DIFFICULTY–DIVERSITY COLLABORATIVE FILTERING FOR DATA-EFFICIENT LLM FINE-TUNING

Anonymous authors
Paper under double-blind review

ABSTRACT

The performance of fine-tuned language models is heavily influenced by the quality and quantity of their fine-tuning data. While scaling laws suggest that larger models benefit from more data during pretraining, the Less-is-More hypothesis highlights that downstream fine-tuning often requires only a small but high-quality dataset to effectively elicit a model's pretrained knowledge. However, identifying such premium data, particularly in terms of difficulty and diversity, typically relies on human expertise, and existing methods offer limited guidance for automatic selection from large unannotated corpora. This work presents a novel quantitative framework that formalizes the interplay between question difficulty and diversity, and introduces *Difficulty–Diversity Collaborative Filtering* (DDCF): an automated approach that tailors data selection to the unique characteristics of each language model via collaborative filtering. By leveraging a small seed dataset to predict correctness across a large unannotated corpus, our method reduces the annotation cost by $100-200\times$, while maintaining downstream performance comparable to full-corpus fine-tuning.

1 INTRODUCTION

The remarkable success of Large Language Models (LLMs) in recent years (Grattafiori et al., 2024b; Yang et al., 2025b) stems largely from their ability to learn rich and generalizable representations from massive pretraining corpora. To further enhance capabilities of these models on downstream tasks, supervised fine-tuning (SFT) has become a popular approach (Wei et al., 2022; Chung et al., 2024). However, SFT typically involves fine-tuning pretrained models on large-scale, human-annotated instruction datasets, often comprising hundreds of thousands of examples.

Despite its effectiveness, fine-tuning on such large datasets presents several challenges. First, data collection and model training incur substantial computational costs. Second, updating a model on a new large corpus may cause catastrophic forgetting, where continual learning of new tasks degrades performance on previously acquired knowledge (Biderman et al., 2024; Wang et al., 2024). Third, scaling up the dataset often leads to over-representation of common patterns, reducing diversity and underrepresenting rare but important examples (Kim et al., 2022; Zhang et al., 2025a).

Recently, the *Less-is-More* hypothesis (Zhou et al., 2023; Ye et al., 2025) has suggested that downstream task adaptation can be achieved through minimal supervision, where the model primarily learns task-specific formatting or styles to reveal knowledge already encoded during pretraining. Empirical studies have shown that fine-tuning on just a few carefully selected examples sometimes outperforms naively using vast annotated corpora (Zhou et al., 2023; Ye et al., 2025; Muennighoff et al., 2025). However, such curated datasets often rely on evolving human expertise, making them labor-intensive, inflexible, and inconvenient to adapt to new models or tasks.

While recent efforts have explored automated methods to improve data quality (Xia et al., 2024; Yang et al., 2024b), the automatic selection without annotated output responses remains an open challenge. For example, Xia et al. (2024) leveraged gradient matching to a target dataset, while Yang et al. (2024b) instead trained the LM on the entire annotated corpus and then selected samples by clustering their loss trajectories. Other approaches (Ye et al., 2025; Muennighoff et al., 2025; Marion et al., 2023) identify challenging examples based on binary correctness, reasoning length, or perplexity, and group them into manually defined categories. However, such approaches are not

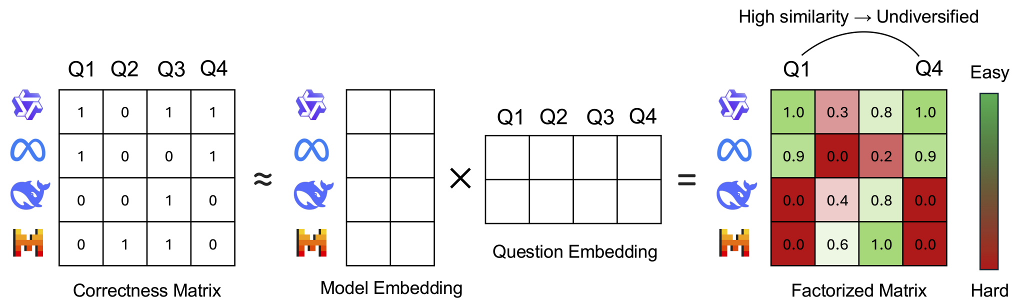

Figure 1: An illustration of our proposed *Difficulty–Diversity Collaborative Filtering* framework. Given a dataset, a binary correctness matrix from model predictions is factorized into model and question embeddings. Difficulty is defined by factorized scores, while diversity is measured by question similarity. These two criteria jointly guide the selection of compact yet effective training subsets, providing strong learning signals while avoiding redundant, overly similar examples.

universal—questions deemed difficult for one model may not be difficult for another, and rarely achieve an optimal balance of difficulty and diversity.

To address these gaps, we propose *Difficulty–Diversity Collaborative Filtering* (DDCF), a framework that reduces both annotating and fine-tuning costs by automatically selecting a compact, high-quality subset of questions tailored to each target model. As illustrated in Figure 1, DDCF measures question difficulty using collaborative filtering over predicted correctness patterns from multiple open-source LLMs, and quantifies diversity based on question similarity. By formulating data selection as a combinatorial optimization that directly trades off these two criteria, we can efficiently approximate the optimal subset using a simple *k-greedy* strategy. Starting from an empty set, k-greedy iteratively adds the question with the greatest marginal gain in our difficulty–diversity objective until exactly k examples are selected. Empirically, DDCF selects compact yet impactful subsets that effectively challenge the target model while maintaining broad coverage, enabling more efficient fine-tuning and improved performance in various downstream tasks.

Our key contributions are as follows:

- We propose *Difficulty–Diversity Collaborative Filtering*, a novel framework that leverages multiple LMs to capture unique characteristics of each target LM, enabling automatic construction of compact, model-specific training subsets *without requiring prior annotation*.
- To the best of our knowledge, this work is the first to systematically quantify and analyze the interplay between difficulty and diversity in data selection, and to demonstrate how their trade-off shapes downstream fine-tuning performance.
- We empirically demonstrate that DDCF outperforms existing data selection baselines across multiple benchmarks, achieving higher accuracy with the same selection budget.

2 Related Work

Numerous approaches have been proposed to curate high-quality training data, which can be grouped into several categories. Influence-based methods estimate each example's impact on a target set via gradient matching—e.g., Grad-Match (Killamsetty et al., 2021), LESS (Xia et al., 2024), NICE (Wang et al., 2025)—or by framing selection as an optimal control problem (Gu et al., 2025). Heuristic approaches often use perplexity as a proxy for difficulty, with medium-perplexity examples found especially informative (Marion et al., 2023). Feedback-driven frameworks leverage closed-source LLMs (such as ChatGPT) to score or prune candidates—exemplified by AlpaGasus (Chen et al., 2024) and Evol (Liu et al., 2024). Diversity-aware sampling ensures broad representational coverage through embedding similarity (e.g., D4 (Tirumala et al., 2023), DiSF (Fan et al., 2025)), while lightweight proxy models cluster examples from loss trajectories, as in S2L (Yang et al., 2024b).

Parallel to data selection, recent work has explored the problem of choosing the most appropriate model for a given question, commonly referred to as *LLM routing*. FrugalGPT (Chen et al., 2023) adaptively queries models in sequence until a reliable answer is obtained. More recent methods

Table 1: Comparison of DDCF with prior data selection methods. "Difficulty-Aware" and "Diversity-Aware" reflect whether these criteria are considered in selection. "No Full-Corpus Fine-Tuning" indicates whether the method avoids training on the full corpus. "Minimal-Annotation" denotes methods that (almost) do not rely on annotations, thereby reducing annotation costs. "No LLM Feedback" indicates the method does not depend on external reward models, e.g,. ChatGPT.

Method	**Difficulty-Aware**	**Diversity-Aware**	**No Full-Corpus Fine-Tuning**	**Minimal Annotation**	**No LLM Feedback**
Perplexity (Marion et al., 2023)	✓	✗	✓	✗	✓
S2L (Yang et al., 2024b)	✓	✓	✗	✗	✓
AlpaGasus (Chen et al., 2024)	✓	✗	✓	✗	✗
LESS (Xia et al., 2024)	✓	✓	✓	✗	✓
DiSF (Fan et al., 2025)	✗	✓	✓	✓	✓
DDCF (ours)	✓	✓	✓	✓	✓

embed models and questions into a shared latent space and learn routing policies using matrix factorization (Ong et al., 2024; Zhuang et al., 2025), while Nguyen et al. (2024) frame the problem as a multi-armed bandit.

