SmallToLarge (S2L): Scalable Data Selection for Fine-tuning Large Language Models by Summarizing Training Trajectories of Small Models

Yu Yang [1] Siddhartha Mishra [1] Jeffrey Chiang [2] Baharan Mirzasoleiman [1]

Abstract

Despite the effectiveness of data selection for large language models (LLMs) during pretraining and instruction fine-tuning phases, improving data efficiency in supervised fine-tuning (SFT) for specialized domains poses significant challenges due to the complexity of fine-tuning data. To bridge this gap, we introduce an effective and scalable data selection method for SFT, SmallToLarge (S2L), which leverages training trajectories from small models to guide the data selection for larger models. We demonstrate through extensive experiments that S2L significantly improves data efficiency in SFT for mathematical problem-solving, reducing the training data to just 11% of the original MathInstruct dataset (Yue et al., 2023) to match full dataset performance while outperforming state-of-the-art data selection algorithms by an average of 4.7% across 6 in- and out-domain evaluation datasets. Remarkably, selecting only 50K data for SFT, S2L achieves a 32.7% accuracy on the challenging MATH (Hendrycks et al., 2021) benchmark, improving Phi-2 (Li et al., 2023b) by 16.6%. In clinical text summarization on the MIMIC-III dataset (Johnson et al., 2016), S2L again outperforms training on the full dataset using only 50% of the data. Notably, S2L can perform data selection using a reference model $40\times$ smaller than the target model, proportionally reducing the cost of data selection.

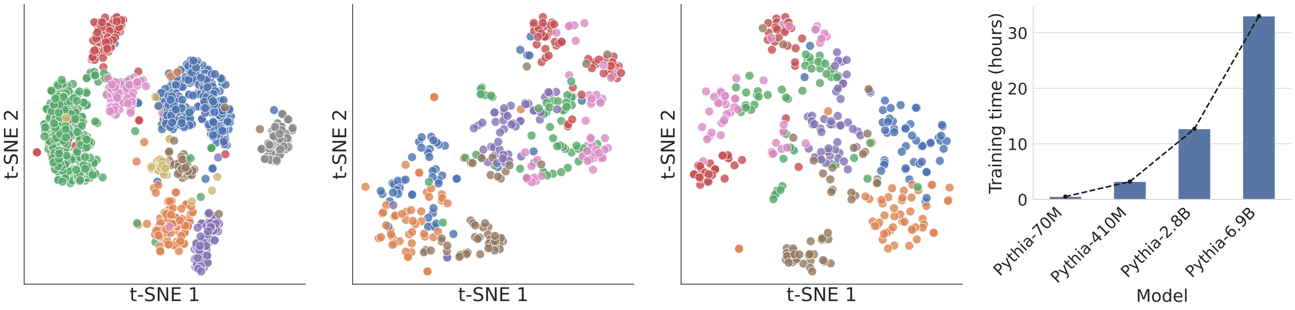
1. Introduction

In recent years, large language models (LLMs) have revolutionized artificial intelligence by demonstrating an unprecedented ability to understand and generate human language (Brown et al., 2020). Applications requiring a balanced performance across the full spectrum of natural text, such as open-ended dialogue (Thoppilan et al., 2022; Touvron et al., 2023) or instruction following (Ouyang et al., 2022; Wei et al., 2022a), favor generalist models. However, to maximize performance in a specialized domain, such as mathematics (Azerbayev et al., 2023; Luo et al., 2023a; Yu et al., 2023; Liu et al., 2023; Yue et al., 2023), code (Roziere et al., 2023; Luo et al., 2023b), medicine (Singhal et al., 2023a;b; Cheng et al., 2023), or finance (Wu et al., 2023a; Cheng et al., 2023), models fine-tuned on domain data offer superior capabilities over generalist models (Jang et al., 2023).

Recent research indicates that LLMs benefit more from training for additional epochs on carefully curated data rather than on larger, uncurated ones during pretraining (Tirumala et al., 2023) and instruction fine-tuning (Zhou et al., 2023a), making data selection one of the most promising means to unlock the next level of LLMs' language capability. Yet, maximizing the data efficiency in supervised fine-tuning (SFT) for specialized domains remains a challenge. Firstly, heuristic approaches that are effective in the instruction fine-tuning stage, like manual curation (Zhou et al., 2023a) or using advanced models such as GPT-4 for dataset evaluation (Chen et al., 2024), are less reliable due to the need for specialized knowledge and become costly with large volumes of uncurated fine-tuning data. Beyond these heuristic methods, other approaches rely on generating representations for each training example using a reference model, often utilizing metrics like perplexity (Marion et al., 2023), confidence (Swayamdipta et al., 2020; Varshney et al., 2022), or hidden states (Abbas et al., 2023; Tirumala et al., 2023; Yang et al., 2023c; Bhatt et al., 2024) as features. However, these techniques also fall short in SFT for specialized domains for two reasons: (1) the significant shift in data characteristics during SFT can render these traditional feature representations less informative (Figure 1), and (2) the computation and memory demands associated with generating representations for each training become prohibitive as these specialized domains often require larger models, some with up to 540 billion parameters (Chowdhery et al., 2023; Singhal et al., 2023a), leading to substantial scalability challenges (Figure 1d).

[1] Department of Computer Science, University of California, Los Angeles, United States [2] Department of Computational Medicine, University of California, Los Angeles, United States. Correspondence to: Yu Yang <yuyang@cs.ucla.edu>, Baharan Mirzasoleiman <baharan@cs.ucla.edu>.

Preprint.



(a) Hidden states of the Pile on pretrained Pythia-410M

(b) Hidden states of MathInstruct on pretrained Pythia-410M

(c) Hidden states of MathInstruct on fully fine-tuned Pythia-410M

(d) Increase in training time as the size of the model scales up

Figure 1. Existing data selection methods often rely on the feature representation from a reference model, which makes their effectiveness very sensitive to how well they have been trained on the target dataset (Marion et al., 2023). For supervised fine-tuning (SFT), while the pretrained model can provide good feature representations for natural language and separate topics in different colors (Figure 1a), it cannot provide a good feature representation if the fine-tuning data has a distribution shift from the pretraining (Figure 1b), which is usually the case when the data is from specialized domains. Even after fine-tuning, the improvement in the feature representation is limited (Figure 1c). Meanwhile, the cost of training a reference model that can be used to get feature representation for data selection increases significantly as the model size increases, making the deployment of data selection for large language models expensive.

To tackle the challenges of data selection in SFT for specialized domains, we present SMALLTOLARGE (S2L), a more effective and scalable data selection method. S2L operates by first gathering training trajectories for each training example using a smaller model. These trajectories are then clustered, and a balanced selection is made from these clusters. This method is grounded in the findings of recent research (Xia et al., 2023), which reveals consistent training dynamics across models of various sizes. Such consistency validates the use of smaller models as effective proxies for data selection in larger models. Additionally, S2L's balanced sampling from clusters ensures data with all learning patterns, some of which might be missed in uniform or weighted sampling, are adequately represented.

To validate S2L's effectiveness, we applied it to the challenging tasks of SFT for mathematical problem-solving and clinical text summarization. Our experiments on MathInstruct (Yue et al., 2023) have shown that S2L can significantly reduce the required training data size to just 11% of the original dataset while still matching the performance levels of the full dataset, outperforming current state-of-the-art one-shot and online data selection algorithms by an average of 4.7% across 6 in- and out-domain evaluation datasets. Remarkably, on the MATH benchmark (Hendrycks et al., 2021), S2L attained a 32.7% accuracy with just 50K data points, improving the best open-sourced model under 3 billion parameters, Phi-2, by 16.6%. For clinical text summarization tasks on the MIMIC-III (Johnson et al., 2016) dataset, S2L outperforms training on the complete dataset, using only half the data. Unlike the existing methods that require training and getting features from large models, S2L achieves superior data efficiency using a model with as few as 70 million parameters, which is 40x smaller than the largest target model we train with 2.8 billion parameters.

2. Related Work

Foundations of Data Selection. Data selection has been well studied for small models and classification tasks. There are one-shot algorithms that select data based on rankings of the proposed training statistics, for example, the L2-norms of error and gradient vectors (EL2N and GraNd) (Paul et al., 2021), confidence and its variability across epochs (Swayamdipta et al., 2020), and the number of times each example is learned but then forgot at the subsequent training step (Toneva et al., 2019). Besides these heuristic indicators, there are embedding-based pruning algorithms (Sorscher et al., 2022) and online selection algorithms with theoretical performance guarantees for efficiency (Mirzasoleiman et al., 2020; Killamsetty et al., 2021a;b; Pooladzandi et al., 2022; Yang et al., 2023b) and robustness (Yang et al., 2022; 2023a; Deng et al., 2023). Coleman et al. proposed to use the intermediate feature representation of a small proxy model to select data for image classification. Most recently, data selection has shown great potential in more substantial training speedup when implemented on near-storage hardware (Prakriya et al., 2023), and data selection beyond supervised learning of image data, e.g., for self-supervised learning (Joshi & Mirzasoleiman, 2023) and multimodal learning (Abbas et al., 2023; Mahmoud et al., 2023), also emerged.

Data Efficient Training of Large Language Models. For the pre-training of LLMs, Marion et al. studied data quality indicators including Perplexity, Error L2-Norm (EL2N) (Paul et al., 2021), and memorization ranking (Biderman et al., 2023a), and found training on examples with middle Perplexity rankings outperforms training on examples selected based on the other two metrics, and sometimes even outperforms training on the entire dataset. Tirumala et al. uses pre-trained model embeddings to select data for

LLM pre-training. The proposed algorithm, D4, first applies an embedding-based data de-duplication algorithm (Abbas et al., 2023) and then discards data points that are the closest to the K-Means cluster centroids in the embedding space (Sorscher et al., 2022) to ensure diversity.

On fine-tuning LLMs, existing work on data efficiency primarily focused on manually curating high-quality instructions (Zhou et al., 2023a), or using strong closed-source models (e.g., GPT-4 (Achiam et al., 2023) or ChatGPT) to rate the quality of each training example (Eldan & Li, 2023; Li et al., 2023a; Chen et al., 2024). Bhatt et al. implemented an experimental design framework to evaluate the existing data selection methods for instruction fine-tuning of LLMs and found selecting facility locations based on hidden representations (i.e., embeddings) is the most effective. As the only data selection algorithm for specialized domains, SCIP (Yang et al., 2023c) focuses on pruning low-quality code data for training code LLMs. Since it relies on breaking the code syntax to understand the characteristics of low-quality code in the embedding (i.e, hidden states) space, adapting SCIP to domains other than Python code data is non-trivial.

Adapting Large Language Models for Specialized Domains. The rapid development of large language models (LLMs) gives rise to new state-of-the-art models in specialized domains. For mathematical reasoning, Galactica (Taylor et al., 2022), MINERVA (Lewkowycz et al., 2022) and Llemma (Azerbayev et al., 2023) continue to train an LLM on large-scale math-related web data to improve a model's general scientific reasoning; WizardMath (Luo et al., 2023a) and TinyGSM (Liu et al., 2023) fine-tune LLMs using supervised data. Similarly for medical LLMs, Cheng et al. continued training pre-trained LLMs on medical text, and (Singhal et al., 2023a;b) fine-tuned PaLM with instruction prompt tuning on medical domain data.

