MIXTURE-OF-SKILLS: Learning to Optimize Data Usage for Fine-Tuning Large Language Models

Anonymous ACL submission

Abstract

Large language models (LLMs) are typically fine-tuned on diverse and extensive datasets sourced from various origins to develop a comprehensive range of skills, such as writing, reasoning, chatting, coding, and more. Each skill has unique characteristics, and these datasets are often heterogeneous and imbalanced, making the fine-tuning process highly challenging. Balancing the development of each skill while ensuring the model maintains its overall performance requires sophisticated techniques and careful dataset curation. In this work, we propose a general, model-agnostic, reinforcement learning framework, MIXTURE-OF-SKILLS (MOS), that learns to optimize data usage automatically during the fine-tuning process. This framework ensures the optimal comprehensive skill development of LLMs by dynamically adjusting the focus on different datasets based on their current learning state. To validate the effectiveness of MOS, we conduct extensive experiments using three diverse LLM backbones on two widely used benchmarks and demonstrate that MOS substantially enhances model performance. Building on the success of MOS, we propose MOSPEC, an adaptation for task-specific fine-tuning, which harnesses the utilities of various datasets for a specific purpose. Our work underlines the significance of dataset rebalancing and present MOS as a powerful, general solution for optimizing data usage in the fine-tuning of LLMs for various purposes.

1 Introduction

Large language models (LLMs) have demonstrated their extraordinary capabilities and are expected to proficiently master a diverse range of skills (Ouyang et al., 2022; Sanh et al., 2022; OpenAI, 2023; Anil et al., 2023b; Touvron et al., 2023a,b; Anil et al., 2023a; Mesnard et al., 2024), such as writing, reasoning, chatting, coding, and more,
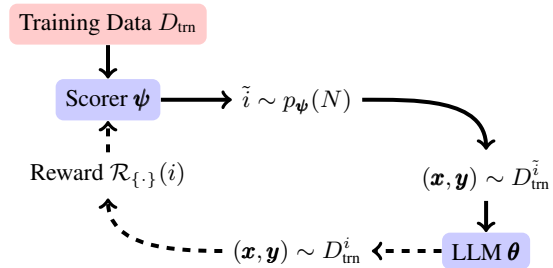

Figure 1: The overview of MIXTURE-OF-SKILLS. The training collection $D_{\text{trn}} = \{D_{\text{trn}}^i\}_{i=1}^N$ consists of various SFT datasets, with D_{trn}^i indicating the i-th dataset. Please refer to Section 3 for more details.

through supervised fine-tuning (SFT) and reinforcement learning from human feedback (RLHF) on an extensive collection of datasets from various sources (Bai et al., 2022; Longpre et al., 2023; Ding et al., 2023). Each dataset contributes unique elements to the model's skill set, but this diversity also brings challenges.

One common challenge in fine-tuning models across multiple datasets is dealing with their **heterogeneity** (where different datasets exhibit distinct characteristics) and **imbalance** (where datasets vary in size and accessibility), making the fine-tuning process highly challenging. To address this challenge, Recent approaches often cap dataset usage to prevent models from being overwhelmed by large datasets, but this limits the utilization of all available data (Raffel et al., 2020; Chung et al., 2022; Iyer et al., 2022; Wei et al., 2022). Previous multilingual research adjusts dataset distribution heuristically with a temperature term τ, which requires extensive hyperparameter tuning and overlooks dataset interactions and model learning dynamics (Arivazhagan et al., 2019; Conneau et al., 2020). This leads us to a critical research question: *Is there a better way to optimize the data usage?*

Building on the research question posed, we first confirm that adjusting the dataset usage properly

can significantly enhance model performance (see Table 1). Moreover, inspired by Differentiable Data Selection (Wang et al., 2020a), we propose a general, model-agnostic reinforcement learning framework MIXTURE-OF-SKILLS (MoS) that *learns* to optimize data usage automatically during the fine-tuning process. To achieve this optimization, we introduce another set of parameters, ψ, known as the scorer network. As shown in Figure 1, this network dynamically adjusts data usage based on the current learning state of the LLM θ. Furthermore, the rewards used to update the scorer network ψ are provided by the LLM θ from three distinct perspectives: *transferability*, *difficulty*, and *learning trajectory*, ensuring that the scorer network can effectively guide the data usage optimization process. All these efforts constitute the success of our MIXTURE-OF-SKILLS framework.

To validate the effectiveness of MoS, we conduct extensive experiments using three diverse model backbones: QWEN1.5-0.5B (Bai et al., 2023), GEMMA-2B (Mesnard et al., 2024), and LLAMA-3-8B (AI@Meta, 2024), on two widely-used benchmarks: MMLU (Hendrycks et al., 2021a) and MT-bench (Zheng et al., 2023). Our empirical results demonstrate that MoS substantially improves the models' overall performance. Our analysis indicates that our model not only effectively learns optimal data utilization but also accelerates training convergence by $2.2\times$. Additionally, it demonstrates robustness against variations in sampling priors and integrates seamlessly with advanced instance selection methods. Furthermore, we explore the application of MoS in task-specific fine-tuning. We show that MoS, with minor reward modifications known as MOSPEC, can be effectively used for task-specific fine-tuning.

Our contributions are summarized as follows:

- We present a general, model-agnostic reinforcement learning framework, MIXTURE-OF-SKILLS (MoS), that learns to automatically optimize data usage during the SFT process with three novel rewards (see Section 3).
- Extensive experiments with three model backbones on two benchmarks demonstrate that MoS significantly enhances model performance. Our analysis reveals that MoS not only effectively learns the optimal data usage but also accelerates training convergence by $2.2\times$. Additionally, it maintains robustness against variations in sampling priors and is compatible with strong instance selection

methods (see Section 4 and Section 5).
- We explore the application of MoS in task-specific fine-tuning, introducing a variant called MOSPEC. This variant, with minor modifications to the rewards, is proven to effectively harness diverse datasets for task-specific fine-tuning (see Section 6).

2 Preliminaries

Supervised Fine-Tuning A large language model (LLM), parameterized by θ, is capable of following and responding to human instructions after supervised fine-tuning (SFT). Given a single training dataset $D_{\mathrm{trn}}^1 = \{(\boldsymbol{x}_j, \boldsymbol{y}_j)\}_{j=1}^{M_1}$, where M_1 is the size of D_{trn}^1 and $\boldsymbol{x}_j$ and $\boldsymbol{y}_j$ are the instruction and response of the j-th example, the objective function during SFT is to minimize the negative log-likelihood with respect to θ:

$$\mathcal{L}_s(D_{\mathrm{trn}}^1; \theta) = -\sum_{j=1}^{M_1} \log p(\boldsymbol{y}_j | \boldsymbol{x}_j; \theta). \quad (1)$$

When fine-tuning the LLM θ over multiple datasets $D_{\mathrm{trn}} = \{D_{\mathrm{trn}}^i\}_{i=1}^N$, where $D_{\mathrm{trn}}^i = \{(\boldsymbol{x}_j^i, \boldsymbol{y}_j^i)\}_{j=1}^{M_i}$, the objective function becomes:

$$\mathcal{L}(D_{\mathrm{trn}}; \theta) = \sum_{i=1}^{N} \mathcal{L}_s(D_{\mathrm{trn}}^i; \theta). \quad (2)$$

