic or task-specific data collection pipelines, limiting their scalability and general applicability.

Several works (e.g., LIMO, s1) emphasize quality-over-quantity curation, showing that small yet carefully selected examples can be effective for reasoning supervision. Nonetheless, such efforts are either domain-specific or manually intensive, and do not scale well to broader alignment settings or new data sources. Beyond SFT, recent applications of DPO and other RL-based methods (e.g., AceReason (Chen et al. 2025), Skywork-OR1 (He et al. 2025)) further refine alignment, but do not prioritize generalizable data pipelines.

In contrast, our work proposes **InfiAlign**, a scalable and data-efficient post-training framework that integrates SFT and DPO under a unified and extensible data selection pipeline. By leveraging multi-dimensional quality metrics, our method enables high-quality alignment with minimal data, achieving competitive performance using substantially less training data compared to strong baselines. This framework provides a practical and generalizable foundation for future work on reasoning alignment, with potential to benefit researchers through more efficient model development at scale.

3 InfiAlign: Scalable and Efficient Post-training for Reasoning

We propose **InfiAlign**, a novel post-training framework that enhances the reasoning capabilities of large language models using minimal data. It integrates three core components: a **scalable data sampling pipeline** that efficiently selects a small yet high-quality subset of data by jointly considering diversity and difficulty, a **balanced SFT strategy** based on cross-domain data mixing for robust generalization, and a **data-efficient DPO recipe** that further strengthens rea-

soning capability. Together, these components enable strong performance with substantially reduced data and computational resources.

3.1 Efficient Alignment Data Sampling Pipeline

We introduce a scalable data pipeline for constructing high-quality QA pairs with controlled diversity and difficulty (see Figure 1). It consists of four components: (1) **Data Collection and Preprocessing**, which standardizes and optionally augments QA pairs with CoT reasoning; (2) **Diversity Sampling**, leveraging topic annotation and semantic clustering to ensure broad coverage; (3) **Difficulty Sampling**, which selects complex examples based on response characteristics; and (4) **Post-sampling Quality Filtering**, applying rule-based checks, sandbox verification, and LLM scoring. The resulting dataset is well-suited for alignment and distillation, especially for enhancing the reasoning abilities of small and medium language models.

Data Collection and Preprocessing Alignment data is primarily collected from large-scale open-source reasoning datasets, with the flexibility to incorporate domain-specific or proprietary sources as needed. All data are formatted into QA pairs to support instruction alignment. For queries lacking CoT reasoning traces, we generate responses using advanced models such as DeepSeek-Distill and Qwen3. Prior work demonstrates that such distillation effectively transfers reasoning abilities from larger models to smaller ones, enhancing alignment performance (Shridhar, Stolfo, and Sachan 2022; Xu et al. 2024).

We begin with rule-based filtering to remove non-English or incomplete QA pairs that may introduce noise. The filtered data are then processed by sampling modules to ensure broad coverage across query types and difficulty levels.

Diversity Sampling To construct a high-quality alignment corpus that supports robust generalization and compositional reasoning, we introduce a dual-granularity diversity sampling strategy. This approach integrates both domain-level and semantic-level signals to capture topical breadth and latent linguistic diversity across QA instances.

- **Domain-Level Sampling:** We begin by assigning domain-specific labels to each question using a prompting-based LLM classifier (see Appendix A). For structured domains such as mathematics and programming, we further decompose the hierarchy into fine-grained subcategories (e.g., *Algebra*, *Geometry*; *Greedy Search*, *Dynamic Programming*).

  Sampling is conducted in a category-balanced manner to avoid skewed distributions and promote balanced domain coverage.
- **Semantic-Level Sampling.** To promote diversity in the latent semantic space, we encode all questions into dense embeddings using a pretrained sentence encoder (e.g., `Alibaba-NLP/gte-base-en-v1.5` (Zhang et al. 2024)). We apply unsupervised clustering (e.g., K-means (Ahmed, Seraj, and Islam 2020)) over the embedding space and sample uniformly across clusters. This latent-space sampling strategy captures the variation in underlying semantics beyond the surface form, complementing domain-level sampling.

To finalize the candidate pool, sampling is performed independently at both levels and the results are merged.

Deduplication is then applied using n-gram overlap matching ($n = 20$), ensuring that samples containing common instructional templates shared across datasets are preserved and not erroneously discarded. Together, these complementary views synergistically enhance diversity across both topical and semantic dimensions, enabling the construction of heterogeneous and representative alignment corpora.

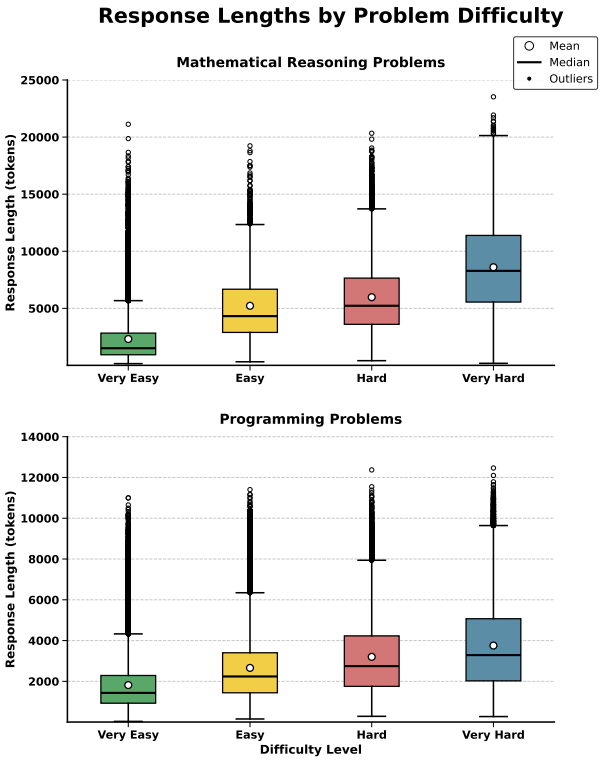

Figure 2: Response lengths increase with problem difficulty across both mathematical and programming domains. Box plots illustrate the distribution of response lengths (in tokens) across four difficulty levels for two problem categories. For both domains, higher difficulty is associated with longer responses, with mathematical problems exhibiting greater variance and heavier tails. This trend suggests that response length can serve as a coarse proxy for reasoning complexity in alignment data.

Difficulty Sampling Figure 2 presents a box plot of response lengths across four difficulty levels in math and programming tasks. Across both domains, we observe a clear positive correlation between task difficulty and model output length, consistent with the findings of OpenCodeReasoning (Ahmad et al. 2025). This empirical trend supports our use of response length as a scalable and domain-agnostic difficulty proxy. Unlike traditional pass@k-based difficulty esti-

mation, which requires costly inference with oracle models, length-based sampling provides a practical alternative that generalizes across symbolic and semi-structured domains. In practice, we prioritize longer responses within each semantic or topical cluster, preserving both difficulty and diversity. This strategy improves reasoning power and generalization in downstream alignment, especially for complex tasks.

Post-sampling Quality Filtering After sampling, we conduct a final quality control phase to ensure that only well-structured, accurate, and reliable QA pairs are retained for alignment training.