3. Problem Formulation

LLM Fine-tuning Objective. Consider a transformer-based language model, parameterized by $\boldsymbol{\theta}$, and denoted as $p_{\boldsymbol{\theta}}$. This model, when provided with a sequence of prompt tokens $\mathbf{x} = (x_1, \ldots, x_M)$, generates a sequence of response tokens $\mathbf{y} = (y_1, \ldots, y_L)$. The conditional probability of generating $\mathbf{y}$ given $\mathbf{x}$ is then formulated as

$$p_{\boldsymbol{\theta}}(\mathbf{y}|\mathbf{x}) = \prod_{l=1}^{L} p_{\boldsymbol{\theta}}(y_l|\mathbf{y}_{1:l-1}, \mathbf{x}). \tag{1}$$

Note that $\mathbf{y}_{1:0}$ is an empty sequence. To adapt the pre-trained LLM for a specialized domain of distribution $\mathcal{D}$, supervised fine-tuning (SFT) is usually employed with a domain-specific training dataset $D_{\text{train}} = \{(\mathbf{x}, \mathbf{y})_i\}_{i=1}^{n} \sim \mathcal{D}$ containing pairs of prompt $\mathbf{x}$ and annotated response $\mathbf{y}$. The fine-tuning objective is thus to minimize the following negative log likelihood loss, expressed as:

$$\min_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}, D_{\text{train}}) = -\frac{1}{n} \sum_{(\mathbf{x},\mathbf{y})_i \in D_{\text{train}}} \big[\log p_{\boldsymbol{\theta}}(\mathbf{y}|\mathbf{x})\big]. \tag{2}$$

Data Selection Objective. In a general setting for data selection, we consider a target language model $p_{\boldsymbol{\theta}}$ with parameters $\boldsymbol{\theta}$. Given a fixed data budget B, which constrains the number of data points that can be used for training, our objective is to select a subset $S \subseteq D_{\text{train}}$ that optimizes the performance of the target model on the full training set D_{train}. This objective can be formally expressed as:

$$\min_{S \subseteq D_{\text{train}}, |S| \leq B} |S| \quad \text{s.t.} \quad \min \mathcal{L}(\boldsymbol{\theta}_s, D_{\text{train}}) \tag{3}$$

where $\boldsymbol{\theta}_s = \arg\min_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}, S)$ is the model trained on the subset S. In practice, the subset S is selected based a reference model $r_{\boldsymbol{\phi}}$ parameterized with $\boldsymbol{\phi}$, which generate representations for each data point $(\mathbf{x}, \mathbf{y})_i \in D_{\text{train}}$ as $r_{\boldsymbol{\phi}}((\mathbf{x}, \mathbf{y})_i)$. These representations are utilized by a data selection algorithm to produce S.

In existing data selection algorithms, $\boldsymbol{\phi}$ is commonly either weights of the pre-trained target model or a target model that has been fully trained on the dataset D_{train} with parameters $\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}, D_{\text{train}})$. However, as evidenced by Figure 1, representations generated by either the pretrained or the fine-tuned model may not always be good enough for data in specialized domains that involve a data distribution shift towards unique notations and terminologies.

4. Methodology

Our proposed method, **SMALLTOLARGE (S2L)**, focuses on optimizing the data selection objective by leveraging *training trajectories* of training examples on a small model.

Training Trajectory. Let $\boldsymbol{\phi}^{(t)}$ be the parameter of a small LM during the training on D_{train} at time $t, t \in \mathcal{T} = \{1, ..., T\}$. The representation of each data point used in S2L is a sequence of losses recorded during training the reference model $(r_{\boldsymbol{\phi}^{(t)}})_{t=1}^{T}$, where

$$r_{\boldsymbol{\phi}^{(t)}}(\mathbf{x}, \mathbf{y}) = -\log p_{\boldsymbol{\phi}^{(t)}}(\mathbf{y}|\mathbf{x}), \tag{4}$$

and T is the length of the training trajectory. Note that $\boldsymbol{\phi}^{(t)}$ is trained for a fixed number of iterations from $\boldsymbol{\phi}^{(t-1)}$, thereby saved sequentially during the training of $\boldsymbol{\phi}$.

Cluster-based Data Selection. Once the loss trajectories are recorded, we apply a clustering algorithm to group examples based on the similarity of their loss trajectories. This results in a set of clusters $\{C_1, C_2, \ldots, C_k\}$, where each cluster C_i contains examples with similar loss behavior over

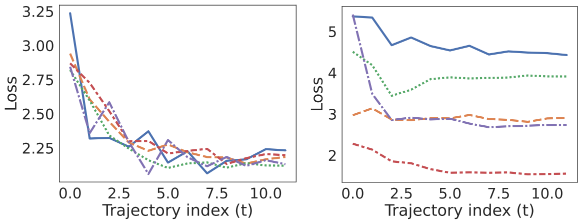
Algorithm 1 Data Selection Based on Training Trajectories

```
Require: Training dataset $D_{\text{train}}$ with corresponding training trajectories, a fixed data budget $B$, number of clusters $K$.
Ensure: Subset $S$ optimizing Equation (3).
Initialize $S$ as an empty set.
Cluster examples in $D_{\text{train}}$ based on their training trajectories to form $\mathcal{C} = \{C_1, C_2, \ldots, C_K\}$, sorted by size.
for each cluster $C_k$ in $\mathcal{C}$ do
    Calculate $R_k$, the number of examples to randomly sample from $C_k$, i.e., $R_k = (B - |S|)/(K - k + 1)$.
    if $|C_k| \leq R_k$ then
        $S \leftarrow S \bigcup C_k$.
    else
        $S \leftarrow S \bigcup S_k$, where $S_k \subset C_k, |S_k| = R_k$
    end if
end for
Return $S$
```

the training epochs:

$$C_i = \{(\mathbf{x}, \mathbf{y}) | i = \arg\min_j d(\mathbf{L}(\mathbf{x}, \mathbf{y}), \mathbf{L}C_j)\}, \tag{5}$$

where $\mathbf{L}(\mathbf{x}, \mathbf{y}) = (r_{\boldsymbol{\phi}^{(1)}}(\mathbf{x}, \mathbf{y}), \ldots, r_{\boldsymbol{\phi}^{(T)}}(\mathbf{x}, \mathbf{y}))$ denotes the loss trajectory of the example $(\mathbf{x}, \mathbf{y})$, $\mathbf{L}_{C_i}$ is the centroid of the loss trajectories in cluster C_i, and $d(\cdot, \cdot)$ is a distance metric, such as Euclidean distance, used for clustering. As shown in Figure 2, clustering algorithms can effectively find groups of examples with similar training dynamics. With the clusters formed, the data selection strategy prioritizes selecting examples from smaller clusters while equally selecting examples from larger clusters, as detailed in Algorithm 1.

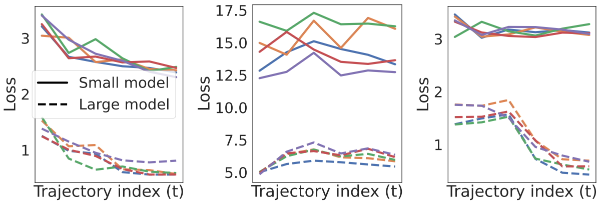


(a) In the same cluster. (b) In different clusters.

Figure 2. Training trajectories ($r_{\boldsymbol{\phi}^{(t)}}(\mathbf{x}, \mathbf{y})$) of 5 random examples in the same cluster vs. in different random clusters with k-means clustering. Examples in the same clusters have very similar training trajectories (Figure 2a) while the training trajectories of examples in different clusters are very different (Figure 2b).

Intuition Behind Loss Trajectory Clustering. Loss trajectories encapsulate information about how the model learns from specific data points. Consider two training examples $(\mathbf{x}, \mathbf{y})_i$ and $(\mathbf{x}, \mathbf{y})_j$ from the dataset D_{train}. If when the model effectively learns from example $(\mathbf{x}, \mathbf{y})_i$, reflected in a decreasing loss trajectory $r_{\boldsymbol{\phi}^{(t)}}(\mathbf{x}, \mathbf{y})_i$ over iterations, a correlated decrease in $r_{\boldsymbol{\phi}^{(t)}}(\mathbf{x}, \mathbf{y})_j$ is observed, it is likely the two examples encode the same knowledge or skills at the same difficulty level. Therefore, by identifying clusters of examples with similar training trajectories and sampling examples from different clusters to cover all topics and difficulties, one can effectively train the model with fewer data points without significantly compromising the training quality. In Section 5.2.3 (Figure 5), we will show quantitatively such similarity between examples in the same cluster using the MathInstruct (Yue et al., 2023) dataset.

Small-to-Large Data Selection. Training a reference model that is of the same size as the target model and getting the feature representation for each example in D_{train} can be costly when the size of the target model is large. Inspired by a recent finding that the training dynamic of most examples is consistent across models of various sizes (Xia et al., 2023), in Figure 3, we empirically show that we can find groups of examples with similar training dynamics on larger models by clustering the training trajectories of D_{train} on a small model, even when the trends are different.



(a) (b) (c)

Figure 3. Examples in the same clusters of training trajectories on a small model (Pythia-70M) also have similar training trajectories on a large model (Pythia-2.8B), even if the trends may not be the same on both models.

5. Experiments

In this section, we present the comprehensive experiments conducted to evaluate the efficacy of the proposed data selection method, SmallToLarge (S2L), across two challenging domains (mathematical reasoning and clinical text summarization) and various model sizes (from 410M to 2.8B) and architectures (Pythia, Phi). Additionally, we incorporate ablation studies to further understand S2L, including the effect of length and timing of the trajectory. We fine-tune all the models with the Huggingface transformers library (Wolf et al., 2020) with Fully Sharded Data Parallel (FSDP) (Zhao et al., 2023) on 4 48G NVIDIA RTX A6000.

5.1. Baselines

We systematically compare S2L against a comprehensive set of open-sourced data selection methods. These methods are categorized based on the type of representation they use and selected as the most representative or best-performing methods as identified in prior work. These include: (1)

Random Sampling; (2) **Least Confidence** selection (Bhatt et al., 2024); (3) selection based on **Middle Perplexity** (Marion et al., 2023); (4) **High Learnability**, determined by the loss decrease before and after full fine-tuning (Zhou et al., 2023b); and (5) **Facility Locations** selection based on hidden states (Bhatt et al., 2024). Additionally, we incorporate two online selection techniques: (6) **DiverseEvol** (Wu et al., 2023b), which progressively adds examples with distinct embeddings, and the (7) **Confidence Curriculum** proposed by Varshney et al., which uses the confidence score averaged over the past few epochs to select examples with decreasing confidence during the training while mixing in a certain fraction of higher confidence examples selected in the previous rounds.

Given that the optimal reference model may vary for each one-shot selection method, we ensure a fair comparison by adopting the approach used in (Marion et al., 2023), which runs each method with both the fully fine-tuned target model and an early fine-tuning checkpoint as the reference model. We report the best results from these setups.

5.2. Mathematical Reasoning

5.2.1. Training Settings

For mathematical reasoning, we focus on fine-tuning using the **MathInstruct** dataset (Yue et al., 2023) with 262,040 training examples for its comprehensive coverage of diverse mathematical fields and its capability in training models to achieve state-of-the-art performance on the standard evaluation benchmarks (Section 5.2.2). We employ the open-source model suites Pythia (Biderman et al., 2023b) and a current state-of-the-art model Phi-2 (Li et al., 2023b) as our base models to validate our S2L algorithm across different scales and directly compare its performance against the state-of-the-art. Following the setup used in (Yue et al., 2023), we adopt a training regimen with a learning rate of 2e-5, a batch size of 128, a maximum length of 512, and a cosine scheduler with a 3% warm-up period. For all experiments on MathInstruct, we standardize the number of training steps to correspond to 3 epochs on the full dataset, also aligning with the setting in (Yue et al., 2023). This allows us to maintain a consistent training schedule across different methods and data budgets, ensuring fair and accurate comparisons of the quality of data selected by different algorithms.

Due to memory and computational constraints, for Facility Locations and DiverseEvol, we calculate pairwise similarity and perform greedy selection on a per-data-source basis. We found this per-source selection approach also yields benefits for S2L as different data sources within MathInstruct exhibit distinct common patterns in their training trajectories. Therefore, we implement S2L also on a per-source basis for MathInstruct, and recommend applying S2L per source when dealing with datasets composed of multiple data sources.