Heuristic Balancing by Temperature Instead of merging all datasets into a single training mixture, a common approach is to adjust the sampling probability of texts in different languages using a temperature term τ (Arivazhagan et al., 2019; Conneau et al., 2020). Specifically, the sampling probability of the i-th dataset is $q(i) = \frac{|M_i|}{\sum_{n=1}^N |M_n|}$ and can be adjusted by the temperature τ as:

$$q_\tau(i) = \frac{q(i)^{1/\tau}}{\sum_{n=1}^N q(n)^{1/\tau}}. \quad (3)$$

Consequently, $\tau = 1$ corresponds to proportional sampling, equivalent to Equation 2. Conversely, $\tau = \infty$ corresponds to uniform sampling, where smaller datasets are up-sampled to match the largest dataset. The loss function becomes:

$$\mathcal{L}(D_{\mathrm{trn}}; \theta, q_\tau(N)) = \mathbb{E}_{i \sim q_\tau(N)} \left[\mathcal{L}_s(D_{\mathrm{trn}}^i; \theta) \right]. \quad (4)$$

Differentiable Data Selection (DDS) Wang et al. (2020a) propose a general framework that automatically re-weighs training instances to enhance model performance, utilizing a validation set D_{dev}. This framework consists of two components: the model $\boldsymbol{\theta}$ and the scorer network $\boldsymbol{\psi}$. The scorer network $\boldsymbol{\psi}$ is designed to calculate a sampling probability for each training instance, which reflects its impact on validation performance. Training instances that have a greater similarity with D_{dev} are allocated a higher probability, thus increasing their likelihood of being selected for model $\boldsymbol{\theta}$ updates. MoS is inspired by DDS but has key differences. Firstly, MoS focuses on rebalancing datasets, unlike DDS, which reweighs training instances. Secondly, MoS does not require prior knowledge of downstream applications, whereas DDS relies on validation set feedback, risking overfitting to that specific validation set. Thirdly, DDS uses the same architecture for the scorer network $\boldsymbol{\psi}$ and model $\boldsymbol{\theta}$, limiting its scalability, while MoS uses a simple MLP model as its scorer network (see Section 3.1).

3 MIXTURE-OF-SKILLS

In this section, we provide a detailed overview of MIXTURE-OF-SKILLS (MoS). We begin by outlining the reinforcement learning framework employed in MoS (Section 3.1). Following this, we discuss the reward functions (Section 3.2).

3.1 Learning to Optimize Data Usage

We propose MIXTURE-OF-SKILLS (MoS) that *learns* to optimize the data usage during the fine-tuning process by training a scorer network, parameterized by $\boldsymbol{\psi}$, within a reinforcement learning (RL) framework. In this setup, the LLM $\boldsymbol{\theta}$ and the training datasets D_{trn} constitute the *environment*, while our scorer network serves as the RL *agent*. In this framework, unlike the static sampling probabilities described in Equation 3, the scorer network $\boldsymbol{\psi}$ dynamically adjusts the sampling probabilities for each dataset in D_{trn} according to the current learning state of the LLM $\boldsymbol{\theta}$. Alternately, the LLM $\boldsymbol{\theta}$ is optimized based on the sampling distribution given by the scorer network $\boldsymbol{\psi}$.

To provide a broader perspective, MoS can be conceptualized as the resolution of a *bi-level optimization* problem (Colson et al., 2007). In this view, the outer level optimizes the parameters of the LLM $\boldsymbol{\theta}$, while the inner level focuses on adjusting the sampling probabilities through the scorer

---

Algorithm 1: MIXTURE-OF-SKILLS

Input : $D_{\text{trn}} = \{\{(\boldsymbol{x}_j^i, \boldsymbol{y}_j^i)\}_{j=1}^{M_i}\}_{i=1}^N$, N training datasets with the size of M_i for the i-th dataset; S, update frequency of $\boldsymbol{\psi}$; T, total training steps; α, learning rate for $\boldsymbol{\theta}$; γ, learning rate for $\boldsymbol{\psi}$;

Output : The converged model $\boldsymbol{\theta}$;

1   Initialize $p_{\boldsymbol{\psi}_0}(N)$ as Equation 3 with $\tau = \infty$;

2   **for** *t=0 to T* **do**

3     $\tilde{i} \sim p_{\boldsymbol{\psi}}(N)$;

4   Sample batch $(\boldsymbol{x}, \boldsymbol{y}) \sim D_{\text{trn}}^{\tilde{i}}$;

5     $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \alpha \cdot \nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{y}|\boldsymbol{x}; \boldsymbol{\theta})$;

6   **if** *t % S == 0* **then**

7   **for** *i=1 to N* **do**

8         $(\boldsymbol{x}', \boldsymbol{y}') \sim D_{\text{trn}}^i$;

9   Compute reward $\mathcal{R}_{\{\cdot\}}(i)$ for D_{trn}^i as in Section 3.2;

10   **end**

11       $\boldsymbol{\psi} \leftarrow$ $\boldsymbol{\psi} + \sum_{i=1}^N \gamma \cdot \mathcal{R}_{\{\cdot\}}(i) \cdot \nabla_{\boldsymbol{\psi}} \log p_{\boldsymbol{\psi}}(i)$

12   **end**

13 **end**

---

network $\boldsymbol{\psi}$. Hence, the training objective becomes:

$$\boldsymbol{\psi} = \operatorname*{argmin}_{\boldsymbol{\psi}} \mathcal{J}(D_{\text{trn}}; \boldsymbol{\theta}(\boldsymbol{\psi})), \text{ where}$$
$$\boldsymbol{\theta}(\boldsymbol{\psi}) = \operatorname*{argmin}_{\boldsymbol{\theta}} \mathbb{E}_{i \sim p_{\boldsymbol{\psi}}(N)}[\mathcal{L}(D_{\text{trn}}^i; \boldsymbol{\theta})]. \quad (5)$$

Specifically, we present the algorithm of MoS in Algorithm 1. MoS initially parameterizes the initial sampling probability distribution with $\boldsymbol{\psi}$ as shown in Equation 3, using a warm-up temperature $\tau = \infty$ (see line 1). When updating the LLM $\boldsymbol{\theta}$, we employ the standard gradient-based optimization method (see line 5). For computational efficiency, the scorer network $\boldsymbol{\psi}$ is updated every S steps (see line 6). During updates of $\boldsymbol{\psi}$, we randomly draw one mini-batch from each training set $\{D_{\text{trn}}^i\}_{i=1}^N$ and compute the corresponding rewards as described in Section 3.2 (see line 9). The training dataset D_{trn}^i that yields a high reward is considered to be relatively more beneficial to the overall performance, and its corresponding sampling probability is increased (see line 11).

A critical issue in Algorithm 1 is that Equation 5 is not directly differentiable with respect to $\boldsymbol{\psi}$. To address this, reinforcement learning (RL)

with suitable reward functions is needed (Wang et al., 2020a). The update rule for ψ becomes:

$$\psi \leftarrow \psi + \sum_{i=1}^{N} \mathcal{R}_{\{\cdot\}}(i) \cdot \nabla_{\psi} \log p_{\psi}(i). \quad (6)$$

Details for the rewards $\mathcal{R}_{\{\cdot\}}(i)$ are in Section 3.2 and the update of the scorer network ψ follows the REINFORCE algorithm (Williams, 1992).

Implementing the scorer network Given that the scorer network ψ is primarily designed to model a relatively simple distribution over the training datasets D_{trn}, we utilize a fully connected 2-layer perceptron network for this task. The network takes as input a vector that specifies which training datasets are accessible. Note that the scorer network ψ is used for adjusting the data usage during the fine-tuning process and is orthogonal to the reward model used in RLHF.

