

A Unified Understanding of Offline Data Selection and Online Self-refining Generation for Post-training LLMs

Quan Xiao

Tianyi Chen

School of Electrical and Computer Engineering
Cornell Tech, Cornell University, New York, NY

Abstract

Offline data selection and online self-refining generation, which enhance the data quality, are crucial steps in adapting large language models (LLMs) to specific downstream tasks. We tackle offline data selection and online self-refining generations through an optimization perspective. Specifically, bilevel data selection is used for offline data selection with respect to the validation dataset, and we treat online self-refining generation as a model adaptation step of selecting the model trained on current responses that best fits the validation data. Our framework offers a unified understanding of offline data selection and self-refining generation by assigning a learned data weight to each question and response, either explicitly or implicitly. For the first time, we theoretically demonstrate the effectiveness of the bilevel data selection framework and demonstrate its performance gains over unfiltered direct mixing baselines. By combining offline data with validation-weighted online generations, our method enhances fine-tuning performance. Experiments on quality enhancement and safety-aware LLM fine-tuning validate its effectiveness.

1 Introduction

Large language models (LLMs) have demonstrated their remarkable empirical success across various domains. To effectively adapt pre-trained LLMs for downstream tasks, the fine-tuning pipeline combines the training stage of supervised fine-tuning (SFT) with a generation stage designed to generalize well on unseen validation data [24, 71].

SFT equips the model with domain-specific knowledge, while we evaluate the generated responses from the model on the validation dataset. The performance of

this pipeline, however, hinges critically on two factors: (i) the quality of the SFT dataset; and (ii) the adaptation of the model to the validation dataset.

While obtaining a high-quality SFT dataset can be costly, it is possible to utilize massive low-quality data through data selection. For example, if the SFT dataset is mixed with safe and unsafe responses, a small amount of high-quality safe validation data can be used to select the data in the SFT dataset [8, 58, 78, 80]. On the other hand, the model adaption on different dataset is naturally a bilevel multi-objective problem, where the lower-level focuses on optimizing SFT performance per-task, and the upper-level guides the model towards the one aligns most with the validation dataset [36, 37, 70].

In this work, we propose a unifying understanding for both data selection and model adaptation. Both of them selects the data to align with the validation dataset — either explicitly, through bilevel data selection, or implicitly, through bilevel multi-objective optimization. However, recent theoretical results have suggested that data selection does not necessarily improve over standard training [81]. Instead, by interpreting the selected data through the minimizer decomposition, we show that bilevel data selection effectively removes low-quality samples with respect to the validation dataset and theoretically justify why selection can outperform B1) naive training on unfiltered SFT dataset, and B2) direct mixing of SFT and validation data.

Beyond this offline data selection setting, we address two practical limitations. First, static data weights for the offline dataset cannot disentangle low-quality questions from poor responses, nor adapt to the evolving state of the model. To overcome this, we extend our framework to an online setting: the model generates on-policy self-refining responses for each question, and we reweight these pairs through bilevel data selection. Second, to reduce the high cost of online data generation, we employ importance sampling (IS) to reuse responses from older policies. We prove that IS weights

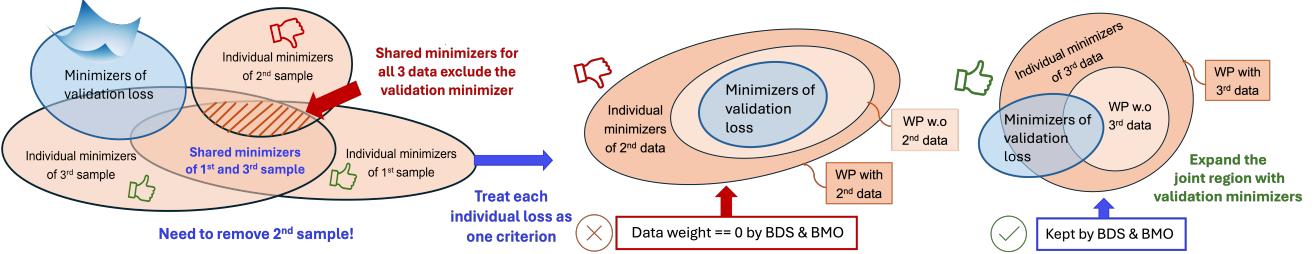


Figure 1: An overview of bilevel data selection principle. ‘WP’ in the figure is short for weak Pareto optimal set, ‘w.o’ is short for without, and 1 – 3 samples are drawn from the lower-level SFT dataset. The orange 2D plates depict the sets of per-sample SFT-loss minimizers, while the blue surface denotes the 3D validation-loss landscape. validation loss achieves minimum on both individual minimizer of 3rd sample and the shared minimum of 1st and 3rd. Optimizing the validation loss at the shared minimum of all lower-level sample losses degrades performance, but it achieves optimum if we remove 2nd sample.

are proportional to the implicit data weights given by bilevel multi-objective optimization, thereby measure the response-level validation performance. Together, we present a unified view of offline data selection and online self-refining generation for LLM fine-tuning: the former uses static, question-level weights to select the training SFT data, while the latter additionally selects the generated responses via response-level weights. Both contribute to the validation performance.

1.1 Related works

Data selection and mixing. Data selection and mixing are two important data curation pipelines to enhance the quality of the SFT dataset. Existing works of data selection rank and filter the question-answer pairs through various criteria, including helpfulness [64], validation alignment [29, 89], influence score [38, 82], safety [8, 58], token-level statistics [51]. Besides, data mixing enhances the diversity of the SFT dataset [40, 77]. Recently, the importance of mixing offline data with online data in preference and instruction tuning has been recognized and extensively studied [43, 60, 62, 84]. However, none of them have justified the theoretical benefits of data selection over direct data mixing. Notably, theoretical results on data selection based on sufficient datasets [3, 19] are closely related to our notion of usefulness. However, instead of identifying a minimal informative set, we target at selecting validation-aligned data samples and refining the non-optimal responses.

Self-training. By viewing the question-only data as unsupervised (unlabeled) data, our approach is closely related to the self-training paradigm [24, 71, 76, 85] which generates pseudo-labels (online responses) for unlabeled data using the current model. Instead of equally leveraging all of the online responses, we assign data weights to measure their validation performance.

Bilevel and multi-objective learning. Bilevel op- timization is powerful to tackle various machine learn-

ing applications [15, 41, 87]. Efficient gradient-based bilevel methods are built upon unrolling differentiation [14, 15, 23], implicit differentiation [7, 20, 25, 30, 52], conjugate gradients [2, 27, 34, 42, 75, 79], and penalty approach [32, 45, 59]. Bilevel multi-objective optimization [6, 10, 11, 26, 47] has board applications on multi-task learning [73] and LLM fine-tuning [54, 69]. A classical result is that bilevel multi-objective optimization is equivalent to bilevel data selection when each lower-level single objective is convex [11], but this is not applicable to LLM fine-tuning because the objectives are nonconvex in general.

1.2 Our contributions

In this paper, we focus on SFT data selection and its self-refinement guided by a small set of high-quality validation data consisting of trusted or human-curated examples. We summarize our contributions below.

- C1) This paper demonstrates the theoretical benefits of data selection against the direct mixing strategy for the first time. We show the effectiveness of bilevel data selection of the LLM fine-tuning problem through the lens of bilevel multi-objective optimization by treating each individual data loss as a competing criterion. See an overview in Figure 1.
- C2) We extend the bilevel data selection framework from offline data selection to online self-refining generation to enhance the evaluation performance. By including the online data, the model continues to improve its on-policy responses iteratively. To reduce the generation cost, we employ an importance sampling strategy to reuse the samples from the old policy, and we prove that the importance ratio is proportional to the implicit weight given by bilevel multi-objective optimization. Thus, it can serve as a per-response data weight for the same question.
- C3) Experiments on quality enhancement and safety aware LLM fine-tuning demonstrate the effective-

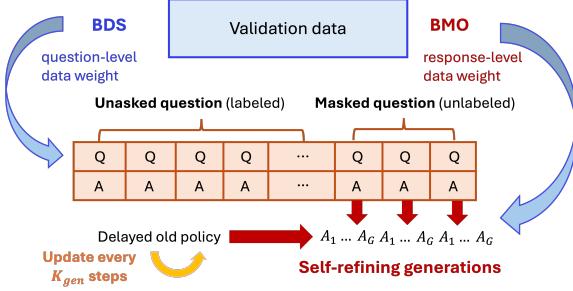


Figure 2: An overview of our online self-refining algorithm design. ‘Q’ and ‘A’ are short for Question and Answer. We masked part of the offline responses to the question and generated on-policy responses instead. We assign both question-level validation score via bilevel data selection (BDS) and response-level data weight via bilevel multi-objective learning (BMO).

ness of the proposed online self-refining algorithm.

We give a preview of our algorithm design in Figure 2.

2 Preliminaries

Notations. We define $\bar{\mathbb{R}}^d := (\mathbb{R} \cup \{\pm\infty\})^d$. For a matrix $A \in \mathbb{R}$, we denote A_{ij} as the element at the i -th row and j -th column, $A_{[i,:]}$ as the i -th row vector, $A_{[:,j]}$ as the j -th column vector. Denote $\mathcal{Y} \times \dots \times \mathcal{Y}$ repeated D times as \mathcal{Y}^D and $[M] = \{1, \dots, M\}$. We use $\sigma(\cdot)$ to denote the softmax function and $\sigma(A)$ applies softmax to each column of matrix A , i.e. $\sigma(A)_{ij} = \frac{\exp(A_{ij})}{\sum_k \exp(A_{kj})}$.

SFT. In the SFT task, the input sequence x is the question, the output sequence $y = (y_1, \dots, y_D)$ is the target response of length D with each token $y_d \in \mathcal{Y}$ from the vocabulary set of size $|\mathcal{Y}| = V$. For any data sample $(x, y) \sim \mathcal{D}_{\text{SFT}}$, the per-sample SFT loss is the negative log-likelihood of the next-token prediction [72]

$$\mathcal{L}_{\text{SFT}}(\theta; x, y) = - \sum_{d=1}^D \mathbf{e}_{y_d}^\top \log \pi_\theta(y_d | x, y_{<d}) \quad (1)$$

where $y_{<d} = \{y_1, \dots, y_{d-1}\}$ and $y_{<1}$ is defined as the empty sequence, $\mathbf{e}_{y_d} \in \mathbb{R}^V$ is the one-hot vector of token y_d , $\theta \in \mathbb{R}^h$ is the LLM parameter and $\pi_\theta(y | x)$ the softmax policy. Due to the auto-regressive nature of SFT loss, a major challenge of (1) compared with the multi-label classification is that both input x and output y are fed into the policy. However, by employing the causal mask [67], even if we provide the whole sequence $(x, y) \sim \mathcal{D}_{\text{SFT}}$ as input to the model, $y_{\geq d}$ remains invisible until the prediction of y_d . Therefore, by denoting the backbone model with causal mask as $\phi_\theta(x, y) \in \mathbb{R}^{V \times D}$, we consider the softmax policy as

$$\pi_\theta(y | x, y) = \sigma(\phi_\theta(x, y)) \quad (2)$$

and then the policy for token y_d can be chosen as [53]

$$\pi_\theta(y_d | x, y_{<d}) = \pi_\theta(y | x, y_{[:d]}) \quad (3)$$

Bilevel data selection. Assume we have a massive low-quality SFT dataset $\mathcal{D}_{\text{SFT}}^- = \{(x^i, y^i)\}_{i=1}^N$ and a small high-quality dataset $\mathcal{D} = \{(\tilde{x}^i, \tilde{y}^i)\}_{i=1}^{N'}$, which can be either SFT or offline reinforcement learning (RL) dataset for evaluation where $N' \ll N$. The goal of bilevel data selection (BDS) is to select data from the low-quality dataset that yields comparable SFT performance on the high-quality dataset. To do so, we solve

$$\begin{aligned} \text{BDS : } \min_{\omega \in \mathbb{R}^N, \theta} \mathcal{L}_0(\theta) &:= \frac{1}{N'} \sum_{i=1}^{N'} \mathcal{L}_0(\theta; \tilde{x}^i, \tilde{y}^i) \\ \text{s.t. } \theta &\in \arg \min_{\theta'} \frac{1}{N} \sum_{i=1}^N \sigma_i(\omega) \mathcal{L}_{\text{SFT}}(\theta'; x^i, y^i) \end{aligned} \quad (4)$$

where the softmax operator $\sigma(\cdot)$ is to ensure the data weight $\sigma_i(\omega)$ on (x^i, y^i) is within the simplex and \mathcal{L}_0 is chosen as the corresponding SFT or rule-based RL loss, depending on the choice of \mathcal{D} . We assume that at least a portion of the low-quality dataset is reusable. Given this precondition, the simplex constraint on ω prevents the trivial all-zero solution, ensuring that some data from the low-quality dataset is selected.

Bilevel multi-objective fine-tuning. We consider the setting where we have M fine-tuning criteria \mathcal{L}_m and a validation dataset $\mathcal{D} = \{(\tilde{x}^i, \tilde{y}^i)\}_{i=1}^{N'}$. Let \mathcal{L}_0 be the corresponding SFT or offline RL loss. The goal of bilevel multi-objective (BMO) fine-tuning is to select LLMs using the validation dataset from the Pareto front of the multiple fine-tuning criteria $\mathcal{L}(\theta) = (\mathcal{L}_1(\theta), \dots, \mathcal{L}_M(\theta))$ [6, 88]

$$\text{BMO : } \min_{\theta} \frac{1}{N'} \sum_{i=1}^{N'} \mathcal{L}_0(\theta; \tilde{x}^i, \tilde{y}^i), \text{ s.t. } \theta \in \text{WP}(\mathcal{L}) \quad (5)$$

where each \mathcal{L}_m can be SFT loss evaluated on different data points or datasets, and $\text{WP}(\mathcal{L})$ denotes the weakly Pareto front of $\mathcal{L}(\theta)$ defined below.

Definition 1. The solution θ is *weakly Pareto optimal* of the vector objective function $\mathcal{L}(\theta)$ if there is no θ' such that $\forall m \in [M], \mathcal{L}_m(\theta') < \mathcal{L}_m(\theta)$.

When the validation function is chosen as the SFT loss on a high-quality dataset, it suggests that BMO can guide the LLM to better align with high-quality data, which appears similar in spirit to BDS. In this paper, we provide a formal justification of this insight and use BMO as a bridge to interpret the weight learned by BDS.

3 Optimization Interpretation of Bilevel Data Selection

To interpret the data weight assigned by bilevel data selection, we will first link BDS to BMO and prove the effectiveness of BDS through the lens of BMO.

3.1 Viewing BDS from BMO

In this section, we will prove that BMO is equivalent to BDS. We focus on the case where the SFT and validation data are separable.

Assumption 1 (Separable data). *There exists $\theta \in \mathbb{R}^h$ such that $\mathcal{L}_{SFT}(\theta; x^i, y^i) = 0$ for $\forall i \in [N]$. Also, there exists $\theta' \in \mathbb{R}^h$ such that $\mathcal{L}_0(\theta'; \tilde{x}^i, \tilde{y}^i) = 0$ for $\forall i \in [N']$.*

This assumption is commonly used in deep learning theory [28, 48, 61, 65, 81], and is empirically justified or used for memory-efficient algorithm design [56, 58, 73]. Moreover, this assumption is likely to hold for an over-parameterized model $\phi_\theta(x, y)$ where we have zero training loss [1, 74, 83]. Although we assume the lower-level SFT and validation datasets are individually separable, we do not assume their joint separability, i.e., $\exists \theta$ yields zero per-sample loss on both. In principle, only a subset of the SFT data is useful, namely, those that can achieve zero loss on validation datasets.

Theorem 1 (Equivalence of BMO and BDS). *Suppose Assumption 1 holds. For BMO in (5) with $\mathcal{L}_i(\theta) = \mathcal{L}_{SFT}(\theta; x^i, y^i)$ for $i \in [N]$, where $(x^i, y^i) \in \mathcal{D}_{SFT}^- = \{(x^i, y^i)\}_{i=1}^N$ and $M = N$, any global (or local) solution θ^* of BMO is also a global (or local) solution of BDS in (4) paired with some ω^* , and vice versa.*

This theorem suggests that BMO can also select the data by treating each individual loss of lower-level SFT data as a separate objective. When guided by the upper-level validation dataset, BMO implicitly assigns data weights to the lower-level SFT dataset.

We will then interpret the optimal data weight $\sigma(\omega^*)$ assigned by BDS through the lens of BMO.