5.2.2. Evaluation Settings

Datasets. We follow the framework established in (Yue et al., 2023) for a comprehensive assessment using several well-regarded datasets, including in-domain and out-of-domain datasets. For the in-domain datasets, we consider **GSM8K** (Cobbe et al., 2021), **MATH** (Hendrycks et al., 2021), and **NumGLUE** (Mishra et al., 2022). For the out-of-domain datasets, we consider **SVAMP** (Patel et al., 2021), **Mathematics** (Davies et al., 2021), **SimulEq** (Koncel-Kedziorski et al., 2016). These datasets collectively span a diverse range of mathematical subjects, such as algebra, probability, number theory, calculus, and geometry. Additionally, some questions in these datasets require the application of commonsense, reading comprehension, and multi-step reasoning. All questions are open-formed.

Metric. We use the standard evaluation metric for open-formed questions, **exact match**, which measures the model's accuracy by comparing its generated answers against the correct solutions. For an answer to be considered correct, it must match the reference solution precisely.

5.2.3. Results

We present a comprehensive analysis of S2L in comparison to the full training baseline, data selection baselines, and the performance of the pretrained model (as a baseline to demonstrate the impact of fine-tuning).

Scaling the Data: SOTA algorithms fail with small data budgets while S2L stands out across data scales. In Table 1, we compare S2L against the baselines from Section 5.1 on Pythia-410M across varying data sizes. The training trajectories used by S2L are based on Pythia-70M, a model approximately 6x smaller than Pythia-410M. With the same number of training steps as the full training, S2L exceeds the full dataset's performance using only 30K examples, only 11% of the full dataset. When the data budget is increased to 100K examples, S2L outperforms the full training by 4% in exact-match accuracy. It leads the runner-up baselines by an average of 4.7%, 4.6% and 2.4% with data budget 30K, 50K and 100K across all six evaluation datasets. We observe that, in this challenging fine-tuning setting with a domain shift, while state-of-the-art data selection algorithms like Facility Locations (Bhatt et al., 2024) and High Learnability (Zhou et al., 2023b) have decent performance with a large enough data budget (i.e., 100K), all SOTA algorithms except S2L cannot even outperform the random sampling baseline when the allowed data size is small (i.e., 30K). Unlike the existing algorithms, S2L consistently outperforms all baselines and even full training across all data sizes. Note that compared to the runner-up

Table 1. Accuracies (↑) on in-domain and out-of-domain datasets using Pythia-410M. Data size refers to the total number of unique training examples used. All experiments in this table share the same total training steps and learning rate schedule (see Section 5.2.1). Results surpassing full training are highlighted in bold. The ranges for confidence, perplexity, and learnability are chosen according to the best-performing intervals reported in prior research (Section 5.1). For one-shot selection methods (excluding S2L), we use representations from either step 1000 or the end of fine-tuning Pythia-410M on MathInstruct, and report the better results, while S2Lselects data based on training trajectories from a smaller Pythia-70M model.

	Data	In-domain				Out-domain			
Selection	Size	GSM8K	MATH	NumGLUE	**Avg**	SVAMP	Mathematics	SimulEq	**Avg**
(Pretrained)		2.0	1.6	10.1	4.6	2.3	2.5	1.4	3.3
Random	30K	3.3	6.2	15.0	8.2	15.0	15.1	1.6	9.4
	50K	3.7	6.4	18.1	9.4	17.0	11.6	1.2	9.7
	100K	5.9	7.6	22.0	11.8	20.5	20.8	2.7	13.3
Least Confidence	30K	2.7	1.3	18.0	7.0	13.7	3.3	1.4	6.7
	50K	2.1	1.7	21.0	8.3	14.5	3.5	1.0	7.3
	100K	2.5	3.3	23.5	9.8	20.8	6.3	3.7	10.0
Middle Perplexity	30K	5.3	3.7	16.2	8.4	14.2	8.7	1.2	8.2
	50K	3.2	5.9	20.5	9.9	18.1	11.3	**5.1**	10.7
	100K	5.4	7.2	20.9	11.2	**23.8**	15.3	3.3	12.6
High Learnability	30K	6.1	1.6	19.1	8.9	10.7	9.9	1.4	8.1
	50K	6.1	2.1	18.6	8.9	14.5	14.0	2.1	8.9
	100K	**7.4**	9.2	**29.8**	**15.5**	20.7	19.4	**10.3**	**16.1**
Facility Locations	30K	4.2	7.7	10.0	7.3	11.8	13.8	1.2	8.1
	50K	5.7	9.1	12.4	9.1	15.4	18.6	1.6	10.5
	100K	**7.4**	**10.9**	**30.5**	**16.3**	**26.2**	21.9	**9.3**	**17.7**
DiverseEvol (Online)	30K	1.9	3.6	8.8	4.8	13.9	3.0	1.6	5.5
	50K	1.6	4.2	12.0	5.9	10.6	7.3	1.9	6.3
	100K	1.3	3.8	12.9	6.0	11.8	8.4	1.0	6.5
Confidence Curriculum (Online)	30K	4.2	6.3	15.4	8.6	18.9	16.5	1.4	10.4
	50K	6.6	3.3	16.9	9.0	19.9	19.6	2.1	11.4
	100K	4.6	6.3	17.1	9.3	21.0	15.2	1.8	11.0
S2L (Ours)	30K	5.0	**10.3**	**29.4**	**14.9**	20.6	17.3	**7.8**	15.1
	50K	**7.6**	**9.7**	**28.1**	**15.1**	22.8	22.9	**5.1**	**16.0**
	100K	**9.3**	**11.8**	**33.6**	**18.3**	**26.6**	**26.5**	**12.8**	**20.1**
None	262K	6.7	9.4	26.6	14.3	23.5	23.3	3.7	15.5

algorithm in this setting, Facility Locations, the cost of S2L is much lower in both training the reference model and data selection stages (Figure 4), and therefore more scalable to both larger target models or larger data sizes.

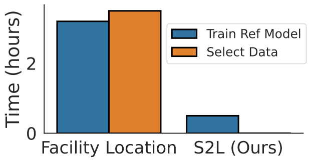


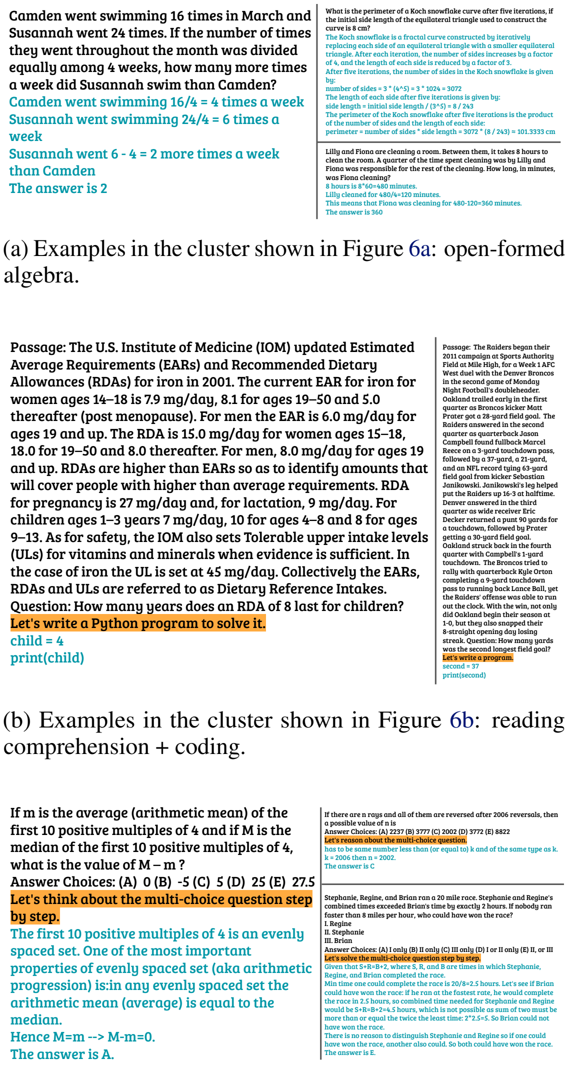
Figure 4. Wall-clock time required to train the reference model and select 100K data from MathInstruct for training Pythia-410M.

Scaling the Model: Data selected by S2L can transfer to larger models in different model suites. We also test whether this subset, chosen using Pythia-70M, can effectively train larger models beyond 410M and models outside the Pythia suite. Since larger models are easier to overfit, we randomly sample 50 examples from each data source as the validation set and use the best model with the lowest validation loss evaluated every 1000 training iterations. Our experiments with Phi-2 reveal that fine-tuning on only 50K S2L-selected data again outperforms full dataset training on the most challenging MATH (Hendrycks et al., 2021) benchmark improving the pretrained Phi-2 by 16.6% and is more data efficient than training on the full MathInstruct dataset to get the same performance.

Examples in Clusters Encode the Same Knowledge/Skill. In Figure 5, we compare actual training examples in MathInstruct that get clustered together due to their similar training trajectories on the small Pythia-70M model. We observe that examples in the same cluster are of the same type and related to the same knowledge/skill: the cluster shown in Figure 5a is comprised of open-formed algebra questions, examples in Figure 5b requires extracting useful information from long text and write programs to answer the question, and Figure 5c shows a cluster of multiple choice questions

Table 2. Accuracies (↑) across model scales and model suites. Data selected by S2L with one small model (Pythia-70M) scale up to training larger models and can also generalize to other pretrained models. Results surpassing full training are highlighted in bold.

Target Model	Fine-tuning Data	GSM8K	MATH	NumGLUE	Avg	Diff to Full
Pythia-410M	(Pretrained)	2.0	1.6	10.1	4.6	−9.7
	S2L-50K(Ours)	**7.6**	**9.7**	**28.1**	**15.1**	+0.8
	Full-262K	6.7	9.4	26.6	14.3	-
Phi-2 (2.7B)	(Pretrained)	53.4	16.1	34.9	34.8	−20.3
	S2L-50K(Ours)	66.8	**32.7**	**65.0**	54.8	−0.3
	Full-262K	68.3	32.6	64.3	55.1	-



Camden went swimming 16 times in March and Susannah went 24 times. If the number of times they went throughout the month was divided equally among 4 weeks, how many more times a week did Susannah swim than Camden?
Camden went swimming 16/4 = 4 times a week
Susannah went swimming 24/4 = 6 times a week
Susannah went 6 - 4 = 2 more times a week than Camden
The answer is 2

What is the perimeter of a Koch snowflake curve after five iterations, if the initial side length of the equilateral triangle used to construct the curve is 8 cm?
The Koch snowflake is a fractal curve constructed by iteratively replacing each side of an equilateral triangle with a smaller equilateral triangle. After each iteration, the number of sides increases by a factor of 4, and the length of each side is reduced by a factor of 3.
After five iterations, the number of sides in the Koch snowflake is given by:
number of sides = 3 * (4^5) = 3 * 1024 = 3072
The length of each side after five iterations is given by:
side length = initial side length / (3^5) = 8 / 243
The perimeter of the Koch snowflake after five iterations is the product of the number of sides and the length of each side:
perimeter = number of sides * side length = 3072 * (8 / 243) ≈ 101.3333 cm

Lilly and Fiona are cleaning a room. Between them, it takes 8 hours to clean the room. A quarter of the time spent cleaning was by Lilly and Fiona was responsible for the rest of the cleaning. How long, in minutes, was Fiona cleaning?
8 hours is 8*60=480 minutes.
Lilly cleaned for 480/4=120 minutes.
This means that Fiona was cleaning for 480-120=360 minutes.
The answer is 360

(a) Examples in the cluster shown in Figure 6a: open-formed algebra.