Our work bridges these two lines of research by recasting model–question interactions as a recommendation problem (Lee & Seung, 2000; He et al., 2017), treating models as users and questions as items. This perspective allows us to learn tailored relevance scores that guide data selection in a large corpus, even without full-annotation labels. Building on this, we propose a lightweight collaborative filtering framework with difficulty–diversity re-ranking to curate a small, high-quality subset from a large unannotated corpus, yielding strong performance in low-resource fine-tuning.

Table 1 summarizes how DDCF compares to representative data selection approaches across five key dimensions. Notably, DDCF only relies on ground-truth answers from a small seeding dataset to construct the binary correctness matrix. This design enables DDCF to uniquely satisfy all five criteria: it selects a compact, challenging, and diverse subset without the need for full-corpus fine-tuning, external annotations, or feedback from closed-source LLMs. As a result, DDCF offers a scalable and domain-agnostic solution for efficient data curation across diverse model families.

3 Data Selection with Minimal Annotation

3.1 Correctness Predictor

Given m language models $\mathcal{M} = \{\mathcal{M}_1, \ldots, \mathcal{M}_m\}$ and a seed dataset of n questions $\mathcal{Q} = \{q_1, \ldots, q_n\}$ with corresponding ground-truth answers, we construct a binary correctness matrix $\mathcal{A} \in \{0,1\}^{m \times n}$. Each entry $\mathcal{A}_{ij}$ indicates whether model $\mathcal{M}_i$ correctly answers question q_j. This matrix captures fine-grained model-question interactions, enabling us to analyze both model capabilities and question difficulty. For instance, certain models may perform well on algebra but poorly on geometry, while questions answered incorrectly by most models likely indicate higher difficulty.

Following the approach in Zhuang et al. (2025), we enrich the binary matrix $\mathcal{A}$ by learning low-dimensional embeddings for both models and questions. Specifically, we learn model embeddings $E_M \in \mathbb{R}^{m \times d}$ and question embeddings $E_Q \in \mathbb{R}^{n \times d}$ such that

$$\mathcal{A} \approx \hat{\mathcal{A}} = E_M E_Q^\top, \tag{1}$$

where d denotes the embedding dimension and $\hat{\mathcal{A}}$ approximates the observed correctness matrix $\mathcal{A}$. This factorization is analogous to those used in collaborative filtering (Lee & Seung, 2000; He et al., 2017), but it is inherently limited to the training set and does not generalize to unseen questions.

To enable generalization, we introduce a correctness predictor $f : \mathcal{M} \times \mathcal{Q} \to [0, 1]$, which estimates whether a given model correctly answers a given question. We instantiate f using an encoder architecture, detailed below.

Encoder The encoder comprises two modules: a model encoder ϕ_M and a question encoder ϕ_Q, both projecting into a shared latent space $\mathbb{R}^d$.

The model encoder $\phi_M : \mathcal{M} \to \mathbb{R}^d$ is defined as a function composition $\phi_M = h_M \circ g_M$, where:

- $g_M : \mathcal{M} \to \mathbb{R}^d$ maps a model index to an initial representation;
- $h_M : \mathbb{R}^d \to \mathbb{R}^d$ refines the initial representation to obtain the model embedding E_{m_i}.

The question encoder $\phi_Q : \mathcal{Q} \to \mathbb{R}^d$ follows the same structure: $\phi_Q = h_Q \circ g_Q$, where:

- $g_Q : \mathcal{Q} \to \mathbb{R}^{\text{dim}_q}$ uses a pre-trained sentence transformer to encode question text into an initial question representation $E^0_{q_j}$;
- $h_Q : \mathbb{R}^{\text{dim}_q} \to \mathbb{R}^d$ projects this representation into the shared latent space, yielding factorized question embeddings E_{q_j} for each $q_j \in \mathcal{Q}$.

In our implementation, h_M and h_Q are multilayer perceptrons.

Classifier Head The classifier predicts correctness from the Hadamard product of embeddings:

$$\psi(E_{m_i} \odot E_{q_j}),$$

where $\psi : \mathbb{R}^d \to \mathbb{R}^2$ is a linear classifier. The overall predictor is thus defined as $f(\mathcal{M}_i, q_j) = \psi(\phi_M(\mathcal{M}_i) \odot \phi_Q(q_j))$, which can be trained using binary cross-entropy loss.

The predicted correctness score for model $\mathcal{M}_i$ on question q_j is defined as:

$$\hat{\mathcal{A}}_{ij} = \sigma(f(\mathcal{M}_i, q_j)_1), \tag{2}$$

where $\sigma(\cdot)$ is the sigmoid function, and the subscript 1 denotes the logit for the "correct" class.

Notably, Equation 2 can be viewed as a parameterized version of the classical matrix factorization in Equation 1. Instead of estimating a single shared difficulty score per question, this formulation allows the difficulty of a question to be *"personalized"* for each model's characteristic. This personalized modeling of correctness underpins our approach in the next section, where we construct *Difficulty–Diversity Collaborative Filtering* strategies tailored to individual models.

3.2 Difficulty-Diversity Collaborative Filtering

Given a target model $\mathcal{M}_i$ and a *large unannotated dataset* $\mathcal{D}$, where $|\mathcal{D}| \gg |\mathcal{Q}|$ and $\mathcal{Q}$ is the introduced seed dataset, our goal is to select a subset $S_i \subset \mathcal{D}$ of k questions that are both (1) *difficult* for the model $\mathcal{M}_i$ and (2) *diverse* to cover a wide range of topics. This ensures that the selected examples provide strong learning signals while avoiding redundancy.

To estimate question difficulty, we leverage the correctness predictor f introduced earlier. For every question $q_j \in \mathcal{D}$, the model $\mathcal{M}_i$'s predicted correctness score is given by $\tilde{\mathcal{A}}_{ij} = \sigma(f(\mathcal{M}_i, q_j)_1)$, and we aggregate these into a vector $\tilde{\mathcal{A}}_i \in \mathbb{R}^{|\mathcal{D}|}$. Lower values of $\tilde{\mathcal{A}}_{ij}$ correspond to questions the model is more likely to get wrong, thus indicating higher difficulty.

To encourage diversity among selected questions, we define a similarity matrix $\Sigma \in \mathbb{R}^{|\mathcal{D}|\times|\mathcal{D}|}$, where each entry is the cosine similarity of the sentence-transformer embeddings of questions q_u and q_j:

$$\Sigma_{uj} = \frac{(E^0_{q_u})(E^0_{q_j})^\top}{\|E^0_{q_u}\|_2 \|E^0_{q_j}\|_2}. \tag{3}$$

We then formulate the selection problem as a combinatorial optimization task:

$$\min_{x \in \{0,1\}^{|\mathcal{D}|}} \lambda(x^\top \tilde{\mathcal{A}}_i) + (1-\lambda)(x^\top \Sigma x), \quad \text{s.t.} \sum_{j=1}^{|\mathcal{D}|} x_j = k. \tag{4}$$

Here, $S_i = \{q_j | x_j = 1\}$ is the curated subset for the model $\mathcal{M}_i$, and $\lambda \in [0, 1]$ balances the trade-off between difficulty ($x^\top \tilde{\mathcal{A}}_i$) and diversity ($x^\top \Sigma x$). Although the objective is convex over the

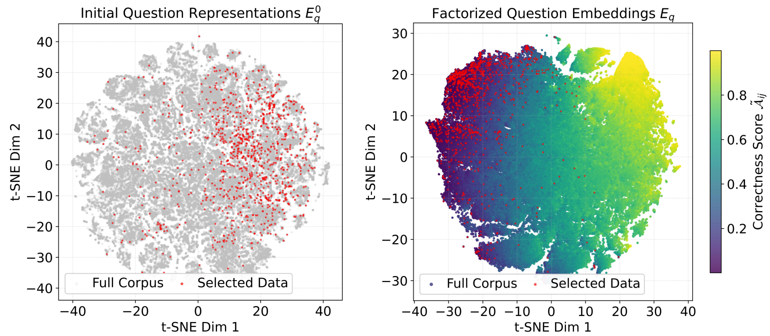

Figure 2: t-SNE visualization of questions selected by DDCF (best viewed in color). **Left:** the selected data in the semantic space encoded by Sentence-Transformer. **Right:** the same data in the factorized space learned by the correctness predictor, with point color indicating difficulty (darker means harder) and selected examples highlighted in red. Our proposed DDCF framework organizes the subset to target challenging questions while preserving diversity across semantic regions.

continuous relaxation of x, the binary constraint renders the problem NP-hard and computationally intensive due to the $O(|\mathcal{D}|^2)$ memory complexity of the similarity term Σ.