3.2   Rewards for Learning

We design the rewards of MoS from three perspectives: transferability (Section 3.2.1), difficulty (Section 3.2.2), and learning trajectory (Section 3.2.3).

3.2.1   Transferability

Transferring knowledge from one problem to another related problem is beneficial, and this transferability is often measured by the similarity between datasets (Du et al., 2018; Zhuang et al., 2021). Datasets with higher similarity are more likely to make more contributions to the targeted performance of the model.

In this work, we represent the training datasets D_{trn} using the mini-batch embeddings and then calculate the pairwise cosine similarities among the mini-batch embeddings for each dataset. We draw a random mini-batch $B^i = \{(\boldsymbol{x}_j^i, \boldsymbol{y}_j^i)\}_{j=1}^{L}$ from D_{trn}^i, where L is the batch size, and the mini-batch embedding $\boldsymbol{z}^i$ is defined as:

$$\boldsymbol{z}^i = \frac{1}{L} \sum_{j=1}^{L} \boldsymbol{e}_j^i, \quad \boldsymbol{e}_j^i = \frac{1}{K} \sum_{k=1}^{K} \boldsymbol{h}_k, \quad (7)$$

where K is the sequence length of the concatenation of $\boldsymbol{x}_j^i$ and $\boldsymbol{y}_j^i$, and $\boldsymbol{h}_k$ is the hidden state of the token w_k in the concatenated sequence from the topmost layer of the LLM $\boldsymbol{\theta}_t$. Consequently, we define the reward $\mathcal{R}_{\text{COSSIM}}(i)$ for D_{trn}^i as the average cosine similarity among all training datasets:

$$\mathcal{R}_{\text{COSSIM}}(i) = \frac{1}{N} \sum_{n=1}^{N} \frac{\boldsymbol{z}^i \cdot \boldsymbol{z}^n}{\|\boldsymbol{z}^i\| \cdot \|\boldsymbol{z}^n\|}, \quad (8)$$

where N is the number of datasets in D_{trn}.

3.2.2   Difficulty

Recent work demonstrates that the transfer of knowledge between datasets is not always guaranteed (Wu et al., 2021). In response, we attempt to design the reward based on the inherent difficulty of the dataset in this section.

Recently, the perplexity is used for measure the dataset difficulty (Li et al., 2023b; Marion et al., 2023). Given a training example $(\boldsymbol{x}_j^i, \boldsymbol{y}_j^i)$ from D_{trn}^i and the LLM $\boldsymbol{\theta}$, the perplexity is defined as:

$$\begin{aligned} &\text{PPL}(\boldsymbol{y}_j^i; \boldsymbol{x}_j^i, \boldsymbol{\theta}) \\ &= \exp\left(-\frac{1}{|\boldsymbol{y}_j^i|} \sum_{k=1}^{|\boldsymbol{y}_j^i|} \log p_{\boldsymbol{\theta}}(y_{j,k}|\boldsymbol{x}_j^i, \boldsymbol{y}_{j,<k}) \right). \end{aligned} \quad (9)$$

However, we argue that perplexity is not a suitable metric for evaluating the difficulty of non-natural language texts, such as mathematical formulas and programming codes. Our preliminary study indicates that the perplexity scores given to mathematical texts by various language models are typically lower than those for natural language texts, despite the common belief that mathematical problems pose significant challenges for LLMs (Yue et al., 2023; Yu et al., 2023). Our preliminary study is presented in Appendix A. Therefore, given a random mini-batch $B^i = \{(\boldsymbol{x}_j^i, \boldsymbol{y}_j^i)\}_{j=1}^{L}$ from D_{trn}^i, the reward $\mathcal{R}_{\text{DIFF}}(i)$ for D_{trn}^i is:

$$\mathcal{R}_{\text{DIFF}}(i) = \frac{1}{L} \sum_{j=1}^{L} \frac{\text{PPL}(\boldsymbol{y}_j^i; \boldsymbol{x}_j^i, \boldsymbol{\theta})}{\text{PPL}(\boldsymbol{y}_j^i; \boldsymbol{x}_j^i, \boldsymbol{\theta}_0)}, \quad (10)$$

where $\boldsymbol{\theta}_0$ is the original LLM backbone and $\boldsymbol{\theta}$ is the fine-tuned LLM. The term $\mathcal{R}_{\text{DIFF}}(i)$ represents the relative decrease in perplexity for D_{trn}^i after fine-tuning. A high value of $\mathcal{R}_{\text{DIFF}}(i)$ suggests that D_{trn}^i is difficult to learn and requires more training efforts, while a lower value indicates the opposite.

3.2.3   Learning Trajectory

We design the rewards $\mathcal{R}_{\text{COSSIM}}(i)$ and $\mathcal{R}_{\text{DIFF}}(i)$ based on the transferability and difficulty of the training dataset D_{trn}^i, as discussed in Section 3.2.1 and Section 3.2.2. However, both rewards ignore the learning trajectory of the fine-tuning process. Therefore, we introduce the exponential moving average (EMA) when estimating the rewards. This approach can both better estimate the reward and stabilize the data usage optimization process.

	μ_{BOTH}	MMLU				MT-bench		
		μ_{MU}	Math	Med.	Others	μ_{MB}	Turn 1	Turn 2
QWEN1.5-0.5B								
PROP. ($\tau = 1$)	32.82	30.95	**23.40**	31.30	31.76	3.47	4.18	2.76
TEMP. ($\tau = 10$)	34.17	32.09	22.88	30.88	33.41	3.63	4.11	3.14
UNI. ($\tau = \infty$)	33.81	31.52	21.45	29.97	33.02	3.61	4.21	3.01
MoS + CosSim	34.30	31.95	21.90	31.18	33.28	3.67	3.96	3.38
MoS + CosSim + EMA	**35.13**	**32.45**	22.27	**31.56**	**33.82**	**3.78**	**4.44**	3.13
MoS + Diff	34.24	31.49	20.71	29.94	33.07	3.70	4.21	3.19
MoS + Diff + EMA	34.83	32.11	21.84	31.01	33.53	3.76	4.11	**3.40**
GEMMA-2B								
PROP. ($\tau = 1$)	42.90	33.61	20.03	33.33	35.25	5.22	5.63	4.81
TEMP. ($\tau = 10$)	41.86	**36.16**	**21.03**	**37.92**	37.55	4.76	5.49	4.03
UNI. ($\tau = \infty$)	43.95	35.95	20.82	35.97	**37.71**	5.19	5.60	4.79
MoS + CosSim	43.84	32.44	20.61	32.35	33.83	5.53	**6.16**	4.89
MoS + CosSim + EMA	44.49	33.86	20.57	33.32	35.52	5.51	6.01	5.01
MoS + Diff	44.93	34.32	20.29	34.06	36.00	5.55	6.04	**5.06**
MoS + Diff + EMA	**45.10**	34.61	20.83	34.64	36.20	**5.56**	6.13	4.99
LLAMA-3-8B								
PROP. ($\tau = 1$)	60.97	56.78	26.61	62.03	59.19	6.52	6.96	6.08
TEMP. ($\tau = 10$)	61.40	56.17	28.36	59.64	58.68	6.66	7.04	6.29
UNI. ($\tau = \infty$)	60.99	55.72	27.65	60.77	57.93	6.63	7.04	6.21
MoS + CosSim	62.49	56.95	28.91	59.91	59.59	6.80	7.11	6.50
MoS + CosSim + EMA	**63.85**	58.08	27.60	61.54	**60.90**	**6.96**	7.28	**6.65**
MoS + Diff	63.00	57.93	31.08	**62.65**	60.07	6.81	6.98	6.64
MoS + Diff + EMA	63.26	**58.34**	**32.81**	62.21	60.49	6.82	**7.34**	6.30

Table 1: Main results given by QWEN1.5-0.5B, GEMMA-2B, and LLAMA-3-8B on MMLU and MT-bench. The best and second-best results are highlighted in **bold** and underline. Note that μ_{MB} is upscaled by $10\times$ to a range from 1 to 100 used for computing μ_{BOTH}.