We begin with format-level validation to eliminate responses that are incomplete, excessively verbose, or missing critical components—such as final answers enclosed in `\boxed` for mathematical problems. Domain-specific automated verifiers (e.g., `MathVerify`, `Sandbox`) are employed to assess response correctness in tasks with well-defined ground truth, such as mathematics and programming. For responses that fail verification, we invoke an LLM to regenerate the answer using a structured correction template. This verification–regeneration process is iterated up to eight times or until all verification checks are passed. Responses that fail all attempts are discarded.

For open-ended or partially verifiable tasks, we employ LLM-based evaluation protocols to assess question clarity, answer redundancy, and overall informativeness. In cases where the response is ambiguous or the confidence is low, the sample is conservatively discarded to maintain the reliability of the dataset.

Dataset Decontamination To avoid data leakage to evaluation benchmarks, we perform data decontamination. Specifically, we filter out QA pairs that exhibit substantial lexical or semantic overlap with publicly available benchmark datasets. This includes removing examples with high n-gram overlap ($n = 15$) or elevated cosine similarity scores (greater than 0.9) based on sentence embeddings. This procedure helps prevent contamination of the test set and ensures that the evaluation metrics accurately reflect the generalization capabilities of the model.

These post-sampling quality control mechanisms, in conjunction with prior filtering and decontamination steps, ensure that the final alignment corpus is clean, diverse, and robust—suitable for high-quality instruction tuning.

3.2 SFT Data Curation and Training Recipe

Data Sources and Composition To enable sample-efficient alignment via supervised fine-tuning, we curate **InfiAlign-SFT-92K** and **InfiAlign-SFT-165K**—two compact yet high-quality instruction corpus consisting of 95K or 165K reasoning-focused QA pairs. These datasets are constructed from over 10M raw alignment examples drawn from ten open-source corpora, including `OpenThoughts-114K`, `OpenThoughts3-1.2M` (Guha et al. 2025a), `AM-DeepSeek-R1-Distilled-1.4M` (Zhao et al. 2025), `data-ablation-full59K` (Muennighoff et al. 2025), `NuminaMath-CoT` (LI et al. 2024), `OpenCodeReasoning`, `Llama-Nemotron-Post-Training-Dataset` (Bercovich et al. 2025), `Mixture-of-Thoughts` (HuggingFace 2025), and `OpenScience` (NVIDIA 2025). Please refer to Appendix B for data composition and proportion.

To ensure that the resulting dataset is both informative and domain-balanced, we apply the proposed multi-dimensional data selection pipeline, which evaluates samples based on diversity, difficulty, and quality. Empirically, we observe that mathematical and coding tasks exhibit strong transferability and are more sensitive to data scaling, whereas general and domain-specific examples offer diminishing returns under increased volume. Based on these findings, we adopt a domain mixing ratio of **Math:Code:Science = 4:4:3**, prioritizing reasoning-rich tasks while maintaining a broad topical spread.

Two-stage Curriculum Learning To further optimize learning dynamics and mitigate data inefficiency, we adopt a curriculum-inspired two-stage fine-tuning strategy that reflects the hierarchical complexity of reasoning tasks. In the first stage, we train the model on 70% relatively simple data of the data (predominantly math and code instructions) which provide structured and relatively accessible reasoning patterns. This early phase allows the model to acquire foundational reasoning skills in a stable optimization regime.

In the second stage, we expand the training set to the full InfiAlign-SFT-165K corpus by incorporating more diverse and domain-specific instructions, particularly from scientific and open-ended domains. Crucially, we retain first-stage samples in this phase to ensure distributional continuity and avoid catastrophic forgetting. This gradual curriculum enables the model to transition smoothly from well-structured to more open-ended reasoning tasks, leading to improved generalization across domains. Together, the domain-aware data composition and curriculum-based training schedule form a unified and principled strategy for effective reasoning alignment under limited data budgets.

3.3 DPO Data Curation

To further enhance the reasoning capability of our SFT model, we continue training it with DPO, one of the most popular preference optimization method. Given a prompt x and a pair of responses (y_w, y_l), where y_w is the correct answer and y_l is the SFT model's incorrect answer, DPO maximizes the log-likelihood gap between the correct answer and incorrect answer. The objective function of DPO is

$$\mathcal{L}_{\text{DPO}} = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\left[\log\sigma\left(\beta\log\frac{\pi_\theta(y_w \mid x)}{\pi_{\text{ref}}(y_w \mid x)} - \beta\log\frac{\pi_\theta(y_l \mid x)}{\pi_{\text{ref}}(y_l \mid x)}\right)\right] \tag{1}$$

where π_θ is the policy model, π_{ref} is the reference policy, typically the SFT model, σ is the sigmoid function and β controls the deviation from the base reference policy.

To build the DPO training dataset, we leverage `OpenMathReasoning` (Moshkov et al. 2025),

`Mixture-of-Thoughts` and `OpenScience`, which provide QA pairs spanning the math, science and code domains. All samples in these datasets contain verified reasoning solutions generated by powerful reasoning models such as DeepSeek-R1 and QwQ-32B. The DPO data curation pipeline includes:

- **Data Decontamination and Deduplication:** We decontaminate data against evaluation benchmarks and deduplicate samples from the SFT training dataset.
- **Data Selection:** We first utilize Qwen2.5-32B-Instruct model (Qwen-Team 2024) to annotate each sample with domain-specific labels. For each category, we select the problems with the longest solution, representing the most challenging problems. Our SFT model then generates responses for these selected problems, which are used in the subsequent rejection sampling step.
- **Reject Sampling:** We employ the Qwen2.5-32B-Instruct model to evaluate the SFT model's responses to math and science questions, and utilize an internal sandbox service to verify the correctness of code-related answers. For each domain, we select false samples with the longest solution lengths from each category, ensuring a balanced number of samples across categories. Previous work (Wen et al. 2025) has discovered that for challenging problems, using chosen responses from significantly stronger models yielded better results. Therefore, we directly use the solutions (generated by strong models such as DeepSeek-R1) as the positive samples, and pair them with the selected false samples to construct training pairs.

4 Experiment

We conduct comprehensive experiments to evaluate the effectiveness of our alignment data sampling pipeline in producing compact yet powerful instruction-tuned models. We first fine-tune a base model using supervised learning (InfiAlign-SFT-7B), and further apply preference optimization (InfiAlign-DPO-7B), both initialized from `Qwen2.5-Math-7B`.

Despite being trained on a relatively fewer alignment samples compared to other state-of-the-art models, both InfiAlign-SFT-7B and InfiAlign-DPO-7B demonstrate competitive or superior performance on general reasoning, math and code benchmarks.

4.1 InfiAlign-7B Training

We use the datasets InfiAlign-SFT-92k and InfiAlign-SFT-165k to train InfiAlign-Qwen-7B-SFT-92K and InfiAlign-Qwen-7B-SFT-165K, respectively, which are constructed through our proposed aligned data sampling pipeline. We fine-tune `Qwen2.5-Math-7B` using a two-stage SFT schedule. The model is trained for 5 epochs using a batch size of 16 and a learning rate of 1e-5. All training is conducted on 8 NVIDIA H800 GPUs using mixed precision. The two-stage training first emphasizes simpler mathematical and code data before introducing more complex and general-domain examples, consistent with our curriculum-inspired strategy.