Passage: The U.S. Institute of Medicine (IOM) updated Estimated Average Requirements (EARs) and Recommended Dietary Allowances (RDAs) for iron in 2001. The current EAR for iron for women ages 14–18 is 7.9 mg/day, 8.1 for ages 19–50 and 5.0 thereafter (post menopause). For men the EAR is 6.0 mg/day for ages 19 and up. The RDA is 15.0 mg/day for women ages 15–18, 18.0 for 19–50 and 8.0 thereafter. For men, 8.0 mg/day for ages 19 and up. RDAs are higher than EARs so as to identify amounts that will cover people with higher than average requirements. RDA for pregnancy is 27 mg/day and, for lactation, 9 mg/day. For children ages 1–3 years 7 mg/day, 10 for ages 4–8 and 8 for ages 9–13. As for safety, the IOM also sets Tolerable upper intake levels (ULs) for vitamins and minerals when evidence is sufficient. In the case of iron the UL is set at 45 mg/day. Collectively the EARs, RDAs and ULs are referred to as Dietary Reference Intakes. Question: How many years does an RDA of 8 last for children?
Let's write a Python program to solve it.
child = 4
print(child)

Passage: The Raiders began their 2011 campaign at Sports Authority Field at Mile High, for a Week 1 AFC West duel with the Denver Broncos in the second game of Monday Night Football's doubleheader. Oakland trailed early in the first quarter as Broncos kicker Matt Prater got a 28-yard field goal. The Raiders answered in the second quarter as quarterback Jason Campbell found fullback Marcel Reece on a 3-yard touchdown pass, followed by a 37-yard, a 21-yard, and an NFL record tying 63-yard field goal from kicker Sebastian Janikowski. Janikowski's leg helped put the Raiders up 16-3 at halftime. Denver answered in the third quarter as wide receiver Eric Decker returned a punt 90 yards for a touchdown, followed by Prater getting a 30-yard field goal. Oakland struck back in the fourth quarter with Campbell's 1-yard touchdown. The Broncos tried to rally with quarterback Kyle Orton completing a 9-yard touchdown pass to running back Lance Ball, yet the Raiders' offense was able to run out the clock. With the win, not only did Oakland begin their season at 1-0, but they also snapped their 8-straight opening day losing streak. Question: How many yards was the second longest field goal?
Let's write a program.
second = 37
print(second)

(b) Examples in the cluster shown in Figure 6b: reading comprehension + coding.

If m is the average (arithmetic mean) of the first 10 positive multiples of 4 and if M is the median of the first 10 positive multiples of 4, what is the value of M – m ?
Answer Choices: (A) 0 (B) -5 (C) 5 (D) 25 (E) 27.5
Let's think about the multi-choice question step by step.
The first 10 positive multiples of 4 is an evenly spaced set. One of the most important properties of evenly spaced set (aka arithmetic progression) is:in any evenly spaced set the arithmetic mean (average) is equal to the median.
Hence M=m --> M-m=0.
The answer is A.

If there are n rays and all of them are reversed after 2006 reversals, then a possible value of n is
Answer Choices: (A) 2237 (B) 3777 (C) 2002 (D) 3772 (E) 8822
Let's reason about the multi-choice question.
has to be same number less than (or equal to) k and of the same type as k.
k = 2006 then n = 2002.
The answer is C

Stephanie, Regine, and Brian ran a 20 mile race. Stephanie and Regine's combined times exceeded Brian's time by exactly 2 hours. If nobody ran faster than 8 miles per hour, who could have won the race?
I. Regine
II. Stephanie
III. Brian
Answer Choices: (A) I only (B) II only (C) III only (D) I or II only (E) II, or III
Let's solve the multi-choice question step by step.
Given that S+R=B+2, where S, R, and B are times in which Stephanie, Regine, and Brian completed the race.
Min time one could complete the race is 20/8=2.5 hours. Let's see if Brian could have won the race: if he ran at the fastest rate, he would complete the race in 2.5 hours, so combined time needed for Stephanie and Regine would be S+R=B+2=4.5 hours, which is not possible as sum of two must be more than or equal the twice the least time: 2*2.5=5. So Brian could not have won the race.
There is no reason to distinguish Stephanie and Regine so if one could have won the race, another also could. So both could have won the race.
The answer is E.

(c) Examples in the cluster shown in Figure 6c: multiple-choice + multi-step reasoning.

Figure 5. Examples (MathInstruct) with similar training trajectories have the same question format. Intuitively, training on one of the questions helps the model learn how to solve the other questions with the same format, and therefore their losses have similar trends. Sampling from all clusters and prioritizing the small ones help ensure different types of examples can be learned. See the full version in Appendix B.

Table 3. A synthetic radiology report (MRI of the brain), generated by the GPT-4 model (Achiam et al., 2023) to demonstrate the typical data format and content used in the clinical text summarization task. It is not suitable for clinical or diagnostic use.

Findings	The brain parenchyma demonstrates normal morphology with no evidence of mass effect or midline shift. No acute infarcts are seen on diffusion-weighted images. There are no signs of intracranial hemorrhage. Mild generalized cerebral atrophy is noted. The ventricles and sulci appear within normal limits for the patient's age. The pituitary gland and sella turcica are unremarkable. There are no abnormal signal intensities within the brain parenchyma. The orbits, paranasal sinuses, and mastoid air cells are clear.
Impression	Normal MRI of the brain. Mild cerebral atrophy, likely age-related. No acute intracranial pathology.

that require multi-step reasoning. Therefore, by sampling from different clusters, we make sure the selected examples cover the knowledge required for all topics and skills required for all types of questions.

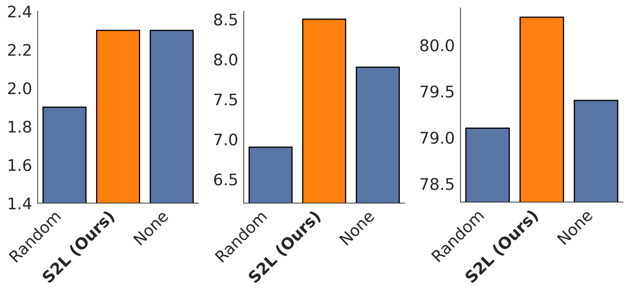
5.3. Clinical Text Summarization

S2L can improve data efficiency not only for fine-tuning data in the mathematical domain but also for other specialized domains. In this subsection, we focus on the task of clinical text summarization within the context of radiology reports. This task involves processing the detailed analysis and results listed in the findings section of a radiology report and distilling them into a concise impression section. Such summaries are crucial for providing clinicians with quick and actionable insights from radiological studies.

Dataset & Setup. We use the **MIMIC-III** dataset (Johnson et al., 2016), a comprehensive collection of radiology reports and findings authored by attending physicians in routine clinical care. We use the same preprocessing procedures as (Delbrouck et al., 2023; Demner-fushman et al., 2023) to extract the findings and impression sections and remove invalid reports. Given that access to MIMIC-III requires specific credentials, we provide a synthetic example of a radiology report generated by GPT-4 (Achiam et al., 2023) for illustrative purposes in Table 3. We employ the Pythia-1B model and keep the training setting consistent with the mathematical reasoning task.

Evaluation. Our evaluation of the generated clinical summaries on the test split of the MIMIC-III dataset uses three key metrics, as suggested in (Van Veen et al., 2023; Tu et al., 2023). (1) **BLEU** (Papineni et al., 2002) calculates the overlap between the generated and reference texts using short sequences of words. (2) **ROUGE-L** (Lin, 2004) examines the longest sequence of words that appears in both texts. (3) **BERTScore** (Zhang et al., 2020), utilizing BERT's contextual embeddings, evaluates how semantically similar the generated text is to the reference. These metrics together offer a comprehensive evaluation, from basic word-level accuracy to deeper semantic alignment, ensuring our summaries are not only precise in language but also meaningful and coherent in the context of clinical information. We compare S2L to random selection, a surprisingly strong baseline as evidenced in Table 1, to check the validity of the data selection problem on this dataset and then compare it to training on the full dataset to assess its effectiveness.

Results. We compare using 30K examples selected by random vs. selected through S2L. Even with only half of the data, the model trained with S2L selected data achieves similar BLEU and significantly higher ROUGE-L and BERTSCore compared to the model trained on the entire 61.5K data. Meanwhile, training on randomly selected 30K examples performs worse than training on the full dataset on all 3 metrics. These results together demonstrate S2L's effectiveness.



(a) BLEU (b) ROUGE-L (c) BERTScore

Figure 6. Performance (↑) of models trained on (1) **random**ly selected 30K examples, (2) **S2L** selected 30K examples, and (3) full 61K examples (**none**) evaluated with 3 different metrics. The minimum value on the y-axis is the performance of the model before fine-tuning. S2L improves the data efficiency for the clinical text summarization task by outperforming training on the full dataset with only less than half of the data.

5.4. Ablation Studies

We conduct ablation studies on MathInstruct and Pythia-410M to further understand the best practices for using S2L.

S2L is robust w.r.t. the length of the trajectories but can benefit more from longer trajectories. Figure 7 compares models trained with data selected by S2L based on training trajectories of different lengths. The shorter trajectories are derived from a uniform sample of the longer trajectories. From the small slopes of the lines, we can conclude that S2L is relatively robust to the length of the training trajectories. Nevertheless, we can also observe a slight increase in the performance on some of the datasets when longer trajectories are used, so having longer trajectories is still preferred.

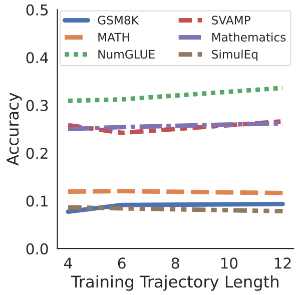


Figure 7. S2L is robust to the length of training trajectories.

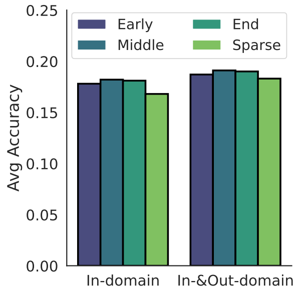


Figure 8. S2L prefers dense trajectories over sparse ones.

S2L can utilize training trajectories collected at any stage of training but preferably denser ones. With the length of the trajectories fixed to 4, we can observe in Figure 8 that denser trajectories recorded at any training stage (early, middle, or late) are more helpful for S2L than trajectories recorded sparsely throughout the training.

6. Conclusion and Limitations

In this work, we introduced SmallToLarge (S2L), a scalable data selection method to improve the data efficiency of supervised fine-tuning (SFT) for large language models (LLMs) in specialized domains. By clustering data points based on their training dynamics on smaller models and balanced sampling from all clusters, S2L significantly reduces the required training data size without compromising performance compared to using the entire training dataset. Our comprehensive experiments across the mathematical problem-solving and clinical text summarization domains demonstrate the effectiveness of S2L.

Our study does come with its limitations. S2L has been only tested within two domains, mathematics and medicine, and on models up to 3 billion parameters, constrained by our computational resources. Additionally, our experiments employed a fixed training schedule across all methods without further optimization or hyperparameter tuning for each method, including S2L. This unified approach, while it ensures a fair comparison, may not fully capture the potential performance improvement that could be achieved with more tailored training strategies. We encourage further research to extend the application of S2L across a broader spectrum of domains and investigate the impact of hyperparameter tuning on its effectiveness.

Impact Statements

This paper introduces a data selection method for large language models (LLMs), aiming to enhance the data efficiency in the supervised fine-tuning (SFT) of these models.

Positive Impacts: Our method, by reducing the data requirements for training LLMs, can make fine-tuning LLMs more effective and accessible. This could lead to broader participation in AI research and application development across various fields, including healthcare and education.

Negative Impacts: Our method does not inherently involve or encourage applications with direct negative societal impacts. The focus is on a generic improvement in the field of machine learning, particularly in the training of LLMs.

References

Abbas, A. K. M., Tirumala, K., Simig, D., Ganguli, S., and Morcos, A. S. Semdedup: Data-efficient learning at web-scale through semantic deduplication. In *ICLR 2023 Workshop on Multimodal Representation Learning: Perks and Pitfalls*, 2023.