To overcome these limitations, we propose a memory- and compute-efficient k-greedy heuristic that incrementally selects questions. Starting with an empty set S_i, at each step, we add the next question $q_j \in \mathcal{D} \setminus S_i$ that minimizes a composite score:

$$q_j = \underset{q_j \in \mathcal{D} \setminus S_i}{\arg\min} \left[\lambda \tilde{A}_{ij} + (1-\lambda) \max_{q_u \in S_i} \Sigma_{uj} \right]. \quad (5)$$

This approach not only relaxes the original NP-hard problem but also significantly improves memory efficiency by computing only $O(k \cdot |\mathcal{D}|)$ pairwise similarities on the fly. As a result, the k-greedy strategy is both fast and scalable, enabling efficient selection over large unannotated corpora while maintaining a strong trade-off between difficulty and diversity.

Figure 2 provides a qualitative validation of our k-greedy sampler's dual objectives. In the initial representation space encoded by Sentence-Transformer (left), the chosen subset spans multiple semantic regions of the full corpus, confirming that the on-the-fly diversity term successfully prevents redundant sampling. After projecting into the factorized embedding space (right), a smooth gradient of question difficulty emerges, and the selection concentrates almost in the most challenging questions. Together, these two views demonstrate that our *Difficulty–Diversity Collaborative Filtering* simultaneously maintains semantic diversity and precisely targets high-difficulty examples.

Therefore, the selected subset S_i provides a highly informative slice of the large corpus for downstream use. In the case of unannotated corpora, DDCF enables cost-effective data preparation by concentrating annotation efforts, either from human experts or strong teacher models, on only the most impactful k examples. Here, DDCF serves as a front-end filter that reduces supervision costs while preserving strong learning signals. For already annotated corpora, DDCF serves as a post-hoc filter that eliminates trivial or redundant examples and tailors the learning path to the strengths and weaknesses of the target model, thereby shortening the training time. In both scenarios, the resulting compact, model-aware dataset S_i can undergo further quality checks—such as expert review of selected questions and annotations— especially in high-stakes domains like medicine or law. Overall, DDCF facilitates a data-efficient tuning paradigm where LLMs can be rapidly adapted with minimal supervision, even when full-corpus annotation is impractical or prohibitively expensive.

4 Experiments with Pre-Annotated Corpus

4.1 Experimental Setup

Training the Correctness Predictor To learn factorized model and question embeddings, we train a correctness predictor using outputs from 23 open-source LMs [1] spanning a wide range of sizes and

[1]Appendix B presents the full list of all 23 LMs and their inference times on the seed datasets.

architectures. Each model was evaluated on the seed dataset of 19,470 questions from the training splits of GSM8K (Cobbe et al., 2021) and MATH (Hendrycks et al., 2021b). Responses were labeled correct or incorrect by a rule-based verifier, resulting in binary supervision for each model–question pair. We use 10% of the questions as a held-out test set (results shown in Appendix C). For the initial question embeddings, we employ the sentence transformer `Qwen/Qwen3-Embedding-4B`.

Data Selection We conduct experiments on the `OpenR1-Math-220K` dataset[2], which contains 225,129 math problems annotated by `DeepSeek-R1-671B` (DeepSeek-AI et al., 2025). We select 1,000 training instances from this corpus using different selection strategies. Based on ablation results, we set the difficulty–diversity trade-off in Equation 5 to $\lambda = 0.2$ by default.

Baselines We compare our approach, *DDCF*, with various baselines:

- Dummy baseline: (1) **Random** randomly samples 1,000 examples; (2) **Longest** selects the 1,000 longest instruction examples; (3) **Binary Hard** randomly samples 1,000 examples that the targeted model incorrectly answers from the seed dataset;
- Annotation dependent: (4) **Least Confidence** (Settles, 2009) measures the model's confidence as the product of probabilities of the data example. (5) **Perplexity** (Marion et al., 2023) selects examples around medium perplexity; (6) SmallToLarge (**S2L**) (Yang et al., 2024b) samples from clusters summarizing the loss trajectory of easy-to-hard questions.
- Annotation independent: (7) **DiSF** (Fan et al., 2025) chooses samples that maximize the diversity of the question embedding space via covariance eigenvalue maximization.
- Manually selected dataset: (8) **LIMO**, 817 instructive examples from Ye et al. (2025); (9) **s1.1-1K**, 1,000 high-quality examples curated by Muennighoff et al. (2025).

It is worth noting that, unlike **Random**, **DiSF**, and our method **DDCF**—which can be applied *prior to annotation*—the remaining baselines require full-corpus annotation to compute selection criteria such as gradients, reasoning length, or perplexity. We do not compare our method with selectors like AlpaGasus (Chen et al., 2024) or LESS (Xia et al., 2024), as they assume different settings, such as reliance on ChatGPT feedback or access to a target dataset for gradient matching.

Evaluation We evaluate on 10 popular reasoning benchmarks, grouped into two categories:

- **In-Distribution**: **MATH500** (Hendrycks et al., 2021b), **OlympiadBench** (He et al., 2024), **GSM8K** (Cobbe et al., 2021), **AIGEval-SAT-Math** (Zhong et al., 2024), and **AIME24**.
- **Out-of-Distribution**: **Minerva** (Lewkowycz et al., 2022), which includes undergraduate-level STEM problems; **Gaokao**, sourced from China's 2024 National College Entrance Exam; and the **STEM** subset of MMLU (Hendrycks et al., 2021a).
- **Development Set**: We use **SVAMP** (Patel et al., 2021) (elementary), and **AMC23** (competition level) to determine hyperparameter λ balancing the difficulty-diversity trade-off.

We report pass@1 accuracy by default, while for **AMC23** and **AIME24** we report pass@32, due to their small size and high difficulty. Experiment details can be found in Appendix A.

4.2 Main Results

In-Distribution Results Table 2 demonstrates that **DDCF** consistently produces the strongest 1K-example subsets among all selection strategies. For *Qwen2.5-Math-7B*, DDCF attains an average score of 70.2, outperforms the best baseline (Perplexity, 69.3) while staying within only -5.6 points of full-data training. Notably, DDCF yields larger gains on the hardest benchmarks: it boosts AIME24 performance to 49.0, +10.4 over random and +14.4 over the base model. For *Qwen3-8B-Base*, DDCF achieves 85.0 on average, outperforming all baselines and reducing the gap to full-data training to just -2.0. Its improvement is most evident on GSM8K, where DDCF reaches 95.9, surpassing all baselines by up to +3.9. These results indicate that DDCF maintains both breadth and depth in coverage, enabling efficient fine-tuning with limited data.

[2]https://huggingface.co/datasets/open-r1/OpenR1-Math-220k, licensed under Apache 2.0.

Table 2: Performance on In-Distribution and Out-of-Distribution benchmarks.

Method	*In-Distribution*						*Out-of-Distribution*			
	AIME24	**MATH**	**OlyBen**	**GSM8k**	**SAT**	**Avg.**	**Miverva**	**Gaokao**	**STEM**	**Avg.**
					Qwen2.5-Math-7B					
Full Dataset	64.5	80.6	42.5	92.6	98.2	75.8	46.3	72.2	79.7	66.1
Base Model	34.6	55.4	16.4	91.6	80.0	55.6	12.9	67.1	67.7	49.2
Random	38.6	76.4	34.8	91.0	98.2	67.8	41.2	64.6	75.7	60.5
Longest	19.7	53.8	18.4	81.1	69.1	48.4	25.7	36.7	50.7	37.7
Binary Hard	29.6	67.2	28.3	89.3	85.5	60.0	31.6	53.2	67.9	50.9
Least Confid.	12.3	42.8	11.7	61.6	58.6	37.4	21.0	20.3	47.0	29.4
Perplexity	44.7	77.8	37.8	89.3	96.8	<u>69.3</u>	46.7	69.6	79.1	<u>65.1</u>
S2L	36.7	74.4	34.8	90.1	98.2	66.9	39.3	58.2	75.4	57.7
DiSF	44.6	76.2	35.4	89.9	96.8	68.6	43.4	68.4	75.6	62.4
LIMO	41.1	76.4	35.7	89.5	94.6	67.4	35.3	58.2	74.5	56.0
s1.1-1K	41.9	76.6	37.4	90.3	96.8	68.5	40.1	67.1	75.9	61.0
DDCF	49.0	77.6	35.0	91.2	98.2	**70.2**	45.6	74.7	75.8	**65.4**
					Qwen3-8B-Base					
Full Dataset	88.6	91.8	60.3	95.0	99.1	87.0	64.3	84.8	92.1	80.4
Base Model	47.8	60.8	36.3	89.8	98.2	66.6	40.8	58.2	84.4	61.1
Random	80.9	89.2	53.8	94.4	99.6	83.6	62.5	83.5	90.8	79.0
Longest	81.4	90.4	54.7	94.4	99.6	<u>84.1</u>	64.3	84.8	90.8	<u>80.0</u>
Binary Hard	75.0	91.4	54.5	94.2	93.6	81.8	60.3	80.0	86.9	75.7
Least Confid.	71.6	89.8	52.6	94.8	99.6	81.7	62.5	81.0	90.6	78.0
Perplexity	79.3	89.8	55.0	94.5	99.6	83.6	60.3	83.5	91.0	78.3
S2L	76.4	91.0	55.0	94.5	99.1	83.2	62.1	78.5	91.3	77.3
DiSF	74.9	90.6	54.8	94.6	99.6	82.9	65.1	83.5	91.1	79.9
LIMO	79.8	89.4	55.3	93.7	98.6	83.4	54.8	82.3	88.7	75.2
s1.1-1K	75.5	86.2	51.9	92.0	98.2	80.7	57.7	77.2	89.3	74.7
DDCF	82.2	91.0	56.0	95.9	100	**85.0**	66.2	84.8	90.6	**80.5**