Specifically, we define the EMA as follows:

$$\mathcal{R}_{\{\cdot\}}(i) = \beta\mathcal{R}'_{\{\cdot\}}(i) + (1 - \beta)\mathcal{R}''_{\{\cdot\}}(i), \quad (11)$$

where β is the smoothing factor, $\mathcal{R}'_{\{\cdot\}}(i)$ indicates the original reward for the current update, $\mathcal{R}''_{\{\cdot\}}(i)$ represents the reward for the previous update, and $\mathcal{R}_{\{\cdot\}}(i)$ is the smoothed reward for the current update. Note that both $\mathcal{R}_{\text{CosSim}}(i)$ and $\mathcal{R}_{\text{Diff}}(i)$ can be applied in Equation 11 and we set $\beta = 0.9$.

4 Experiments

We present our experimental setup (Section 4.1) and main results (Section 4.2) in this section.

4.1 Experimental Setup

Datasets In this work, we collect supervised fine-tuning (SFT) datasets for four distinct skills: **Mathematics** (Yue et al., 2023), **Medicine** (Zhang et al., 2023a), **General** (ShareGPT),[1] and **NLP** (Sanh et al., 2022). Due to the constraint on compute, we sample 10% of examples from each dataset. More details can be found in Appendix B.

---

[1] https://huggingface.co/datasets/anon8231489123/ShareGPT_Vicuna_unfiltered

Model Backbones We apply MoS to three diverse model backbones, including QWEN1.5-0.5B (Bai et al., 2023), GEMMA-2B (Mesnard et al., 2024), and LLAMA-3-8B (AI@Meta, 2024). Optimization details are in Appendix C.

Baselines We compare MoS with three heuristic baselines based on Equation 3: proportional sampling (PROP., $\tau = 1$), temperature sampling (TEMP., $\tau = 10$), and uniform sampling (UNI., $\tau = \infty$). We do not use the maximum cap and other instance selection methods as our baselines because they fail to fully utilize all available data.

Evaluation In this work, we evaluate our models on two widely-used benchmarks that are highly correlated with human judgments: **MMLU** (Hendrycks et al., 2021a) and **MT-bench** (Zheng et al., 2023). We **conduct zero-shot evaluations** on MMLU and report the average accuracy across all the subjects as μ_{MU}. To better understand the model performance, we categorize the 57 subjects of MMLU into three groups: mathematics, medicine, and others, and also report the average accuracy of each group. Moreover, we report the average score across all eight skills of MT-bench as μ_{MB}. The overall per-
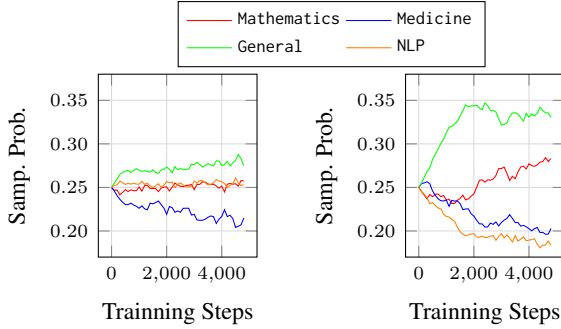
(a) MoS + CosSim + EMA (b) MoS + Diff + EMA

Figure 2: Learned dataset distribution given by Llama-3-8B with different variations of MoS. The x-axis indicates the training steps, and the y-axis indicates the sampling probabilities of datasets.

Figure 3: Training loss curves of heuristic baselines and MoS + Diff + EMA.

formance is reported as the average score of both μ_{MU} and μ_{MB}, denoted as μ_{BOTH}. Note that when computing μ_{BOTH}, MT-bench scores are upscaled by $10\times$ to range from 1 to 100, maintaining consistency with MMLU. More details are in Appendix D.

4.2 Main Results

We present the main results in Table 1.

An optimal temperature τ boosts performance, but no universally optimal τ exists. When comparing the heuristic baselines, we observe that there is no universally optimal τ that consistently works well for all model backbones. As shown in Table 1, Temp. ($\tau = 10$) performs best for Qwen1.5-0.5B and Llama-3-8B, but is least effective for Gemma-2B. This variability confirms the motivation behind this work.

MoS outperforms heuristic baselines, with larger models showing greater improvements. Our method consistently outperforms heuristic baselines across all three model backbones in terms of μ_{BOTH}. Notably, larger models show greater improvements with our approach. As shown in Table 1, the best variant of our method surpasses the best heuristic baseline by +0.96, +1.15, and +2.45 in μ_{BOTH} for Qwen1.5-0.5B, Gemma-2B, and Llama-3-8B, respectively. This is particularly significant in the era of **LARGE** language models.

Different rewards work better for different models, and EMA always helps. As shown in Table 1, MoS with CosSim outperforms MoS with Diff for Qwen1.5-0.5B and Llama-3-8B, while Diff-based MoS yields be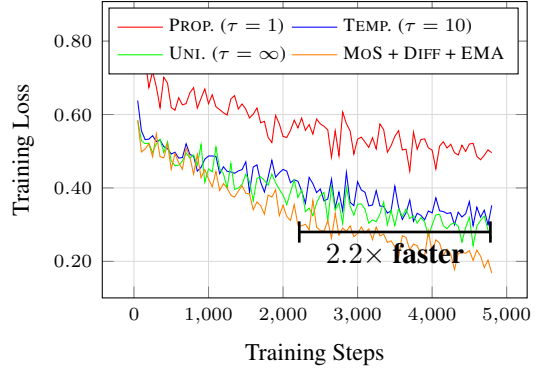tter results for Gemma-2B. Additionally, EMA consistently enhances overall performance in terms of μ_{BOTH}, supporting our rationales concerning learning trajectory in Section 3.2.3.

5 Analysis

In this section, we conduct an in-depth analysis of MoS using Llama-3-8B. Our analysis encompasses the learned dataset distribution of MoS, the training convergence speed of MoS, the impact of sampling priors on MoS, and its compatibility with the instance selection methods.

MoS with different rewards learns different dataset distributions. We visualize the dataset distribution learned by MoS using Llama-3-8B as shown in Figure 2. Starting with equal sampling probabilities, both CosSim and Diff variations of MoS increase the probability for General and decrease it for Medicine. However, CosSim maintains the probabilities for Mathematics and NLP, whereas Diff upsamples Mathematics but downsamples NLP.

MoS speeds up the convergence. We illustrate the training dynamics of heuristic baselines and MoS + Diff + EMA in Figure 3. When compared to Temp. ($\tau = 10$), which is the best-performing heuristic baseline, MoS + Diff + EMA demonstrates notable improvements. Specifically, it converges approximately $2.2\times$ faster, as shown in Figure 3, and achieves a +1.86 improvement in terms of μ_{BOTH}, as detailed in Table 1.

MoS demonstrates robustness to changes in sampling priors. As indicated in line 1 in Algorithm 1, we initialize our sampling probability distribution with $\tau = \infty$. Consequently, we investigate the effects of various sampling priors on MoS.