Achiam, O. J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., Avila, R., Babuschkin, I., Balaji, S., Balcom, V., Baltescu, P., Bao, H., Bavarian, M., Belgum, J., Bello, I., Berdine, J., Bernadett-Shapiro, G., Berner, C., Bogdonoff, L., Boiko, O., Boyd, M., Brakman, A.-L., Brockman, G., Brooks, T., Brundage, M., Button, K., Cai, T., Campbell, R., Cann, A., Carey, B., Carlson, C., Carmichael, R., Chan, B., Chang, C., Chantzis, F., Chen, D., Chen, S., Chen, R., Chen, J., Chen, M., Chess, B., Cho, C., Chu, C., Chung, H. W., Cummings, D., Currier, J., Dai, Y., Decareaux, C., Degry, T., Deutsch, N., Deville, D., Dhar, A., Dohan, D., Dowling, S., Dunning, S., Ecoffet, A., Eleti, A., Eloundou, T., Farhi, D., Fedus, L., Felix, N., Fishman, S. P., Forte, J., Fulford, I., Gao, L., Georges, E., Gibson, C., Goel, V., Gogineni, T., Goh, G., Gontijo-Lopes, R., Gordon, J., Grafstein, M., Gray, S., Greene, R., Gross, J., Gu, S. S., Guo, Y., Hallacy, C., Han, J., Harris, J., He, Y., Heaton, M., Heidecke, J., Hesse, C., Hickey, A., Hickey, W., Hoeschele, P., Houghton, B., Hsu, K., Hu, S., Hu, X., Huizinga, J., Jain, S., Jain, S., Jang, J., Jiang, A., Jiang, R., Jin, H., Jin, D., Jomoto, S., Jonn, B., Jun, H., Kaftan, T., Kaiser, L., Kamali, A., Kanitscheider, I., Keskar, N. S., Khan, T., Kilpatrick, L., Kim, J. W., Kim, C., Kim, Y., Kirchner, H., Kiros, J. R., Knight, M., Kokotajlo, D., Kondraciuk, L., Kondrich, A., Konstantinidis, A., Kosic, K., Krueger, G., Kuo, V., Lampe, M., Lan, I., Lee, T., Leike, J., Leung, J., Levy, D., Li, C. M., Lim, R., Lin, M., Lin, S., Litwin, M., Lopez, T., Lowe, R., Lue, P., Makanju, A. A., Malfacini, K., Manning, S., Markov, T., Markovski, Y., Martin, B., Mayer, K., Mayne, A., McGrew, B., McKinney, S. M., McLeavey, C., McMillan, P., McNeil, J., Medina, D., Mehta, A., Menick, J., Metz, L., Mishchenko, A., Mishkin, P., Monaco, V., Morikawa, E., Mossing, D. P., Mu, T., Murati, M., Murk, O., M'ely, D., Nair, A., Nakano, R., Nayak, R., Neelakantan, A., Ngo, R., Noh, H., Long, O., O'Keefe, C., Pachocki, J. W., Paino, A., Palermo, J., Pantuliano, A., Parascandolo, G., Parish, J., Parparita, E., Passos, A., Pavlov, M., Peng, A., Perelman, A., de Avila Belbute Peres, F., Petrov, M., de Oliveira Pinto, H. P., Pokorny, M., Pokrass, M., Pong, V. H., Powell, T., Power, A., Power, B., Proehl, E., Puri, R., Radford, A., Rae, J., Ramesh, A., Raymond, C., Real, F., Rimbach, K., Ross, C., Rotsted, B., Roussez, H., Ryder, N., Saltarelli, M. D., Sanders, T., Santurkar, S., Sastry, G., Schmidt, H., Schnurr, D., Schulman, J., Selsam, D., Sheppard, K., Sherbakov, T., Shieh, J., Shoker, S., Shyam, P., Sidor, S., Sigler, E., Simens, M., Sitkin, J., Slama, K., Sohl, I., Sokolowsky, B. D., Song, Y., Staudacher, N., Such, F. P., Summers, N., Sutskever, I., Tang, J., Tezak, N. A., Thompson, M., Tillet, P., Tootoonchian, A., Tseng, E., Tuggle, P., Turley, N., Tworek, J., Uribe, J. F. C., Vallone, A., Vijayvergiya, A., Voss, C., Wainwright, C., Wang, J. J., Wang, A., Wang, B., Ward, J., Wei, J., Weinmann, C., Welihinda, A., Welinder, P., Weng, J., Weng, L., Wiethoff, M., Willner, D., Winter, C., Wolrich, S., Wong, H., Workman, L., Wu, S., Wu, J., Wu, M., Xiao, K., Xu, T., Yoo, S., Yu, K., Yuan, Q., Zaremba, W., Zellers, R., Zhang, C., Zhang, M., Zhao, S., Zheng, T., Zhuang, J., Zhuk, W., and Zoph, B. Gpt-4 technical report. 2023. URL https://api.semanticscholar.org/CorpusID:257532815.

Azerbayev, Z., Schoelkopf, H., Paster, K., Santos, M. D., McAleer, S., Jiang, A. Q., Deng, J., Biderman, S., and Welleck, S. Llemma: An open language model for mathematics. *arXiv preprint arXiv:2310.10631*, 2023.

Bhatt, G., Chen, Y., Das, A. M., Zhang, J., Truong, S. T., Mussmann, S., Zhu, Y., Bilmes, J., Du, S. S., Jamieson, K., et al. An experimental design framework for label-efficient supervised finetuning of large language models. *arXiv preprint arXiv:2401.06692*, 2024.

Biderman, S., Prashanth, U. S., Sutawika, L., Schoelkopf, H., Anthony, Q., Purohit, S., and Raf, E. Emergent and predictable memorization in large language models. *arXiv preprint arXiv:2304.11158*, 2023a.

Biderman, S., Schoelkopf, H., Anthony, Q. G., Bradley, H., O'Brien, K., Hallahan, E., Khan, M. A., Purohit, S., Prashanth, U. S., Raff, E., et al. Pythia: A suite for analyzing large language models across training and scaling. In *International Conference on Machine Learning*, pp. 2397–2430. PMLR, 2023b.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. Language models are few-shot learners. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 1877–1901. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf.

Chen, L., Li, S., Yan, J., Wang, H., Gunaratna, K., Yadav, V., Tang, Z., Srinivasan, V., Zhou, T., Huang, H., et al. Alpagasus: Training a better alpaca with fewer data. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=FdVXgSJhvz.

Cheng, D., Huang, S., and Wei, F. Adapting large language models via reading comprehension. *arXiv preprint arXiv:2309.09530*, 2023.

Chowdhery, A., Narang, S., Devlin, J., Bosma, M., Mishra, G., Roberts, A., Barham, P., Chung, H. W., Sutton, C., Gehrmann, S., et al. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113, 2023.

Cobbe, K., Kosaraju, V., Bavarian, M., Chen, M., Jun, H., Kaiser, L., Plappert, M., Tworek, J., Hilton, J., Nakano, R., Hesse, C., and Schulman, J. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Coleman, C., Yeh, C., Mussmann, S., Mirzasoleiman, B., Bailis, P., Liang, P., Leskovec, J., and Zaharia, M. Selection via proxy: Efficient data selection for deep learning. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=HJg2b0VYDr.

Davies, A., Veličković, P., Buesing, L., Blackwell, S., Zheng, D., Tomašev, N., Tanburn, R., Battaglia, P., Blundell, C., Juhász, A., et al. Advancing mathematics by guiding human intuition with ai. *Nature*, 600(7887):70–74, 2021.

Delbrouck, J.-B., Varma, M., Chambon, P., and Langlotz, C. Overview of the RadSum23 shared task on multimodal and multi-anatomical radiology report summarization. In Demner-fushman, D., Ananiadou, S., and Cohen, K. (eds.), *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pp. 478–482, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.bionlp-1.45. URL https://aclanthology.org/2023.bionlp-1.45.

Demner-fushman, D., Ananiadou, S., and Cohen, K. (eds.). *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Toronto, Canada, July 2023. Association for Computational Linguistics. URL https://aclanthology.org/2023.bionlp-1.0.

Deng, Y., Yang, Y., Mirzasoleiman, B., and Gu, Q. Robust learning with progressive data expansion against spurious correlation. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=9QEVJ9qm46.

Douze, M., Guzhva, A., Deng, C., Johnson, J., Szilvasy, G., Mazaré, P.-E., Lomeli, M., Hosseini, L., and Jégou, H. The faiss library. 2024.

Eldan, R. and Li, Y. Tinystories: How small can language models be and still speak coherent english? *arXiv preprint arXiv:2305.07759*, 2023.

Hendrycks, D., Burns, C., Kadavath, S., Arora, A., Basart, S., Tang, E., Song, D., and Steinhardt, J. Measuring mathematical problem solving with the MATH dataset. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021. URL https://openreview.net/forum?id=7Bywt2mQsCe.

Jang, J., Kim, S., Ye, S., Kim, D., Logeswaran, L., Lee, M., Lee, K., and Seo, M. Exploring the benefits of training expert language models over instruction tuning. In Krause, A., Brunskill, E., Cho, K., Engelhardt, B., Sabato, S., and Scarlett, J. (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 14702–14729. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/jang23a.html.

Johnson, A. E., Pollard, T. J., Shen, L., Lehman, L.-w. H., Feng, M., Ghassemi, M., Moody, B., Szolovits, P., Anthony Celi, L., and Mark, R. G. Mimic-iii, a freely accessible critical care database. *Scientific data*, 3(1):1–9, 2016.

Joshi, S. and Mirzasoleiman, B. Data-efficient contrastive self-supervised learning: Most beneficial examples for supervised learning contribute the least. In *International conference on machine learning*, pp. 15356–15370. PMLR, 2023.

Killamsetty, K., Durga, S., Ramakrishnan, G., De, A., and Iyer, R. Grad-match: Gradient matching based data subset selection for efficient deep model training. In *International Conference on Machine Learning*, pp. 5464–5474. PMLR, 2021a.

Killamsetty, K., Sivasubramanian, D., Ramakrishnan, G., and Iyer, R. Glister: Generalization based data subset selection for efficient and robust learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 8110–8118, 2021b.

Koncel-Kedziorski, R., Roy, S., Amini, A., Kushman, N., and Hajishirzi, H. MAWPS: A math word problem repository. In Knight, K., Nenkova, A., and Rambow, O. (eds.), *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 1152–1157, San Diego, California, June 2016. Association for Computational Linguistics. doi: 10.18653/v1/N16-1136. URL https://aclanthology.org/N16-1136.

Lewkowycz, A., Andreassen, A., Dohan, D., Dyer, E., Michalewski, H., Ramasesh, V., Slone, A., Anil, C., Schlag, I., Gutman-Solo, T., et al. Solving quantitative reasoning problems with language models. *Advances in Neural Information Processing Systems*, 35:3843–3857, 2022.

Li, Y., Bubeck, S., Eldan, R., Del Giorno, A., Gunasekar, S., and Lee, Y. T. Textbooks are all you need ii: phi-1.5 technical report. *arXiv preprint arXiv:2309.05463*, 2023a.

Li, Y., Bubeck, S., Eldan, R., Del Giorno, A., Gunasekar, S., and Lee, Y. T. Textbooks are all you need ii: **phi-1.5** technical report. *arXiv preprint arXiv:2309.05463*, 2023b.

Lin, C.-Y. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pp. 74–81, Barcelona, Spain, July 2004. Association for Computational Linguistics. URL https://aclanthology.org/W04-1013.

Liu, B., Bubeck, S., Eldan, R., Kulkarni, J., Li, Y., Nguyen, A., Ward, R., and Zhang, Y. Tinygsm: achieving¿ 80% on gsm8k with small language models. *arXiv preprint arXiv:2312.09241*, 2023.

Luo, H., Sun, Q., Xu, C., Zhao, P., Lou, J., Tao, C., Geng, X., Lin, Q., Chen, S., and Zhang, D. Wizardmath: Empowering mathematical reasoning for large language models via reinforced evol-instruct. *arXiv preprint arXiv:2308.09583*, 2023a.

Luo, Z., Xu, C., Zhao, P., Sun, Q., Geng, X., Hu, W., Tao, C., Ma, J., Lin, Q., and Jiang, D. Wizardcoder: Empowering code large language models with evol-instruct. *arXiv preprint arXiv:2306.08568*, 2023b.

Mahmoud, A., Elhoushi, M., Abbas, A., Yang, Y., Ardalani, N., Leather, H., and Morcos, A. Sieve: Multimodal dataset pruning using image captioning models. *arXiv preprint arXiv:2310.02110*, 2023.