Out-of-Distribution Results Under distribution shifts, DDCF also demonstrates strong generalization. For the 7B model, it records a 65.4 average—closing the gap to the full dataset down to 0.7 and surpassing every other subset strategy by margins ranging from +0.3 to +8.0. On Gaokao, DDCF not only outperforms all baselines but also exceeds the full-data performance by +2.5 (74.7 vs 72.2), suggesting that efficient fine-tuning might preserve generalization in multi-lingual settings. For the 8B model, DDCF achieves the highest OOD average (80.5), slightly ahead of full-data fine-tuning (80.4). This edge comes primarily from Minerva, where DDCF improves by +1.9. Together, these findings highlight that compact, model-aware subsets selected by DDCF can preserve or even enhance out-of-distribution robustness relative to training on the full corpus.

Owing to space limitations, we report the results of `Falcon-10B-Base` in Appendix D.

4.3 Data Size Ablation

To assess the effect of training set size, we vary the number of selected questions k from 0 to 225,129 and evaluate both in-distribution (ID) and out-of-distribution (OOD) performance averaged across the benchmarks introduced earlier (Figure 3). We compare two settings: a strong base model `Qwen2.5-Math-7B` and a weaker base model `Qwen2.5-Math-1.5B`.

On one hand, for `Qwen2.5-Math-7B`, ID accuracy improves almost monotonically, rising from 55.6 at $k = 0$ to 75.8 at full scale, with the sharpest gain achieved within the first 1,000 samples (70.2). OOD performance, however, exhibits a non-monotonic trend: it peaks early at 65.4 for $k = 1{,}000$, declines to around 61 at $k \in [4{,}000, 8{,}000]$, and then recovers steadily to 66.1 at full scale. This mid-range dip suggests that while small curated sets provide strong generalization, enlarging the pool without sufficient coverage may initially dilute transferability before larger sets restore

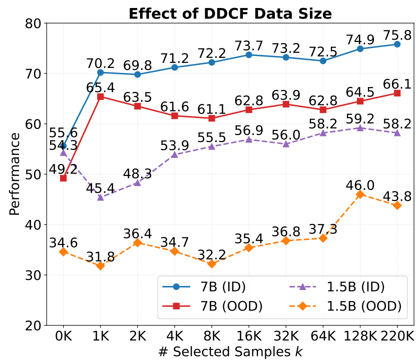

Figure 3: Performance of the fine-tuned models with different data sizes.

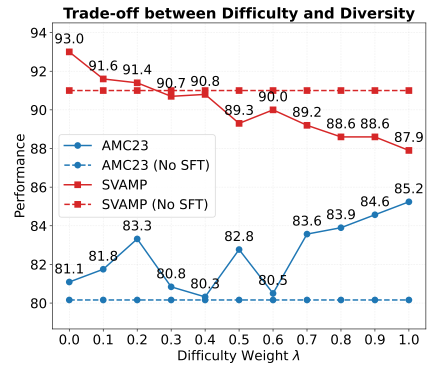

Figure 4: Difficulty-Diversity trade-off in data selection.

robustness. Notably, selecting only 1,000 samples already secures over 70% of the ID improvement and nearly the full OOD benefit, highlighting the data efficiency of our DDCF framework.

On the other hand, fine-tuning on small yet highly complex datasets can degrade the performance of weaker language models—a phenomenon referred to as the *Small Model Learnability Gap* (Li et al., 2025) or *Long CoT Degradation* (Luo et al., 2025). Figure 3 illustrates this effect for `Qwen2.5-Math-1.5B` fine-tuned on DDCF subsets of size k. With only $k = 1{,}000$ examples, ID accuracy drops sharply from 54.3 to 45.4 (-8.9) and OOD falls from 34.6 to 31.8 (-2.8), illustrating the known phenomenon. Increasing to $k = 4{,}000$ largely mitigates this effect—ID is only 0.4 below its pre-fine-tuning level while at $k = 8{,}000$ both curves recover fully and begin to climb.

Beyond $k = 8{,}000$, performance increases steadily: by $k = 16{,}000$ we attain 56.9 ID and 35.4 OOD, and by $k = 128{,}000$ the model culminates at 59.2 ID and 46.0 OOD. Notably, this represents 1.0 ID / 2.2 OOD improvements over a conventional full-corpus fine-tuning on all 225,129 available samples, demonstrating that our DDCF strategy can overcome initial degradation and ultimately yield superior results with far fewer examples.

4.4 Difficulty-Diversity Trade-off

To determine the optimal difficulty weight λ in Equation 5, we perform an ablation study on elementary-level SVAMP and competition-level AMC23 with selection size $k = 1000$ using `Qwen2.5-Math-7B`, as shown in Figure 4. As λ increases from 0 (pure diversity) to 0.2, AMC23 performance jumps from 81.1 to 83.3% while SVAMP remains at its pre-trained baseline of $\sim 91\%$. Further increasing λ continues to boost AMC23, peaking at 85.2 for $\lambda = 1.0$, but with diminishing returns, SVAMP performance declines by about 2 points at $\lambda = 0.5$ and 4 points at $\lambda = 1.0$, indicating that excessive emphasis on difficult examples undermines proficiency on simpler tasks.

Since our ultimate goal is to elicit the model's full problem-solving ability from a small, curated fine-tuning set without eroding its pre-trained knowledge, we adopt $\lambda = 0.2$ as the default parameter for our *Difficulty–Diversity Collaborative Filtering* framework, striking a balanced trade-off between difficulty and diversity. Beyond this default, DDCF enables the difficulty weight λ to be adjusted on the fly, allowing users to instantly tailor data selection to their priorities without additional retraining or redesigning the framework. This adaptability makes the framework both convenient and versatile, supporting a wide spectrum of selection strategies within a unified formulation.

4.5 Does DDCF Framework Learn Model Characteristics?

Let $\mathcal{S}_a$ and $\mathcal{S}_b$ be subsets selected by models $\mathcal{M}_a$ and $\mathcal{M}_b$. We quantify their overlap via the Jaccard index: $J(\mathcal{S}_a, \mathcal{S}_b) = \frac{|\mathcal{S}_a \cap \mathcal{S}_b|}{|\mathcal{S}_a \cup \mathcal{S}_b|}$, which measures the fraction of questions chosen by both models relative to the total unique questions. A higher J indicates greater similarity in the subsets, reflecting closer alignment in model behavior. We hypothesize that models within the same family, sharing architecture and pre-training data, will exhibit higher Jaccard similarity than those from different families. Indeed, our analysis shows an average intra-family index of $\mathbf{0.224}$ versus $\mathbf{0.169}$ inter-family, demonstrating that our framework captures meaningful model-specific characteristics.

Figure 5 shows the topic-wise composition of each model's selected subset alongside the full dataset distribution. Although the full corpus is dominated by Algebra (48.1%), our framework tailors sampling to each model's behavior. In particular, `Qwen2.5-Math-7B` and `Qwen2.5-32B` exhibit almost identical distributions: Combinatorics holds the largest share, while Algebra, Geometry, Number Theory, and Logic & Puzzles each retain substantial and balanced proportions. By contrast, `Llama-3.1-8B` diverges markedly, de-emphasizing Algebra and boosting Combinatorics and Logic & Puzzles. This divergence shows that DDCF tailors question selection to each model's specific strengths and weaknesses, targeting areas for improvement rather than sampling uniformly. The experiment on data transferability across models is in Appendix E.