	μ_{BOTH}	μ_{MU}	μ_{MB}
TEMP. ($\tau = 10$)	61.40	56.17	6.66
MoS + CosSim + EMA			
$+\ \tau = 1$	62.49	56.95	6.80
$+\ \tau = 10$	**63.94**	57.98	**6.99**
$+\ \tau = \infty$	63.85	**58.08**	6.96
MoS + Diff + EMA			
$+\ \tau = 1$	62.29	56.51	6.81
$+\ \tau = 10$	**63.66**	58.22	**6.91**
$+\ \tau = \infty$	63.26	**58.34**	6.82

Table 2: Results of MoS with different sampling priors τ. The best results are highlighted in **bold**.

	μ_{BOTH}	μ_{MU}	μ_{MB}
RANDOM			
PROP. ($\tau = 1$)	59.78	54.39	6.52
TEMP. ($\tau = 10$)	60.62	54.89	6.63
UNI. ($\tau = \infty$)	59.21	54.21	6.42
IFD			
PROP. ($\tau = 1$)	60.43	55.03	6.58
TEMP. ($\tau = 10$)	61.02	55.13	6.69
UNI. ($\tau = \infty$)	60.62	54.92	6.63
MoS + CosSim + EMA	62.01	**56.81**	6.72
MoS + Diff + EMA	**62.05**	56.43	**6.77**

Table 3: Compatibility between MoS and IFD. RANDOM and IFD indicate the 10% data from each dataset selected by random sampling and IFD selection, respectively. The best results are highlighted in **bold**.

As shown in Table 2, MoS with different sampling priors consistently outperforms the best heuristic baseline TEMP. ($\tau = 10$). Moreover, selecting the appropriate sampling prior for MoS can further enhance performance. These results underscore the robustness and effectiveness of our approach.

MoS is compatible with the instance selection method. Following Li et al. (2023b), we leverage QWEN1.5-0.5B to calculate the Instruction-Following Difficulty (IFD) scores for each training instance and select the top 10% of training instances with the highest scores from each dataset. As shown in Table 3, combining MoS with IFD further improves the model performance, indicating the successful combination of MoS and IFD.

6 Fine-tuning from Generalist to Specialist

Large, general-purpose models offer broad capabilities but can be costly to deploy in real-world applications. Many scenarios require only a narrow set of functionalities, making smaller, specialized models more effective for specific tasks than larger,

	μ_{ALL}	GSM8K 5-shot	MATH 5-shot	M-math 0-shot
Generalist				
TEMP. ($\tau = 10$)	29.17	49.62	9.54	28.36
MoS + CosSim + EMA	29.26	50.40	9.78	27.60
MoS + Diff + EMA	30.88	49.58	10.26	**32.81**
Math Specialist				
MATHLLAMA-3-8B	27.04	41.02	9.76	30.34
MOSPEC + C.S. + E.	30.63	51.10	10.64	30.16
MOSPEC + D. + E.	**31.91**	**52.10**	**11.40**	32.24

Table 4: Results on GSM8K, MATH, and M-math given by generalists and math specialists. μ_{ALL} indicates the average performance over all three benchmarks. MOSPEC + C.S. + E. and MOSPEC + D. + E. indicates MOSPEC + CosSim + EMA and MOSPEC + Diff + EMA, respectively. The best results are highlighted in **bold**.

general-purpose ones (Luo et al., 2023; Azerbayev et al., 2023; Wu et al., 2024a). MIXTURE-OF-SKILLS (MoS) is a framework designed to optimize data usage for various fine-tuning purposes, including task-specific fine-tuning. This section explores the application of MoS in this context.

We aim to fine-tune LLAMA-3-8B to specialize in mathematics using datasets from Section 4.1, referring to this modified version as MOSPEC. For MOSPEC with CosSim, we compute the cosine similarity between Mathematics and other datasets, including Mathematics itself. For MOSPEC with Diff, we double the reward for the Mathematics dataset. For comparison, we fine-tune LLAMA-3-8B directly on Mathematics dataset in Section 4.1, denoted as MATHLLAMA-3-8B. Both MOSPEC and MATHLLAMA-3-8B use the identical hyperparameters from Appendix C, except MATHLLAMA-3-8B is fine-tuned for 12 epochs for fairness. We evaluate the models on math-related subjects in MMLU (0-shot, denoted as M-math), GSM8K (5-shot) (Cobbe et al., 2021), and MATH (5-shot) (Hendrycks et al., 2021b).

SFT datasets from other sources are beneficial for the specific target task. As shown in Table 4, the MATHLLAMA-3-8B model trained solely on the Mathematics subset performs the worst among all models. This indicates that incorporating additional SFT datasets is advantageous. The performance gap is particularly evident on the GSM8K dataset, which requires step-by-step reasoning. We believe this discrepancy arises from the Mathematics subset's incompleteness, while other SFT datasets can address these shortcomings.
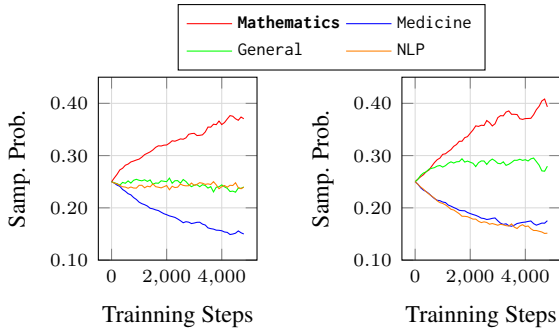
7

Figure 4: Learned dataset distribution given by LLAMA-3-8B with MOSPEC + COSSIM + EMA (left) and MO-SPEC + DIFF + EMA (right).

MOSPEC can harness diverse datasets to enhance task-specific performance. When comparing MOS and MOSPEC with the same reward type, MOSPEC consistently outperforms MOS on mathematical benchmarks in Table 4. By assigning a higher reward, MOSPEC effectively learns the optimal dataset distribution for learning the mathematical capabilities. As shown in Figure 4, COSSIM and DIFF in MOSPEC effectively guide the scorer network ψ to increase the sampling probability of Mathematics.

We believe this property of MOS is particularly meaningful when the task-specific dataset is not sufficiently large or comprehensive.

7 Related Work

Data Engineering for LLMs The success of large language models (LLMs) heavily relies on their training datasets. Researchers gather or create extensive datasets (Raffel et al., 2020; Gao et al., 2021; Penedo et al., 2023; Wang et al., 2023; Li et al., 2023a; Cui et al., 2023; Wu et al., 2024b) Recent efforts focus on selecting data subsets to enhance training efficiency. Xie et al. (2023) estimate the quality of each subset in the pretraining dataset mixture using a small proxy model. Recent approaches filter out low-quality examples using perplexity (Li et al., 2023b; Marion et al., 2023).

Dataset Rebalancing The standard practice for dataset rebalancing in fine-tuning large language models (LLMs) involves capping the number of examples per dataset (Raffel et al., 2020; Chung et al., 2022; Iyer et al., 2022; Wei et al., 2022). However, this approach does not fully utilize all available data. Previous multilingual research often rebalances datasets for multiple languages using

a temperature term τ (Arivazhagan et al., 2019; Conneau et al., 2020). Furthermore, Wang et al. (2020a) reweigh training examples based on their similarity with the validation set. Inspired by Wang et al. (2020a), Wang et al. (2020b) and Wu et al. (2021) propose rebalancing the dataset distribution for machine translation tasks.