3.2 Bilevel data selection selects useful data

To interpret the data weight, we first need to define the *useful* and *useless* data samples from the optimization perspective. If an LLM trained on one dataset achieves the minimal validation loss, it suggests that this dataset is high-quality. Similarly, we can define the notion of *usefulness* of each data sample by evaluating its optimal LLM on validation dataset.

Since the SFT loss for each data sample may admit multiple minimizers, we first define the individual minimizer to eliminate the coupling effects between the SFT losses of different data samples.

Definition 2 (Individual minimizer). For i -th data sample, we say $\theta_i^* \in \arg \min_{\theta} \mathcal{L}_{SFT}(\theta; x^i, y^i)$ is one individual minimizer of $\mathcal{L}_{SFT}(\theta; x^i, y^i)$ if there exists no $j \neq i$ that $\theta_i^* \in \arg \min_{\theta} \mathcal{L}_{SFT}(\theta; x^j, y^j)$.

Without loss of generality, we assume at least one individual minimizer exists for each i . Otherwise, the

SFT loss for a given sample can be reduced by another sample that fully contains its information.

Definition 3 (Useful samples). Define $\Theta_i := \arg \min_{\theta} \mathcal{L}_{SFT}(\theta; x^i, y^i)$ as the set of *individual minimizers* for the i -th SFT sample defined in Def. 2, and define $\Theta_{\text{val}} := \arg \min_{\theta} \mathcal{L}_0(\theta)$ as the set of validation minimizers. We say that (x^i, y^i) is

- *useful* if $\Theta_i \cap \Theta_{\text{val}} \neq \emptyset$;
- *useless* otherwise, i.e., $\Theta_i \cap \Theta_{\text{val}} = \emptyset$.

With this definition, a sample is considered *useful* if there exists at least one model that fits it perfectly and is also optimal on the validation data. If no such model exists, fitting this sample perfectly inevitably hurts validation performance, so it is *useless*.

Remark 1. According to Definition 3, each data sample in the low-quality SFT dataset can be labeled by evaluating the validation loss at the LLM optimally fine-tuned on that sample, i.e. $\mathcal{L}_0(\theta_i^*)$. However, this approach is impractical because it is impossible to obtain all of the individual minimizers of one data sample.

The next theorem demonstrates that BDS in (4) is effective to remove all of *useless* and select only *useful* data samples from the low-quality dataset.

Theorem 2 (BDS can select useful data). *Suppose Assumption 1 holds and, under Definition 3, there exists at least one useful sample in the low-quality dataset \mathcal{D}_{SFT}^- . If (x^i, y^i) is useless, then for any optimal solution (ω^*, θ^*) for BDS, we have $\sigma_i(\omega^*) = 0$. Conversely, for any optimal solution for BDS, if $\sigma_i(\omega^*) > 0$, then (x^i, y^i) is useful.*

Theorem 2 shows that BDS can remove all of *useless* data points with respect to the validation dataset. Building upon Theorem 2, we will show that BDS (and BMO) yield a model that *strictly improves* validation loss over (1) training on the lower-level SFT dataset alone and (2) training on a mixture of lower-level SFT and validation data. The key reason is that both (1) and (2) are using the full low-quality SFT dataset, which does not have a shared minimizer with the validation dataset due to the distraction of *useless* samples. By removing those useless samples, BDS and BMO are able to achieve better validation performance.

Theorem 3. *Under the conditions in Theorem 2 and denoting the optimal model given by BDS and BMO as θ^* , then for any mixing parameter $0 < \rho \leq 1$, we have*

$$\mathcal{L}_0(\theta^*) < \min_{\tilde{\theta} \in \mathcal{S}_{\text{mix}}} \mathcal{L}_0(\tilde{\theta})$$

where $\mathcal{S}_{\text{mix}} := \arg \min_{\theta'} \frac{\rho}{N} \sum_{i=1}^N \mathcal{L}_{SFT}(\theta'; x^i, y^i) + (1 - \rho) \mathcal{L}_0(\theta')$ denotes the set of optimal models using the weighted mixture of upper-level and lower-level data.

Moreover, when $\rho = 1$, \mathcal{S}_{mix} denotes the set of optimal models using the full unfiltered lower-level SFT dataset.

Theorem 3 suggests that by removing *useless* samples, the optimal model given by both **BDS** and **BMO** achieves lower empirical validation loss. The proof of Theorem 3 is deferred to Appendix C.3.

Theorem 3 quantifies the fine-tuning performance of the two formulations compared with the direct mixing baseline. Building upon Theorem 3, we can also quantify their generalization performance over an evaluation dataset whose underlying distribution is close to that of the validation dataset. This is the case for safety-aware LLM fine-tuning, where the validation and evaluation datasets are drawn from similar safety-focused distributions. Define the population losses for the validation and evaluation dataset as

$$\begin{aligned}\mathcal{L}_{\text{val}}(\theta) &:= \mathbb{E}_{(x,y) \sim p_{\text{val}}} [\mathcal{L}_0(\theta; x, y)] \\ \mathcal{L}_{\text{eval}}(\theta) &:= \mathbb{E}_{(x,y) \sim p_{\text{eval}}} [\mathcal{L}_0(\theta; x, y)]\end{aligned}$$

where p_{val} and p_{eval} denote the underlying validation and evaluation distributions, and \mathcal{L}_0 denotes the SFT loss when these datasets contain SFT samples, or an RL rule-based loss (e.g., direct preference optimization loss) when they contain preferred and unpreferred data. Then empirical validation loss $\mathcal{L}_0(\theta)$ is the i.i.d. realization of $\mathcal{L}_{\text{val}}(\theta)$ and we can similarly define the empirical evaluation loss as $\hat{\mathcal{L}}_{\text{eval}}(\theta) = \frac{1}{N^{\dagger}} \sum_{i=1}^{N^{\dagger}} \mathcal{L}_{\text{SFT}}(\theta; \bar{x}^i, \bar{y}^i)$.

The following theorem shows that when the number of validation and evaluation data samples is relatively large, and the population evaluation distribution is close to the validation distribution, the empirical evaluation losses given by the **BDS** and **BMO** will be strictly smaller than those of the direct mixing method.

Theorem 4. Suppose that the conditions in Theorem 2 hold, and let us denote the optimal model given by **BDS** and **BMO** as θ^* . Suppose that the Kullback–Leibler (KL) divergence $\text{KL}(p_{\text{eval}} || p_{\text{val}}) \leq \epsilon$ and per-sample losses $0 \leq \mathcal{L}_0(\theta; x, y) \leq B$. Then there exists $\bar{N} > 0$ and $\delta' > 0$ such that, for any $\delta \leq \delta'$, $N, N^{\dagger} \geq \bar{N}$, and mixing parameter $0 < \rho \leq 1$, we have in high probability,

$$\hat{\mathcal{L}}_{\text{eval}}(\theta^*) < \min_{\tilde{\theta} \in \mathcal{S}_{\text{mix}}} \hat{\mathcal{L}}_{\text{eval}}(\tilde{\theta})$$

where $\mathcal{S}_{\text{mix}} := \arg \min_{\theta'} \frac{\rho}{N} \sum_{i=1}^N \mathcal{L}_{\text{SFT}}(\theta'; x^i, y^i) + (1 - \rho) \mathcal{L}_0(\theta')$ denotes the set of optimal models using the weighted mixture of upper-level and lower-level data. Moreover, when $\rho = 1$, \mathcal{S}_{mix} denotes the set of optimal models using the full unfiltered lower-level SFT dataset.

Theorem 4 shows that the empirical evaluation loss for the obtained model trained via **BDS** and **BMO** using empirical validation dataset is still guaranteed to be lower than that of direct mixing method, if the population validation and evaluation distributions are close to each other, and the number of samples are large. The proof

is given in Appendix C.4.

3.3 Proof sketch and core idea

In this section, we provide the proof sketch of our main theorems in Section 3 and distill the core ideas, which are broadly reusable for establishing theoretical foundations of LLM beyond our setting.

Let $\mathcal{S}(\omega) = \arg \min_{\theta'} \frac{1}{N} \sum_{i=1}^N \sigma_i(\omega) \mathcal{L}_{\text{SFT}}(\theta'; x^i, y^i)$ denote the lower-level optimal set for θ and $\mathcal{L}(\theta) = (\mathcal{L}_{\text{SFT}}(\theta; x^1, y^1), \dots, \mathcal{L}_{\text{SFT}}(\theta; x^N, y^N))$ denote the vector objective of individual SFT loss at each data sample. The main insight to prove Theorem 1 is observing

$$\text{WP}(\mathcal{L}) = \bigcup_{\omega \in \mathbb{R}^N} \mathcal{S}(\omega)$$

which is guaranteed by the convexity of SFT loss with respect to the embedding of the backbone model $\phi_\theta(x, y)$ and the image space of the backbone model $\text{Im}(\phi_\theta(x, y))$ suffices to characterize $\mathcal{S}(\omega)$ and $\text{WP}(\mathcal{L})$.

With Theorem 1, if the optimal **BDS** weight for i -th sample is $\sigma(w_i^*) = 0$, then it suggests that optimizing a validation over the Pareto front is unchanged by removing i -th sample. We denote $\mathcal{L}^{\setminus i}$ as the vector objective removing i -th data objective.

Lemma 5. Under same conditions in Theorem 2, we have $\text{WP}(\mathcal{L}) \setminus \text{WP}(\mathcal{L}^{\setminus i})$ equals to the individual minimizer set of $\mathcal{L}_{\text{SFT}}(\theta; x^i, y^i)$.

This lemma indicates that if i -th data sample is *useless*, then validation loss is positive for all individual minimizer of i -th sample so that the validation loss achieves minimum purely on $\text{WP}(\mathcal{L}^{\setminus i})$, which is the case of 2nd sample in Figure 2 and will be eliminated by **BDS** to avoid conflict with validation minimizers. Conversely, if $\sigma(w_i^*) > 0$, then validation loss is positive for some individual minimizer of i -th sample, which is the case of 3rd sample in Figure 2 and will be kept by **BDS** to enlarge the joint region of the minimizer set of selected data samples with the validation minimizers. The complete proof for Theorem 2 is deferred to Appendix C.2.

Since we only select *useful* data samples which share minimizers with the validation loss, **BDS** is able to achieve a smaller validation loss in Theorem 2.

Due to the space limit, we highlight other core ideas in Appendix A. Here we give a summary of the key steps and relations of three theorems in Figure 3.

4 From Offline Data Selection to Online Self-refining Generation

In the previous section, we focused on selecting offline SFT data. However, it remains unclear how to (i) recursively generate responses that are aligned with val-

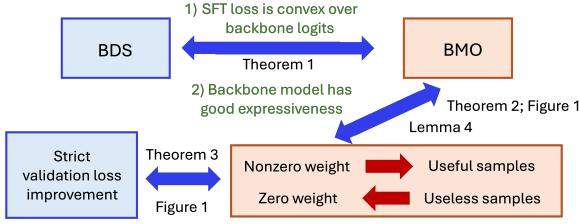


Figure 3: An overview of key steps for establishing the theorems and the relations of each theorem.

idation data and (ii) improve the model by discarding misleading questions based on the model’s current responses. In this section, we generalize our approach to the online SFT setting via self-refining generation.

4.1 Self-refining generation

We mask the offline responses for a subset of questions in the low-quality SFT dataset, replace them with online responses generated under the current policy, and evaluate the resulting question-response pairs on the high-quality validation dataset. In this way, the responses for the masked questions are kept updating, which we refer to as online self-refining generations.

Following the same setting as BDS, assume we are given the masked question set $\mathcal{I}_M \subset [N]$ with $|\mathcal{I}_M| = N_M$. Letting $y_s^{i,g} \sim \pi_\theta(y | x^i)$, $g \in [G]$ be a set of generated responses for masked question $i \in \mathcal{I}_M$, self-refining generation replaces the previous offline responses y_i with

$$\frac{1}{GN_M} \sum_{i \in \mathcal{I}_M} \sum_{g=1}^G \sigma_i(\omega) \mathcal{L}_{\text{SFT}}(\theta'; x^i, y_s^{i,g}) \quad (6)$$

in BDS formulation.

The benefits of bilevel online self-refining generation are twofold: (1) rather than discarding questions based on potentially suboptimal offline answers, it allows exploration for self-tuning responses, which provides more reliable question-level validation scores; and (2) the improved online answers and question-level scores, in turn, strengthen the model fine-tuning.

4.2 Algorithm design

We can generalize the memory-efficient offline penalty-based stochastic gradient descent method in [56, 58, 73] to the online self-refining generation setting. The key idea is that with Assumption 1, online self-refining generation is equivalent to its penalty reformulation

$$\begin{aligned} \min_{\omega \in \mathbb{R}^N, \theta'} \mathcal{L}_0(\theta') + \frac{\gamma_k}{N - N_M} \sum_{i \notin \mathcal{I}_M} \sigma_i(\omega) \mathcal{L}_{\text{SFT}}(\theta'; x^i, y^i) \\ + \frac{\gamma_k}{GN_M} \sum_{i \in \mathcal{I}_M} \sum_{g=1}^G \sigma_i(\omega) \mathcal{L}_{\text{SFT}}(\theta'; x^i, y_s^{i,g}) \end{aligned} \quad (7)$$

where γ_k is an enlarging penalty constant and when $\gamma_k \rightarrow \infty$, the solution of the penalty problem recovers

Algorithm 1 Bilevel online self-tuning generation

```

1: validation dataset  $\mathcal{D}$  and low-quality SFT dataset  $\mathcal{D}_{\text{SFT}}^-$ . Initial LLM parameter  $\theta_0$  and data selector parameter  $\omega$ . Step sizes  $\alpha_k, \beta_k$ , penalty strength  $\gamma_k$ , and generation frequency  $K_{\text{gen}}$ .
2: Generated masked question index set  $\mathcal{I}_M \subset [N]$ 
3: for  $k = 1$  to  $K$  do
4:   Sample question-answer data pair  $(\tilde{x}^{j_k}, \tilde{y}^{j_k}) \sim \mathcal{D}$  and  $(x^{i_k}, y^{i_k}) \sim \mathcal{D}_{\text{SFT}}^-$ .
5:   if  $k \bmod K_{\text{gen}} = 0$  then
6:     Generate  $G$  responses  $y_s^{i,g}$  from the current  $\pi_{\theta^k}$  for each masked question  $i \in \mathcal{I}_M$ 
7:     Update  $\pi_{\text{old}} \leftarrow \pi_{\theta^k}$  and  $y_{\text{old}}^{i,g} \leftarrow y_s^{i,g}$ 
8:   end if
9:   if  $i_k \in \mathcal{I}_M$  then
10:    Calculate importance ratio  $r^g = \frac{\pi_\theta(x^i, y_s^{i,g})}{\pi_{\text{old}}(x^i, y_s^{i,g})}$ 
11:    Calculate per-response gradient  $\nabla_k^g$  via (9)
12:    Average gradient using importance ratio
13:     $\nabla_\theta^k = \frac{1}{G} \sum_{g=1}^G r^g \nabla_k^g$ 
14:   else
15:     Use offline gradient  $\nabla_\theta^k = \nabla \mathcal{L}_{\text{SFT}}(\theta^k; x^{i_k}, y^{i_k})$ 
16:   end if
17:   Update  $\theta^{k+1}$  via (10a) and update  $\omega^{k+1}$  via (10b)
17: end for

```

that for the online self-refining generation problem.

However, self-refining generations in (7) lead to an inefficient algorithm because we need to generate on-policy responses at each iteration. Instead, leveraging importance sampling (IS) [66], we can estimate the expected gradient on the current policy using samples generated by the old policy because $\mathbb{E}_{\pi_\theta}[A] = \mathbb{E}_{\pi_{\text{old}}} \left[\frac{\pi_\theta}{\pi_{\text{old}}} A \right]$, where A is any quantity such as gradient, π_{old} is the old policy updated in a slower fashion, and $\frac{\pi_\theta}{\pi_{\text{old}}}$ is the importance ratio which compensates the delayed effect of using old samples.