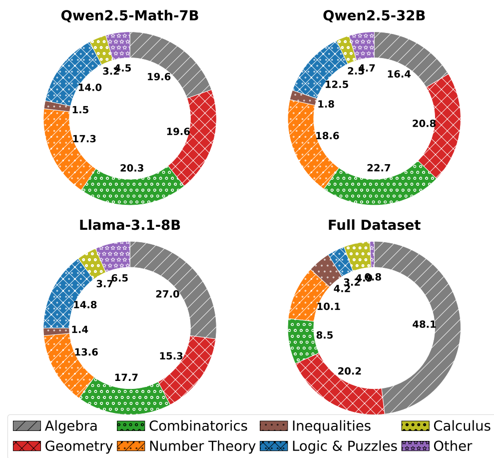

Figure 5: Topic distribution of DDCF datasets.

5 Experiments with Unannotated Corpus at Selection Time

To illustrate capabilities of DDCF on large corpora that are initially unannotated, we evaluate on the MMLU benchmark (Hendrycks et al., 2021a), which includes 99,842 training, 1,531 validation, and 14,042 test questions. While the corpus spans many disciplines, it lacks reasoning annotations. We therefore train a *correctness predictor* on the validation split and use it with DDCF to select 1,000 high-quality training examples, which are then distilled into reasoning traces using `Qwen/Qwen3-32B` in long-thinking mode. By filtering *before* annotation, DDCF reduces distillation cost nearly 100×, whereas prior methods require annotating the full corpus in advance.

Table 3: Performance on the MMLU benchmark.

Method	Qwen2.5-7B			Qwen3-8B-Base			Falcon-10B-Base		
	Humanities	**Social Science**	**STEM**	**Humanities**	**Social Science**	**STEM**	**Humanities**	**Social Science**	**STEM**
Base Model	59.0	77.1	71.5	62.8	82.3	84.4	66.1	80.9	81.4
Random	61.4	80.5	79.3	**69.0**	86.7	88.8	69.1	85.5	85.9
DiSF	62.6	81.3	74.7	68.9	87.1	89.1	69.5	85.8	84.7
DDCF	**63.5**	**81.4**	**81.1**	68.2	**87.3**	**89.7**	**69.9**	**86.0**	**87.2**

Table 3 compares **DDCF** with annotation-free baselines (**Random**, **DiSF**) across three MMLU domains. On *Qwen2.5-7B*, **DDCF** achieves the best accuracy in all domains, improving the average by +6.2 over the base model and +2.5 over the strongest baseline, with the largest gain in STEM (81.1; +9.6 over base, +6.4 over DiSF). For *Qwen3-8B-Base*, DDCF again excels, setting new highs in Social Science (87.3) and STEM (89.7), and raising the average by +5.2 over base with only a slight drop in Humanities. On *Falcon-10B-Base*, DDCF outperforms all baselines, boosting Humanities, Social Science, and STEM by +3.8, +5.1, and +5.8, respectively.

Overall, these results show that DDCF strengthens not only quantitative reasoning (STEM) but also inferential reasoning (Humanities, Social Science), even with scarce annotations. Beyond in-domain performance, OOD validation (Appendix F) reveals that fine-tuning on just 1,000 distilled MMLU examples transfers effectively to general tasks such as commonsense, reading comprehension, and instruction following, where DDCF achieves the best average across all backbones.

6 Conclusion

Difficulty–Diversity Collaborative Filtering is a novel concept for curating small, high-quality fine-tuning subsets from large unannotated corpora by balancing question difficulty, via a learned correctness predictor, and semantic diversity in embedding space. Empirically, DDCF reduces annotation costs by $100-200\times$ while maintaining performance comparable to the full-data baseline, and our analysis shows it naturally tailors data selection to each model's unique strengths and weaknesses.

REFERENCES

Dan Biderman, Jacob Portes, Jose Javier Gonzalez Ortiz, Mansheej Paul, Philip Greengard, Connor Jennings, Daniel King, Sam Havens, Vitaliy Chiley, Jonathan Frankle, Cody Blakeney, and John Patrick Cunningham. LoRA learns less and forgets less. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=aloEru2qCG. Featured Certification.

Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, et al. Pythia: A suite for analyzing large language models across training and scaling. In *Proceedings of International Conference on Machine Learning*, pp. 2397–2430. PMLR, 2023.

Lichang Chen, Shiyang Li, Jun Yan, Hai Wang, Kalpa Gunaratna, Vikas Yadav, Zheng Tang, Vijay Srinivasan, Tianyi Zhou, Heng Huang, and Hongxia Jin. Alpagasus: Training a better alpaca with fewer data. In *Proceedings of The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=FdVXgSJhvz.

Lingjiao Chen, Matei Zaharia, and James Zou. Frugalgpt: How to use large language models while reducing cost and improving performance. *arXiv preprint arXiv:2305.05176*, 2023.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53, 2024.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jiawei Wang, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Shengfeng Ye, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wanjia Zhao, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanhong Xu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning, 2025. URL https://arxiv.org/abs/2501.12948.

Yann Dubois, Balázs Galambosi, Percy Liang, and Tatsunori B Hashimoto. Length-controlled alpacaeval: A simple way to debias automatic evaluators. *arXiv preprint arXiv:2404.04475*, 2024.

Ziqing Fan, Siyuan Du, Shengchao Hu, Pingjie Wang, Li Shen, Ya Zhang, Dacheng Tao, and Yanfeng Wang. Combatting dimensional collapse in LLM pre-training data via submodular file selection. In *Proceedings of The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=f4gF6AIHRy.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vítor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. The llama 3 herd of models, 2024a. URL `https://arxiv.org/abs/2407.21783`.

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

Yuxian Gu, Li Dong, Hongning Wang, Yaru Hao, Qingxiu Dong, Furu Wei, and Minlie Huang. Data selection via optimal control for language models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL `https://openreview.net/forum?id=dhAL5fy8wS`.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, Jie Liu, Lei Qi, Zhiyuan Liu, and Maosong Sun. OlympiadBench: A challenging benchmark for promoting AGI with olympiad-level bilingual multimodal scientific problems. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3828–3850, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.211. URL `https://aclanthology.org/2024.acl-long.211/`.

Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. Neural collaborative filtering. In *Proceedings of the Twenty-Sixth International Conference on World Wide Web*, pp. 173–182, 2017.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *Proceedings of The Ninth International Conference on Learning Representations*, 2021a. URL https://openreview.net/forum?id=d7KBjmI3GmQ.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. *Advances in Thirty-Fifth Conference on Neural Information Processing Systems*, 2021b.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7b, 2023. URL https://arxiv.org/abs/2310.06825.

Krishnateja Killamsetty, Sivasubramanian Durga, Ganesh Ramakrishnan, Abir De, and Rishabh Iyer. Grad-match: Gradient matching based data subset selection for efficient deep model training. In *Proceedings of The Thirty-eighth International Conference on Machine Learning*, pp. 5464–5474. PMLR, 2021.

Jang-Hyun Kim, Jinuk Kim, Seong Joon Oh, Sangdoo Yun, Hwanjun Song, Joonhyun Jeong, Jung-Woo Ha, and Hyun Oh Song. Dataset condensation via efficient synthetic-data parameterization. In *Proceedings of the Tenth International Conference on Machine Learning*, pp. 11102–11118. PMLR, 2022.

Daniel Lee and H Sebastian Seung. Algorithms for non-negative matrix factorization. *Advances in The Thirteenth Annual Conference on Neural Information Processing Systems*, 13, 2000.

Aitor Lewkowycz, Anders Johan Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Venkatesh Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, Yuhuai Wu, Behnam Neyshabur, Guy Gur-Ari, and Vedant Misra. Solving quantitative reasoning problems with language models. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=IFXTZERXdM7.

Yuetai Li, Xiang Yue, Zhangchen Xu, Fengqing Jiang, Luyao Niu, Bill Yuchen Lin, Bhaskar Ramasubramanian, and Radha Poovendran. Small models struggle to learn from strong reasoners, 2025. URL https://arxiv.org/abs/2502.12143.

Jian Liu, Leyang Cui, Hanmeng Liu, Dandan Huang, Yile Wang, and Yue Zhang. Logiqa: A challenge dataset for machine reading comprehension with logical reasoning. In Christian Bessiere (ed.), *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pp. 3622–3628. International Joint Conferences on Artificial Intelligence Organization, 7 2020. doi: 10.24963/ijcai.2020/501. URL https://doi.org/10.24963/ijcai.2020/501. Main track.