We conduct DPO training on both InfiAlign-Qwen-7B-SFT-92K and InfiAlign-Qwen-7B-SFT-165K models. To maintain the same data mixing strategy as used during SFT training, we construct two separate DPO training sets: InfiAlign-DPO-9K (comprising 4k math, 3k code, and 2k science samples) for training InfiAlign-Qwen-7B-SFT-92K model, and InfiAlign-DPO-10K (comprising 3.5k math, 3.5k code, and 3k science samples) for training InfiAlign-Qwen-7B-SFT-165K model.

We utilize 360-LLaMA-Factory framework (Zou et al. 2025) with sequence parallelism to train our DPO model on 16 NVIDIA H800 GPUs with the following settings: epoch as 3, batch size as 16, learning rate as 5e-7, cosine learning rate scheduler, warm-up ratio as 0.1, sequence parallelism as 4. Training minimizes the sigmoid preference loss with β as 0.1.

4.2 Evaluation

Benchmarks We evaluate our models on a diverse set of benchmarks covering four key domains: mathematical reasoning (AIME24/25 (AIME 2024, 2025), MATH500 (Lightman et al. 2023)), code generation (LiveCodeBench (Jain et al. 2024)), general reasoning (MMLU-Pro (Wang et al. 2024)), and scientific QA (GPQA-Diamond (Team et al. 2025)). This suite provides a comprehensive evaluation of both domain-specific and general instruction-following capabilities.

Baselines We compare our approach against multiple strong reasoning baselines, including DeepSeek-Distill-Qwen-7B, OpenThoughts2-7B (Guha et al. 2025b), and Light-R1-7B-DS (Wen et al. 2025). These models are either trained on substantially larger datasets or built upon more powerful base models.

During evaluation, we use a sampling temperature of 0.6 and top-p of 0.95 across all benchmarks. The maximum generation length is set to 32,768 tokens for all tasks. To address variability in reasoning outputs, we report pass@1 performance averaged over multiple runs (denoted as avg@n): $n = 64$ for AIME 24/25, $n = 4$ for MATH500, $n = 8$ for GPQA-Diamond, LiveCodeBench, and $n = 1$ for MMLU-Pro.

4.3 Main Results

Table 1 presents the performance of our models across six representative reasoning benchmarks. InfiAlign-Qwen-7B-SFT-92K achieves an average accuracy of 54.70, matching or slightly exceeding `DeepSeek-Distill-Qwen-7B` (54.43) while using only **12%** of the training data (92K vs. 800K). Notably, it generalizes well to both mathematical (AIME 2025: 43.39 vs. 38.70) and scientific domains (GPQA: 48.48 vs. 47.00), outperforming several baselines trained on substantially larger datasets or with stronger backbones (e.g., OpenThoughts2-7B). These results highlight the effectiveness of our sample-efficient alignment pipeline in achieving strong reasoning generalization under minimal supervision.

To evaluate scalability, we further apply the same sampling pipeline to scale up the training set to 165K QA

Model	Initial CKPT	Data Size	AIME 2025 (avg@64)	AIME 2024 (avg@64)	MATH500 (avg@4)	GPQA Diamond (avg@8)	MMLU-Pro (pass@1)	LCB-v5 (avg@8)	Avg.
Qwen2.5-7B-Instruct	Qwen2.5-7B-Base	1M	8.80	11.93	76.15	38.70	**57.49**	15.77	34.80
Qwen2.5-Math-7B-Instruct	Qwen2.5-7B-Math-Base	2.5M	6.72	6.67	82.40	31.12	43.06	2.68	28.78
DeepSeek-Distill-Qwen-7B	Qwen2.5-7B-Math-Base	800K	37.97	55.50*	92.80*	49.10*	54.16	37.60*	54.43
OpenThinker2-7B	Qwen2.5-7B-Instruct	1M	38.70*	60.70*	87.60*	47.00*	40.60*	37.50	52.01
Light-R1-7B-DS	DeepSeek-Distill-Qwen-7B	3K	44.30*	59.10*	91.35	49.40*	54.95	**38.40**	56.25
InfiAlign-Qwen-7B-SFT-92K	Qwen2.5-7B-Math-Base	92K	43.39	56.46	92.35	48.48	53.51	34.05	54.70
InfiAlign-Qwen-7B-DPO-9K	InfiAlign-Qwen-7B-SFT-92K	9K	44.06	61.04	91.95	48.17	49.90	34.54	54.94
InfiAlign-Qwen-7B-SFT-165K	Qwen2.5-7B-Math-Base	165K	42.19	**63.75**	92.70	**53.60**	56.68	36.20	**57.52**
InfiAlign-Qwen-7B-DPO-10K	InfiAlign-Qwen-7B-SFT-165K	10K	**47.45**	61.25	**93.45**	51.77	53.95	35.30	57.20

Table 1: Main evaluation results of our InfiAlign models on six representative reasoning benchmarks spanning mathematics, code, science, and general knowledge domains. All experiments are conducted under a unified evaluation setup (temperature=0.6, top_p=0.95, max_tokens=32,768). Results marked with * are self-reported by the model developers; the rest are reproduced using the same settings.

pairs. The resulting model, InfiAlign-Qwen-7B-SFT-165K, achieves a higher average accuracy of 57.52, with consistent improvements over the 92K variant across most benchmarks—including +7.29 on AIME 2024, +5.12 on GPQA, and +2.15 on LCB-v5. This upward trend underscores the robustness and scalability of our method, allowing practitioners to balance training cost and performance based on resource availability.

Finally, lightweight preference tuning via DPO could further boost math reasoning ability. On math domain benchmarks, compared to their respective SFT baselines, InfiAlign-Qwen-7B-DPO-9K and InfiAlign-Qwen-7B-DPO-10K achieve average improvements of 1.62% and 1.18%, respectively. Specifically, InfiAlign-Qwen-7B-DPO-9K improves the AIME 2024 score with a +4.58 gain (61.04 vs. 56.46). While InfiAlign-Qwen-7B-DPO-10K achieves 47.45 (+5.26) on AIME 2025 and 93.45 on MATH500, outperforming all baseline models. This highlights the complementary benefits of minimal yet targeted preference data in enhancing reasoning alignment.

4.4 Ablation Studies and Analysis

Ablation Studies on Data Sampling Strategy In this section, we conduct ablation studies to evaluate the impact of different data sampling strategies on the alignment performance of our model. To facilitate reproducibility, we set the random seed to 42 for all experiments.

Effectiveness on General Reasoning. To evaluate the impact of general domain sampling strategies on alignment performance, we performed ablation experiments using fixed subsets of 17.1K QA pairs sampled from the AM-1.4M dataset (Zhao et al. 2025). Models trained on these subsets are evaluated on four representative benchmarks: MATH500, GPQA-Diamond, MMLU-Pro, and the more comprehensive SuperGPQA (Team et al. 2025).