Marion, M., Üstün, A., Pozzobon, L., Wang, A., Fadaee, M., and Hooker, S. When less is more: Investigating data pruning for pretraining llms at scale. *arXiv preprint arXiv:2309.04564*, 2023.

Mirzasoleiman, B., Bilmes, J., and Leskovec, J. Coresets for data-efficient training of machine learning models. In III, H. D. and Singh, A. (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 6950–6960. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/mirzasoleiman20a.html.

Mishra, S., Mitra, A., Varshney, N., Sachdeva, B., Clark, P., Baral, C., and Kalyan, A. NumGLUE: A suite of fundamental yet challenging mathematical reasoning tasks. In Muresan, S., Nakov, P., and Villavicencio, A. (eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3505–3523, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.246. URL https://aclanthology.org/2022.acl-long.246.

Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744, 2022.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pp. 311–318, 2002.

Patel, A., Bhattamishra, S., and Goyal, N. 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.

Paul, M., Ganguli, S., and Dziugaite, G. K. Deep learning on a data diet: Finding important examples early in training. *Advances in Neural Information Processing Systems*, 34: 20596–20607, 2021.

Pooladzandi, O., Davini, D., and Mirzasoleiman, B. Adaptive second order coresets for data-efficient machine learning. In Chaudhuri, K., Jegelka, S., Song, L., Szepesvari, C., Niu, G., and Sabato, S. (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 17848–17869. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/pooladzandi22a.html.

Prakriya, N., Yang, Y., Mirzasoleiman, B., Hsieh, C.-J., and Cong, J. Nessa: Near-storage data selection for accelerated machine learning training. In *Proceedings of the 15th ACM Workshop on Hot Topics in Storage and File Systems*, HotStorage '23, pp. 8–15, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9798400702242. doi: 10.1145/3599691.3603404. URL https://doi.org/10.1145/3599691.3603404.

Roziere, B., Gehring, J., Gloeckle, F., Sootla, S., Gat, I., Tan, X. E., Adi, Y., Liu, J., Remez, T., Rapin, J., et al. Code llama: Open foundation models for code. *arXiv preprint arXiv:2308.12950*, 2023.

Singhal, K., Azizi, S., Tu, T., Mahdavi, S. S., Wei, J., Chung, H. W., Scales, N., Tanwani, A., Cole-Lewis, H., Pfohl, S., et al. Large language models encode clinical knowledge. *Nature*, 620(7972):172–180, 2023a.

Singhal, K., Tu, T., Gottweis, J., Sayres, R., Wulczyn, E., Hou, L., Clark, K., Pfohl, S., Cole-Lewis, H., Neal, D., et al. Towards expert-level medical question answering with large language models. *arXiv preprint arXiv:2305.09617*, 2023b.

Sorscher, B., Geirhos, R., Shekhar, S., Ganguli, S., and Morcos, A. Beyond neural scaling laws: beating power law scaling via data pruning. *Advances in Neural Information Processing Systems*, 35:19523–19536, 2022.

Swayamdipta, S., Schwartz, R., Lourie, N., Wang, Y., Hajishirzi, H., Smith, N. A., and Choi, Y. Dataset cartography: Mapping and diagnosing datasets with training dynamics. In Webber, B., Cohn, T., He, Y., and Liu, Y. (eds.), *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 9275–9293, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.746. URL https://aclanthology.org/2020.emnlp-main.746.

Taylor, R., Kardas, M., Cucurull, G., Scialom, T., Hartshorn, A., Saravia, E., Poulton, A., Kerkez, V., and Stojnic, R. Galactica: A large language model for science. *arXiv preprint arXiv:2211.09085*, 2022.

Thoppilan, R., De Freitas, D., Hall, J., Shazeer, N., Kulshreshtha, A., Cheng, H.-T., Jin, A., Bos, T., Baker, L., Du, Y., et al. Lamda: Language models for dialog applications. *arXiv preprint arXiv:2201.08239*, 2022.

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

Toneva, M., Sordoni, A., des Combes, R. T., Trischler, A., Bengio, Y., and Gordon, G. J. An empirical study of example forgetting during deep neural network learning. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=BJlxm30cKm.

Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

Tu, T., Azizi, S., Driess, D., Schaekermann, M., Amin, M., Chang, P.-C., Carroll, A., Lau, C., Tanno, R., Ktena, I., et al. Towards generalist biomedical ai. *arXiv preprint arXiv:2307.14334*, 2023.

Van Veen, D., Van Uden, C., Blankemeier, L., Delbrouck, J.-B., Aali, A., Bluethgen, C., Pareek, A., Polacin, M., Collins, W., Ahuja, N., et al. Clinical text summarization: adapting large language models can outperform human experts. *arXiv preprint arXiv:2309.07430*, 2023.

Varshney, N., Mishra, S., and Baral, C. Let the model decide its curriculum for multitask learning. In Cherry, C., Fan, A., Foster, G., Haffari, G. R., Khadivi, S., Peng, N. V., Ren, X., Shareghi, E., and Swayamdipta, S. (eds.), *Proceedings of the Third Workshop on Deep Learning for Low-Resource Natural Language Processing*, pp. 117–125, Hybrid, July 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.deeplo-1.13. URL https://aclanthology.org/2022.deeplo-1.13.

Wei, J., Bosma, M., Zhao, V., Guu, K., Yu, A. W., Lester, B., Du, N., Dai, A. M., and Le, Q. V. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*, 2022a. URL https://openreview.net/forum?id=gEZrGCozdqR.

Wei, J., Wang, X., Schuurmans, D., Bosma, M., brian ichter, Xia, F., Chi, E. H., Le, Q. V., and Zhou, D. Chain of thought prompting elicits reasoning in large language models. In Oh, A. H., Agarwal, A., Belgrave, D., and Cho, K. (eds.), *Advances in Neural Information Processing Systems*, 2022b. URL https://openreview.net/forum?id=_VjQlMeSB_J.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Scao, T. L., Gugger, S., Drame, M., Lhoest, Q., and Rush, A. M. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 38–45, Online, October 2020. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/2020.emnlp-demos.6.

Wu, S., Irsoy, O., Lu, S., Dabravolski, V., Dredze, M., Gehrmann, S., Kambadur, P., Rosenberg, D., and Mann, G. Bloomberggpt: A large language model for finance. *arXiv preprint arXiv:2303.17564*, 2023a.

Wu, S., Lu, K., Xu, B., Lin, J., Su, Q., and Zhou, C. Self-evolved diverse data sampling for efficient instruction tuning. *arXiv preprint arXiv:2311.08182*, 2023b.

Xia, M., Artetxe, M., Zhou, C., Lin, X. V., Pasunuru, R., Chen, D., Zettlemoyer, L., and Stoyanov, V. Training trajectories of language models across scales. In Rogers, A., Boyd-Graber, J., and Okazaki, N. (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 13711–13738, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.767. URL https://aclanthology.org/2023.acl-long.767.

Yang, Y., Liu, T. Y., and Mirzasoleiman, B. Not all poisons are created equal: Robust training against data poisoning. In Chaudhuri, K., Jegelka, S., Song, L., Szepesvari, C., Niu, G., and Sabato, S. (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 25154–25165. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/yang22j.html.

Yang, Y., Gan, E., Dziugaite, G. K., and Mirzasoleiman, B. Identifying spurious biases early in training through the lens of simplicity bias. *arXiv preprint arXiv:2305.18761*, 2023a.

Yang, Y., Kang, H., and Mirzasoleiman, B. Towards sustainable learning: Coresets for data-efficient deep learning. In Krause, A., Brunskill, E., Cho, K., Engelhardt, B., Sabato, S., and Scarlett, J. (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 39314–39330. PMLR, 23–29 Jul 2023b.

Yang, Y., Singh, A. K., Elhoushi, M., Mahmoud, A., Tirumala, K., Gloeckle, F., Rozière, B., Wu, C.-J., Morcos, A. S., and Ardalani, N. Decoding data quality via synthetic corruptions: Embedding-guided pruning of code data. *arXiv preprint arXiv:2312.02418*, 2023c.

Yu, L., Jiang, W., Shi, H., Yu, J., Liu, Z., Zhang, Y., Kwok, J. T., Li, Z., Weller, A., and Liu, W. Metamath: Bootstrap your own mathematical questions for large language models. *arXiv preprint arXiv:2309.12284*, 2023.

Yue, X., Qu, X., Zhang, G., Fu, Y., Huang, W., Sun, H., Su, Y., and Chen, W. Mammoth: Building math generalist models through hybrid instruction tuning. *arXiv preprint arXiv:2309.05653*, 2023.

Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q., and Artzi, Y. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=SkeHuCVFDr.

Zhao, Y., Gu, A., Varma, R., Luo, L., Huang, C.-C., Xu, M., Wright, L., Shojanazeri, H., Ott, M., Shleifer, S., et al. Pytorch fsdp: experiences on scaling fully sharded data parallel. *arXiv preprint arXiv:2304.11277*, 2023.

Zhou, C., Liu, P., Xu, P., Iyer, S., Sun, J., Mao, Y., Ma, X., Efrat, A., Yu, P., YU, L., Zhang, S., Ghosh, G., Lewis, M., Zettlemoyer, L., and Levy, O. LIMA: Less is more for alignment. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023a. URL https://openreview.net/forum?id=KBMOKmX2he.

Zhou, H., Liu, T., Ma, Q., Yuan, J., Liu, P., You, Y., and Yang, H. Lobass: Gauging learnability in supervised fine-tuning data. *arXiv preprint arXiv:2310.13008*, 2023b.

A. Experiment Details

A.1. Models

Pythia. The Pythia models (Biderman et al., 2023b) are a suite of large language models (LLMs) developed by EleutherAI licensed under the Apache License 2.0. These models range in size from 70 million to 12 billion parameters and are designed to enable controlled scientific research on transparently trained LLMs across various scales.

Phi. The Phi models (Li et al., 2023b) developed by Microsoft are under the MIT License. Phi-1.5, a transformer-based model with 1.3 billion parameters, and its successor, Phi-2, with 2.7 billion parameters, have been trained on a diverse set of data sources, including synthetic texts and curated websites. The Phi models underscore the potential of small yet powerful language models in understanding and generating human language, empowering a range of NLP tasks. Phi-2, in particular, has raised the bar for reasoning and language understanding among foundation models, matching or even exceeding the performance of models 25 times its size on complex benchmarks.

A.2. Datasets

MathInstruct. The MathInstruct dataset (Yue et al., 2023) is compiled from 13 diverse math rationale datasets, using both chain-of-thought (CoT) and program-of-thought (PoT) rationales. It ensures comprehensive coverage across various mathematical fields in the 262K training examples, making it a popular resource for fine-tuning large language models (LLMs) for general math problem-solving. MathInstruct is licensed under the MIT license.

MIMIC-III. The MIMIC-III (Medical Information Mart for Intensive Care III) dataset (Johnson et al., 2016) is a comprehensive collection of de-identified health data associated with over 40,000 patients who stayed in critical care units of the Beth Israel Deaconess Medical Center in Boston, Massachusetts. This large dataset includes information such as demographics, vital signs, laboratory tests, medications, and more, making it an invaluable resource for a wide range of research in healthcare, including clinical decision support systems, medical procedure efficacy studies, and patient care optimization strategies.

The MIMIC-III dataset is made freely available to the research community under the Health Insurance Portability and Accountability Act (HIPAA) compliance, ensuring patient confidentiality and data protection. Access to the dataset is granted under a data use agreement (DUA) to individuals affiliated with an institution that approves the use of the data for research purposes. Researchers seeking to utilize the MIMIC-III dataset must complete a required training course on human research protections, which ensures that all researchers are aware of the responsibilities involved in handling sensitive patient data.