Therefore, at each iteration k , we can randomly sample $(\tilde{x}^{j_k}, \tilde{y}^{j_k})$ and (x^{i_k}, y^{i_k}) from validation dataset \mathcal{D} and low-quality SFT dataset $\mathcal{D}_{\text{SFT}}^-$, respectively. If $i_k \in \mathcal{I}_M$, the gradient on the low-quality SFT dataset should be calculated using generated responses; otherwise, we are using the offline response. Let ∇^k be the gradient estimator of a low-quality SFT dataset, then

$$\nabla_\theta^k = \begin{cases} \nabla_\theta \mathcal{L}_{\text{SFT}}(\theta^k; x^{i_k}, y^{i_k}), & \text{if } i_k \notin \mathcal{I}_M, \\ \frac{1}{G} \sum_{g=1}^G r^g \nabla_k^g, & \text{otherwise.} \end{cases} \quad (8)$$

where $r^g = \frac{\pi_\theta(x^i, y_s^{i,g})}{\pi_{\text{old}}(x^i, y_s^{i,g})}$ is the importance ratio and ∇_k^g is the gradient for g -th response and i_k -th question

$$\nabla_k^g = \nabla_\theta \mathcal{L}_{\text{SFT}}(\theta; x^{i_k}, y_{\text{old}}^{i_k, g}). \quad (9)$$

After obtaining the gradient estimator for the low-quality SFT dataset, we reweight it with the upper-level

gradient estimator to update θ as

$$\theta^{k+1} = \theta^k - \beta_k \left(\nabla \mathcal{L}_0(\theta^k; \tilde{x}^{j_k}, \tilde{y}^{j_k}) + \gamma_k \sigma_{i_k}(\omega) \nabla_\theta^k \right) \quad (10a)$$

and update ω via

$$\omega^{k+1} = \omega^k - \alpha_k \gamma_k \nabla \sigma_{i_k}(\omega^k) C_\omega^k \quad (10b)$$

with the coefficient either the SFT loss on the offline data or the generated response

$$C_\omega^k = \begin{cases} \mathcal{L}_{\text{SFT}}(\theta^{k+1}; x^{i_k}, y^{i_k}), & \text{if } i_k \notin \mathcal{I}_k, \\ \frac{1}{G} \sum_{g=1}^G \mathcal{L}_{\text{SFT}}(\theta^{k+1}; x^{i_k}, y_{\text{old}}^{i,g}), & \text{otherwise.} \end{cases} \quad (10c)$$

Our complete bilevel online self-tuning algorithm is summarized in Algorithm 1.

4.3 Importance ratio as response weight

Interestingly, we find that the importance ratio for each generated response can be interpreted as the response weight with respect to the validation loss because it is proportional to the implicit weight given by BMO.

Lemma 6 (Implicit response weight given by BMO). *Under Assumption 1, the implicit weight of g-th response for i-th question assigned by BMO is*

$$\lambda_{i,g} = \frac{\exp(-\mathcal{L}_{\text{SFT}}(\theta; x^i, y_{\text{old}}^{i,g}))}{\sum_{g'=1}^G \exp(-\mathcal{L}_{\text{SFT}}(\theta; x^i, y_{\text{old}}^{i,g'}))}.$$

The proof of Lemma 6 can be found in Appendix D.

Since $\mathcal{L}_{\text{SFT}}(\theta; x^m, y_{\text{old}}^{i,g}) = -\log \pi_\theta(x^m, y_{\text{old}}^{i,g})$ and the denominator of $\lambda_{i,g}$ is same to each g-th response for i-th question, we have the importance ratio of each response

$$r^g = \frac{\pi_\theta(x^i, y_{\text{old}}^{i,g})}{\pi_{\text{old}}(x^i, y_{\text{old}}^{i,g})} \propto \pi_\theta(x^i, y_{\text{old}}^{i,g}) \propto \lambda_{i,g}$$

where we treat $\pi_{\text{old}}(x^i, y_{\text{old}}^{i,g})$ as the scaling factor. Note that from the old model π_{old} to the current model π_θ , we jointly optimize on both validation and selected lower-level data. Therefore, if the model's prediction on g-th response is higher on the current model π_θ than the old model π_{old} , it suggests that g-th response is more aligned with the validation data and we assign a higher response weight to it.

In summary, the data weight for g-th response with i-th question given by Algorithm 1 is the multiplication of question weight $\sigma_i(\omega)$ learned by BDS and the response weight r^g given by BMO.

5 Experimental Verifications

In this section, we verify our analysis and test our algorithm in two LLM post-training tasks: 1) quality enhancement, and 2) safety-aware fine-tuning task. We consider two base LLM models: PYTHIA-1B [4] and LLAMA-3-8B-INSTRUCT [21] model. We compare our online sampling Algorithm 1 with two baselines:

B1) Direct data mixing approach uses the weighted sum of upper-level and lower-level objectives $(1 - \rho)\mathcal{L}_0(\theta) + \rho\mathcal{L}_{\text{SFT}}(\theta; x^i, y^i)$ as objective function. When $\rho = 1$, it means SFT on the lower-level dataset only.

B2) Vanilla gradient-based BDS offline selection approach to BSG without online sampling by applying [58].

All experiments are conducted on one NVIDIA H100 GPU with 96 GB memory, and results are averaged over 4 runs. Our code is adapted from the bilevel LLM post-training library <https://github.com/Post-LLM/BIPOST> and experiment details are referred to Appendix F. We evaluate the performance on unseen data samples split from the validation dataset before fine-tuning, which we refer to as the evaluation loss. Additional evaluations on general question answering (QA) dataset and win-rate comparisons using ALPACAEVAL [35] are provided in Tables 5, 6, and Table 10 in Appendix F.

5.1 Quality enhancement tuning

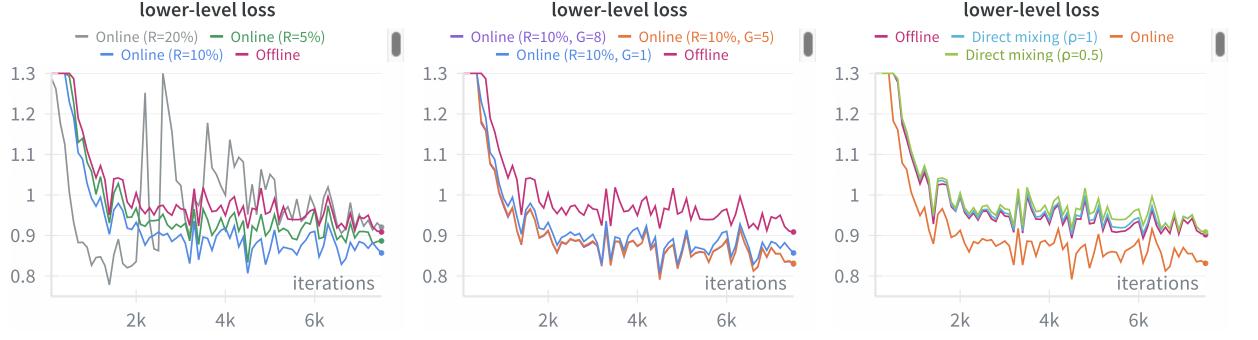
In this task, we use OPENORCA dataset in the upper-level, which has higher per-sample quality for complex, chain-of-thought style instructions [44, 50], and utilize ALPACA-CLEANED dataset in the lower-level, which is a small and tidy instruction-following dataset [71].

Method	PYTHIA-1B↓	LLAMA-8B↓
Direct mixing ($\rho = 1$)	1.56 ± 0.008	0.92 ± 0.012
Direct mixing ($\rho = 0.5$)	1.41 ± 0.011	0.84 ± 0.008
Offline selection	1.38 ± 0.005	0.80 ± 0.007
Online ($R = 5\%$, $G = 1$)	1.37 ± 0.005	0.80 ± 0.004
Online ($R = 10\%$, $G = 1$)	1.34 ± 0.003	0.78 ± 0.006
Online ($R = 10\%$, $G = 5$)	1.32 ± 0.004	0.76 ± 0.003

Table 1: Evaluation loss (upper-level) on OPENORCA dataset fine-tuned on PYTHIA-1B and LLAMA-8B model. **Bold** indicates the best result (lower is better). Online sample ratio is defined as $R = \frac{N_M}{N}$.

Enhancing lower-level response quality. We report the evaluation loss of our algorithm with different online sampling ratios $R = \frac{N_M}{N}$ and the number of online samples G on the lower-level OPENORCA dataset in Figure 4. We find that adding a moderate fraction of online samples (5%, 10%) to the offline dataset boosts lower-level performance, but a larger share (20%) hurts performance, likely because it adds early training with noisy signals. Generating multiple candidates per question improves response quality on the lower-level dataset, but there is no clear improvement beyond $G = 5$. For most of the hyperparameters, online self-tuning consistently outperforms direct mixing and the offline selection approach, which demonstrates its stability.

Guiding generations towards validation data. The test losses (upper-level) are shown in Table 1,



(a) different online ratio with $G = 1$ (b) different online sample size (c) comparison with baselines

Figure 4: Ablation study of our algorithm (online) and comparisons with other baselines. Fine-tuning loss on ALPACA-CLEANED dataset (lower-level) finetuned with LLAMA-3-8B-INSTRUCT model on validation tuning task. $R = N_M/N$ denotes the online sample ratio, G is the number of responses generated per question, and ρ is the mixing ratio of upper-level and lower-level datasets for the direct mixing approach.

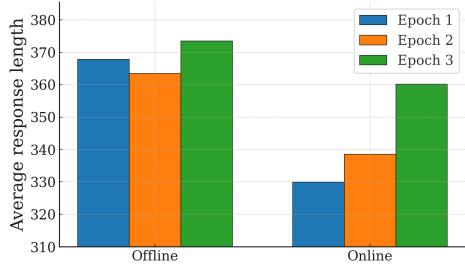


Figure 5: Average response length of top 10% questions ranked by learned data weights via offline selection and online self-refining approach. Self-refining approach tends to learn from simple to hard questions.

which suggests that the lower-level performance gain of the online self-refining strategy does not come at the expense of the upper-level performance or even slightly improve it as well. This may be due to the data weights assigned to generated responses, which encourage the model to produce appropriate outputs aligned with the validation dataset. Some example responses for the validation OPENORCA dataset are shown in Table 7 in Appendix F.2, which suggests the output produced by the online self-refining method has conciser reasoning and more accurate responses. The concision comes from the help of a tidier lower-level dataset, which in turn benefits the upper-level dataset training when questions are reweighted appropriately with a validation score.

Learning from simple to hard questions. To further compare the learning behavior of offline strategy and online self-refining, we analyze the top 10% and bottom 10% questions at each epoch given by them. According to Table 8 in Appendix F.2, the online self-refining tends to focus on easy questions at the first epoch and gradually progress from simple to hard questions. The knowledge gained from simpler questions builds a solid backbone model for subsequent adapta-

tion to the hard questions [39, 46, 86], which is also evidenced by the selected questions via online self-refining in Table 8. To get more sense of the statistics of the top-ranked data, we also report response length as a partial measure for question difficulty in Figure 5 (longer responses indicating longer reasoning and thus, harder questions; see evidence in Table 9). Notably, the online self-refining algorithm learns from shorter-response questions first, while the offline selection continues to tackle longer-response questions throughout all epochs.

Comparable runtime. We report the runtime of different algorithms in Table 2. With the number of online samples per question $G = 1$, the computational overhead of online sampling is not significant, especially for the larger LLAMA-3-8B-INSTRUCT model. However, generating $G = 5$ samples per question introduces additional $1 \times$ computational overhead while only leading to slight improvement on the model performance (see Figure 4 (b)). Therefore, $G = 1$ is an ideal choice.

Method	PYTHIA-1B	LLAMA-8B
Direct mixing	0.24	10.53
Offline selection	0.31	11.75
Online ($R = 5\%$, $G = 1$)	0.43	13.28
Online ($R = 10\%$, $G = 1$)	0.47	13.78
Online ($R = 10\%$, $G = 5$)	0.86	30.46

Table 2: Average running time (measured in hours) using OPENORCA dataset on PYTHIA-1B and LLAMA-8B model. Online sample ratio is defined as $R = \frac{N_M}{N}$.

5.2 Safety-aware fine-tuning

To test our algorithm on a safety-aware LLM fine-tuning task, we follow the setup from [58]. We use the BLUEORCA and REDORCA datasets, where the BLUEORCA dataset contains only safe data from the SLIMORCA, but REDORCA dataset is mixed with 22k potentially unsafe instructions and responses picked

from the ANTHROPIC RED-TEAMING dataset [16].

Giving questions another chance. With online self-refining generation, the model can guide the generated response by the upper-level safe data, even if the initial responses are harmful. Instead of throwing out the question-answer pair directly, we improve the response, but still keep the diverse information from the questions. The evaluation losses on the safe and unsafe data for different methods are shown in Table 3. While both offline selection and online self-refining improve the validation performance, online self-refining generation further enhances the fine-tuning performance on the lower-level unsafe dataset, which suggests that we also improve the quality of lower-level dataset.

Dynamically updating masked questions. So far, we sample the masked question index set \mathcal{I}_M once and keep it fixed for online responses. However, we can further improve lower-level performance by generating online responses only for questions currently flagged as unsafe, i.e., dynamically construct \mathcal{I}_M informed by the data weight. We call this a dynamic online strategy. In Table 3, we test the performance of the dynamic online strategy when \mathcal{I}_M at step k is chosen as

$\mathcal{I}_M^k = \{i \mid \sigma_i(\omega_t) \text{ is ranked bottom } R \text{ among all data}\}$ where $R = \frac{N_M}{N}$ is the online sample ratio. Table 3 shows that the dynamic strategy significantly improves the performance on the unsafe dataset REDORCA.

6 Conclusions and Limitations

We study the data selection problem BDS and the validation adaptation problem formulated as BMO in LLM fine-tuning. We prove that they are equivalent optimization problem that assign a validation weight to the low-quality SFT dataset either explicitly or implicitly. Moreover, we prove the effectiveness of BDS for selecting useful data samples through the lens of BMO, and thus both of them improve over the naive data mixing strategy. To further improve the model performance, we incorporate online generations to refine the offline responses using the current policy. We assign implicit validation weights given by BMO to generated responses to further enhance the validation alignment. Experiments on validation tuning and safety-aware fine-tuning validate the effectiveness of the proposed algorithm. Our study is currently limited to SFT data selection in the lower-level. Extending our approach to RLHF data, token-level selection, and large-scale multi-domain corpora is a promising direction for future work.

Acknowledgement

The work was supported by the National Science Foundation Projects 2401297, 2532349 and 2532653, and by

Method	BLUEORCA \downarrow	REDORCA \downarrow
Direct mixing ($\rho = 0.5$)	0.88 ± 0.010	1.31 ± 0.012
Direct mixing ($\rho = 1$)	0.94 ± 0.007	1.27 ± 0.008
Offline selection	0.85 ± 0.006	1.25 ± 0.005
Online ($R = 5\%$, $G = 1$)	0.84 ± 0.004	1.22 ± 0.003
Online ($R = 10\%$, $G = 1$)	0.83 ± 0.005	1.20 ± 0.005
Online ($R = 10\%$, $G = 2$)	0.83 ± 0.003	1.17 ± 0.004
Online dynamic	0.82 ± 0.004	1.02 ± 0.006

Table 3: Evaluation loss on BLUEORCA dataset and selected REDORCA dataset fine-tuned with LLAMA-8B model. **Bold** indicates the best result (lower is better). Online sample ratio is defined as $R = \frac{N_M}{N}$. For online dynamic strategy, we choose $R = 10\%$ and $G = 1$.

the Cisco Research Award.