Wei Liu, Weihao Zeng, Keqing He, Yong Jiang, and Junxian He. What makes good data for alignment? a comprehensive study of automatic data selection in instruction tuning. In *Proceedings of The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=BTKAeLqLMw.

Renjie Luo, Jiaxi Li, Chen Huang, and Wei Lu. Through the valley: Path to effective long cot training for small language models, 2025. URL https://arxiv.org/abs/2506.07712.

Max Marion, Ahmet Üstün, Luiza Pozzobon, Alex Wang, Marzieh Fadaee, and Sara Hooker. When less is more: Investigating data pruning for pretraining LLMs at scale. In *NeurIPS Workshop on Attributing Model Behavior at Scale*, 2023. URL https://openreview.net/forum?id=XUIYn3jo5T.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. Can a suit of armor conduct electricity? a new dataset for open book question answering. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii (eds.), *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2381–2391, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1260. URL https://aclanthology.org/D18-1260/.

Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori Hashimoto. s1: Simple test-time scaling. *arXiv preprint arXiv:2501.19393*, 2025.

Quang H Nguyen, Thinh Dao, Duy C Hoang, Juliette Decugis, Saurav Manchanda, Nitesh V Chawla, and Khoa D Doan. Metallm: A high-performant and cost-efficient dynamic framework for wrapping llms. *arXiv preprint arXiv:2407.10834*, 2024.

Isaac Ong, Amjad Almahairi, Vincent Wu, Wei-Lin Chiang, Tianhao Wu, Joseph E Gonzalez, M Waleed Kadous, and Ion Stoica. Routellm: Learning to route llms from preference data. In *Proceedings of The Thirteenth International Conference on Learning Representations*, 2024.

Arkil Patel, Satwik Bhattamishra, and Navin Goyal. Are NLP models really able to solve simple math word problems? In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 2080–2094, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.168. URL https://aclanthology.org/2021.naacl-main.168.

Burr Settles. Active learning literature survey. 2009.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models, 2024. URL https://arxiv.org/abs/2402.03300.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, Johan Ferret, Peter Liu, Pouya Tafti, Abe Friesen, Michelle Casbon, Sabela Ramos, Ravin Kumar, Charline Le Lan, Sammy Jerome, Anton Tsitsulin, Nino Vieillard, Piotr Stanczyk, Sertan Girgin, Nikola Momchev, Matt Hoffman, Shantanu Thakoor, Jean-Bastien Grill, Behnam Neyshabur, Olivier Bachem, Alanna Walton, Aliaksei Severyn, Alicia Parrish, Aliya Ahmad, Allen Hutchison, Alvin Abdagic, Amanda Carl, Amy Shen, Andy Brock, Andy Coenen, Anthony Laforge, Antonia Paterson, Ben Bastian, Bilal Piot, Bo Wu, Brandon Royal, Charlie Chen, Chintu Kumar, Chris Perry, Chris Welty, Christopher A. Choquette-Choo, Danila Sinopalnikov, David Weinberger, Dimple Vijaykumar, Dominika Rogozińska, Dustin Herbison, Elisa Bandy, Emma Wang, Eric Noland, Erica Moreira, Evan Senter, Evgenii Eltyshev, Francesco Visin, Gabriel Rasskin, Gary Wei, Glenn Cameron, Gus Martins, Hadi Hashemi, Hanna Klimczak-Plucińska, Harleen Batra, Harsh Dhand, Ivan Nardini, Jacinda Mein, Jack Zhou, James Svensson, Jeff Stanway, Jetha Chan, Jin Peng Zhou, Joana Carrasqueira, Joana Iljazi, Jocelyn Becker, Joe Fernandez, Joost van Amersfoort, Josh Gordon, Josh Lipschultz, Josh Newlan, Ju yeong Ji, Kareem Mohamed, Kartikeya Badola, Kat Black, Katie Millican, Keelin McDonell, Kelvin Nguyen, Kiranbir Sodhia, Kish Greene, Lars Lowe Sjoesund, Lauren Usui, Laurent Sifre, Lena Heuermann, Leticia Lago, Lilly McNealus, Livio Baldini Soares, Logan Kilpatrick, Lucas Dixon, Luciano Martins, Machel Reid, Manvinder Singh, Mark Iverson, Martin Görner, Mat Velloso, Mateo Wirth, Matt Davidow, Matt Miller, Matthew Rahtz, Matthew Watson, Meg Risdal, Mehran Kazemi, Michael Moynihan, Ming Zhang, Minsuk Kahng, Minwoo Park, Mofi Rahman, Mohit Khatwani, Natalie Dao, Nenshad Bardoliwalla, Nesh Devanathan, Neta Dumai, Nilay Chauhan, Oscar Wahltinez, Pankil Botarda, Parker Barnes, Paul Barham, Paul Michel, Pengchong Jin, Petko Georgiev, Phil Culliton, Pradeep Kuppala, Ramona Comanescu, Ramona Merhej, Reena Jana, Reza Ardeshir Rokni, Rishabh Agarwal, Ryan Mullins, Samaneh Saadat, Sara Mc Carthy, Sarah Cogan, Sarah Perrin, Sébastien M. R. Arnold, Sebastian Krause, Shengyang Dai, Shruti Garg, Shruti Sheth, Sue Ronstrom, Susan Chan, Timothy Jordan, Ting Yu, Tom Eccles, Tom Hennigan, Tomas Kocisky, Tulsee Doshi, Vihan Jain, Vikas Yadav, Vilobh Meshram, Vishal Dharmadhikari, Warren Barkley, Wei Wei, Wenming Ye, Woohyun Han, Woosuk Kwon, Xiang Xu, Zhe Shen, Zhitao Gong, Zichuan

Wei, Victor Cotruta, Phoebe Kirk, Anand Rao, Minh Giang, Ludovic Peran, Tris Warkentin, Eli Collins, Joelle Barral, Zoubin Ghahramani, Raia Hadsell, D. Sculley, Jeanine Banks, Anca Dragan, Slav Petrov, Oriol Vinyals, Jeff Dean, Demis Hassabis, Koray Kavukcuoglu, Clement Farabet, Elena Buchatskaya, Sebastian Borgeaud, Noah Fiedel, Armand Joulin, Kathleen Kenealy, Robert Dadashi, and Alek Andreev. Gemma 2: Improving open language models at a practical size, 2024. URL https://arxiv.org/abs/2408.00118.

Kushal Tirumala, Daniel Simig, Armen Aghajanyan, and Ari S. Morcos. D4: Improving LLM pretraining via document de-duplication and diversification. In *Proceedings of Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023. URL https://openreview.net/forum?id=CG0L2PFrb1.

Jingtan Wang, Xiaoqiang Lin, Rui Qiao, Pang Wei Koh, Chuan-Sheng Foo, and Bryan Kian Hsiang Low. NICE data selection for instruction tuning in LLMs with non-differentiable evaluation metric. In *Proceedings of The Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=2wt8m5HUBs.

Liyuan Wang, Xingxing Zhang, Hang Su, and Jun Zhu. A comprehensive survey of continual learning: Theory, method and application. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V Le. Finetuned language models are zero-shot learners. In *Proceedings of the Tenth International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=gEZrGCozdqR.

Mengzhou Xia, Sadhika Malladi, Suchin Gururangan, Sanjeev Arora, and Danqi Chen. LESS: Selecting influential data for targeted instruction tuning. In *Proceedings of The Forty-First International Conference on Machine Learning*, 2024.

An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, Keming Lu, Mingfeng Xue, Runji Lin, Tianyu Liu, Xingzhang Ren, and Zhenru Zhang. Qwen2.5-math technical report: Toward mathematical expert model via self-improvement, 2024a. URL https://arxiv.org/abs/2409.12122.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, 2025a. URL https://arxiv.org/abs/2505.09388.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025b. URL https://arxiv.org/abs/2412.15115.

Yu Yang, Siddhartha Mishra, Jeffrey N Chiang, and Baharan Mirzasoleiman. Smalltolarge (s2l): Scalable data selection for fine-tuning large language models by summarizing training trajectories of small models. In *Advances in The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024b. URL https://openreview.net/forum?id=K9IGlMQpif.

Yixin Ye, Zhen Huang, Yang Xiao, Ethan Chern, Shijie Xia, and Pengfei Liu. LIMO: Less is more for reasoning. In *Second Conference on Language Modeling*, 2025. URL https://openreview.net/forum?id=T2TZ0RY4Zk.