Table 2 presents a comparison of eight sampling strategies, including random selection, length- and complexity-based filtering, and combinations with diversity mechanisms such as semantic-level embeddings, domain-level categorization, and our proposed dual-granularity approach.

Sampling based on **response length** exhibits a strong correlation with enhanced mathematical reasoning, yielding a +7.7 point improvement over random sampling on MATH500 and outperforming all other diversity-driven strategies. This confirms response length as a reliable and efficient proxy for reasoning complexity in symbolic domains. In contrast, **complexity-aware sampling**—guided by model-estimated prompt difficulty—achieves superior performance on scientific tasks such as GPQA-Diamond, effectively capturing nuanced, knowledge-intensive challenges that length alone fails to reflect.

Strategy	MATH500	GPQA-Diamond	SuperGPQA	MMLU-Pro
Random	75.60	33.21	22.96	50.31
Dual diverse only	76.25	32.13	23.43	48.49
Length only	83.30	35.81	30.02	**56.88**
Complexity & Dual diverse	73.55	**42.17**	24.54	50.15
IFD & Dual diverse	78.85	33.65	26.07	52.55
Length & Embedding diverse	**84.55**	40.91	29.84	55.07
Length & Category diverse	81.25	31.12	28.74	53.29
Length & Dual diverse (Ours)	82.21	37.82	**30.20**	**55.13**

Table 2: Ablation study on general data sampling strategies. Each strategy samples 17.1K instances from AM-1.4M. SFT was performed on the Qwen2.5-7B-Base model.

Regarding **diversity**, our **Length & Dual diverse** approach, which integrates response-length heuristics with both domain-level and semantic-level diversity, consistently delivers balanced gains across all benchmarks. It achieves top performance on SuperGPQA and remains competitive elsewhere, outperforming single-axis diversity strategies (Length & Embedding or Length & Category). This underscores the importance of hybrid multi-granularity diversity in covering the heterogeneity of real-world instruction distributions.

Collectively, these findings validate the central hypothesis of our framework: that a simple yet principled combination of response length and principled diversity is sufficient to construct compact, high-quality reasoning datasets. In contrast to approaches that rely on expensive difficulty estimators or task-specific heuristics, our method is lightweight, domain-agnostic, and empirically robust across diverse reasoning benchmarks.

Effectiveness on Science and Math Reasoning. To further evaluate the domain-specific utility of our sampling strategy, we conduct ablation studies in two reasoning-intensive domains: **science** and **math**. For science, we sample 10K instances from the `OpenScience` dataset; for math, we consider `NuminaMath-CoT`, `s1-59K`,

and their mixture. All models are fine-tuned from the `Qwen2.5-7B-Base` checkpoint under consistent settings.

Strategy (Science)	MATH500	GPQA	MMLU-Pro
Random	**72.00**	40.23	53.10
Dual diverse only	**70.05**	**42.74**	53.70
Length only	65.90	38.51	**57.35**
Length & Dual diverse (Ours)	69.95	**41.28**	**56.94**

Table 3: Ablation on science-domain sampling strategies. Each subset is drawn from `OpenScience`. 10k samples are used for each group.

Strategy (Math)	AIME25	AIME24	MATH500	GPQA
NuminaMath-CoT Easy&diverse	14.27	12.97	73.25	27.97
NuminaMath-CoT Hard&diverse	20.57	15.05	76.45	33.90
s1-59K Hard&diverse	**23.85**	22.71	**84.40**	32.51
Mix Hard&diverse	21.72	**26.82**	82.00	**35.54**

Table 4: Ablation on math-domain sampling. "Easy"/"Hard" defined by response length. "Mix" combines `NuminaMath-CoT` and `s1-59K`. Each group has 10K samples.

In the *science domain*, unlike general data, diversity is a more critical factor due to the unique characteristics of different scientific subfields. Although Dual Diverse achieves only a slightly higher score on GPQA than our method, our Length & Dual Diverse approach consistently yields balanced performance across other benchmarks (Table 3).

As shown in Table 4, performance in the *math domain* improves with both data quality and instance difficulty. Longer, more diverse samples from `NuminaMath-CoT` outperform shorter ones, with notable gains on AIME25 (+6.3%) and GPQA (+5.9%). Samples drawn from `s1-59K` further exceed those from `NuminaMath-CoT` alone, indicating higher source quality. Importantly, combining both sources using our dual-heuristic strategy achieves the best overall results, highlighting the approach's robustness and scalability in multi-source alignment settings.

Scaling InfiAlign to 32B: Robustness Across Model Sizes We evaluate the scalability of **InfiAlign** beyond 7B by fine-tuning **Qwen2.5-32B-Instruct** on 1K-sample subsets drawn from a shared 59K data pool, with strict de-duplication via 15-gram filtering and embedding similarity ($>$0.9). All responses are generated by a high-capacity teacher model (QwQ-32B) and evaluated on four reasoning benchmarks.

Data	AIME 2025	AIME 2024	MATH500	GPQA Diamond	Avg.
s1.1	56.70	60.00	**95.40**	63.60	68.93
s1K-QwQ	59.38	**67.29**	94.35	**67.31**	72.08
Random 1k	55.78	63.44	93.75	64.52	69.37
Random 1k	56.15	64.53	94.00	66.60	70.32
Random 1k	57.55	65.05	94.30	64.65	70.39
InfiAlign 1k	**63.70**	66.61	94.75	64.07	**72.28**

Table 5: Ablation study on Qwen2.5-32B-Instruct using different 1k-sample subsets from the same 59K data pool. The responses of all samples were generated using QwQ-32B.

Table 5 reveals key insights:

High-quality supervision is crucial. Replacing DeepSeek-R1 with QwQ-32B supervision consistently improves s1K-QwQ over s1.1 across all benchmarks, notably +7.29 on AIME 2024. Linguistic analysis of reasoning-related discourse markers—such as deliberation cues ("let me think," "hmm"), verification phrases ("let me double-check"), and supplemental expressions ("for example," "on the other hand")—shows that QwQ-32B responses are on average 20% longer and contain 78% more reasoning-indicative phrases (Table 6). This suggests richer, more structured reasoning aligned with our hypothesis that longer responses encode stronger introspective signals, enhancing downstream distillation.

Teacher Model	Avg. Length (chars)	Total Feature Frequency
s1.1 (DeepSeek-R1)	27,535.5	75.83
s1K-QwQ (QwQ-32B)	**33,053.3(+20%)**	**135.26(+78%)**

Table 6: Linguistic characteristics of 1k responses generated by different teacher models.

InfiAlign demonstrates robustness and scalability. It matches s1K-QwQ performance without task-specific heuristics and consistently outperforms random baselines. Compared to the manual, resource-intensive filtering in s1, our automated pipeline offers a scalable, generalizable solution across model sizes and domains. These results underscore the effectiveness of combining scalable quality assessment with principled data sampling to build high-performance alignment models.