References

- [1] Zeyuan Allen-Zhu, Yuanzhi Li, and Zhao Song. A convergence theory for deep learning via over-parameterization. In *Proc. International Conference on Machine Learning*, pages 242–252, 2019.
- [2] Michael Arbel and Julien Mairal. Amortized implicit differentiation for stochastic bilevel optimization. In *Proc. International Conference on Learning Representations*, virtual, 2022.
- [3] Omar Bennouna, Amine Bennouna, Saurabh Amin, and Asuman Ozdaglar. What data enables optimal decisions? an exact characterization for linear optimization. *arXiv preprint arXiv:2505.21692*, 2025.
- [4] Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O’Brien, Eric Hallahan, Mohammad Aflah Khan, Shivan-shu Purohit, USVSN Sai Prashanth, Edward Raff, et al. Pythia: A suite for analyzing large language models across training and scaling. In *Proc. International Conference on Machine Learning*, 2023.
- [5] Stephen Boyd, Stephen P Boyd, and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.
- [6] Lisha Chen, Quan Xiao, Ellen Hidemi Fukuda, Xinyi Chen, Kun Yuan, and Tianyi Chen. Efficient first-order optimization on the pareto set for multi-objective learning under preference guidance. In *Proc. International Conference on Machine Learning*, Vancouver, BC, Canada, 2025.
- [7] Tianyi Chen, Yuejiao Sun, and Wotao Yin. Closing the gap: Tighter analysis of alternating stochastic gradient methods for bilevel problems. In *Proc.*

- Advances in Neural Information Processing Systems*, virtual, 2021.
- [8] Hyeong Kyu Choi, Xuefeng Du, and Yixuan Li. Safety-aware fine-tuning of large language models. In *Neurips Safe Generative AI Workshop*, 2024.
- [9] Mathieu Dagréou, Pierre Ablin, Samuel Vaïter, and Thomas Moreau. A framework for bilevel optimization that enables stochastic and global variance reduction algorithms. In *Proc. Advances in Neural Information Processing Systems*, New Orleans, LA, 2022.
- [10] Kalyanmoy Deb and Ankur Sinha. Solving bilevel multi-objective optimization problems using evolutionary algorithms. In *International conference on evolutionary multi-criterion optimization*, pages 110–124. Springer, 2009.
- [11] S. Dempe and P. Mehlitz. Semivectorial bilevel programming versus scalar bilevel programming. *Optimization*, 69(4):657–679, 2020.
- [12] Matthias Ehrgott. *Multicriteria optimization*. Springer, 2005.
- [13] Heshan Fernando, Han Shen, Parikshit Ram, Yi Zhou, Horst Samulowitz, Nathalie Baracaldo, and Tianyi Chen. Mitigating forgetting in llm supervised fine-tuning and preference learning. *arXiv preprint arXiv:2410.15483*, 2024.
- [14] Luca Franceschi, Michele Donini, Paolo Frasconi, and Massimiliano Pontil. Forward and reverse gradient-based hyperparameter optimization. In *Proc. International Conference on Machine Learning*, Sydney, Australia, 2017.
- [15] Luca Franceschi, Paolo Frasconi, Saverio Salzo, Riccardo Grazzi, and Massimiliano Pontil. Bilevel programming for hyperparameter optimization and meta-learning. In *Proc. International Conference on Machine Learning*, Stockholm, Sweden, 2018.
- [16] Deep Ganguli, Liane Lovitt, Jackson Kernion, Amanda Askell, Yuntao Bai, Saurav Kadavath, Ben Mann, Ethan Perez, Nicholas Schiefer, Kamal Ndousse, et al. Red teaming language models to reduce harms: Methods, scaling behaviors, and lessons learned. *arXiv preprint arXiv:2209.07858*, 2022.
- [17] Bolin Gao and Lacra Pavel. On the properties of the softmax function with application in game theory and reinforcement learning. *arXiv preprint arXiv:1704.00805*, 2017.
- [18] Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lin-tang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. The language model evaluation harness, 07 2024. URL <https://zenodo.org/records/12608602>.
- [19] Hossein Shokri Ghadikolaei, Hadi Ghauch, Carlo Fischione, and Mikael Skoglund. Learning and data selection in big datasets. In *Proc. International Conference on Machine Learning*, 2019.
- [20] Saeed Ghadimi and Mengdi Wang. Approximation methods for bilevel programming. *arXiv preprint arXiv:1802.02246*, 2018.
- [21] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.
- [22] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.
- [23] Riccardo Grazzi, Luca Franceschi, Massimiliano Pontil, and Saverio Salzo. On the iteration complexity of hypergradient computation. In *Proc. International Conference on Machine Learning*, virtual, 2020.
- [24] Caglar Gulcehre, Tom Le Paine, Srivatsan Srinivasan, Ksenia Konyushkova, Lotte Weerts, Abhishek Sharma, Aditya Siddhant, Alex Ahern, Miaosen Wang, Chenjie Gu, et al. Reinforced self-training (rest) for language modeling. *arXiv preprint arXiv:2308.08998*, 2023.
- [25] Mingyi Hong, Hoi-To Wai, Zhaoran Wang, and Zhuoran Yang. A two-timescale stochastic algorithm framework for bilevel optimization: Complexity analysis and application to actor-critic. *SIAM Journal on Optimization*, 33(1):147–180, 2023.
- [26] Joshua Hang Sai Ip, Ankush Chakrabarty, Ali Mesbah, and Diego Romeres. User preference meets pareto-optimality in multi-objective bayesian optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 20246–20254, 2025.
- [27] Kaiyi Ji, Junjie Yang, and Yingbin Liang. Bilevel optimization: Convergence analysis and enhanced design. In *Proc. International Conference on Machine Learning*, virtual, 2021.

- [28] Ziwei Ji and Matus Telgarsky. Directional convergence and alignment in deep learning. In *Proc. Advances in Neural Information Processing Systems*, virtual, 2020.
- [29] Feiyang Kang, Hoang Anh Just, Yifan Sun, Hrimanshu Jahagirdar, Yuanzhi Zhang, Rongxing Du, Anit Kumar Sahu, and Ruoxi Jia. Get more for less: Principled data selection for warming up fine-tuning in llms. In *Proc. International Conference on Learning Representations*, Vienna, Austria, 2024.
- [30] Prashant Khanduri, Siliang Zeng, Mingyi Hong, Hoi-To Wai, Zhaoran Wang, and Zhuoran Yang. A near-optimal algorithm for stochastic bilevel optimization via double-momentum. In *Proc. Advances in Neural Information Processing Systems*, virtual, 2021.
- [31] Diederik P Kingma. Adam: A method for stochastic optimization. In *Proc. International Conference on Learning Representations*, 2015.
- [32] Jeongyeol Kwon, Dohyun Kwon, Stephen Wright, and Robert D Nowak. A fully first-order method for stochastic bilevel optimization. In *Proc. International Conference on Machine Learning*, Honolulu, HI, 2023.
- [33] Jeongyeol Kwon, Dohyun Kwon, Steve Wright, and Robert Nowak. On penalty methods for non-convex bilevel optimization and first-order stochastic approximation. In *Proc. International Conference on Learning Representations*, Vienna, Austria, 2024.
- [34] Junyi Li, Bin Gu, and Heng Huang. A fully single loop algorithm for bilevel optimization without hessian inverse. In *Proc. Association for the Advancement of Artificial Intelligence*, virtual, 2022.
- [35] Xuechen Li, Tianyi Zhang, Yann Dubois, Rohan Taori, Ishaan Gulrajani, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Alpacaeval: An automatic evaluator of instruction-following models. https://github.com/tatsu-lab/alpaca_eval, 5 2023.
- [36] Yuan Li, Zhengzhong Liu, and Eric Xing. Data mixing optimization for supervised fine-tuning of large language models. In *Proc. International Conference on Machine Learning*, Vancouver, BC, Canada, 2025.
- [37] Ziniu Li, Congliang Chen, Tian Xu, Zeyu Qin, Jiancong Xiao, Zhi-Quan Luo, and Ruoyu Sun. Preserving diversity in supervised fine-tuning of large language models. In *Proc. International Conference on Learning Representations*, Singapore, Singapore, 2025.
- [38] Xinyu Lin, Wenjie Wang, Yongqi Li, Shuo Yang, Fuli Feng, Yinwei Wei, and Tat-Seng Chua. Data-efficient fine-tuning for llm-based recommendation. In *Proceedings of the 47th international ACM SIGIR conference on research and development in information retrieval*, pages 365–374, 2024.
- [39] Yong Lin, Shange Tang, Bohan Lyu, Ziran Yang, Jui-Hui Chung, Haoyu Zhao, Lai Jiang, Yihan Geng, Jiawei Ge, Jingruo Sun, et al. Goedel-prover-v2: Scaling formal theorem proving with scaffolded data synthesis and self-correction. *arXiv preprint arXiv:2508.03613*, 2025.
- [40] Qian Liu, Xiaosen Zheng, Niklas Muennighoff, Guangtao Zeng, Longxu Dou, Tianyu Pang, Jing Jiang, and Min Lin. Regmix: Data mixture as regression for language model pre-training. In *Proc. International Conference on Learning Representations*, Singapore, Singapore, 2025.
- [41] Risheng Liu, Pan Mu, Xiaoming Yuan, Shangzhi Zeng, and Jin Zhang. A generic first-order algorithmic framework for bi-level programming beyond lower-level singleton. In *Proc. International Conference on Machine Learning*, virtual, 2020.
- [42] Risheng Liu, Yaohua Liu, Wei Yao, Shangzhi Zeng, and Jin Zhang. Averaged method of multipliers for bi-level optimization without lower-level strong convexity. In *Proc. International Conference on Machine Learning*, Honolulu, HI, 2023.
- [43] Wei Liu, Weihao Zeng, Keqing He, Yong Jiang, and Junxian He. What makes good data for alignment? a comprehensive study of automatic data selection in instruction tuning. In *Proc. International Conference on Learning Representations*, Vienna, Austria, 2024.
- [44] Shayne Longpre, Le Hou, Tu Vu, Albert Webson, Hyung Won Chung, Yi Tay, Denny Zhou, Quoc V Le, Barret Zoph, Jason Wei, et al. The flan collection: Designing data and methods for effective instruction tuning. In *Proc. International Conference on Machine Learning*, Honolulu, HI, 2023.
- [45] Zhaosong Lu and Sanyou Mei. First-order penalty methods for bilevel optimization. *SIAM Journal on Optimization*, 34(2):1937–1969, 2024.
- [46] Lu Ma, Hao Liang, Meiyi Qiang, Lexiang Tang, Xiaochen Ma, Zhen Hao Wong, Junbo Niu, Chengyu Shen, Runming He, Bin Cui, et al. Learning what reinforcement learning can't: Interleaved online fine-tuning for hardest questions. *arXiv preprint arXiv:2506.07527*, 2025.
- [47] Debabrata Mahapatra and Vaibhav Rajan. Multi-task learning with user preferences: Gradient descent with controlled ascent in pareto optimiza-

- tion. In *Proc. International Conference on Machine Learning*, virtual, 2020.
- [48] Jonathan Mamou, Hang Le, Miguel Del Rio, Cory Stephenson, Hanlin Tang, Yoon Kim, and Sueyeon Chung. Emergence of separable manifolds in deep language representations. In *Proc. International Conference on Machine Learning*, virtual, 2020.
- [49] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of machine learning*. MIT press, 2018.
- [50] Subhabrata Mukherjee, Arindam Mitra, Ganesh Jawahar, Sahaj Agarwal, Hamid Palangi, and Ahmed Awadallah. Orca: Progressive learning from complex explanation traces of gpt-4. *arXiv preprint arXiv:2306.02707*, 2023.
- [51] Jinlong Pang, Na Di, Zhaowei Zhu, Jiaheng Wei, Hao Cheng, Chen Qian, and Yang Liu. Token cleaning: Fine-grained data selection for llm supervised fine-tuning. In *Proc. International Conference on Machine Learning*, Vancouver, BC, Canada, 2025.
- [52] Fabian Pedregosa. Hyperparameter optimization with approximate gradient. In *Proc. International Conference on Machine Learning*, New York City, NY, 2016.
- [53] Yi Ren and Danica J Sutherland. Learning dynamics of LLM Finetuning. In *Proc. International Conference on Learning Representations*, Singapore, Singapore, 2025.
- [54] Yinuo Ren, Tesi Xiao, Michael Shavlovsky, Lexing Ying, and Holakou Rahamanian. Cos-dpo: Conditioned one-shot multi-objective fine-tuning framework. In *The 41st Conference on Uncertainty in Artificial Intelligence*.
- [55] R Tyrrell Rockafellar and Roger J-B Wets. *Variational analysis*, volume 317. Springer Science & Business Media, 2009.
- [56] Baturay Saglam, Paul Kassianik, Blaine Nelson, Sajana Weerawardhena, Yaron Singer, and Amin Karbasi. Large language models encode semantics in low-dimensional linear subspaces. *arXiv preprint arXiv:2507.09709*, 2025.
- [57] Benjamin Scellier and Yoshua Bengio. Equilibrium propagation: Bridging the gap between energy-based models and backpropagation. *Frontiers in computational neuroscience*, 11:24, 2017.
- [58] Han Shen, Pin-Yu Chen, Payel Das, and Tianyi Chen. Seal: Safety-enhanced aligned llm fine-tuning via bilevel data selection. In *Proc. International Conference on Learning Representations*, 2025.
- [59] Han Shen, Quan Xiao, and Tianyi Chen. On penalty-based bilevel gradient descent method. *Mathematical Programming*, pages 1–51, 2025.
- [60] Yuda Song, Gokul Swamy, Aarti Singh, J Bagnell, and Wen Sun. The importance of online data: Understanding preference fine-tuning via coverage. In *Proc. Advances in Neural Information Processing Systems*, Vancouver, BC, Canada, 2024.
- [61] Daniel Soudry, Elad Hoffer, Mor Shpigel Nacson, Suriya Gunasekar, and Nathan Srebro. The implicit bias of gradient descent on separable data. *Journal of Machine Learning Research*, 19(70):1–57, 2018.
- [62] Fahim Tajwar, Anikait Singh, Archit Sharma, Rafael Rafailov, Jeff Schneider, Tengyang Xie, Stefano Ermon, Chelsea Finn, and Aviral Kumar. Preference fine-tuning of llms should leverage sub-optimal, on-policy data. In *Proc. International Conference on Machine Learning*, Vienna, Austria, 2024.
- [63] Hiroki Tanabe, Ellen H. Fukuda, and Nobuo Yamashita. New merit functions for multiobjective optimization and their properties. *arXiv preprint arXiv:2010.09333*, 2022.
- [64] Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. Stanford alpaca: An instruction-following llama model, 2023.
- [65] Davoud Ataee Tarzanagh, Yingcong Li, Christos Thrampoulidis, and Samet Oymak. Transformers as support vector machines. *arXiv preprint arXiv:2308.16898*, 2023.
- [66] Surya T Tokdar and Robert E Kass. Importance sampling: a review. *Wiley Interdisciplinary Reviews: Computational Statistics*, 2(1):54–60, 2010.
- [67] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proc. Advances in Neural Information Processing Systems*, 2017.
- [68] Paul Vicol, Jonathan P Lorraine, Fabian Pedregosa, David Duvenaud, and Roger B Grosse. On implicit bias in overparameterized bilevel optimization. In *Proc. International Conference on Machine Learning*, Baltimore, MD, 2022.
- [69] Haoxiang Wang, Wei Xiong, Tengyang Xie, Han Zhao, and Tong Zhang. Interpretable preferences via multi-objective reward modeling and mixture-of-experts. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 10582–10592, 2024.