Xin Zhang, Jiawei Du, Ping Liu, and Joey Tianyi Zhou. Breaking class barriers: Efficient dataset distillation via inter-class feature compensator. In *Proceedings of The Thirteenth International Conference on Learning Representations*, 2025a. URL https://openreview.net/forum?id=X0CxfByJog.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. Qwen3 embedding: Advancing text embedding and reranking through foundation models, 2025b. URL https://arxiv.org/abs/2506.05176.

Lianmin Zheng, Liangsheng Yin, Zhiqiang Xie, Chuyue Sun, Jeff Huang, Cody Hao Yu, Shiyi Cao, Christos Kozyrakis, Ion Stoica, Joseph E. Gonzalez, Clark Barrett, and Ying Sheng. SGLang: Efficient execution of structured language model programs. In *Proceedings of The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=VqkAKQibpq.

Wanjun Zhong, Ruixiang Cui, Yiduo Guo, Yaobo Liang, Shuai Lu, Yanlin Wang, Amin Saied, Weizhu Chen, and Nan Duan. AGIEval: A human-centric benchmark for evaluating foundation models. In Kevin Duh, Helena Gomez, and Steven Bethard (eds.), *Findings of the Association for Computational Linguistics: NAACL 2024*, pp. 2299–2314, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-naacl.149. URL https://aclanthology.org/2024.findings-naacl.149/.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, et al. Lima: Less is more for alignment. *Advances in Neural Information Processing Systems*, 36:55006–55021, 2023.

Richard Zhuang, Tianhao Wu, Zhaojin Wen, Andrew Li, Jiantao Jiao, and Kannan Ramchandran. EmbedLLM: Learning compact representations of large language models. In *Proceedings of The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=Fs9EabmQrJ.

A EXPERIMENT DETAILS

A.1 CORRECTNESS PREDICTORS

MLP Block Both the model encoder h_M and the question encoder h_Q use a residual multilayer perceptron (MLP) block for refinement. Given an input $E^0 \in \mathbb{R}^d$, the block is defined as

$$\text{MLPBlock}(E^0) = E^0 + u, \tag{6}$$

where

$$u = W_2\big(\text{Dropout}(\gamma(W_1\,\text{LN}(E^0)))\big). \tag{7}$$

Here, LN denotes LayerNorm, $W_1 \in \mathbb{R}^{d\times d'}$ and $W_2 \in \mathbb{R}^{d'\times d}$ are linear layers with hidden dimension $d' = 0.1 * d$, and γ is a ReLU activation. Dropout with rate 0.8 is applied during training. To stabilize optimization, the final projection W_2 is zero-initialized, making the block behave as the identity map at initialization.

Noise Regularization To reduce overfitting to the limited set of models and questions, we inject Gaussian noise into both model and question embeddings during training. For model embedding $E^0_{m_i}$ and question embedding $E^0_{q_j}$, the perturbed representations are

$$E^0_{m_i} := E^0_{m_i} + \epsilon_p, \quad E^0_{q_j} := E^0_{q_j} + \epsilon_q, \tag{8}$$

where

$$\epsilon_p, \epsilon_q \sim \mathcal{N}(0, \alpha^2 I_d), \tag{9}$$

and α is a scalar hyperparameter controlling the perturbation scale. This stochastic perturbation acts as embedding-level data augmentation, preventing the predictor from memorizing spurious correlations in the binary correctness matrix. During inference, noise is disabled and the raw embeddings are used.

Table 4: Epochs and batch sizes used for supervised fine-tuning across dataset sizes.

Hyperparameter	*Dataset size k*								
	1K	**2K**	**4K**	**8K**	**16K**	**32K**	**64K**	**128K**	**220K**
Epochs	4	4	4	4	4	3	3	3	3
Batch size	32	32	32	32	32	64	128	128	128

Training Hyperparameters We train the correctness predictor with the Adam optimizer (weight decay 1×10^{-5}) and a cosine learning rate schedule with a warmup ratio of 0.03. The initial learning rate is set to 1×10^{-3}, and training runs for 30 epochs. For the *OpenR1-Math-220K* dataset, we use a batch size of 1028 and set the regularization parameter to $\alpha = 1 \times 10^{-2}$. For the *MMLU* experiment, we use a smaller batch size of 64 and increase the regularization strength to $\alpha = 3 \times 10^{-2}$.

A.2 Supervised Fine-tuning

We fine-tune LLMs using the `TRL`[3] library with a maximum sequence length of 16,384 tokens. Training is performed in `bfloat16` precision with the Adam optimizer, a cosine learning rate schedule, and a warmup ratio of 0.03. Table 4 summarizes the epoch and batch size configurations across different datasizes. Experiments in this paper can be done with 2 H100 gpus.

A.3 Data Selection Procedures

Baseline Details. Most baselines in our SFT experiments are described in Section 4.1, but we highlight additional implementation details here. For the **Binary Hard** baseline, we randomly sample 1,000 questions that the target model answers incorrectly from the seed datasets (GSM8K and MATH). Since GSM8K and MATH are annotated with short-CoT rationales of lower quality compared to OpenR1-Math-220K—which provides long-CoT annotations with structured reasoning and rigorous reflections—we re-annotate these 1,000 questions using `Qwen/Qwen3-32B` in long-thinking mode.

For the **S2L** method, we follow Yang et al. (2024b) and train a Pythia-70M model (Biderman et al., 2023) on the full OpenR1-Math-220K corpus as a proxy model to record loss trajectories. Samples are then clustered into 1,000 groups, from which representative examples are selected to form the training subset.

Finally, **DiSF** requires converting text samples into embedding vectors prior to selection. For a fair comparison with our proposed DDCF, we use `Qwen/Qwen3-Embedding-4B` (Zhang et al., 2025b) as the sentence encoder for DiSF.

Data Cleaning. Due to computational constraints, we restrict training to a maximum sequence length of 16,384 tokens. Accordingly, we discard all examples exceeding this length (fewer than 1% of OpenR1-Math-220K). To enhance the diversity of the selected dataset, we further remove duplicated questions, retaining only the instance with the shortest completion. After cleaning, the OpenR1-Math-220K dataset contains 189,257 examples, which we use for all experiments involving data selection.

A.4 Inference Hyperparameters

To improve efficiency, we accelerate inference with the SGLang framework (Zheng et al., 2024). By default, we report pass@1 accuracy, generating a single sampled response per query with `temperature=0.6`, `top_p=0.95`, `top_k=20`, `min_p=0`, and a maximum sequence length of 16,384 tokens. For the **AMC23** and **AIME24** benchmarks, we sample 64 responses per query and report pass@32 due to their small size and high difficulty.

[3] `https://github.com/huggingface/trl`

B List of Models in DDCF and Efficient DDCF Runtime

Table 5: Models used in the DDCF framework.

deepseek-ai/deepseek-math-7b-base	Qwen/Qwen2.5-Math-1.5B
google/gemma-2-2b	Qwen/Qwen2.5-Math-7B
google/gemma-2-9b	Qwen/Qwen3-0.6B-Base
google/gemma-2-27b	Qwen/Qwen3-1.7B-Base
meta-llama/Llama-3.2-1B	Qwen/Qwen3-14B-Base
meta-llama/Llama-3.2-3B	Qwen/Qwen3-4B-Base
meta-llama/Llama-3.1-8B	Qwen/Qwen3-8B-Base
mistralai/Mistral-7B-v0.3	tiiuae/Falcon3-1B-Base
mistralai/Mistral-Nemo-Base-2407	tiiuae/Falcon3-3B-Base
Qwen/Qwen2.5-7B	tiiuae/Falcon3-7B-Base
Qwen/Qwen2.5-14B	tiiuae/Falcon3-10B-Base
Qwen/Qwen2.5-32B	

Table 5 lists the 23 models included in our DDCF framework. These are models from Qwen 2.5 (Yang et al., 2025b), Qwen 2.5 Math (Yang et al., 2024a), Qwen3 (Yang et al., 2025a), Falcon 3, Mistral (Jiang et al., 2023), Llama 3 (Grattafiori et al., 2024a), Gemma 2 (Team et al., 2024), and Deepseek Math (Shao et al., 2024).

Running inference with all 23 LMs on 19,470 seeding questions from GSM8K and MATH requires approximately 11 H100 GPU hours, while processing 1,531 questions from the MMLU validation set takes around 2 H100 GPU hours, both accelerated by SGLang Zheng et al. (2024). With access to 8 H100 GPUs, the entire seed dataset—comprising triplets of the form `(model, question, binary correctness)`—can be constructed in under 2 wall-clock hours. Since the correctness predictors are lightweight in architecture, their training time is negligible. Likewise, the k-greedy selection procedure is computationally efficient, with minimal overhead across k iterations. As a result, the full DDCF pipeline—from seeding to subset selection—can be executed rapidly, making it efficient and effective even for large unannotated corpora, flexibly adaptable across various tasks and scenarios.