5 Conclusion and Limitation

We propose **InfiAlign**, a scalable and data-efficient post-training framework that combines supervised fine-tuning and reinforcement learning to align large language models for complex reasoning tasks. Central to our approach is a robust data selection pipeline that leverages multi-dimensional quality metrics—diversity, difficulty, and alignment quality—to automatically curate high-value instruction data from open sources. Applied to Qwen2.5-Math-7B-Base, InfiAlign matches the performance of DeepSeek-R1-Distill-Qwen-7B while using only $\sim$12% of the training data. Incorporating DPO further improves mathematical reasoning ability, with an 3.89% average gain on AIME 24/25. The modularity of our pipeline allows for seamless integration of new tasks and data sources, supporting efficient scaling and continuous improvement.

Limitations. Although our selection framework is domain-agnostic, it relies on manually defined metrics that may require tuning for unseen domains. Furthermore, while response length and reasoning-indicative markers are positively correlated with model performance, we have not yet systematically investigated how these surface-level characteristics—particularly response diversity and linguistic markers—impact the effectiveness of student model distillation.

References

Ahmad, W. U.; Narenthiran, S.; Majumdar, S.; Ficek, A.; Jain, S.; Huang, J.; Noroozi, V.; and Ginsburg, B. 2025. Opencodereasoning: Advancing data distillation for competitive coding. *arXiv preprint arXiv:2504.01943*.

Ahmed, M.; Seraj, R.; and Islam, S. M. S. 2020. The k-means algorithm: A comprehensive survey and performance evaluation. *Electronics*, 9(8): 1295.

AIME. 2024, 2025. AIME problems and solutions.

Bercovich, A.; Levy, I.; Golan, I.; Dabbah, M.; El-Yaniv, R.; Puny, O.; Galil, I.; Moshe, Z.; Ronen, T.; Nabwani, N.; Shahaf, I.; Tropp, O.; Karpas, E.; Zilberstein, R.; Zeng, J.; Singhal, S.; Bukharin, A.; Zhang, Y.; Konuk, T.; Shen, G.; Mahabaleshwarkar, A. S.; Kartal, B.; Suhara, Y.; Delalleau, O.; Chen, Z.; Wang, Z.; Mosallanezhad, D.; Renduchintala, A.; Qian, H.; Rekesh, D.; Jia, F.; Majumdar, S.; Noroozi, V.; Ahmad, W. U.; Narenthiran, S.; Ficek, A.; Samadi, M.; Huang, J.; Jain, S.; Gitman, I.; Moshkov, I.; Du, W.; Toshniwal, S.; Armstrong, G.; Kisacanin, B.; Novikov, M.; Gitman, D.; Bakhturina, E.; Scowcroft, J. P.; Kamalu, J.; Su, D.; Kong, K.; Kliegl, M.; Karimi, R.; Lin, Y.; Satheesh, S.; Parmar, J.; Gundecha, P.; Norick, B.; Jennings, J.; Prabhumoye, S.; Akter, S. N.; Patwary, M.; Khattar, A.; Narayanan, D.; Waleffe, R.; Zhang, J.; Su, B.-Y.; Huang, G.; Kong, T.; Chadha, P.; Jain, S.; Harvey, C.; Segal, E.; Huang, J.; Kashirsky, S.; McQueen, R.; Putterman, I.; Lam, G.; Venkatesan, A.; Wu, S.; Nguyen, V.; Kilaru, M.; Wang, A.; Warno, A.; Somasamudramath, A.; Bhaskar, S.; Dong, M.; Assaf, N.; Mor, S.; Argov, O. U.; Junkin, S.; Romanenko, O.; Larroy, P.; Katariya, M.; Rovinelli, M.; Balas, V.; Edelman, N.; Bhiwandiwalla, A.; Subramaniam, M.; Ithape, S.; Ramamoorthy, K.; Wu, Y.; Velury, S. V.; Almog, O.; Daw, J.; Fridman, D.; Galinkin, E.; Evans, M.; Luna, K.; Derczynski, L.; Pope, N.; Long, E.; Schneider, S.; Siman, G.; Grzegorzek, T.; Ribalta, P.; Katariya, M.; Conway, J.; Saar, T.; Guan, A.; Pawelec, K.; Prayaga, S.; Kuchaiev, O.; Ginsburg, B.; Olabiyi, O.; Briski, K.; Cohen, J.; Catanzaro, B.; Alben, J.; Geifman, Y.; Chung, E.; and Alexiuk, C. 2025. Llama-Nemotron: Efficient Reasoning Models. arXiv:2505.00949.

Bukharin, A.; Li, S.; Wang, Z.; Yang, J.; Yin, B.; Li, X.; Zhang, C.; Zhao, T.; and Jiang, H. 2024. Data Diversity Matters for Robust Instruction Tuning. In Al-Onaizan, Y.; Bansal, M.; and Chen, Y., eds., *Findings of the Association for Computational Linguistics: EMNLP 2024, Miami, Florida, USA, November 12-16, 2024*, 3411–3425. Association for Computational Linguistics.

Chen, L.; Li, S.; Yan, J.; Wang, H.; Gunaratna, K.; Yadav, V.; Tang, Z.; Srinivasan, V.; Zhou, T.; Huang, H.; and Jin, H. 2024. AlpaGasus: Training A Better Alpaca with Fewer Data. arXiv:2307.08701.

Chen, Y.; Yang, Z.; Liu, Z.; Lee, C.; Xu, P.; Shoeybi, M.; Catanzaro, B.; and Ping, W. 2025. Acereason-nemotron: Advancing math and code reasoning through reinforcement learning. *arXiv preprint arXiv:2505.16400*.

DeepSeek-AI. 2025. DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning. arXiv:2501.12948.

Ge, Y.; Liu, Y.; Hu, C.; Meng, W.; Tao, S.; Zhao, X.; Ma, H.; Zhang, L.; Chen, B.; Yang, H.; Li, B.; Xiao, T.; and Zhu, J. 2024. Clustering and Ranking: Diversity-preserved Instruction Selection through Expert-aligned Quality Estimation. arXiv:2402.18191.

Guha, E.; Marten, R.; Keh, S.; Raoof, N.; Smyrnis, G.; Bansal, H.; Nezhurina, M.; Mercat, J.; Vu, T.; Sprague, Z.; Suvarna, A.; Feuer, B.; Chen, L.; Khan, Z.; Frankel, E.; Grover, S.; Choi, C.; Muennighoff, N.; Su, S.; Zhao, W.; Yang, J.; Pimpalgaonkar, S.; Sharma, K.; Ji, C. C.-J.; Deng, Y.; Pratt, S.; Ramanujan, V.; Saad-Falcon, J.; Li, J.; Dave, A.; Albalak, A.; Arora, K.; Wulfe, B.; Hegde, C.; Durrett, G.; Oh, S.; Bansal, M.; Gabriel, S.; Grover, A.; Chang, K.-W.; Shankar, V.; Gokaslan, A.; Merrill, M. A.; Hashimoto, T.; Choi, Y.; Jitsev, J.; Heckel, R.; Sathiamoorthy, M.; Dimakis, A. G.; and Schmidt, L. 2025a. OpenThoughts: Data Recipes for Reasoning Models. arXiv:2506.04178.