- [70] Weixuan Wang, Minghao Wu, Barry Haddow, and Alexandra Birch. Hbo: Hierarchical balancing optimization for fine-tuning large language models. *arXiv preprint arXiv:2505.12300*, 2025.
- [71] Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A Smith, Daniel Khashabi, and Hannaneh Hajishirzi. Self-instruct: Aligning language models with self-generated instructions. *arXiv preprint arXiv:2212.10560*, 2022.
- [72] Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. Finetuned language models are zero-shot learners. In *Proc. International Conference on Learning Representations*, 2022.
- [73] Peiyao Xiao, Chaosheng Dong, Shaofeng Zou, and Kaiyi Ji. Ldc-mtl: Balancing multi-task learning through scalable loss discrepancy control. *arXiv preprint arXiv:2502.08585*, 2025.
- [74] Quan Xiao and Tianyi Chen. Unlocking global optimality in bilevel optimization: A pilot study. *arXiv preprint arXiv:2408.16087*, 2024.
- [75] Quan Xiao, Songtao Lu, and Tianyi Chen. A generalized alternating method for bilevel optimization under the polyak-łojasiewicz condition. In *Proc. Advances in Neural Information Processing Systems*, New Orleans, LA, 2023.
- [76] Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. Self-training with noisy student improves imagenet classification. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10687–10698, 2020.
- [77] Sang Michael Xie, Hieu Pham, Xuanyi Dong, Nan Du, Hanxiao Liu, Yifeng Lu, Percy S Liang, Quoc V Le, Tengyu Ma, and Adams Wei Yu. Doremi: Optimizing data mixtures speeds up language model pretraining. In *Proc. Advances in Neural Information Processing Systems*, New Orleans, LA, 2023.
- [78] Sang Michael Xie, Shibani Santurkar, Tengyu Ma, and Percy S Liang. Data selection for language models via importance resampling. In *Proc. Advances in Neural Information Processing Systems*, New Orleans, LA, 2023.
- [79] Junjie Yang, Kaiyi Ji, and Yingbin Liang. Provably faster algorithms for bilevel optimization. In *Proc. Advances in Neural Information Processing Systems*, virtual, 2021.
- [80] Yang Yu, Kai Han, Hang Zhou, Yehui Tang, Kaiqi Huang, Yunhe Wang, and Dacheng Tao. Llm data selection and utilization via dynamic bi-level optimization. *arXiv preprint arXiv:2507.16178*, 2025.
- [81] Runtian Zhai, Chen Dan, J Zico Kolter, and Pradeep Kumar Ravikumar. Understanding why generalized reweighting does not improve over ERM. In *Proc. International Conference on Learning Representations*, Kigali, Rwanda, 2023.
- [82] Chi Zhang, Huaping Zhong, Kuan Zhang, Chengliang Chai, Rui Wang, Xinlin Zhuang, Tianyi Bai, Qiu Jiantao, Lei Cao, Ju Fan, et al. Harnessing diversity for important data selection in pretraining large language models. In *Proc. International Conference on Learning Representations*, Singapore, Singapore, 2025.
- [83] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. *arXiv preprint arXiv:1611.03530*, 2016.
- [84] Dylan Zhang, Qirun Dai, and Hao Peng. The best instruction-tuning data are those that fit. In *Proc. Advances in Neural Information Processing Systems*, San Diego, CA, 2025.
- [85] Shuai Zhang, Meng Weng, Sijia Liu, Pin-Yu Chen, and Jinjun Xiong. How unlabeled data improve generalization in self-training? a one-hidden-layer theoretical analysis. In *Proc. International Conference on Learning Representations*, virtual, 2022.
- [86] Wenhao Zhang, Yuexiang Xie, Yuchang Sun, Yanxi Chen, Guoyin Wang, Yaliang Li, Bolin Ding, and Jingren Zhou. On-policy rl meets off-policy experts: Harmonizing supervised fine-tuning and reinforcement learning via dynamic weighting. *arXiv preprint arXiv:2508.11408*, 2025.
- [87] Yihua Zhang, Prashant Khanduri, Ioannis Tsaknakis, Yuguang Yao, Mingyi Hong, and Sijia Liu. An introduction to bi-level optimization: Foundations and applications in signal processing and machine learning. *arXiv preprint arXiv:2308.00788*, 2023.
- [88] Yifan Zhong, Chengdong Ma, Xiaoyuan Zhang, Ziran Yang, Haojun Chen, Qingfu Zhang, Siyuan Qi, and Yaodong Yang. Panacea: Pareto alignment via preference adaptation for llms. In *Proc. Advances in Neural Information Processing Systems*, Vancouver, BC, Canada, 2024.
- [89] Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, et al. Lima: Less is more for alignment. In *Proc. Advances in Neural Information Processing Systems*, New Orleans, LA, 2023.

Supplementary Material

Contents

A Proof Roadmap and Core Idea	14
B Proof of Lemma 7	15
C Proof of theorems in Section 3	16
C.1 Proof of Theorem 1	16
C.2 Proof of Theorem 2 and Lemma 5	17
C.3 Proof of Theorem 3	18
C.4 Proof of Theorem 4	18
D Implicit weight assigned by BMO	19
E Connections between algorithm designs for BDS and BMO	21
F Additional experiments	22
F.1 Details of experimental setup	22
F.2 Quality comparisons of model outputs	23

A Proof Roadmap and Core Idea

In this section, we provide the proof sketch of our main theorems in Section 3 and distill the core ideas, which are broadly reusable for establishing theoretical foundations of LLM beyond our setting. An overview of proof sketch is highlighted in Figure 1.

The first building block of our main theories is the curvature property of SFT loss. With causal masking in attention [67], the backbone model $\phi_\theta(x, y)$ is unable to see the future tokens before predicting, even if we input the whole sequence [53]. Therefore, the SFT loss can be viewed as the token-level cross-entropy loss of a sequential multi-class classification problem, where the label is the next token in the response y . Due to the nonlinearity of $\phi_\theta(x, y)$, SFT loss might not be convex with respect to θ , but it is convex with respect to the backbone representation $z = \phi_\theta(x, y) \in \mathbb{R}^{V \times D}$.

Lemma 7. *Per-sample SFT loss in (1) is convex with respect to the backbone model $z = \phi_\theta(x, y) \in \mathbb{R}^{V \times D}$.*

The complete proof is deferred in Appendix B.

Remark 2. As far as we know, this is the first convexity results of SFT loss with respect to the backbone model $z = \phi_\theta(x, y)$. Although Ren and Sutherland [53] also consider the SFT loss with softmax parameterization and express it in terms of the backbone model, they do not analyze its curvature properties. Compared with [13], we focus on a general setting with token-level NLL loss with nonlinear backbone model, which is more aligned with practical LLM fine-tuning, i.e. the result in [13] is a special case of ours when $D = 1$ and $\phi_\theta(x, y)$ is linear in θ .

On the other hand, the classical result shows that BDS and BMO are equivalent when each lower-level objective is convex [11] because in this setting, traversing over all (nonnegative) linear combinations of the lower-level objectives recovers the entire Pareto set. Therefore, given SFT loss is convex over the backbone model, a natural question is whether the optimization parameter space of BDS and BMO can be converted from $\theta \in \mathbb{R}^h$ into backbone representation space $z \in \mathbb{R}^{V \times D}$. Since varying $\theta \in \mathbb{R}^h$ only moves z within the subspace $\text{Im}(\phi_\theta(x, y))$, the key for Theorem 1 is to prove the image space of the backbone model $\text{Im}(\phi_\theta(x, y))$ is enough to characterize the optimal model using weighted sum of loss in BDS and Pareto optimal solutions for BMO. These are possible under Assumption 1 because the existence of zero-loss shared model parameter θ for lower-level SFT dataset implies

that the zero-loss backbone is realized within $\text{Im}(\phi_\theta(x, y))$, which in turn determines the optimal backbone with respect to **BDS** and **BMO**. See details of proof for Theorem 1 in Appendix 3.

With Theorem 1, if the optimal **BDS** weight for i -th sample is $\sigma(w_i^*) = 0$, then it suggests that optimizing the validation loss over the Pareto front is unchanged by removing i -th sample. We characterize the difference of two weak Pareto optimal sets, which coincides with the individual minimizer sets of i -th data sample, and thereby indicates that if $\sigma(w_i^*) = 0$, then validation function is positive for all individual minimizer of i -th sample, which gives Theorem 2. The complete proof for Theorem 2 is deferred to Appendix C.2.

B Proof of Lemma 7

Due to the nonlinear backbone model $\phi_\theta(x, y)$, SFT loss might not be convex with respect to θ , which makes the optimization landscape complicated. Nevertheless, the building block of our theory lies in the convexity of the SFT loss with respect to the backbone representation $z = \phi_\theta(x, y) \in \mathbb{R}^{V \times D}$, which makes the SFT loss a composite convex function over θ .

Proof. According to (1)–(3), SFT loss takes the form of

$$\begin{aligned}\mathcal{L}_{\text{SFT}}(\theta; x, y) &= -\sum_{d=1}^D \mathbf{e}_{y_d}^\top \log \pi_\theta(y_d | x, y_{<d}) = -\sum_{d=1}^D \mathbf{e}_{y_d}^\top \log \sigma(\phi_\theta(x, y))_{[:,d]} \\ &= -\sum_{d=1}^D \mathbf{e}_{y_d}^\top \log \sigma(z)_{[:,d]} = -\sum_{d=1}^D \mathbf{e}_{y_d}^\top \log \sigma(z_{[:,d]}).\end{aligned}\quad (11)$$

The Jacobian of softmax function is well-known [17] and is given by

$$\nabla \sigma(z_{[:,d]}) = \text{diag}(\sigma(z_{[:,d]})) - \sigma(z_{[:,d]})\sigma(z_{[:,d]})^\top.$$

Therefore, the derivative of $\mathcal{L}_{\text{SFT}}(\theta; x, y)$ with respect to $z_{[:,d]}$ is given by

$$\begin{aligned}\frac{\partial \mathcal{L}_{\text{SFT}}(\theta; x, y)}{\partial z_{[:,d]}} &= \nabla \sigma(z_{[:,d]}) \frac{\partial \mathcal{L}_{\text{SFT}}(\theta; x, y)}{\partial \sigma(z_{[:,d]})} \\ &= -[\text{diag}(\sigma(z_{[:,d]})) - \sigma(z_{[:,d]})\sigma(z_{[:,d]})^\top] \left[\frac{(\mathbf{e}_{y_d})_1}{\sigma(z_{[:,d]})_1}, \dots, \frac{(\mathbf{e}_{y_d})_V}{\sigma(z_{[:,d]})_V} \right]^\top \\ &\stackrel{(a)}{=} \sigma(z_{[:,d]}) - \mathbf{e}_{y_d}\end{aligned}\quad (12)$$

where $(\mathbf{e}_{y_d})_i$ denotes the i -the element in the one-hot vector $\mathbf{e}_{y_d} \in \mathbb{R}^V$ and $\sigma(z_{[:,d]})_i$ denotes the i -the element in the softmax vector $\sigma(z_{[:,d]}) \in \mathbb{R}^V$, and (a) uses the fact that $\sum_i (\mathbf{e}_{y_d})_i = 1$. Moreover, the Jacobian of $\mathcal{L}_{\text{SFT}}(\theta; x, y)$ with respect to z is given by

$$\begin{aligned}\frac{\partial \mathcal{L}_{\text{SFT}}(\theta; x, y)}{\partial z} &= \left[\frac{\partial \mathcal{L}_{\text{SFT}}(\theta; x, y)}{\partial z_{[:,1]}}, \dots, \frac{\partial \mathcal{L}_{\text{SFT}}(\theta; x, y)}{\partial z_{[:,D]}} \right] \\ &= [\sigma(z_{[:,1]}) - \mathbf{e}_{y_1}, \dots, \sigma(z_{[:,D]}) - \mathbf{e}_{y_D}]\end{aligned}\quad (13)$$

On the other hand, the second-order derivative of $\mathcal{L}_{\text{SFT}}(\theta; x, y)$ with respect to $z_{[:,d]}$ is given by

$$\frac{\partial^2 \mathcal{L}_{\text{SFT}}(\theta; x, y)}{\partial^2 z_{[:,d]}} = \nabla \sigma(z_{[:,d]}), \quad \text{and} \quad \frac{\partial^2 \mathcal{L}_{\text{SFT}}(\theta; x, y)}{\partial z_{[:,d]} \partial z_{[:,\tilde{d}]}} = \mathbf{0}^{V \times D}, \quad \text{if } \tilde{d} \neq d.\quad (14)$$

where $\mathbf{0}^{V \times D} \in \mathbb{R}^{V \times D}$ is the zero matrix. The Hessian of $\mathcal{L}_{\text{SFT}}(\theta; x, y)$ over z is defined as the Hessian of the vectorized z , i.e. $\frac{\partial^2 \mathcal{L}_{\text{SFT}}(\theta; x, y)}{\partial z} \in \mathbb{R}^{Vd \times Vd}$. We define the vectorized z as

$$\text{vec}(z) = [z_{11}, z_{21}, \dots, z_{V1}, \dots, z_{1D}, z_{2D}, \dots, z_{VD}]\quad (15)$$

Let $i = (d_1 - 1)V + v_1$ and $j = (d_2 - 1)V + v_2$, then each element of the Hessian is

$$\left[\frac{\partial^2 \mathcal{L}_{\text{SFT}}(\theta; x, y)}{\partial z} \right]_{ij} = \left[\frac{\partial^2 \mathcal{L}_{\text{SFT}}(\theta; x, y)}{\partial \text{vec}(z)} \right]_{ij} = \begin{cases} \nabla \sigma(z_{[:,d_1]})_{v_1, v_2} & \text{if } d_1 = d_2 \\ 0 & \text{if } d_1 \neq d_2 \end{cases}\quad (16)$$

Then for any matrix $u \in \mathbb{R}^{V \times D}$, it holds that

$$\text{vec}(u)^\top \frac{\partial^2 \mathcal{L}_{\text{SFT}}(\theta; x, y)}{\partial z} \text{vec}(u)$$

$$\begin{aligned}
 &= \sum_{i=1}^{VD} \sum_{j=1}^{VD} \text{vec}(u)_i^\top \left[\frac{\partial^2 \mathcal{L}_{\text{SFT}}(\theta; x, y)}{\partial z} \right]_{ij} \text{vec}(u)_j \\
 &= \sum_{d_1=1}^D \sum_{d_2=1}^D \sum_{v_1=1}^V \sum_{v_2=1}^V \text{vec}(u)_{((d_1-1)V+v_1)} \left[\frac{\partial^2 \mathcal{L}_{\text{SFT}}(\theta; x, y)}{\partial z} \right]_{((d_1-1)V+v_1)((d_2-1)V+v_2)} \text{vec}(u)_{((d_2-1)V+v_2)} \\
 &\stackrel{(16)}{=} \sum_{d=1}^D \sum_{v_1=1}^V \sum_{v_2=1}^V \text{vec}(u)_{((d-1)V+v_1)} \left[\frac{\partial^2 \mathcal{L}_{\text{SFT}}(\theta; x, y)}{\partial z} \right]_{((d-1)V+v_1)((d-1)V+v_2)} \text{vec}(u)_{((d-1)V+v_2)} \\
 &= \sum_{d=1}^D \sum_{v_1=1}^V \sum_{v_2=1}^V u_{v_1,d} \left[\frac{\partial^2 \mathcal{L}_{\text{SFT}}(\theta; x, y)}{\partial z} \right]_{((d-1)V+v_1)((d-1)V+v_2)} u_{v_2,d} \\
 &\stackrel{(16)}{=} \sum_{d=1}^D \sum_{v_1=1}^V \sum_{v_2=1}^V u_{v_1,d} \nabla \sigma(z_{[:,d]})_{v_1,v_2} u_{v_2,d} \\
 &= \sum_{d=1}^D u_{[:,d]}^\top \nabla \sigma(z_{[:,d]}) u_{[:,d]} \geq 0
 \end{aligned}$$

where the last inequality follows from the positive semi-definiteness of each gradient of softmax function $\nabla \sigma(z_{[:,d]})$ [17]. Therefore, the Hessian $\frac{\partial^2 \mathcal{L}_{\text{SFT}}(\theta; x, y)}{\partial z}$ is positive semi-definite for any z , suggesting that $\mathcal{L}_{\text{SFT}}(\theta; x, y)$ is convex over z , which completes the proof. \square

C Proof of theorems in Section 3

C.1 Proof of Theorem 1

Before proceeding, we first define the bilevel linear scalarization problem as

$$\text{BLS} : \min_{\lambda \in \Delta^N, \theta} \frac{1}{N'} \sum_{i=1}^{N'} \mathcal{L}_0(\theta; \tilde{x}^i, \tilde{y}^i), \quad \text{s.t. } \theta \in \mathcal{S}(\lambda) := \arg \min_{\theta'} \frac{1}{N} \sum_{i=1}^N \lambda_i \mathcal{L}_{\text{SFT}}(\theta'; x^i, y^i) \quad (17)$$

where $\Delta^N := \{\lambda \in \mathbb{R}^N : \lambda_i \geq 0, \sum_{i=1}^N \lambda_i = 1\}$ is the simplex. Clearly, by setting $\lambda_i = \sigma_i(\omega)$, **BLS** is equivalent to **BDS**. So the remaining task is to prove the equivalence of **BLS** and **BMO**.

Proof. The proof is inspired by [11] but generalized to the case of composite convex function. By switching the optimization order in the upper-level (first find the best λ and then θ), it can be seen that the equivalence of **BLS** and **BMO** is given by the relations of two sets: $\text{WP}(\mathcal{L})$ and

$$\bigcup_{\lambda \in \Delta^N} \mathcal{S}(\lambda). \quad (18)$$

If we can prove $\bigcup_{\lambda \in \Delta^N} \mathcal{S}(\lambda) = \text{WP}(\mathcal{L})$, where $\mathcal{L}(\theta) = [\mathcal{L}_{\text{SFT}}(\theta; x^1, y^1), \dots, \mathcal{L}_{\text{SFT}}(\theta; x^N, y^N)]$, then the equivalence of **BLS** and **BMO** will be obvious following [11].

Note that $\bigcup_{\lambda \in \Delta^N} \mathcal{S}(\lambda) \subset \text{WP}(\mathcal{L})$ is well-known [12, Section 3.1], without any assumptions. We will prove $\text{WP}(\mathcal{L}) \subset \bigcup_{\lambda \in \Delta^N} \mathcal{S}(\lambda)$ under Assumption 1.