Multi-Task Learning Our work is also related to multi-task learning (Ruder, 2017; Crawshaw, 2020; Zhang et al., 2023b). Both transferability and difficulty are commonly used for reweighting the importance of tasks to achieve better overall performance and mitigate the conflicts between tasks (Kendall et al., 2018; Chen et al., 2018; Yu et al., 2020; Wang et al., 2021). We highlight that *tasks* are the specific goals the model works towards, while *skills* are the broader abilities that allow the model to perform a wide range of tasks.

Ours In this work, MIXTURE-OF-SKILLS (MOS) is inspired by Wang et al. (2020a) and related to Wang et al. (2020b) and Wu et al. (2021), but offers several key advancements. Unlike previous methods, MOS does not require knowledge of downstream applications, avoiding the risk of overfitting to validation sets. Additionally, MOS introduces novel rewards tailored for LLMs and considers the learning trajectory during fine-tuning, enhancing overall performance. Finally, MOS is highly adaptable for specific fine-tuning needs, setting it apart from prior works.

8 Conclusion

In this work, we address the critical challenge of optimizing data usage during the fine-tuning process of LLMs. We propose a general, model-agnostic reinforcement learning framework, MIXTURE-OF-SKILLS (MOS), that dynamically adjusts dataset usage to enhance model performance with three novel rewards. Through extensive experiments on three diverse model backbones and two widely-used benchmarks, we demonstrate that MOS significantly improves overall model performance. Additionally, we explore the application of MOS in task-specific fine-tuning, leading to the development of MOSPEC. Our experiments show that models fine-tuned with MOSPEC on various datasets outperform those trained solely on task-specific datasets. In summary, MOS provides a powerful and flexible solution to the challenges of dataset heterogeneity and imbalance in the fine-tuning of LLMs.

9 Limitations

Computational Overhead In this study, the scorer network ψ and the large language model (LLM) θ are updated in an alternating fashion. Although the scorer network ψ is a relatively simple two-layer MLP model, the overall training duration increases by approximately 20%, compared with the heuristic baselines, when the LLM θ is updated for the same number of steps.

Number of Datasets Our experiments are limited to four datasets due to computational resource constraints. The performance of our approach as the dataset count increases remains unexplored.

These limitations are acknowledged and we leave them to the future work.

References

AI@Meta. 2024. Llama 3 model card.

Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M. Dai, Anja Hauth, Katie Millican, David Silver, Slav Petrov, Melvin Johnson, Ioannis Antonoglou, Julian Schrittwieser, Amelia Glaese, Jilin Chen, Emily Pitler, Timothy P. Lillicrap, Angeliki Lazaridou, Orhan Firat, James Molloy, Michael Isard, Paul Ronald Barham, Tom Hennigan, Benjamin Lee, Fabio Viola, Malcolm Reynolds, Yuanzhong Xu, Ryan Doherty, Eli Collins, Clemens Meyer, Eliza Rutherford, Erica Moreira, Kareem Ayoub, Megha Goel, George Tucker, Enrique Piqueras, Maxim Krikun, Iain Barr, Nikolay Savinov, Ivo Danihelka, Becca Roelofs, Anaïs White, Anders Andreassen, Tamara von Glehn, Lakshman Yagati, Mehran Kazemi, Lucas Gonzalez, Misha Khalman, Jakub Sygnowski, and et al. 2023a. Gemini: A family of highly capable multimodal models. *CoRR*, abs/2312.11805.

Rohan Anil, Andrew M. Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, Eric Chu, Jonathan H. Clark, Laurent El Shafey, Yanping Huang, Kathy Meier-Hellstern, Gaurav Mishra, Erica Moreira, Mark Omernick, Kevin Robinson, Sebastian Ruder, Yi Tay, Kefan Xiao, Yuanzhong Xu, Yujing Zhang, Gustavo Hernández Ábrego, Junwhan Ahn, Jacob Austin, Paul Barham, Jan A. Botha, James Bradbury, Siddhartha Brahma, Kevin Brooks, Michele Catasta, Yong Cheng, Colin Cherry, Christopher A. Choquette-Choo, Aakanksha Chowdhery, Clément Crepy, Shachi Dave, Mostafa Dehghani, Sunipa Dev, Jacob Devlin, Mark Díaz, Nan Du, Ethan Dyer, Vladimir Feinberg, Fangxiaoyu Feng, Vlad Fienber, Markus Freitag, Xavier Garcia, Sebastian Gehrmann, Lucas Gonzalez, and et al. 2023b. Palm 2 technical report. *CoRR*, abs/2305.10403.

Naveen Arivazhagan, Ankur Bapna, Orhan Firat, Dmitry Lepikhin, Melvin Johnson, Maxim Krikun, Mia Xu Chen, Yuan Cao, George F. Foster, Colin Cherry, Wolfgang Macherey, Zhifeng Chen, and Yonghui Wu. 2019. Massively multilingual neural machine translation in the wild: Findings and challenges. *CoRR*, abs/1907.05019.

Zhangir Azerbayev, Hailey Schoelkopf, Keiran Paster, Marco Dos Santos, Stephen McAleer, Albert Q. Jiang, Jia Deng, Stella Biderman, and Sean Welleck. 2023. Llemma: An open language model for mathematics. *CoRR*, abs/2310.10631.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, Jianxin Ma, Rui Men, Xingzhang Ren, Xuancheng Ren, Chuanqi Tan, Sinan Tan, Jianhong Tu, Peng Wang, Shijie Wang, Wei Wang, Shengguang Wu, Benfeng Xu, Jin Xu, An Yang, Hao Yang, Jian Yang, Shusheng Yang, Yang Yao, Bowen Yu, Hongyi Yuan, Zheng Yuan, Jianwei Zhang, Xingxuan Zhang, Yichang Zhang, Zhenru Zhang, Chang Zhou, Jingren Zhou, Xiaohuan Zhou, and Tianhang Zhu. 2023. Qwen technical report. *CoRR*, abs/2309.16609.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, Nicholas Joseph, Saurav Kadavath, Jackson Kernion, Tom Conerly, Sheer El Showk, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Tristan Hume, Scott Johnston, Shauna Kravec, Liane Lovitt, Neel Nanda, Catherine Olsson, Dario Amodei, Tom B. Brown, Jack Clark, Sam McCandlish, Chris Olah, Benjamin Mann, and Jared Kaplan. 2022. Training a helpful and harmless assistant with reinforcement learning from human feedback. *CoRR*, abs/2204.05862.

Zhao Chen, Vijay Badrinarayanan, Chen-Yu Lee, and Andrew Rabinovich. 2018. Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 793–802. PMLR.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Y. Zhao, Yanping Huang, Andrew M. Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling instruction-finetuned language models. *CoRR*, abs/2210.11416.

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. 2021. Training verifiers to solve math word problems. *CoRR*, abs/2110.14168.

Benoît Colson, Patrice Marcotte, and Gilles Savard. 2007. An overview of bilevel optimization. *Ann. Oper. Res.*, 153(1):235–256.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Michael Crawshaw. 2020. Multi-task learning with deep neural networks: A survey. *CoRR*, abs/2009.09796.

Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Wei Zhu, Yuan Ni, Guotong Xie, Zhiyuan Liu, and Maosong Sun. 2023. Ultrafeedback: Boosting language models with high-quality feedback. *CoRR*, abs/2310.01377.

Ning Ding, Yulin Chen, Bokai Xu, Yujia Qin, Shengding Hu, Zhiyuan Liu, Maosong Sun, and Bowen Zhou. 2023. Enhancing chat language models by scaling high-quality instructional conversations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3029–3051, Singapore. Association for Computational Linguistics.