Guha, E.; Marten, R.; Keh, S.; Raoof, N.; Smyrnis, G.; Bansal, H.; Nezhurina, M.; Mercat, J.; Vu, T.; Sprague, Z.; et al. 2025b. OpenThoughts: Data Recipes for Reasoning Models. *arXiv preprint arXiv:2506.04178*.

He, J.; Liu, J.; Liu, C. Y.; Yan, R.; Wang, C.; Cheng, P.; Zhang, X.; Zhang, F.; Xu, J.; Shen, W.; et al. 2025. Skywork open reasoner 1 technical report. *arXiv preprint arXiv:2505.22312*.

HuggingFace. 2025. Open R1: A fully open reproduction of DeepSeek-R1.

Jain, N.; Han, K.; Gu, A.; Li, W.-D.; Yan, F.; Zhang, T.; Wang, S.; Solar-Lezama, A.; Sen, K.; and Stoica, I. 2024. Livecodebench: Holistic and contamination free evaluation of large language models for code. *arXiv preprint arXiv:2403.07974*.

LI, J.; Beeching, E.; Tunstall, L.; Lipkin, B.; Soletskyi, R.; Huang, S. C.; Rasul, K.; Yu, L.; Jiang, A.; Shen, Z.; Qin, Z.; Dong, B.; Zhou, L.; Fleureau, Y.; Lample, G.; and Polu, S. 2024. Numina-Math. [https://huggingface.co/AI-MO/NuminaMath-CoT](https://github.com/project-numina/aimo-progress-prize/blob/main/report/numina_dataset.pdf).

Li, X.; Gao, M.; Zhang, Z.; Yue, C.; and Hu, H. 2024. Rule-based data selection for large language models. *arXiv preprint arXiv:2410.04715*.

Lightman, H.; Kosaraju, V.; Burda, Y.; Edwards, H.; Baker, B.; Lee, T.; Leike, J.; Schulman, J.; Sutskever, I.; and Cobbe, K. 2023. Let's verify step by step. In *The Twelfth International Conference on Learning Representations*.

Moshkov, I.; Hanley, D.; Sorokin, I.; Toshniwal, S.; Henkel, C.; Schifferer, B.; Du, W.; and Gitman, I. 2025. AIMO-2 Winning Solution: Building State-of-the-Art Mathematical Reasoning Models with OpenMathReasoning dataset. *arXiv preprint arXiv:2504.16891*.

Muennighoff, N.; Yang, Z.; Shi, W.; Li, X. L.; Fei-Fei, L.; Hajishirzi, H.; Zettlemoyer, L.; Liang, P.; Candès, E.; and Hashimoto, T. 2025. s1: Simple test-time scaling. arXiv:2501.19393.

NVIDIA. 2025. OpenScience Dataset.

Pan, X.; Huang, L.; Kang, L.; Liu, Z.; Lu, Y.; and Cheng, S. 2024. G-DIG: Towards Gradient-based DIverse and hiGh-quality Instruction Data Selection for Machine Translation. In Ku, L.; Martins, A.; and Srikumar, V., eds., *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, 15395–15406. Association for Computational Linguistics.

Qwen-Team. 2024. Qwen2.5: A Party of Foundation Models.

Qwen-Team. 2025. QwQ-32B: Embracing the Power of Reinforcement Learning.

Rafailov, R.; Sharma, A.; Mitchell, E.; Manning, C. D.; Ermon, S.; and Finn, C. 2023. Direct preference optimization: Your language model is secretly a reward model. In *Advances in neural information processing systems*, volume 36, 53728–53741.

Shridhar, K.; Stolfo, A.; and Sachan, M. 2022. Distilling reasoning capabilities into smaller language models. *arXiv preprint arXiv:2212.00193*.

Team, M.-A.; Du, X.; Yao, Y.; Ma, K.; Wang, B.; Zheng, T.; Zhu, K.; Liu, M.; Liang, Y.; Jin, X.; et al. 2025. SuperGPQA: Scaling LLM Evaluation across 285 Graduate Disciplines. *CoRR*.

Wang, Y.; Ma, X.; Zhang, G.; Ni, Y.; Chandra, A.; Guo, S.; Ren, W.; Arulraj, A.; He, X.; Jiang, Z.; et al. 2024. MMLU-Pro: A more robust and challenging multi-task language understanding benchmark. *Advances in Neural Information Processing Systems*, 37: 95266–95290.

Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Xia, F.; Chi, E.; Le, Q. V.; Zhou, D.; et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35: 24824–24837.

Wen, L.; Cai, Y.; Xiao, F.; He, X.; An, Q.; Duan, Z.; Du, Y.; Liu, J.; Tang, L.; Lv, X.; Zou, H.; Deng, Y.; Jia, S.; and Zhang, X. 2025. Light-R1: Curriculum SFT, DPO and RL for Long COT from Scratch and Beyond. *arXiv preprint arXiv:2503.10460*.

Wu, S.; Lu, K.; Xu, B.; Lin, J.; Su, Q.; and Zhou, C. 2023. Self-Evolved Diverse Data Sampling for Efficient Instruction Tuning. arXiv:2311.08182.

Xia, M.; Malladi, S.; Gururangan, S.; Arora, S.; and Chen, D. 2024. LESS: Selecting Influential Data for Targeted Instruction Tuning. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net.

Xu, X.; Li, M.; Tao, C.; Shen, T.; Cheng, R.; Li, J.; Xu, C.; Tao, D.; and Zhou, T. 2024. A survey on knowledge distillation of large language models. *arXiv preprint arXiv:2402.13116*.

Ye, Y.; Huang, Z.; Xiao, Y.; Chern, E.; Xia, S.; and Liu, P. 2025. LIMO: Less is More for Reasoning. arXiv:2502.03387.

Yin, M.; Wu, C.; Wang, Y.; Wang, H.; Guo, W.; Wang, Y.; Liu, Y.; Tang, R.; Lian, D.; and Chen, E. 2024. Entropy Law: The Story Behind Data Compression and LLM Performance. arXiv:2407.06645.

Zhang, J.; Qin, Y.; Pi, R.; Zhang, W.; Pan, R.; and Zhang, T. 2025. TAGCOS: Task-agnostic Gradient Clustered Coreset Selection for Instruction Tuning Data. In Chiruzzo, L.; Ritter, A.; and Wang, L., eds., *Findings of the Association for Computational Linguistics: NAACL 2025*, 4671–4686. Albuquerque, New Mexico: Association for Computational Linguistics. ISBN 979-8-89176-195-7.

Zhang, X.; Zhang, Y.; Long, D.; Xie, W.; Dai, Z.; Tang, J.; Lin, H.; Yang, B.; Xie, P.; Huang, F.; Zhang, M.; Li, W.; and Zhang, M. 2024. mGTE: Generalized Long-Context Text Representation and Reranking Models for Multilingual Text Retrieval. arXiv:2407.19669.

Zhao, H.; Wang, H.; Peng, Y.; Zhao, S.; Tian, X.; Chen, S.; Ji, Y.; and Li, X. 2025. 1.4 Million Open-Source Distilled Reasoning Dataset to Empower Large Language Model Training. arXiv:2503.19633.