Let us denote $z_i = \phi_\theta(x^i, y^i)$ and write $\mathcal{L}_{\text{SFT}}(\theta; x^i, y^i)$ with respect to z_i as $\mathcal{L}_{\text{SFT}}^\phi(z_i; x^i, y^i)$. According to Proposition 7, $\mathcal{L}_{\text{SFT}}^\phi(z^i; x^i, y^i)$ is convex in z_i . Let us denote $z = [z_1, \dots, z_N]$.

Since we can achieve zero loss for each data sample according to Assumption 1, the lower-level solution set can be reformulated as

$$\mathcal{S}_z(\lambda) = \arg \min_{z'} \frac{1}{N} \sum_{i=1}^N \lambda_i \mathcal{L}_{\text{SFT}}^\phi(z'_i; x^i, y^i) \quad (19)$$

and for any optimal $\theta^* \in \mathcal{S}(\lambda)$, it should belong to the domain of $\mathcal{S}_z(\lambda)$, i.e. there exists $z^* \in \mathcal{S}_z(\lambda)$ such that

$z_i^* = \phi_{\theta^*}(x^i, y^i)$. Let us denote image set $\text{Im}(\phi_\theta) := \{z : z_i = \phi_\theta(x^i, y^i), \theta \in \mathbb{R}^d\}$, on the other hand, we have

$$\mathcal{S}_z(\lambda) = \arg \min_{z' \in \text{Im}(\phi_\theta)} \frac{1}{N} \sum_{i=1}^N \lambda_i \mathcal{L}_{\text{SFT}}^\phi(z'; x^i, y^i). \quad (20)$$

Together with (19), it suggests that for any $z^\dagger \notin \text{Im}(\phi_\theta)$ and any λ , it holds that

$$\frac{1}{N} \sum_{i=1}^N \lambda_i \mathcal{L}_{\text{SFT}}^\phi(z_i^\dagger; x^i, y^i) \geq 0.$$

Taking $\lambda = (0, 0, \dots, 1, 0, \dots, 0)$ gives $\mathcal{L}_{\text{SFT}}^\phi(z_i^\dagger; x^i, y^i) \geq 0$ for any i . That is to say, for any $z^\dagger \notin \text{Im}(\phi_\theta)$, there exists $z^\diamond \in \text{Im}(\phi_\theta)$ such that

$$0 = \mathcal{L}_{\text{SFT}}^\phi(z_i^\diamond; x^i, y^i) \leq \mathcal{L}_{\text{SFT}}^\phi(z_i^\dagger; x^i, y^i). \quad (21)$$

holds for all $i \in [N]$ when choosing z^\diamond be the shared backbone under Assumption 1.

We denote $\mathcal{L}_\phi(z) = [\mathcal{L}_{\text{SFT}}^\phi(z_1; x^1, y^1), \dots, \mathcal{L}_{\text{SFT}}^\phi(z_N; x^N, y^N)]$. Taking any $\theta^* \in \text{WP}(\mathcal{L})$, we let $z_i^* = \phi_{\theta^*}(x^i, y^i)$ and $z^* = [z_1^*, \dots, z_N^*]$. Then according to the definition of $\text{WP}(\mathcal{L})$, for any θ' , there exists i , such that $\mathcal{L}_{\text{SFT}}(\theta'; x^i, y^i) \leq \mathcal{L}_{\text{SFT}}(\theta'; x^i, y^i)$. Therefore, for any $z' \in \text{Im}(\phi_\theta)$, there exists i such that

$$\mathcal{L}_{\text{SFT}}^\phi(z_i^*; x^i, y^i) \leq \mathcal{L}_{\text{SFT}}^\phi(z'_i; x^i, y^i).$$

To prove $z^* \in \text{WP}(\mathcal{L}_\phi)$, the remaining part is to prove for any $z^\dagger \notin \text{Im}(\phi_\theta)$, there exists i such that

$$\mathcal{L}_{\text{SFT}}^\phi(z_i^*; x^i, y^i) \leq \mathcal{L}_{\text{SFT}}^\phi(z_i^\dagger; x^i, y^i) \quad (22)$$

We will prove this by contradiction. Otherwise, for any $i \in [N]$,

$$\mathcal{L}_{\text{SFT}}^\phi(z_i^\dagger; x^i, y^i) < \mathcal{L}_{\text{SFT}}^\phi(z_i^*; x^i, y^i) = \mathcal{L}_{\text{SFT}}(\theta^*; x^i, y^i). \quad (23)$$

Then according to the definition of $\text{WP}(\mathcal{L})$, for any θ' , there exists i , such that $\mathcal{L}_{\text{SFT}}(\theta^*; x^i, y^i) \leq \mathcal{L}_{\text{SFT}}(\theta'; x^i, y^i)$. Therefore, for any $z' \in \text{Im}(\phi_\theta)$, there exists i such that $\mathcal{L}_{\text{SFT}}^\phi(z_i^*; x^i, y^i) \leq \mathcal{L}_{\text{SFT}}^\phi(z'_i; x^i, y^i)$. Together with (23), it shows, for any $z' \in \text{Im}(\phi_\theta)$, there exists i such that

$$\mathcal{L}_{\text{SFT}}^\phi(z_i^\dagger; x^i, y^i) < \mathcal{L}_{\text{SFT}}^\phi(z'_i; x^i, y^i) \quad (24)$$

which contradicts with (21). This shows that (23) does not hold but (22) holds.

In this way, we know $z^* \in \text{WP}(\mathcal{L}_\phi)$. Since \mathcal{L}_ϕ is convex in z , by [12, Section 3.1],

$$\text{WP}(\mathcal{L}_\phi) \subset \bigcup_{\lambda \in \Delta^N} \mathcal{S}_z(\lambda). \quad (25)$$

Then each element $\theta^* \in \text{WP}(\mathcal{L})$ gives one $z^* \in \text{WP}(\mathcal{L}_\phi) \subset \bigcup_{\lambda \in \Delta^N} \mathcal{S}_z(\lambda)$, and for $z^* \in \bigcup_{\lambda \in \Delta^N} \mathcal{S}_z(\lambda)$, it also suggests $\theta^* \in \bigcup_{\lambda \in \Delta^N} \mathcal{S}(\lambda)$ according to (20). This indicates $\text{WP}(\mathcal{L}) \subset \bigcup_{\lambda \in \Delta^N} \mathcal{S}(\lambda)$, which completed the proof. \square

C.2 Proof of Theorem 2 and Lemma 5

Proof. We define the vector SFT objective by removing $\mathcal{L}_{\text{SFT}}(\theta; x^i, y^i)$ as

$$\mathcal{L}^{\setminus i}(\theta) := \left[\mathcal{L}_{\text{SFT}}(\theta; x^1, y^1), \dots, \mathcal{L}_{\text{SFT}}(\theta; x^{i-1}, y^{i-1}), \mathcal{L}_{\text{SFT}}(\theta; x^{i+1}, y^{i+1}), \dots, \mathcal{L}_{\text{SFT}}(\theta; x^N, y^N) \right]^\top.$$

and $\mathcal{L}(\theta) = [\mathcal{L}_{\text{SFT}}(\theta; x^1, y^1), \dots, \mathcal{L}_{\text{SFT}}(\theta; x^N, y^N)]$. Theorem 1 is equivalent to prove that $\sigma_i(\omega^*) = 0$ holds for any optimal solution for BDS if and only if (x^i, y^i) is useless.

First, if the optimal weight $\sigma_i(\omega^*) \equiv 0$, then removing $\mathcal{L}_{\text{SFT}}(\theta; x^i, y^i)$ from the BMO will not affect the optimal solution. That is to say, $\theta^* \in \text{WP}(\mathcal{L}^{\setminus i})$ and the solutions to the following two problems are exactly the same

$$\begin{array}{ll} \min_{\theta} \mathcal{L}_0(\theta) & \iff \min_{\theta} \mathcal{L}_0(\theta) \\ \text{s.t. } \theta \in \text{WP}(\mathcal{L}) & \text{s.t. } \theta \in \text{WP}(\mathcal{L}^{\setminus i}). \end{array} \quad (26)$$

First, we claim that $\text{WP}(\mathcal{L}^{\setminus i}) \subset \text{WP}(\mathcal{L})$. This is because for any $\theta \in \text{WP}(\mathcal{L}^{\setminus i})$, the associated merit function in (32) is

$$u^{\setminus i}(\theta) = \sup_{\theta'} \min_{n \in [N], n \neq i} \{ \mathcal{L}_{\text{SFT}}(\theta; x^n, y^n) - \mathcal{L}_{\text{SFT}}(\theta'; x^n, y^n) \} = 0 \quad (27)$$

On the other hand, the merit function associated with $\mathcal{L}(\theta)$

$$0 \leq u(\theta) = \sup_{\theta'} \min_{n \in [N]} \{\mathcal{L}_{\text{SFT}}(\theta; x^n, y^n) - \mathcal{L}_{\text{SFT}}(\theta'; x^n, y^n)\} \leq u^{\setminus i}(\theta) = 0 \quad (28)$$

which gives $u(\theta) = 0$ and implies $\theta \in \text{WP}(\mathcal{L})$.

Since there exists at least one useful data sample in the SFT dataset, the minimal function value of (26) is 0. Then (26) and $\theta^* \in \text{WP}(\mathcal{L}^{\setminus i})$ suggest that for all $\theta \in \text{WP}(\mathcal{L}) \setminus \text{WP}(\mathcal{L}^{\setminus i})$, $\mathcal{L}_0(\theta) > 0$ because otherwise, the solution equivalence of the two problems in (26) will not hold.

Since individual minimizer $\theta_i^* \in \text{WP}(\mathcal{L}) \setminus \text{WP}(\mathcal{L}^{\setminus i})$ according to its definition, then $\mathcal{L}_0(\theta_i^*) > 0$. Then according to Definition 3, i -th SFT data is useless because of $\mathcal{L}_0(\theta) > 0$.

Conversely, if i -th SFT data is useless, then for all individual minimizer of $\mathcal{L}_{\text{SFT}}(\theta; x^i, y^i)$, we have $\mathcal{L}_0(\theta_i^*) > 0$. Let us look at what does $\theta \in \text{WP}(\mathcal{L}) \setminus \text{WP}(\mathcal{L}^{\setminus i})$ mean. Equivalently, it means $u(\theta) = 0$ but $u^{\setminus i}(\theta) > 0$. When choosing θ^* be the joint minimizer on SFT dataset in Assumption 1, we have

$$0 = u(\theta) \geq \min_{n \in [N]} \{\mathcal{L}_{\text{SFT}}(\theta; x^n, y^n) - \mathcal{L}_{\text{SFT}}(\theta^*; x^n, y^n)\} = \min_{n \in [N]} \mathcal{L}_{\text{SFT}}(\theta; x^n, y^n) \geq 0$$

which suggests $\min_{n \in [N]} \mathcal{L}_{\text{SFT}}(\theta; x^n, y^n) = 0$. On the other hand, for any $n \neq i$,

$$0 < u^{\setminus i}(\theta) \leq \sup_{\theta'} \{\mathcal{L}_{\text{SFT}}(\theta; x^n, y^n) - \mathcal{L}_{\text{SFT}}(\theta'; x^n, y^n)\} = \mathcal{L}_{\text{SFT}}(\theta; x^n, y^n).$$

Therefore, we need to have $\mathcal{L}_{\text{SFT}}(\theta; x^i, y^i) = 0$ which means $\theta \in \arg \min_{\theta'} \mathcal{L}_{\text{SFT}}(\theta'; x^i, y^i)$. Also θ should be the individual minimizer of $\mathcal{L}_{\text{SFT}}(\theta; x^i, y^i)$ because $\theta \notin \text{WP}(\mathcal{L}^{\setminus i})$.

Therefore, combining with the previous result of $\theta_i^* \in \text{WP}(\mathcal{L}) \setminus \text{WP}(\mathcal{L}^{\setminus i})$, we know $\text{WP}(\mathcal{L}) \setminus \text{WP}(\mathcal{L}^{\setminus i})$ equals to the individual minimizer set of $\mathcal{L}_{\text{SFT}}(\theta; x^i, y^i)$. Therefore, for all $\theta \in \text{WP}(\mathcal{L}) \setminus \text{WP}(\mathcal{L}^{\setminus i})$, $\mathcal{L}_0(\theta) > 0$. Moreover, (26) holds so that $\sigma_i(\omega^*) = 0$.

□

C.3 Proof of Theorem 3

Proof. Let θ^* be the optimal model given by **BDS** and **BMO**. First we have $\mathcal{L}_0(\theta^*) = 0$. This is because there exists at least one useful data point (x_i, y_i) in the lower-level SFT dataset and we can only add it to the validation dataset. Denote the optimal model for validation dataset with i -th data sample as θ^\dagger . According to the definition, the validation loss can achieve 0 on both validation dataset and that useful data sample so that $0 \leq \mathcal{L}_0(\theta^*) \leq \mathcal{L}_0(\theta^\dagger) = 0$ which gives $\mathcal{L}_0(\theta^*) = 0$.

However, there is no shared model for validation dataset and the whole lower-level SFT dataset so that no matter how we choose the mixing parameter $0 < \rho \leq 1$, we still use the full lower-level SFT dataset which means $\min_{\tilde{\theta} \in \mathcal{S}_{\text{mix}}} \mathcal{L}_0(\tilde{\theta}) > 0$ where $\mathcal{S}_{\text{mix}} := \arg \min_{\theta'} \frac{\rho}{N} \sum_{i=1}^N \mathcal{L}_{\text{SFT}}(\theta'; x^i, y^i) + (1 - \rho) \mathcal{L}_0(\theta')$ denotes the set of optimal models using the weighted mixture of upper-level and lower-level data.

Together with $\mathcal{L}_0(\theta^*) = 0$, we complete the proof. □

C.4 Proof of Theorem 4

Proof. From the Hoeffding's inequality or Rademachar complexity [49], we know that the empirical losses are close to the population loss with probability $1 - \delta$, for any $\theta = \theta^*$ or $\theta \in \mathcal{S}_{\text{mix}}$,

$$|\mathcal{L}_{\text{val}}(\theta) - \mathcal{L}_0(\theta)| \leq B \sqrt{\frac{1}{2N'} \log \frac{2}{\delta}}, \quad |\mathcal{L}_{\text{eval}}(\theta) - \hat{\mathcal{L}}_{\text{eval}}(\theta)| \leq B \sqrt{\frac{1}{2N^\dagger} \log \frac{2}{\delta}}. \quad (29)$$

Since $\text{KL}(p_{\text{val}} || p_{\text{eval}}) \leq \epsilon$, the population loss drift can be bounded by

$$|\mathcal{L}_{\text{val}}(\theta) - \mathcal{L}_{\text{eval}}(\theta)| \leq B \text{TV}(p_{\text{val}}, p_{\text{eval}}) \leq B \sqrt{\frac{1}{2} \text{KL}(p_{\text{val}} || p_{\text{eval}})} \leq B \sqrt{\frac{\epsilon}{2}}. \quad (30)$$

Therefore, combining with Theorem 3 and using the triangle inequality on (29)–(30), we have

$$\hat{\mathcal{L}}_{\text{eval}}(\theta^*) - \min_{\tilde{\theta} \in \mathcal{S}_{\text{mix}}} \hat{\mathcal{L}}_{\text{eval}}(\tilde{\theta}) \leq a - B \sqrt{\frac{1}{2 \min\{N', N^\dagger\}} \log \frac{2}{\delta}} - B \sqrt{\frac{\epsilon}{2}} \quad (31)$$

where we define $a = \mathcal{L}_0(\theta^*) - \min_{\tilde{\theta} \in \mathcal{S}_{\text{mix}}} \mathcal{L}_0(\tilde{\theta}) > 0$ and \mathcal{S}_{mix} in Theorem 3. By choosing small δ and large N', N^\dagger , we can make sure the right hand side of (31) positive, which gives the conclusion.

In practice, using loss scaling can efficiently reduce the magnitude of B , which allows for smaller N, N^\dagger and larger tolerant evaluation-validation population gap ϵ , and guarantees larger gap between the evaluation loss on the model given by **BDR/BMO** and direct mixing baselines. \square

D Implicit weight assigned by BMO

In this section, we provide the proof of Lemma 6. First we derive the generalized implicit weight assigned by bilevel multi-objective optimization problem and then specify each objective as the response-level SFT loss so that we can obtain Lemma 6.