A.3. Implementation Details

S2L The training trajectories for both MathInstruct and MIMIC-III are gathered from training a Pythia-70M model, the smallest model in the Pythia model suite, recorded every 500 training iterations. We utilize the Faiss library (Douze et al., 2024) to perform efficient K-means clustering of loss trajectories with Euclidean distance with $K = 100$ and 20 iterations. The hyperparameter K is tuned in the range of $\{50, 100, 200\}$ based on the average accuracy of the model trained on $30K$ selected data. We found $K = 100$ worked the best for both datasets we studied in this paper. Ablations studies on the length and the best time in the training to record the trajectories can be found in Section 5.4.

Comparing Reference Models for the Baselines For one-shot selection methods (excluding S2L), we use representations from either step 1000 or the end of fine-tuning Pythia-410M on MathInstruct and reported the better result in Table 1. In Table 4, we include the complete comparison between using early-fine-tuning vs. end-of-fine-tuning model checkpoints as the inference model. For Facility Locations, we further compared using the first hidden states as the feature representation as suggested in (Bhatt et al., 2024) to using the last hidden states (Wu et al., 2023b) for the tasks we studied.

A.4. Evaluation

A.4.1. MATHINSTRUCT

Datasets. We utilize 6 diverse datasets with open-formed questions for evaluating the mathematical reasoning capabilities of models trained with both the full MathInstruct dataset and selected subsets. These datasets, detailed in Table 5, span a

Table 4. Complete results used for selecting the best reference model for each one-shot data selection baseline. The choice of early-fine-tuning (step 1000) and end-of-fine-tuning checkpoint follows (Marion et al., 2023). The best results selected for Table 1 are highlighted in cyan.

	Ref	Data	In-domain				Out-domain			
Selection	Model	Size	GSM8K	MATH	NumGLUE	**Avg**	SVAMP	Mathematics	SimulEq	**Avg**
Least Confidence	Early	30K	2.3	1.7	15.5	6.5	13.6	1.2	0.5	5.8
		50K	1.7	2.6	20.5	8.3	16.0	4.0	1.8	7.8
		100K	3.9	2.7	22.5	9.7	19.2	8.0	3.3	9.9
	End	30K	2.7	1.3	18.0	7.0	13.7	3.3	1.4	6.7
		50K	2.1	1.7	21.0	8.3	14.5	3.5	1.0	7.3
		100K	2.5	3.3	23.5	9.8	20.8	6.3	3.7	10.0
Middle Perplexity	Early	30K	3.3	3.8	17.5	8.2	11.8	1.2	1.2	6.5
		50K	2.9	4.1	19.6	8.9	15.6	7.6	2.9	8.8
		100K	4.8	7.1	20.4	10.8	19.6	16.1	3.9	12.0
	End	30K	5.3	3.7	16.2	8.4	14.2	8.7	1.2	8.2
		50K	3.2	5.9	20.5	9.9	18.1	11.3	**5.1**	10.7
		100K	5.4	7.2	20.9	11.2	**23.8**	15.3	3.3	12.6
High Learnability	Early	30K	6.1	1.6	19.1	8.9	10.7	9.9	1.4	8.1
		50K	6.1	2.1	18.6	8.9	14.5	14.0	2.1	8.9
		100K	**7.4**	9.2	**29.8**	**15.5**	20.7	19.4	**10.3**	**16.1**
	End	30K	3.0	1.4	14.7	6.4	2.1	6.8	1.8	5.0
		50K	1.3	2.1	16.0	6.5	4.7	6.9	3.1	5.7
		100K	4.3	7.2	23.0	11.5	16.7	16.1	4.3	11.9
Facility Location	Early (First)	50K	3.9	7.6	12.4	8.0	11.1	14.6	1.9	8.6
	Early (Last)	50K	5.7	9.1	12.4	9.1	15.4	18.6	1.6	10.5
	End (First)	50K	3.8	7.7	14.8	8.7	19.2	11.4	2.3	9.9
	End (Last)	50K	5.2	**9.7**	11.8	8.9	12.4	18.2	1.0	9.7

Table 5. Types of questions in the evaluation datasets for the mathematical reasoning task.

Dataset	Size	Level	Tasks
GSM8K	1319	Early Algebra	Multi-step reasoning using basic arithmetic operations
MATH	5000	Early Algebra, Intermediate Algebra, Algebra, Probability, NumTheory, Calculus, Geometry	Problems from mathematics competitions, including the AMC 10, AMC 12, AIME
NumGLUE	1042	Early Algebra	Commonsense, Domain-specific, Arithmetic Reasoning, Quantitative Comparison, Fill-in-the-blanks Format, Reading Comprehension, Numerical Reasoning, Quantitative NLI, Arithmetic Word Problems
SVAMP	1000	Early Algebra	Arithmetic Word Problems
Mathematics	1000	Early Algebra, Intermediate Algebra, NumTheory, Calculus	Arithmetic Reasoning
SimulEq	514	Linear Algebra	Single and multiple equation word problems

range of mathematical disciplines from early algebra to calculus and linear algebra, covering various types of questions such as multi-step reasoning, arithmetic word problems, and problems from mathematics competitions. This variety ensures a comprehensive assessment across both in-domain and out-domain tasks.

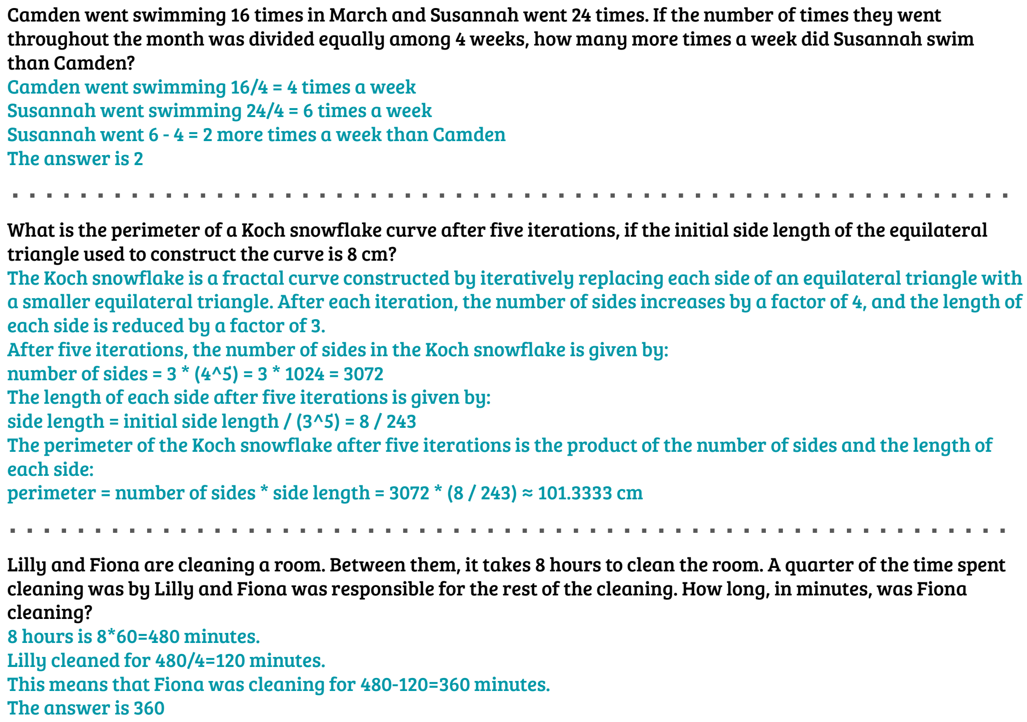
Pipeline. We utilize the pipeline provided by (Yue et al., 2023), designed to first determine whether the model can be prompted to generate a code snippet. This code snippet, if successfully generated, should be executable and produce the correct answer when run. This code-based evaluation is also used for Phi models (Li et al., 2023b). In cases where the model does not directly produce a viable code solution, we employ a "think step-by-step" prompting strategy (Wei et al., 2022b). This method prompts the model to break down its reasoning process, a technique that has been widely proven effective in fully exploiting the model's problem-solving capacity.

A.4.2. MIMIC-III

Following (Delbrouck et al., 2023; Demner-fushman et al., 2023), we include the six most common modality/anatomy pairs: CT head, CT abdomen, CT chest, MRI head, CT spine, and CT neck, and five less common pairs in the text data: MRI spine, CT sinus, MRI abdomen, MRI pelvis, and MRI neck in the evaluation. There are in total 13.7K test examples after data preprocessing and train-test splitting.

B. Examples in Different Clusters

We provide the full version of Figure 5, which compares data points in the same and different clusters based on training trajectories, in Figure 9, Figure 10 and Figure 11. We can observe that examples with similar training trajectories have the same question format. Therefore, balanced sampling from all clusters can ensure different types of examples can be covered in the selected subset of training data.



Camden went swimming 16 times in March and Susannah went 24 times. If the number of times they went throughout the month was divided equally among 4 weeks, how many more times a week did Susannah swim than Camden?

Camden went swimming 16/4 = 4 times a week
Susannah went swimming 24/4 = 6 times a week
Susannah went 6 - 4 = 2 more times a week than Camden
The answer is 2

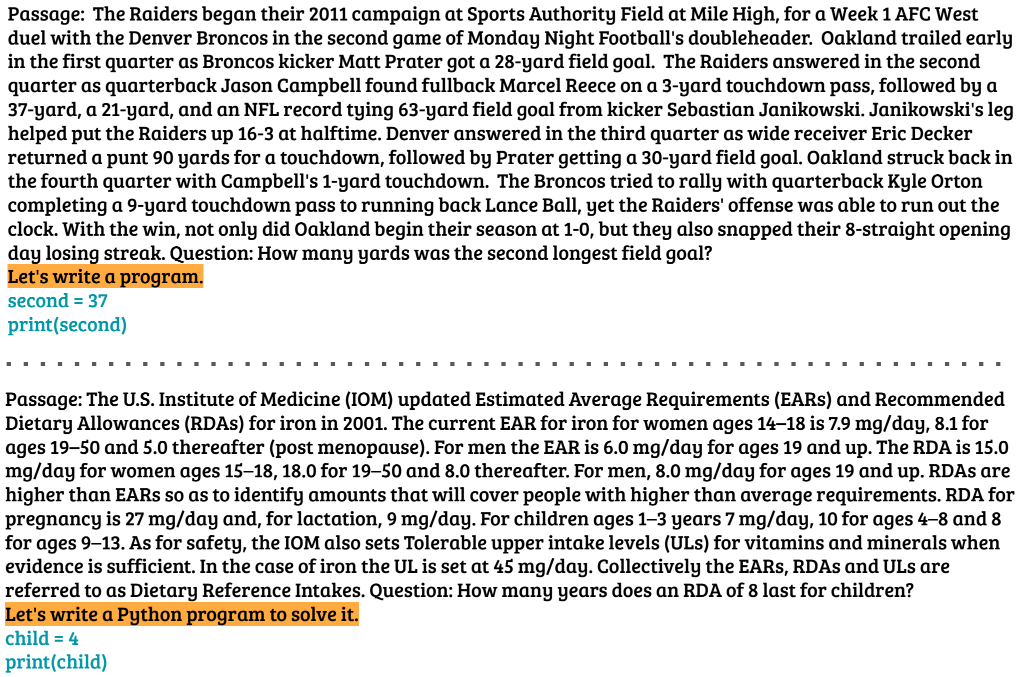
What is the perimeter of a Koch snowflake curve after five iterations, if the initial side length of the equilateral triangle used to construct the curve is 8 cm?

The Koch snowflake is a fractal curve constructed by iteratively replacing each side of an equilateral triangle with a smaller equilateral triangle. After each iteration, the number of sides increases by a factor of 4, and the length of each side is reduced by a factor of 3.
After five iterations, the number of sides in the Koch snowflake is given by:
number of sides = 3 * (4^5) = 3 * 1024 = 3072
The length of each side after five iterations is given by:
side length = initial side length / (3^5) = 8 / 243
The perimeter of the Koch snowflake after five iterations is the product of the number of sides and the length of each side:
perimeter = number of sides * side length = 3072 * (8 / 243) ≈ 101.3333 cm

Lilly and Fiona are cleaning a room. Between them, it takes 8 hours to clean the room. A quarter of the time spent cleaning was by Lilly and Fiona was responsible for the rest of the cleaning. How long, in minutes, was Fiona cleaning?