C How Reliable is the Correctness Predictor?

Table 6: Effect of the number of participating models on the correctness predictor's accuracy.

# Models	1	2	4	8	16	23
Accuracy	81.5	81.7	81.8	82.2	82.5	82.7

Table 7: Effect of the number of seeding questions on the correctness predictor's accuracy.

# Questions	1K	2K	4K	8K	16K	17.5K
Accuracy	80.0	80.1	80.8	81.9	82.4	82.7

As described in Section 4.1, we trained our Correctness Predictor on a seed dataset of triplets, `(model, question, binary correctness)`, comprising 23 open-source language models and 19,470 questions, with 1,947 questions (10%) held out for testing. To evaluate its reliability, we measured the predictor's accuracy on unseen test questions for the `Qwen2.5-Math-7B` model under two conditions: (1) fixing the number of models at 23 while varying the number of training questions, and (2) fixing the number of training questions while varying the number of models.

Overall, the Correctness Predictor exhibits strong sample efficiency in low-data regimes alongside steady improvements as more models or questions are added. When trained with just one model, it attains 81.5% accuracy, rising to 81.7% with two models and peaking at 82.7% when all 23 models are included (Table 6). Likewise, increasing the number of seeding questions boosts accuracy from 80.0% with 1,000 examples to 81.9% with 8,000 examples, and ultimately to 82.7% with 17,523 examples (Table 7). These results confirm that our predictor is reliable even with minimal data and scales effectively: most gains emerge early, with incremental benefits thereafter.

Table 8: Performance on In-Distribution and Out-of-Distribution benchmarks.

Method	*In-Distribution*						*Out-of-Distribution*			
	AIME24	**MATH**	**OlyBen**	**GSM8k**	**SAT**	**Avg.**	**Miverva**	**Gaokao**	**STEM**	**Avg.**
				Falcon-10B-Base						
Full Dataset	83.8	90.4	56.3	95.2	99.1	85.0	64.3	82.3	91.7	79.4
Base Model	41.1	68.6	34.2	81.4	93.6	63.8	39.7	55.7	81.4	58.9
Random	65.5	82.2	47.0	93.1	98.6	<u>77.3</u>	58.1	79.8	89.2	75.7
Longest	68.3	82.0	45.8	91.1	88.6	75.2	56.6	58.2	82.8	65.9
Binary Hard	67.4	83.4	49.0	94.2	77.8	74.3	57.4	50.6	76.2	61.4
Least Confid.	49.4	79.0	40.3	94.3	97.7	72.1	54.4	43.0	89.1	62.2
Perplexity	60.1	82.8	45.2	93.5	99.1	76.1	61.8	78.5	89.6	**76.6**
S2L	62.2	82.4	49.2	94.0	98.6	<u>77.3</u>	61.0	74.7	90.0	75.2
DiSF	63.2	83.0	47.7	93.4	98.6	77.2	62.1	72.2	89.2	74.5
LIMO	66.5	81.4	48.7	93.5	57.3	69.5	51.5	48.1	68.3	55.9
s1.1-1K	54.8	80.0	46.7	93.0	91.8	73.3	58.1	65.8	85.8	69.9
DDCF	66.6	83.0	46.1	93.9	98.1	**77.6**	60.3	78.5	88.9	<u>75.9</u>

D Experiment Results on OpenR1-Math-220K for Falcon-10B-Base

In-Distribution Table 8 shows that for *Falcon-10B-Base*, **DDCF** delivers the strongest overall subset, reaching an average of 77.6. This slightly surpasses the best-performing baselines (*Random* and *S2L*, both 77.3) and narrows the gap to the full-data upper bound (85.0) to just -7.4. Performance gains are especially visible on MATH500 (83.0) and GSM8k (93.9), where DDCF matches or exceeds competing selectors. On the most challenging benchmark, AIME24, DDCF secures 66.6—well above *Perplexity* (60.1) and *Least Confidence* (49.4), underscoring its ability to capture harder examples without sacrificing breadth.

Out-of-Distribution On OOD tasks, DDCF remains highly competitive. It achieves an average of 75.9, ranking just behind *Perplexity* (76.6) but outperforming all other baselines, including Random (75.7) and S2L (75.2). Notably, DDCF preserves strong performance across datasets: it improves over Random on Gaokao (+-0.7 vs +13.9 over weaker baselines) and stays close to the top scorer on Minerva (60.3 vs 61.8 with Perplexity). Again, DDCF consistently produces a compact subset that balances difficulty and diversity, yielding competitive results with only 1,000 examples.

E Data Transferability Between Models

Figure 6 shows the performance of `Qwen2.5-Math-7B` after fine-tuning on DDCF datasets curated for other models. Fine-tuning on its own curated data yields the highest combined performance of 67.5%. Substituting the dataset from `Qwen2.5-32B` incurs a modest 0.4 point drop (to 67.1%), while using `Gemma-2-9B` and `Mistral-7B-v0.3` subsets leads to declines of 2.0 and 2.3 points, respectively. Beyond these, we observe a gradually larger drop of 2.5 points with `Qwen2.5-Math-1.5B` and `Falcon3-7B-Base`, and 2.7 points with `Llama-3.1-8B`. Overall, this pattern hints that datasets drawn from models with closer architectural or training kinship may transfer more effectively, although more extensive experiments would be needed to confirm the precise nature of this relationship.

F OOD Performance on General Tasks of LLMs Fine-Tuned on MMLU Subsets

While DDCF is tailored for reasoning-centric MMLU tasks, Table 9 shows it also transfers effectively to out-of-distribution (OOD) general tasks. Fine-tuning on just 1,000 distilled MMLU examples leads to strong performance across diverse benchmarks, including commonsense reasoning

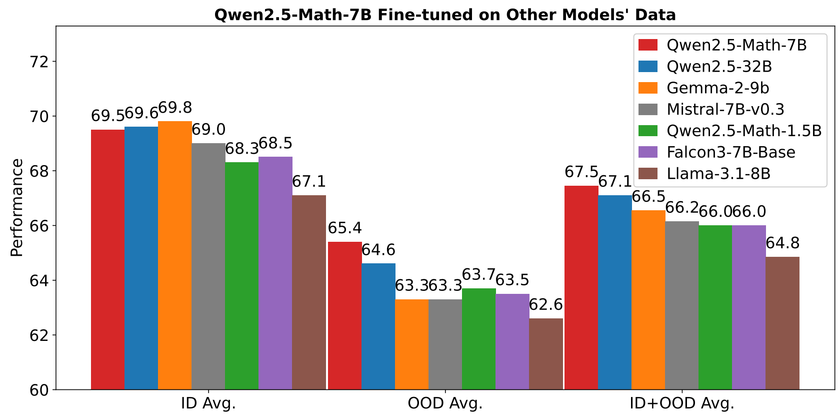

Figure 6: Each model has its own datasets. Using other models' datasets yields suboptimal results.

Table 9: Performance on OOD general tasks.

Method	Qwen2.5-7B			Qwen3-8B-Base			Falcon-10B-Base			**Avg.**
	LogiQA	**OpenBookQA**	**AlpacaEval2.0**	**LogiQA**	**OpenBookQA**	**AlpacaEval2.0**	**LogiQA**	**OpenBookQA**	**AlpacaEval2.0**	
Base Model	47.3	83.6	5.6	51.8	82.6	16.5	48.1	80.8	7.0	47.0
Random	**50.7**	89.4	33.3	61.0	93.2	59.3	53.5	90.4	49.7	64.5
DiSF	47.0	88.4	**36.6**	60.8	**94.4**	58.3	52.8	90.0	46.0	63.8
DDCF	48.2	**90.4**	32.5	**61.3**	92.0	**59.5**	**56.2**	**92.0**	**53.3**	**65.0**

(LogiQA (Liu et al., 2020)), reading comprehension (OpenBookQA (Mihaylov et al., 2018)), and instruction following (AlpacaEval 2.0 (Dubois et al., 2024)), without using any target-task labels.

DDCF outperforms the base models by an average of +18.0 points and achieves the highest overall average (65.0) among all methods. On average, it improves commonsense reasoning by +6.2 over Base, delivers state-of-the-art reading comprehension on Qwen2.5 and Falcon (+9.1 avg), and shows the largest gains in instruction following (+38.7), surpassing Random and DiSF on stronger backbones. These results underscore DDCF's broad generalization ability beyond its intended domain.