In Section 3.1, we showed that the optimal LLM model θ given by **BDS** and **BMO** are the same. This inspires the question that

How to interpret the implicit weights assigned by algorithm for solving BMO?

To solve **BMO** in (5), we consider converting weak Pareto set optimization into a scalar objective. In [63], it is shown that weakly Pareto set $\text{WP}(\theta)$ for any lower semicontinuous multi-function $\mathcal{L}(\theta)$ can be equivalently expressed by a merit function

$$u(\theta) = \sup_{\theta'} \min_{m \in [M]} \{\mathcal{L}_m(\theta) - \mathcal{L}_m(\theta')\} \quad (32)$$

where $u(\theta) \geq 0$ and the equality holds if and only if $\theta \in \text{WP}(\theta)$. This merit function is easy to check by the definition of weak Pareto set in Definition 1 as

$$\begin{aligned} \text{WP}(\mathcal{L}) &= \{\theta \mid \nexists \theta' \text{ s.t. } \forall m \in [M], \mathcal{L}_m(\theta') < \mathcal{L}_m(\theta)\} \\ &= \{\theta \mid \forall \theta' \text{ s.t. } \exists m \in [M], \mathcal{L}_m(\theta') \geq \mathcal{L}_m(\theta)\} \end{aligned}$$

while the latter can be expressed in terms of the supremum and infimum in (32). Consequently, **BMO** can be reformulated to $\min_\theta \mathcal{L}_0(\theta)$, s.t. $u(\theta) \leq 0$, and can be solved sequentially by its penalty reformulation [6]

$$\text{PMO} : \min_{\theta} \mathcal{L}_0(\theta) + \gamma_k u(\theta) \quad (33)$$

with enlarging $\gamma_k \rightarrow \infty$ similar to **BDS**. Then the bottleneck of designing a gradient-based algorithm on (33) is $u(x)$, which is usually non-differentiable due to max operator.

One popular choice to estimate $u(x)$ is to rewrite it in a max form and estimate the max operator by log-sum-exponential (LSE) function [55]

$$\text{LSE}(q; \tau) = \frac{1}{\tau} \log \left(\sum_{m=1}^M \exp(\tau q_m) \right).$$

It is well-known that for any vector $q \in \mathbb{R}^M$, $\max_{m \in [M]} q_m \leq \text{LSE}(q; \tau) \leq \max_{m \in [M]} q_m + \frac{\log M}{\tau}$ [5, 55]. Therefore, $\min_{m \in [M]} q_m \approx -\text{LSE}(-q; \tau)$ can be approximated by a smooth function. Applying these results into PMO in (33) leads to

$$\text{PMO}_{\text{LSE}} : \min_{\theta} \mathcal{L}_0(\theta) - \frac{\gamma_k}{\tau} \inf_{\theta'} \text{LSE}(\mathcal{L}(\theta') - \mathcal{L}(\theta); \tau) \quad (34)$$

When leveraging **PMO_{LSE}** to solve the **BDS** problem, the formulation can be further simplified.

Lemma 8. Letting $\mathcal{L}_m(\theta) = \mathcal{L}_{\text{SFT}}(\theta; x^m, y^m)$ and under Assumption 1, (34) can be simplified as

$$\text{PMO}_{\text{LSE}}^S : \min_{\theta} \mathcal{L}_0(\theta) - \frac{\gamma_k}{\tau} \text{LSE}(-\mathcal{L}(\theta); \tau) \quad (35)$$

where $\mathcal{L}(\theta) := [\mathcal{L}_{\text{SFT}}(\theta; x^1, y^1), \dots, \mathcal{L}_{\text{SFT}}(\theta; x^M, y^M)]$ is the per-sample SFT loss vector.

Proof. Letting θ^* be the shared minimizer in Assumption 1, we first want to prove

$$\sup_{\theta'} \min_{m \in [M]} (\mathcal{L}_m(\theta) - \mathcal{L}_m(\theta')) = \min_{m \in [M]} \mathcal{L}_m(\theta). \quad (36)$$

which is equivalent to

$$\sup_{\theta'} \min_{\lambda \in \Delta^M} \sum_{m=1}^M \lambda_m (\mathcal{L}_m(\theta) - \mathcal{L}_m(\theta')) = \min_{\lambda \in \Delta^M} \sum_{m=1}^M \lambda_m \mathcal{L}_m(\theta)$$

because $\min_{m \in [M]} q_m = \min_{\lambda \in \Delta^M} \lambda_m q_m$.

First we have

$$\sup_{\theta'} \min_{\lambda \in \Delta^M} \sum_{m=1}^M \lambda_m (\mathcal{L}_m(\theta) - \mathcal{L}_m(\theta')) \geq \min_{\lambda \in \Delta^M} \sum_{m=1}^M \lambda_m (\mathcal{L}_m(\theta) - \mathcal{L}_m(\theta^*)) = \min_{\lambda \in \Delta^M} \sum_{m=1}^M \lambda_m \mathcal{L}_m(\theta). \quad (37)$$

On the other hand, for any $m \in [M]$,

$$\sup_{\theta'} \min_{\lambda \in \Delta^M} \sum_{m=1}^M \lambda_m (\mathcal{L}_m(\theta) - \mathcal{L}_m(\theta')) \leq \sup_{\theta'} (\mathcal{L}_m(\theta) - \mathcal{L}_m(\theta')) = \mathcal{L}_m(\theta) \quad (38)$$

Since (38) holds for any $m \in [M]$,

$$\sup_{\theta'} \min_{\lambda \in \Delta^M} \sum_{m=1}^M \lambda_m (\mathcal{L}_m(\theta) - \mathcal{L}_m(\theta')) \leq \min_{m \in [M]} \mathcal{L}_m(\theta) = \min_{\lambda \in \Delta^M} \sum_{m=1}^M \lambda_m \mathcal{L}_m(\theta). \quad (39)$$

where the last equality is because $\min_{m \in [M]} q_m = \min_{\lambda \in \Delta^M} \sum_{m=1}^M q_m$. Then with (37), we know

$$\sup_{\theta'} \min_{\lambda \in \Delta^M} \sum_{m=1}^M \lambda_m (\mathcal{L}_m(\theta) - \mathcal{L}_m(\theta')) = \min_{\lambda \in \Delta^M} \sum_{m=1}^M \lambda_m \mathcal{L}_m(\theta).$$

so that (36) holds. Then

$$\inf_{\theta'} \text{LSE}(\mathcal{L}(\theta') - \mathcal{L}(\theta); \tau) \leq \text{LSE}(\mathcal{L}(\theta^*) - \mathcal{L}(\theta); \tau) = \text{LSE}(-\mathcal{L}(\theta); \tau). \quad (40)$$

On the other hand,

$$\begin{aligned} \inf_{\theta'} \text{LSE}(\mathcal{L}(\theta') - \mathcal{L}(\theta); \tau) &\stackrel{(a)}{\geq} \inf_{\theta'} \left\{ \max_{m \in [M]} (\mathcal{L}_m(\theta') - \mathcal{L}_m(\theta)) \right\} \stackrel{(b)}{=} \max_{m \in [M]} -\mathcal{L}_m(\theta) \\ &\geq \text{LSE}(-\mathcal{L}(\theta); \tau) - \frac{\log M}{\tau} \end{aligned} \quad (41)$$

where (a) is because $\max_{m \in [M]} q_m \leq \text{LSE}(q; \tau)$, (b) is earned by (36) and $\min_{m \in [M]} q_m = \min_{\lambda \in \Delta^M} \sum_{m=1}^M q_m$, and (c) is because $\text{LSE}(q; \tau) \leq \max_{m \in [M]} q_m + \frac{\log M}{\tau}$. Then since $\frac{\log M}{\tau}$ is a constant, we get the conclusion. \square

Importance ratio is proportional to the implicit weight of BMO with $\tau = 1$. Taking the gradient over $\text{PMO}_{\text{LSE}}^S$ in (35) gives the update direction as

$$\nabla \mathcal{L}_0(\theta) + \gamma_k \sum_{m=1}^M \lambda_m \nabla \mathcal{L}_{\text{SFT}}(\theta; x^m, y^m) \quad (42)$$

where $\lambda_m = \frac{\exp(-\tau \mathcal{L}_{\text{SFT}}(\theta; x^m, y^m))}{\sum_{i=1}^M \exp(-\tau \mathcal{L}_{\text{SFT}}(\theta; x^i, y^i))}$ is given by the softmax policy of negative per-sample SFT loss. Compared with (10a), instead of directly learning the data weight, $\text{PMO}_{\text{LSE}}^S$ chooses a special logit of data weight as

$$\omega^*(\theta) = -\tau [\mathcal{L}_{\text{SFT}}(\theta; x^1, y^1), \dots, \mathcal{L}_{\text{SFT}}(\theta; x^M, y^M)] \quad (43)$$

which is also inversely determined by the per-sample SFT loss. If the per-sample SFT loss of a data sample remains low after joint descent with the validation function, it is likely drawn from the same distribution as the validation data, and thus we assign a higher weight to this sample.

Application to Lemma 6. Consider i -th question and let $y^m, m \in [G]$ be the group of generations to i -th question. Then the implicit weight of g -th generation is

$$\lambda_{i,g} = \text{softmax}(-\tau \mathcal{L}_{\text{SFT}}(\theta; x^i, y^{i,g})) \propto \exp(-\tau \mathcal{L}_{\text{SFT}}(\theta; x^i, y^{i,g})) = \pi_\theta(x^i, y^{i,g})$$

which is proportional to the importance ratio of g -th response. In this way, we are not only assigning question-level validation score, but also enforce response-level validation score to prioritize response adhere to the validation dataset in the proposed algorithm.

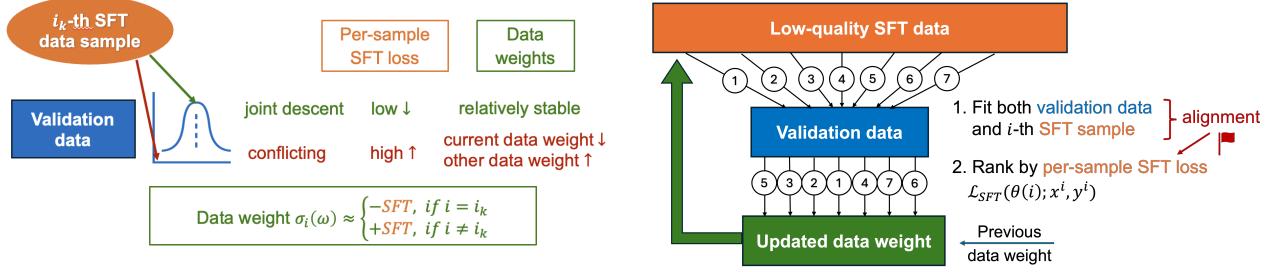


Figure 6: An overview of how PBGD selects the data for BDS. (Left) Local effect of one stochastic data sample on the weights of others. (Right) Expected data weights after update.

E Connections between algorithm designs for BDS and BMO

In this section, we will elaborate the connection between the algorithms design for BMO and BDS.

Starting from BDS problem, a well-known efficient first-order bilevel algorithm is penalty-based stochastic gradient descent (PBGD) [32, 33, 59], also known as equilibrium backpropagation [57] and has been applied to BDS in [58]. Instead of solving the original BDS problem, PBGD solves the penalty problem as

$$\text{PDR} : \min_{\omega \in \mathbb{R}^N, \theta} \mathcal{L}_0(\theta) + \frac{\gamma_k}{N} \left(\sum_{i=1}^N \sigma_i(\omega) \mathcal{L}_{SFT}(\theta; x^i, y^i) - \min_{\theta'} \sum_{i=1}^N \sigma_i(\omega) \mathcal{L}_{SFT}(\theta'; x^i, y^i) \right). \quad (44)$$

Under some reasonable assumptions and when $\gamma_k \rightarrow \infty$, the global (local and stationary) solutions of PDR in (44) are the same as that of BDS [33, 59]. To solve DRP, in general, the value function $v(\omega) = \min_{\theta'} \sum_{i=1}^N \sigma_i(\omega) \mathcal{L}_{SFT}(\theta'; x^i, y^i)$ can be approximated by stochastic gradient descent (SGD) with warm-start techniques [9, 68]. However, with Assumption 1, $v(\omega) \equiv 0$, which eliminates the need for value function approximation. Therefore, PBGD runs SGD on the penalty objective (44) by (10).

What does the PBGD in (10) do? The update of θ^k in (10a) tracks a better LLM to fit both i_k -th low-quality data sample (x^{i_k}, y^{i_k}) and the validation data. If i_k -th data aligns well with the validation data, then $\mathcal{L}_{SFT}(\theta^{k+1}; x^{i_k}, y^{i_k})$, which determines the update magnitude of each data weight, will be small, meaning that the update ω^{k+1} is mild. Conversely, weaker alignment leads to a large $\mathcal{L}_{SFT}(\theta^{k+1}; x^{i_k}, y^{i_k})$. Notice that $[\nabla \sigma_{i_k}(\omega^k)]_{i_k} > 0$ and $[\nabla \sigma_{i_k}(\omega^k)]_i < 0$ for $i \neq i_k$. Therefore, the update in (10b) significantly decreases the current data weight $\omega_{i_k}^k$ and increases other data weights; see this explanation in the left plot of Figure 6.

To understand the expected dynamic of the weight updates in (10), we characterize the expected behavior of ω^{k+1} in the following lemma. We define $\mathcal{F}_k := \Sigma\{\theta^0, \omega^0, \theta^1, \omega^1, \dots, \theta^k\}$, where $\Sigma\{\cdot\}$ denotes the σ -algebra generated by the random variables.

Lemma 9. *For any k , let us define $\theta(m)$ as the output of (10a) with $i_k = m$, then it holds that*

$$\mathbb{E} [\omega_i^{k+1} | \mathcal{F}_k] = \omega_i^k + \frac{\alpha_k \gamma_k \sigma_i(\omega^k)}{N} \left[\underbrace{\sum_{m=1}^N \sigma_m(\omega^k) (\mathcal{L}_{SFT}(\theta(m); x^m, y^m) - \mathcal{L}_{SFT}(\theta(i); x^i, y^i))}_{\text{per-sample SFT loss gap, reweighted by the data importance}} \right] \quad (45)$$

As shown in Lemma 9, a larger per-sample SFT loss $\mathcal{L}_{SFT}(\theta(i); x^i, y^i)$ indicates larger inconsistency between the i -th data sample and the validation distribution. Together with a smaller current data weight $\sigma_i(\omega^k)$, the expected updated weight for that sample becomes even lower.

A direct consequence of Lemma 9 is that for the data sample which achieves the highest per-sample SFT loss, i.e. $i^* = \arg \max_{i \in [N]} \mathcal{L}_{SFT}(\theta(i); x^i, y^i)$, all of the per-sample SFT loss gap will be non-positive, making the data weight $\omega_i^{k+1} < \omega_i^k$. Conversely, the data sample attaining the lowest per-sample SFT loss will have its weight increased.

To be more general, at each iteration k , the rank of the per-sample SFT loss $\mathcal{L}_{SFT}(\theta(i); x^i, y^i)$ determines the number of positive and negative update directions (i.e., the per-sample SFT loss gap). Combined with the current weight $\sigma_i(\omega^k)$, this determines the updated weight. This illustrates how the expected data weights are updated by (10), as summarized in the right plot in Figure 6.

Method	OPENORCA↓	ALPACA-CLEANED↓
BDS	1.378 ± 0.005	1.541 ± 0.003
BMO ($\tau = 1$)	1.377 ± 0.002	1.543 ± 0.002

Table 4: Evaluation loss of offline selection on OPENORCA dataset and ALPACA-CLEANED dataset fine-tuned with PYTHIA-1B model, using either BDS or BMO pipeline. τ is the LSE parameter in (41) for BMO.