8 hours is 8*60=480 minutes.
Lilly cleaned for 480/4=120 minutes.
This means that Fiona was cleaning for 480-120=360 minutes.
The answer is 360

Figure 9. A full version of Figure 5a: open-formed algebra. Questions are in black and answers are in cyan.



Passage: The Raiders began their 2011 campaign at Sports Authority Field at Mile High, for a Week 1 AFC West duel with the Denver Broncos in the second game of Monday Night Football's doubleheader. Oakland trailed early in the first quarter as Broncos kicker Matt Prater got a 28-yard field goal. The Raiders answered in the second quarter as quarterback Jason Campbell found fullback Marcel Reece on a 3-yard touchdown pass, followed by a 37-yard, a 21-yard, and an NFL record tying 63-yard field goal from kicker Sebastian Janikowski. Janikowski's leg helped put the Raiders up 16-3 at halftime. Denver answered in the third quarter as wide receiver Eric Decker returned a punt 90 yards for a touchdown, followed by Prater getting a 30-yard field goal. Oakland struck back in the fourth quarter with Campbell's 1-yard touchdown. The Broncos tried to rally with quarterback Kyle Orton completing a 9-yard touchdown pass to running back Lance Ball, yet the Raiders' offense was able to run out the clock. With the win, not only did Oakland begin their season at 1-0, but they also snapped their 8-straight opening day losing streak. Question: How many yards was the second longest field goal?

Let's write a program.

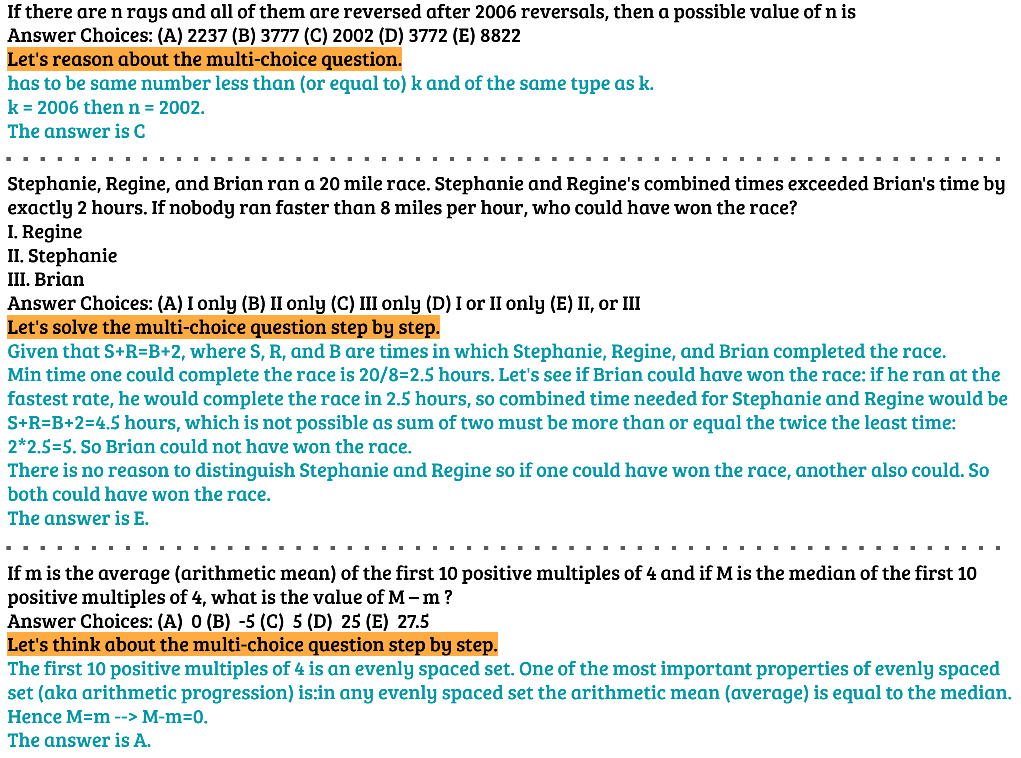
```
second = 37
print(second)
```

Passage: The U.S. Institute of Medicine (IOM) updated Estimated Average Requirements (EARs) and Recommended Dietary Allowances (RDAs) for iron in 2001. The current EAR for iron for women ages 14–18 is 7.9 mg/day, 8.1 for ages 19–50 and 5.0 thereafter (post menopause). For men the EAR is 6.0 mg/day for ages 19 and up. The RDA is 15.0 mg/day for women ages 15–18, 18.0 for 19–50 and 8.0 thereafter. For men, 8.0 mg/day for ages 19 and up. RDAs are higher than EARs so as to identify amounts that will cover people with higher than average requirements. RDA for pregnancy is 27 mg/day and, for lactation, 9 mg/day. For children ages 1–3 years 7 mg/day, 10 for ages 4–8 and 8 for ages 9–13. As for safety, the IOM also sets Tolerable upper intake levels (ULs) for vitamins and minerals when evidence is sufficient. In the case of iron the UL is set at 45 mg/day. Collectively the EARs, RDAs and ULs are referred to as Dietary Reference Intakes. Question: How many years does an RDA of 8 last for children?

Let's write a Python program to solve it.

```
child = 4
print(child)
```

Figure 10. A full version of Figure 5b: reading comprehension + coding. Questions are in black and answers are in cyan; instructions are highlighted in orange.



If there are n rays and all of them are reversed after 2006 reversals, then a possible value of n is
Answer Choices: (A) 2237 (B) 3777 (C) 2002 (D) 3772 (E) 8822

Let's reason about the multi-choice question.

has to be same number less than (or equal to) k and of the same type as k.
k = 2006 then n = 2002.
The answer is C

Stephanie, Regine, and Brian ran a 20 mile race. Stephanie and Regine's combined times exceeded Brian's time by exactly 2 hours. If nobody ran faster than 8 miles per hour, who could have won the race?
I. Regine
II. Stephanie
III. Brian
Answer Choices: (A) I only (B) II only (C) III only (D) I or II only (E) II, or III

Let's solve the multi-choice question step by step.

Given that S+R=B+2, where S, R, and B are times in which Stephanie, Regine, and Brian completed the race.
Min time one could complete the race is 20/8=2.5 hours. Let's see if Brian could have won the race: if he ran at the fastest rate, he would complete the race in 2.5 hours, so combined time needed for Stephanie and Regine would be S+R=B+2=4.5 hours, which is not possible as sum of two must be more than or equal the twice the least time: 2*2.5=5. So Brian could not have won the race.
There is no reason to distinguish Stephanie and Regine so if one could have won the race, another also could. So both could have won the race.
The answer is E.

If m is the average (arithmetic mean) of the first 10 positive multiples of 4 and if M is the median of the first 10 positive multiples of 4, what is the value of M – m ?
Answer Choices: (A) 0 (B) -5 (C) 5 (D) 25 (E) 27.5

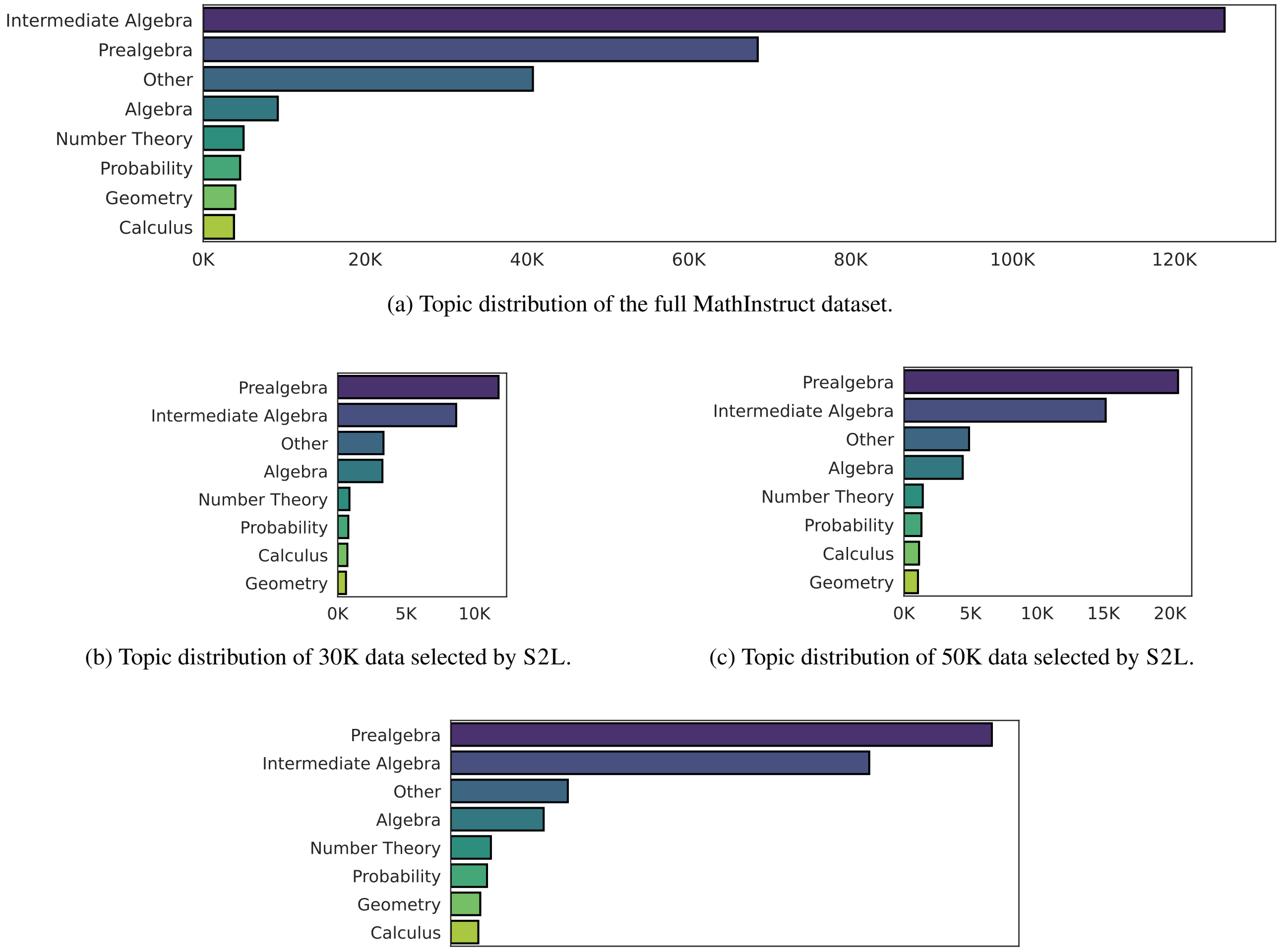
Let's think about the multi-choice question step by step.

The first 10 positive multiples of 4 is an evenly spaced set. One of the most important properties of evenly spaced set (aka arithmetic progression) is:in any evenly spaced set the arithmetic mean (average) is equal to the median.
Hence M=m --> M-m=0.
The answer is A.

Figure 11. A full version of Figure 5c: multiple-choice + multi-step reasoning. Questions are in black and answers are in cyan; instructions are highlighted in orange.

C. Topic Distribution of Data Selected by S2L

Beyond qualitative examples from different clusters, we study how S2L changes the data distribution to outperform using the full fine-tuning dataset as well as using random subsets of the same size that have the same distribution as the original dataset. In Figure 12, we can observe that S2L not only guarantees a thorough and balanced coverage across the spectrum of topics but also ensures sufficient representation of foundational topics, such as pre-algebra, which lays the groundwork for tackling more complex subjects.



(a) Topic distribution of the full MathInstruct dataset.

(b) Topic distribution of 30K data selected by S2L.

(c) Topic distribution of 50K data selected by S2L.

(d) Topic distribution of 100K data selected by S2L.

Figure 12. Compared to the original topic distribution, S2L prioritized easier topics (e.g., pre-algebra over intermediate algebra, algebra over other more advanced topics) while always ensuring complete and more balanced coverage of all topics.