Yunshu Du, Wojciech M. Czarnecki, Siddhant M. Jayakumar, Razvan Pascanu, and Balaji Lakshminarayanan. 2018. Adapting auxiliary losses using gradient similarity. *CoRR*, abs/1812.02224.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. 2021. The pile: An 800gb dataset of diverse text for language modeling. *CoRR*, abs/2101.00027.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021a. Measuring massive multitask language understanding. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021b. Measuring mathematical problem solving with the MATH dataset. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021, December 2021, virtual*.

Srinivasan Iyer, Xi Victoria Lin, Ramakanth Pasunuru, Todor Mihaylov, Daniel Simig, Ping Yu, Kurt Shuster, Tianlu Wang, Qing Liu, Punit Singh Koura, Xian Li, Brian O'Horo, Gabriel Pereyra, Jeff Wang, Christopher Dewan, Asli Celikyilmaz, Luke Zettlemoyer, and Ves Stoyanov. 2022. OPT-IML: scaling language model instruction meta learning through the lens of generalization. *CoRR*, abs/2212.12017.

Alex Kendall, Yarin Gal, and Roberto Cipolla. 2018. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 7482–7491. Computer Vision Foundation / IEEE Computer Society.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Haonan Li, Fajri Koto, Minghao Wu, Alham Fikri Aji, and Timothy Baldwin. 2023a. Bactrian-x : A multilingual replicable instruction-following model with low-rank adaptation. *CoRR*, abs/2305.15011.

Ming Li, Yong Zhang, Zhitao Li, Jiuhai Chen, Lichang Chen, Ning Cheng, Jianzong Wang, Tianyi Zhou, and Jing Xiao. 2023b. From quantity to quality: Boosting LLM performance with self-guided data selection for instruction tuning. *CoRR*, abs/2308.12032.

Shayne Longpre, Le Hou, Tu Vu, Albert Webson, Hyung Won Chung, Yi Tay, Denny Zhou, Quoc V. Le, Barret Zoph, Jason Wei, and Adam Roberts. 2023. The flan collection: Designing data and methods for effective instruction tuning. In *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pages 22631–22648. PMLR.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Ziyang Luo, Can Xu, Pu Zhao, Qingfeng Sun, Xiubo Geng, Wenxiang Hu, Chongyang Tao, Jing Ma, Qingwei Lin, and Daxin Jiang. 2023. Wizardcoder: Empowering code large language models with evol-instruct. *CoRR*, abs/2306.08568.

Max Marion, Ahmet Üstün, Luiza Pozzobon, Alex Wang, Marzieh Fadaee, and Sara Hooker. 2023. When less is more: Investigating data pruning for pretraining llms at scale. *CoRR*, abs/2309.04564.

Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, Pouya Tafti, Léonard Hussenot, Aakanksha Chowdhery, Adam Roberts, Aditya Barua, Alex Botev, Alex

Castro-Ros, Ambrose Slone, Amélie Héliou, Andrea Tacchetti, Anna Bulanova, Antonia Paterson, Beth Tsai, Bobak Shahriari, Charline Le Lan, Christopher A. Choquette-Choo, Clément Crepy, Daniel Cer, Daphne Ippolito, David Reid, Elena Buchatskaya, Eric Ni, Eric Noland, Geng Yan, George Tucker, George-Christian Muraru, Grigory Rozhdestvenskiy, Henryk Michalewski, Ian Tenney, Ivan Grishchenko, Jacob Austin, James Keeling, Jane Labanowski, Jean-Baptiste Lespiau, Jeff Stanway, Jenny Brennan, Jeremy Chen, Johan Ferret, Justin Chiu, and et al. 2024. Gemma: Open models based on gemini research and technology. *CoRR*, abs/2403.08295.

OpenAI. 2023. GPT-4 technical report. *CoRR*, abs/2303.08774.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *NeurIPS*.

Guilherme Penedo, Quentin Malartic, Daniel Hesslow, Ruxandra Cojocaru, Hamza Alobeidli, Alessandro Cappelli, Baptiste Pannier, Ebtesam Almazrouei, and Julien Launay. 2023. The refinedweb dataset for falcon LLM: outperforming curated corpora with web data only. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67.

Sebastian Ruder. 2017. An overview of multi-task learning in deep neural networks. *CoRR*, abs/1706.05098.

Victor Sanh, Albert Webson, Colin Raffel, Stephen H. Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Arun Raja, Manan Dey, M Saiful Bari, Canwen Xu, Urmish Thakker, Shanya Sharma Sharma, Eliza Szczechla, Taewoon Kim, Gunjan Chhablani, Nihal V. Nayak, Debajyoti Datta, Jonathan Chang, Mike Tian-Jian Jiang, Han Wang, Matteo Manica, Sheng Shen, Zheng Xin Yong, Harshit Pandey, Rachel Bawden, Thomas Wang, Trishala Neeraj, Jos Rozen, Abheesht Sharma, Andrea Santilli, Thibault Févry, Jason Alan Fries, Ryan Teehan, Teven Le Scao, Stella Biderman, Leo Gao, Thomas Wolf, and Alexander M. Rush. 2022. Multitask prompted training enables zero-shot task generalization. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurélien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023a. Llama: Open and efficient foundation language models. *CoRR*, abs/2302.13971.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton-Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurélien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023b. Llama 2: Open foundation and fine-tuned chat models. *CoRR*, abs/2307.09288.

Xinyi Wang, Hieu Pham, Paul Michel, Antonios Anastasopoulos, Jaime G. Carbonell, and Graham Neubig. 2020a. Optimizing data usage via differentiable rewards. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 9983–9995. PMLR.

Xinyi Wang, Yulia Tsvetkov, and Graham Neubig. 2020b. Balancing training for multilingual neural machine translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8526–8537, Online. Association for Computational Linguistics.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. Self-instruct: Aligning language models with self-generated instructions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13484–13508, Toronto, Canada. Association for Computational Linguistics.

Zirui Wang, Yulia Tsvetkov, Orhan Firat, and Yuan Cao. 2021. Gradient vaccine: Investigating and improving multi-task optimization in massively multilingual models. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. 2022. Finetuned

language models are zero-shot learners. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Ronald J. Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Mach. Learn.*, 8:229–256.

Minghao Wu, Yitong Li, Meng Zhang, Liangyou Li, Gholamreza Haffari, and Qun Liu. 2021. Uncertainty-aware balancing for multilingual and multi-domain neural machine translation training. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7291–7305, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Minghao Wu, Thuy-Trang Vu, Lizhen Qu, George F. Foster, and Gholamreza Haffari. 2024a. Adapting large language models for document-level machine translation. *CoRR*, abs/2401.06468.

Minghao Wu, Abdul Waheed, Chiyu Zhang, Muhammad Abdul-Mageed, and Alham Aji. 2024b. LaMini-LM: A diverse herd of distilled models from large-scale instructions. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 944–964, St. Julian's, Malta. Association for Computational Linguistics.

Sang Michael Xie, Hieu Pham, Xuanyi Dong, Nan Du, Hanxiao Liu, Yifeng Lu, Percy Liang, Quoc V. Le, Tengyu Ma, and Adams Wei Yu. 2023. Doremi: Optimizing data mixtures speeds up language model pretraining. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Longhui Yu, Weisen Jiang, Han Shi, Jincheng Yu, Zhengying Liu, Yu Zhang, James T. Kwok, Zhenguo Li, Adrian Weller, and Weiyang Liu. 2023. Metamath: Bootstrap your own mathematical questions for large language models. *CoRR*, abs/2309.12284.