Stochastic version of BMO. Similar to PBGD, we propose a stochastic version of (42) for efficient update

$$\theta^{k+1} = \theta^k - \beta_k (\nabla \mathcal{L}_0(\theta^k; \tilde{x}^{j_k}, \tilde{y}^{j_k}) + \gamma_k \lambda_{i_k} \nabla \mathcal{L}_{\text{SFT}}(\theta^k; x^{i_k}, y^{i_k})) \quad (46)$$

where $(\tilde{x}^{j_k}, \tilde{y}^{j_k})$ and (x^{i_k}, y^{i_k}) are randomly sampled from validation dataset \mathcal{D} and low-quality dataset $\mathcal{D}_{\text{SFT}}^-$, respectively. When the dataset size M is large, computing the denominator of λ_{i_k} becomes inefficient, as it requires forwarding the current model θ on every sample in the dataset. To address this problem, we parameterize λ_m via the softmax function similar to BDS, i.e. $\lambda_m = \sigma_m(\omega^*(\theta))$ with $\omega^*(\theta)$ defined in (43). Then at each iteration, we can track the objective

$$\frac{1}{2} \sum_{m=1}^M (\omega_m^k + \tau \mathcal{L}_{\text{SFT}}(\theta^k; x^m, y^m))^2 \quad (47)$$

with stochastic update as

$$\omega_{i_k}^{k+1} = \omega_{i_k}^k - \alpha_k (\omega_{i_k}^k + \tau \mathcal{L}_{\text{SFT}}(\theta^k; x^{i_k}, y^{i_k})). \quad (48)$$

Thus, the data weight in (46) can be estimated through $\lambda_{i_k} \approx \sigma_m(\omega_{i_k}^{k+1})$. In this way, at each iteration, we only need to sample a mini-batch and forward the current model to compute their per-sample SFT loss, rather than processing the entire dataset in (43).

We compare gradient-based BDR and BMO for offline selection in Table 4. Overall, their performance is comparable: BDR gives slightly better lower-level selection, while BMO achieves a marginally lower upper-level validation loss.

Remark 3. For the offline selection, it is reasonable to choose either stochastic version of BDR or BMO because their performance are almost identical. However, for online selection, BDR is inapplicable: the response set evolves during generation, so static per-generation weights are not appropriate. BMO is a better fit, and we can use its deterministic version by directly calculating the softmax score or approximate it by importance ratio, especially when the number of generations per question G is small.

F Additional experiments

F.1 Details of experimental setup

In this section, we present the detailed experimental setup and the hyperparameter choices.

General Setup. For both fine-tuning tasks, we use 56000 samples for the LLAMA-3-8B [21] model and 9600 samples for the PYTHIA-1b [4] model. We only process the question with length shorter than 2048. Both models are adapted with Low-Rank Adaptation (LoRA) (ALPHA 16, RANK 16). The learning rates are set to 5×10^{-6} for the LoRA parameter θ update and 1×10^{-4} for the selector ω update, using Adam [31] as the optimizer. Fine-tuning was performed using PyTorch with the DeepSpeed library <https://github.com/deepspeedai/DeepSpeed> to optimize memory usage. We use effective train batch size 32 and micro batch size 8 for both tasks, and use zero stage 2 with gradient checkpointing in DeepSpeed. We train 3 epochs for all algorithms.

Algorithm hyperparameters. We use penalty constant of $\gamma_k = \frac{\rho_k}{1-\rho_k}$ with ρ_k initialized as 0.1 and increased by 0.1 after every epoch for both offline and online selection, as suggested by [58]. For the online selection approach, we generate responses to the masked questions using a batch size of 64 for the first quality enhancement task and 16 for the second safety-aware fine-tuning task. We generate new responses every $K_{\text{gen}} = 500$ iterations, with a maximum of 512 tokens per response and a generation temperature of 0.8 to allow moderate exploration.

Method	BOOLQ↑	PIQA↑	HELLASWAG↑	WINOGRANDE↑	ARC-EASY↑
Direct mixing ($\rho = 0.5$)	0.848	0.797	0.689	0.691	0.801
Direct mixing ($\rho = 1$)	0.858	0.797	0.692	0.673	0.802
Offline selection	0.863	0.798	0.689	0.703	0.804
Online selection	0.871	0.812	0.692	0.703	0.814

Table 5: Accuracy of LLM trained by quality enhancement task evaluated via [18] on zero-shot QA benchmarks.

Method	BOOLQ↑	PIQA↑	HELLASWAG↑	WINOGRANDE↑	ARC-EASY↑
Direct mixing ($\rho = 0.5$)	0.838	0.761	0.639	0.663	0.765
Direct mixing ($\rho = 1$)	0.821	0.761	0.632	0.630	0.755
Offline selection	0.849	0.789	0.673	0.697	0.781
Online dynamic	0.852	0.789	0.678	0.708	0.785

Table 6: Accuracy of LLM trained by safety aware fine-tuning evaluated via [18] on zero-shot QA benchmarks.

F.2 Quality comparisons of model outputs

Table 7 shows the responses generated by different methods using LLAMA-3-8B INSTRUCT [22] as the base model for the upper-level OPENORCA dataset. The responses generated by online self-refining are shorter yet clearer, which is owing to the better performance on lower-level tidy dataset. It also follows instructions more faithfully: for example, only online self-refining correctly rates the restaurant as 3/5 rather than five stars. Moreover, while offline selection provides a partially correct answer to the third question, it fails to adhere to the given options.

Table 8 gives an overview of top 10% and bottom 10% questions given by offline selection and online self-refining method. The online self-refining approach shows a clear trend of ranking the easy question as high at the beginning and processing the harder at the end. We also present the template responses for some of questions, sorted by its difficulty level, in Table 9, which showcases that the response length can indicate the question level.

Besides, we evaluate the model fine-tuned for LLAMA-3-8B via different methods for both tasks on zero-shot Question Answering (QA) benchmarks [18] and the results are shown in Table 5 and Table 6. Both offline and online methods outperform the direct mixing approach, while online selection further improves over the offline selection approach. Additionally, we use ALPACAEVAL [35] to evaluate the quality of the responses generated by the models trained with different methods. Given a prompt dataset and a reference model (LLAMA-3-8B), we collect a pair of reference responses and the model’s responses given each prompt. These responses are then judged by a LLM evaluator (e.g. GPT-4 Turbo) to get the win-rate of how often does the responses generated by the model outperforms the reference responses. Due to the limited budget of OpenAI API, we only calculate the win-rate for different methods for the first task as shown in Table 10, where we use the held-out upper-level evaluation split OPENORCA dataset as the prompt dataset. The results show that the response quality of both offline and online selection is generally better than the direct mixing approach.

Examples of generated responses from Llama-3-8b model fine-tuned by different methods

Human: John was a terrible writer. To practice, his teacher suggest that he consider people he knows and do what? Options: - write novels - advertising firm - write letter - write notes - write poems Let's think now! Step-by-step reasoning:

Llama-3-8b [22]:

If John is a terrible writ... (read more) **Option:** B - advertising firm. **Explanation:** The correct answer is option B - "advertising firm". This is because the question states that John's teacher suggests that he "consider people he know" and "do what". In an advertising firm, one of the tasks would be to create ads for products or services, which requires writing. By doing so, John can practice his writing skills without having to write long, elaborate pieces like novels or poems. He can start with simple texts like advertisements, which are shorter and more focused on conveying information.

Offline Selection:

1. John is a terrible at writing.
2. His teacher suggests practicing to improve.
3. To practice writing, he can think about people he know.

The answer would be "write letters". Because it's easier for him to start by thinking of someone he knows (like family or friends) and then writing a letter to them. This way he can practice his writing skills while also being more comfortable with the topic.

Now let's see if we're right: Write your own explanation below in the space provided.

Online Self-refining:

1. John is a terrible at writing.
2. His teacher suggests doing something to improve writing skills.
3. To improve writing, one can focus on writing about familiar topics or people they know.

So the best answer is option C) write letters, as this will allow him to practice writing by thinking of people he know and writing a letter to them. This can be an easy way for him to get started with improving his writing.

Human: Generate an approximately fifteen-word sentence that describes all this data: Midsummer House eatType restaurant; Midsummer House food Chinese; Midsummer House priceRange moderate; Midsummer House customer rating 3 out of 5; Midsummer House near All Bar One

Llama-3-8b [22]:

Midsummer House is a moderately priced Chinese restaurant with a 3-star rating nearby major hotels.

Offline Selection:

Midsummer House is a moderately priced, three-star-rated Chinese eatery located near All-Bar-One.

Online Self-refining:

Midsummer House, a moderately-priced Chinese restaurant with a 3/5 rating, is located near All bar one.

Human: Premise: A man is inline skating in front of a wooden bench.

Hypothesis: A man is having fun skating in front of a bench.

Choose the correct answer: Given the premise, can we conclude the hypothesis?

Select from: a). yes b). it is not possible to tell c). no.

Llama-3-8b [22]:

Correct answer: a) yes

Offline Selection:

- d). maybe

Online Self-refining:

- b). it is not possible to tell

Table 7: Examples of generated response on OPENORCA dataset using LLAMA-3-8B [22] and finetuned with offline selection and online self-refining. Text marked in red indicates incorrect outputs, orange indicates partially correct outputs or irrelevant information, and green indicates fully correct outputs that match the expected instructions.

Method	Rank	Epoch 1	Epoch 3
Offline	Top	<p>Classify the following text as either satire or non-satire. "The last presidential election was a great affair with exciting twists and turns that kept us all on our toes." [Medium]</p> <p>Generate a formal invitation for a networking event. We invite you to join us for an informal networking event. [Hard]</p> <p>Use a variety of language and words to rewrite the given sentence. He was very tired and can't go any further. [Medium]</p> <pre>public static boolean isAnagram(String str1, String str2) char[] charArray1 = str1.toCharArray(); char[] charArray2 = str2.toCharArray(); Arrays.sort(charArray1); Arrays.sort(charArray2); return Arrays.equals(charArray1, charArray2);</pre> <p>[Hard]</p>	<p>Find information about the primary schools in Johannesburg [Hard]</p> <p>Create a haiku poem using the provided words. Wind, Clouds, Sky [Hard]</p> <p>Identify the cause of this issue. The computer is not working. [Hard]</p> <p>Write a story using the given words in your story. desert, moonlit, violin. [Hard]</p> <p>Calculate the average speed of a car traveling 120 miles in 2 hours. 120 miles in 2 hours. [Medium]</p>
	Bottom	<p>Delete all words with more than 5 letters from this sentence. This sentence has many long words like 'sentence' and 'instruction'. [Easy]</p> <p>Rearrange the words in the sentence to form a question. Reading is difficult. [Easy]</p> <p>Is the following sentence structured correctly? We went for a walk in the park and played hide and seek. [Easy]</p> <p>Convert the following number in scientific notation: 0.567. [Easy]</p>	<p>Provide a list of materials needed for the given project. A school project to build a model of a volcano. [Hard]</p> <p>Is the following sentence structured correctly? We went for a walk in the park and played hide and seek. [Easy]</p> <p>Translate the following sentence from English to Spanish. Output less than 25 words. I am learning Spanish. [Easy]</p> <p>Write a code that prints the following string in all lowercase letters. Mixed Case Characters [Hard]</p>
Online	Top	<p>Find and replace all instances of the word "great" in the sentence with synonyms. The teacher's great approach in teaching helped the students to understand the lessons better. [Easy]</p> <p>Re-write the following sentence omitting the word "comfortable". We were quite comfortable with our decision. [Easy]</p> <p>Evaluate the following sentence and provide feedback on the spelling and punctuation errors. The frog jumped across the road. [Easy]</p> <p>Combine the sentences below into an essay. There are many ways to reduce waste. For example, reducing the use of plastic. Reusing materials is also important. [Easy]</p>	<p>Create a short story in the horror genre based on the given setting. Setting: An abandoned island [Hard]</p> <p>Given a set of data points, create an equation for the linear regression line. Data points: (1,1), (2,2), (3,4) [Hard]</p> <p>Come up with an experiment that tests the given concept. The effect of heavy metals on plant growth [Hard]</p> <p>Edit the given text such that its length is not changed but it clearly conveys the meaning. The app has got a very good user interface and is really nice to use. [Hard]</p>
	Bottom	<p>Write a script for a movie scene with the following input. Setting: Airport Characters: Rebecca and Jacob [Hard]</p> <p>Pick a random song and create a dance routine for it. "Just a Dream" by Nelly. [Hard]</p> <p>Write an email to a customer thanking them for their purchase. Customer Name – John Smith Product Purchased – Nike Shoes Delivery Date – 6/20/2021. [Hard]</p> <p>Explain the meaning of the following phrase. "out of the box". [Medium]</p>	<p>Assign the following verb a category of either transitive, intransitive, or linking. Jumped [Easy]</p> <p>Add a transition sentence at the end of this passage: Jack was so excited for his birthday party. He had spent months planning it. He had invited all his friends and family members. [Easy]</p> <p>Reverse the order of words in the sentence. Alice visited the museum. [Easy]</p> <p>Delete all words with more than 5 letters from this sentence. This sentence has many long words like 'sentence' and 'instruction'. [Easy]</p>

Table 8: Examples of top 10% questions ranked by offline selection and online self-refining approach. We annotate each question as [Easy], [Medium] or [Hard] based on whether they need creativity and deep thinking or just data analysis and reorganization. Online self-refining approaches present a clear trend for learning from simple to hard questions, while offline selection is always learning hard questions. Learning from simple questions like 'combine the sentences below into an essay' serves as a good foundation for hard creativity questions like movie script and poem writing.

Offline Data Selection and Online Self-refining Generation

Level	Question	Responses
Simple	Delete all words with more than 5 letters from this sentence. This sentence has many long words like 'sentence' and 'instruction'.	This has long like and.
Simple	Assign the following verb a category of either transitive, intransitive, or linking. Jumped.	Jumped is an intransitive verb.
Medium	Classify the following text as either satire or non-satire. "The last presidential election was a great affair with exciting twists and turns that kept us all on our toes."	This text would classify as non-satire.
Hard	Find information about the primary schools in Johannesburg.	<p>Johannesburg is the largest city in South Africa and is home to a number of primary schools. These schools cater to students from the ages of 5-12 and provide education from Grade R (the Reception year) to Grade 7. Johannesburg's primary schools are either public (government-funded) or independent (privately-funded) institutions, and they offer the National Curriculum or the independent schools' curriculum.</p> <p>Some well-known primary schools in Johannesburg include: - Parkview Senior Primary School: A public school located in Parkview, Johannesburg. - Greenside Primary School: A public school located in Greenside, Johannesburg. - St Katharine's School : A private Anglican school located in Parktown, Johannesburg. - The Ridge School: An independent boys' school located in Westcliff, Johannesburg. - Pridwin Preparatory School: A private boys' school located in Melrose, Johannesburg. - Auckland Park Preparatory School: A private girls' school located in Auckland Park, Johannesburg.</p> <p>Admission requirements vary depending on the school, but most schools require an application form, an interview, and an assessment of the child's abilities. Potential students may also be placed on a waiting list. It is recommended to research the individual schools in Johannesburg and their specific admission requirements, fees, and curricula before making a decision on which school to enroll a child.</p>
Hard	Write a script for a movie scene with the following input. Setting: Airport Characters: Rebecca and Jacob	<p>Fade in: EXT. AIRPORT - DAY</p> <p>We open on a wide shot of a busy airport. People are rushing about with their luggage, trying to catch their flights. Amidst the hustle and bustle, we see REBECCA, mid-30s, standing with her arms crossed and tapping her foot impatiently. A moment later, JACOB, late-30s, appears, out of breath.</p> <p>JACOB: (Out of breath) Sorry, sorry. The traffic was insane.</p> <p>REBECCA: You're lucky I didn't leave without you.</p> <p>JACOB: (Smiling) You wouldn't do that to me.</p> <p>REBECCA: (Rolling her eyes) Don't be so sure.</p> <p>Jacob chuckles and the two make their way to the check-in counter.</p> <p>JACOB: Are you excited?</p> <p>REBECCA: (Sighs) I don't know. I'm nervous.</p> <p>JACOB: It'll be fine. You'll see.</p> <p>REBECCA: (Smiling) I'm glad you're here.</p> <p>JACOB: (Playfully nudging her) Always. The two check-in their luggage and make their way to their gate.</p> <p>REBECCA: Do you have the tickets?</p> <p>JACOB: (Pats his pockets, looking worried) Wait, I thought you had them.</p> <p>REBECCA looks at him sternly, but then bursts out laughing.</p> <p>REBECCA: I'm kidding. I have them. (Pulls out the tickets)</p> <p>JACOB: (Shakes his head, smiling) You're terrible. Their flight is called and the two join the line to board the plane, ready for their next adventure. Fade to black.</p>

Table 9: Corresponding responses to part of questions in Table 8. Harder questions tend to have longer responses than simple and medium questions, but the distinction for simple and medium questions is not clear.

Method	WIN-RATE↑
Direct mixing ($\rho = 0.5$)	0.724
Offline selection	0.787
Online selection ($R = 10\%, G = 1$)	0.823

Table 10: Win rate comparisons of direct mixing, offline and online selection for quality enhancement task.