Zou, H.; Lv, X.; Jia, S.; and Zhang, X. 2025. 360-LLaMA-Factory: Plug & Play Sequence Parallelism for Long Post-Training. arXiv:2505.22296.

Appendix A: Domain Classification Prompts

We use prompting-based LLM classification to annotate each QA pair with fine-grained domain labels. Below, we present the exact prompts used for data annotation, including category descriptions for mathematics, code, science, and general instruction tasks.

Math Domain Classification Prompt

Mathematics Domain Annotation Prompt

You are a mathematics expert tasked with classifying math problems into the correct categories:1. **Algebra**: Problems involving equations, expressions, polynomials, factorization, and number manipulation. 2. **Geometry and Topology**: Questions about shapes, angles, distances, volumes, or geometric relationships. 3. **Analysis**: Problems involving limits, calculus, differential equations, integrals, or series. 4. **Number Theory**: Focused on divisibility, primes, modular arithmetic, factorials, and integer properties. 5. **Probability and Statistics**: Questions involving likelihood, counting, distributions, expected values, or averages. 6. **Discrete Mathematics and Combinatorics**: Problems involving counting, arrangements, graphs, or logical structures. 7. **Logic**: Problems involving formal reasoning, truth tables, proof systems, predicates, and logical structure.
According to the item['instruction'], DO NOT PROVIDE ANY EXPLANATION. JUST output ONLY THE MOST RELEVANT category name from the list above. Put the most concise answer inside `\boxed{}`.

Code Domain Classification Prompt

Code Domain Annotation Prompt

You are a programming expert tasked with classifying coding problems into the appropriate category:1. **Array **: Problems focused on manipulating and traversing arrays, including operations like insertion, deletion, and searching within linear data structures. 2. ** String **: Challenges involving text processing, such as substring manipulation, pattern matching, and character encoding. 3. ** Hash Table **: Tasks leveraging key-value pair structures for efficient data lookup, insertion, or frequency counting. 4. ** Dynamic Programming **: Optimization problems requiring breaking down into overlapping subproblems and storing intermediate results. 5. ** Math **: Problems solvable using mathematical concepts, including arithmetic, algebra, or combinatorics. 6. ** Sorting **: Algorithms to reorder data based on specific criteria, often as a preprocessing step for other operations. 7. ** Greedy **: Problems where locally optimal choices at each step lead to a globally optimal solution. 8. ** Depth-First Search (DFS) **: Tree or graph traversal exploring as far as possible along each branch before backtracking. 9. ** Binary Search **: Efficient search algorithm for sorted datasets by repeatedly dividing the search interval in half. 10. ** Matrix **: Operations on 2D grids, such as traversal, rotation, or element-wise computations. 11. ** Bit Manipulation **: Problems requiring operations at the bit level, like masking or shifting. 12. ** Breadth-First Search (BFS) **: Level-order traversal for trees or graphs, often used to find shortest paths. 13. ** Two Pointers **: Technique using paired references to traverse data structures, often for pairwise comparisons or windowing. 14. ** Tree **: Hierarchical data structure problems involving nodes, paths, or subtree properties. 15. ** Prefix Sum: Preprocessing arrays to enable rapid range sum queries or cumulative calculations. 16. ** Heap (Priority Queue) **: Problems requiring efficient access to the highest/lowest priority element, often for scheduling. 17. ** Simulation **: Step-by-step emulation of real-world processes or system behaviors. 18. ** Stack **: Last-in-first-out (LIFO) structure problems, such as parsing or backtracking. 19. ** Counting **: Frequency analysis of elements, often combined with hash tables or arrays. 20. ** Graph **: Problems involving nodes and edges, such as pathfinding or cycle detection. 21. ** Sliding Window **: Technique to maintain a dynamic subset of data (e.g., contiguous subarrays) for efficiency. 22. ** Backtracking **: Exhaustive search by incrementally building candidates and abandoning invalid paths. 23. ** Enumeration **: Systematic listing of all possible solutions or configurations. 24. ** Union Find **: Disjoint-set operations for dynamic connectivity and component merging. 25. ** Monotonic Stack **: Stack variant maintaining elements in sorted order for next-greater/smaller problems. 26. ** Number Theory **: Mathematical problems focusing on integers, primes, divisibility, or modular arithmetic. 27. ** Linked List **: Linear data structure problems involving node manipulation and pointer traversal. 28. ** Bitmask **: Compact representation of sets or states using binary digits, enabling efficient bitwise operations. 29. ** Divide and Conquer **: Problems split into independent subproblems, solved recursively (e.g., merge sort). 30. ** Trie **: Tree-like structure for efficient string storage and retrieval (e.g., autocomplete). 31. ** Memoization **: Optimization technique caching intermediate results to avoid redundant computations. 32. ** Ordered Set **: Data structure maintaining sorted elements for rank/range queries. 33. ** Recursion **: Function calling itself to solve problems with repetitive substructures (e.g., Fibonacci).
According to the item['instruction'], DO NOT PROVIDE ANY EXPLANATION. JUST output ONLY THE MOST RELEVANT category name from the list above. Put the most concise answer inside `\boxed{}`.

Science Domain Classification Prompt

Science Domain Annotation Prompt

You are a science expert tasked with classifying this question into a specific science discipline:1. **Molecular Biology**: Studies molecular processes in cells, such as DNA transcription, RNA translation, and protein synthesis.2. **Genetics**: Focuses on heredity and genetic variation, including gene inheritance, mutations, and trait transmission.3. **Other Biology**: Covers all other biological topics, such as cell biology, physiology, neuroscience, evolution, and ecology.4. **Quantum Mechanics**: Explores matter and energy behavior at atomic and subatomic scales, including superposition and entanglement.5. **High-Energy Particle Physics**: Investigates fundamental particles and forces using particle accelerators and the Standard Model.6. **Physics (general)**: Covers broad or foundational topics in classical or modern physics, not tied to a specific subfield.7. **Astrophysics**: Applies physical principles to celestial phenomena like stars, galaxies, and black holes.8. **Electromagnetism and Photonics**: Studies electric/magnetic fields, electromagnetic waves, light, and optics technologies like lasers.9. **Relativistic Mechanics**: Examines motion and gravity under Einstein's relativity, especially at near-light speeds.10. **Statistical Mechanics**: Uses probability to connect microscopic particle behavior to macroscopic physical laws like thermodynamics.11. **Condensed Matter Physics**: Studies the physical properties of solids and liquids, including semiconductors and superconductors.12. **Optics and Acoustics**: Examines the behavior of light and sound, including reflection, diffraction, and wave propagation.13. **Organic Chemistry**: Focuses on the structure and reactions of carbon-containing compounds, including biomolecules.14. **Chemistry (general)**: Covers fundamental chemical principles not limited to any specific subfield.15. **Inorganic Chemistry**: Studies compounds without carbon-hydrogen bonds, including metals and minerals.16. **Analytical Chemistry**: Involves methods to detect and quantify substances, such as spectroscopy and titration.17. **Physical Chemistry**: Combines physics and chemistry to study energy, thermodynamics, kinetics, and molecular behavior.18. **Others**: Use this only if the content clearly does not fit any of the categories above.
According to the item['instruction'], DO NOT PROVIDE ANY EXPLANATION. JUST output ONLY THE MOST RELEVANT category name from the list above. Put the most concise answer inside `\boxed{}`.