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. 2020. Gradient surgery for multi-task learning. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Xiang Yue, Xingwei Qu, Ge Zhang, Yao Fu, Wenhao Huang, Huan Sun, Yu Su, and Wenhu Chen. 2023. Mammoth: Building math generalist models through hybrid instruction tuning. *CoRR*, abs/2309.05653.

Xinlu Zhang, Chenxin Tian, Xianjun Yang, Lichang Chen, Zekun Li, and Linda Ruth Petzold. 2023a. Alpacare: Instruction-tuned large language models for medical application. *CoRR*, abs/2310.14558.

Zhihan Zhang, Wenhao Yu, Mengxia Yu, Zhichun Guo, and Meng Jiang. 2023b. A survey of multi-task learning in natural language processing: Regarding task relatedness and training methods. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 943–956, Dubrovnik, Croatia. Association for Computational Linguistics.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *CoRR*, abs/2306.05685.

Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, and Qing He. 2021. A comprehensive survey on transfer learning. *Proc. IEEE*, 109(1):43–76.

A Preliminary Study on Perplexity

Perplexity measures how well a probability model predicts a sample, quantifying the model's uncertainty in making these predictions. It is designed for natural language texts because it relies on the probability distributions typical to human languages. However, non-natural language texts, such as mathematical formulas or programming code, often involve symbols and structures whose relationships are governed by logical or mathematical rules rather than linguistic context. We hypothesize that using perplexity to measure the difficulty in these contexts does not capture the essential aspects of understanding or generating such texts.

To verify our hypothesis, we conduct a preliminary study on perplexity and present the results in Table 5. We observe that the perplexity of Mathematics is commonly lower than that of other datasets given by QWEN1.5-0.5B, GEMMA-2B, and LLAMA-3-8B, regardless of whether the models are fine-tuned, while the perplexity of NLP is the highest among all the datasets. However, Mathematics is associated with a higher value of Δ, while NLP achieves the lowest value of Δ, suggesting that Mathematics is difficult to learn while NLP is easy to learn. If we utilize the perplexity as a measure of difficulty in Section 3.2.2, MOS incorrectly assigns a higher sampling probability to NLP.

B Training Datasets

In this work, we collect four distinct supervised fine-tuning (SFT) datasets:

- **Mathematics**: Yue et al. (2023) introduce MathInstruct, a comprehensive collection of mathematical SFT datasets.[2]
- **Medicine**: Zhang et al. (2023a) introduce a medical SFT dataset MedInstruct.[3]
- **General**: The ShareGPT dataset serves as our general SFT dataset, contributed by the general public and characterized by a high degree of diversity and quality.[4]
- **NLP**: Sanh et al. (2022) open-source P3, which is a collection of prompted English datasets covering a diverse set of NLP tasks.[5] Note

that the P3 collection consists of 660 subsets, totaling 122M examples. To ensure task diversity, we initially randomly select 1K examples from each subset.

Due to the constraint on compute, we sample 10% of examples from each dataset and present the dataset statistics in Table 6.

C Optimization

We fine-tune all the parameters of large language models (LLMs) using the AdamW optimizer (Kingma and Ba, 2015; Loshchilov and Hutter, 2019) with a learning rate of 1×10^{-5} and a batch size of 64. We fine-tune all the models for 3 epochs, or the equivalent number of steps. During the training process, we apply the linear learning rate schedule, which includes a warm-up phase comprising 10% of the total training steps. For MOS, ψ is updated for every 100 steps with the learning rate of 1×10^{-4} and the batch size of 64. ψ is initialized by $\tau = \infty$ in Equation 3.

D Evaluation

In this work, we evaluate two widely-used benchmarks that are highly correlated with human judgments:

- **MMLU**: Hendrycks et al. (2021a) propose the MMLU benchmark, covering 57 subjects across STEM, humanities, social sciences, and more. We categorize the subjects into three groups: mathematics, medicine, and others, and **conduct zero-shot evaluations**. We report the average accuracy for each group and the overall accuracy across all subjects, denoted as μ_{MU}. Detailed subject categorization is available in Table 7.
- **MT-bench**: Zheng et al. (2023) propose the MT-bench, a multi-turn conversational benchmark designed to measure large language models' capabilities. This benchmark covers eight key skills, including coding, writing, roleplay, and more. LLM responses are scored by GPT-4 on a scale from 1 to 10. The overall score across all eight skills is denoted as μ_{MB}.

The overall performance is reported as the average score of both μ_{MU} and μ_{MB}, denoted as μ_{BOTH}. Note that when computing μ_{BOTH}, MT-bench scores are upscaled by $10\times$ to range from 1 to 100, maintaining consistency with MMLU.

---

[2]https://huggingface.co/datasets/TIGER-Lab/MathInstruct
[3]https://huggingface.co/datasets/casey-martin/MedInstruct
[4]https://huggingface.co/datasets/anon8231489123/ShareGPT_Vicuna_unfiltered
[5]https://huggingface.co/datasets/bigscience/P3

	QWEN1.5-0.5B			GEMMA-2B			LLAMA-3-8B		
	PPL_{θ_0}	PPL_{θ}	Δ	PPL_{θ_0}	PPL_{θ}	Δ	PPL_{θ_0}	PPL_{θ}	Δ
Mathematics	4.18	2.94	0.70	5.85	2.31	0.39	5.65	2.94	0.52
Medicine	8.38	4.45	0.53	8.60	2.95	0.34	5.86	2.51	0.43
General	6.05	4.01	0.66	9.66	3.51	0.36	4.25	2.51	0.59
NLP	37.70	7.98	0.21	49.28	4.78	0.10	29.79	4.19	0.14

Table 5: Preliminary results on perplexity. PPL_{θ_0} and PPL_{θ} are the average perplexity scores on each subset given by the original LLM backbone and the fine-tuned LLM with PROP. ($\tau = 1$), respectively. $\Delta = \frac{\text{PPL}_{\theta}}{\text{PPL}_{\theta_0}}$ indicates the relative decrease in perplexity. A high value of Δ indicates the dataset is difficult to learn, while a lower value indicates the opposite.

	#exam.	#words	Inst.L	Resp.L	Turns
Mathematics	26.2K	3.4M	47.6	84.0	1.0
Medicine	5.2K	1.0M	36.5	147.4	1.0
General	9.3K	9.3M	54.3	202.4	3.6
NLP	62.6K	8.6M	127.9	9.7	1.0
Total	103.4K	22.3M	88.4	84.3	1.2

Table 6: Dataset statistics of the training datasets in this work. Inst.L, Resp.L, and Turns indicate the average of instruction length (in words), response length (in words), and number of conversation turns.

	Subjects
Mathematics	Abstract Algebra, College Mathematics, Elementary Mathematics, High School Mathematics, High School Statistics
Medicine	Anatomy, Clinical Knowledge, College Medicine, Human Aging, Human Sexuality, Medical Genetics, Nutrition, Professional Medicine, Virology
Others	Astronomy, Business Ethics, College Biology, College Chemistry, College Computer Science, College Physics, Computer Security, Conceptual Physics, Econometrics, Electrical Engineering, Formal Logic, Global Facts, High School Biology, High School Chemistry, High School Computer Science, High School European History, High School Geography, High School Government And Politics, High School Macroeconomics, High School Microeconomics, High School Physics, High School Psychology, High School US History, High School World History, International Law, Jurisprudence, Logical Fallacies, Machine Learning, Management, Marketing, Miscellaneous, Moral Disputes, Moral Scenarios, Philosophy, Prehistory, Professional Accounting, Professional Law, Professional Psychology, Public Relations, Security Studies, Sociology, US Foreign Policy, World Religions

Table 7: MMLU subject categorization.