General Instruction Classification Prompt

General Domain Annotation Prompt

You are a classification assistant tasked with identifying the general topic of a given instruction. Choose from the following categories:1. **Logic and Reasoning**: Problems involving deductive reasoning, logical puzzles, critical thinking, and formal systems. 2. **Mathematical Ability**: Questions requiring numerical computation, problem-solving, mathematical concepts, and quantitative analysis. 3. **Programming and Software Development**: Challenges related to coding, algorithms, data structures, debugging, and software engineering. 4. **Natural Language Processing and Understanding**: Tasks involving text analysis, language modeling, syntax, semantics, and machine comprehension. 5. **Information Processing and Integration**: Problems about data organization, knowledge synthesis, and combining multiple sources of information. 6. **Problem Solving and Support**: Questions that require troubleshooting, decision-making, and providing solutions to user queries. 7. **Data Science and Analytics**: Challenges involving statistical analysis, data visualization, predictive modeling, and machine learning. 8. **Creativity and Design**: Tasks related to ideation, artistic expression, UX/UI design, and innovative problem-solving. 9. **STEM Knowledge**: Interdisciplinary STEM applications (e.g., physics problem-solving, engineering principles) 10. **Humanities, History, Philosophy, and Sociology Knowledge**: Topics related to cultural studies, historical events, ethical theories, and social dynamics. 11. **Open Knowledge Q&A**: General factual queries spanning a wide range of subjects without a specific domain focus. 12. **Life Knowledge and Skills**: Practical advice on everyday activities, DIY tasks, personal development, and lifestyle tips. 13. **Education and Consulting**: Questions about learning strategies, academic guidance, tutoring, and professional advice. 14. **Linguistic Knowledge, Multilingual and Multicultural Understanding**: Topics involving languages, translation, cultural nuances, and comparative linguistics. 15. **Medical, Pharmaceutical, and Health Knowledge**: Questions related to diseases, treatments, pharmacology, and wellness. 16. **Communication and Social Media**: Challenges involving interpersonal skills, digital marketing, content creation, and online engagement. 17. **Task Generation**: Requests for generating structured tasks, prompts, or problem sets for various applications. 18. **Literary Creation and Artistic Knowledge**: Topics related to writing, poetry, storytelling, visual arts, and creative expression. 19. **Analysis and Research**: Tasks requiring deep investigation, literature review, experimental design, and evidence synthesis. 20. ** Project and Task Management**: Questions about planning, organization, workflow optimization, and productivity strategies. 21. **Financial, Economic, and Business Knowledge**: Topics covering investments, market trends, accounting, and corporate strategies. 22. **Psychological Knowledge**: Questions related to cognitive processes, mental health, behavioral theories, and emotional intelligence. 23. **Open Task Completion**: Miscellaneous requests that don't fit neatly into other categories but require structured execution. According to the item['instruction'], DO NOT PROVIDE ANY EXPLANATION. JUST output ONLY THE MOST RELEVANT category name from the list above. Put the most concise answer inside `\boxed{}`.

Appendix B: Data Composition and Proportion

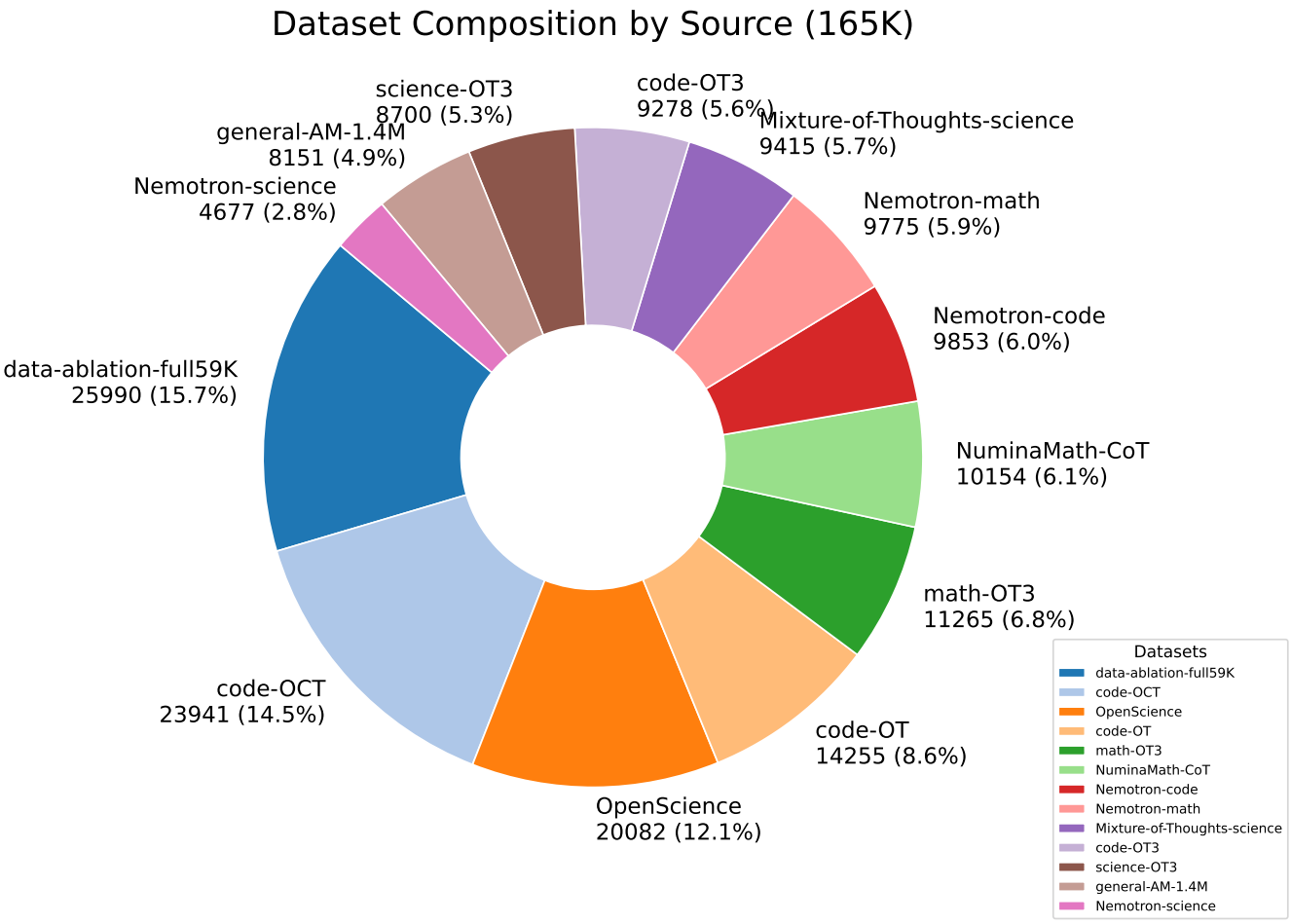

Figure 3: Data Composition and Proportion of full 165K SFT data.