Towards Tax-Free Safety Alignment for Large Reasoning Models

Shen Jiarun
The Chinese University of Hong Kong, Shenzhen
225045001@link.cuhk.edu.cn

Gu Wei
Zhejiang University
weigu@zju.edu.cn

Zheng Tianhang
Zhejiang University
zthzheng@zju.edu.cn

Abstract

Safety alignment of Large Reasoning Models (LRMs) often exhibits a *safety tax*: improving safety and refusal behavior can degrade general capabilities. We study a data-centric hypothesis: the tax is largely driven by *distribution shift* (proxied by perplexity under the base model) and *safety constraints* (proxied by toxicity of base-model generations). We propose **LOMO-CV**, a lightweight and gradient-free pipeline that learns an interpretable linear scorer over dataset descriptors and uses it to rank and filter safety data. LOMO-CV is trained from benchmark-induced *pairwise dataset preferences* pooled across multiple model families via leave-one-model-out cross-validation. On DeepSeek-R1-Distill-Qwen-7B, LOMO-CV improves Math Score by +6.66 points over the best single-dataset baseline while maintaining near-maximal Safety Score (99.0). Across five holdout models, LOMO-CV yields Safety Score within 1 point of the best baseline in 4/5 cases, while capability retention is model-dependent, highlighting both the promise and current failure modes of data-only safety tax reduction.

1 Introduction

Large Reasoning Models (LRMs) achieve strong performance on complex tasks, yet safety alignment often degrades their reasoning ability. Recent evidence suggests this trade-off is systematic for LRMs, motivating the term *safety tax* (Huang et al., 2025). A key open question is *why* some safety datasets incur large tax while others do not, under the same alignment recipe.

We hypothesize that safety tax is largely *data-driven*. First, safety responses can be far from the base model's output distribution, inducing optimization pressure that overwrites reasoning-relevant behavior. Second, safety datasets differ in the structure and verbosity of safe responses (e.g., templated refusals vs. detailed safe reasoning), which affects both safety gains and capability retention. This mirrors findings that models fine-tune more effectively on responses that are more "familiar" under the base model (e.g., lower perplexity), suggesting distribution shift is a key predictor of training outcomes.

Motivated by this, we introduce a lightweight pipeline to *rank and filter* safety datasets before alignment. Our method computes a compact set of dataset-level metrics and learns an interpretable linear scorer from benchmark-induced rankings over candidate datasets. Importantly, we pool rankings across multiple LRMs to learn a single "general" weight vector that transfers to unseen models and corpora, enabling best-first data selection without gradients or expensive rewriting.

Contributions. (1) We formalize safety tax under a fixed alignment recipe and show it varies substantially across safety datasets and LRMs. (2) We propose a gradient-free, metric-based dataset scoring function and learn its weights from benchmark rankings via pairwise preference learning. (3) We demonstrate that pooled (cross-model) supervision yields transferable "general" weights that improve safety while reducing reasoning degradation. (4) We provide interpretability via feature attributions and quantify the compute savings relative to gradient-based data selection.

2 Related Work

Aligning large language models to be helpful and harmless is commonly framed as preference optimization, most prominently via reinforcement learning from human feedback (RLHF) (Ouyang et al., 2022) and, more recently, direct preference optimization (DPO) and related offline preference-learning objectives (Rafailov et al., 2023). Safety-focused tasks often introduce additional constraints (e.g., KL regularization to a reference policy) or

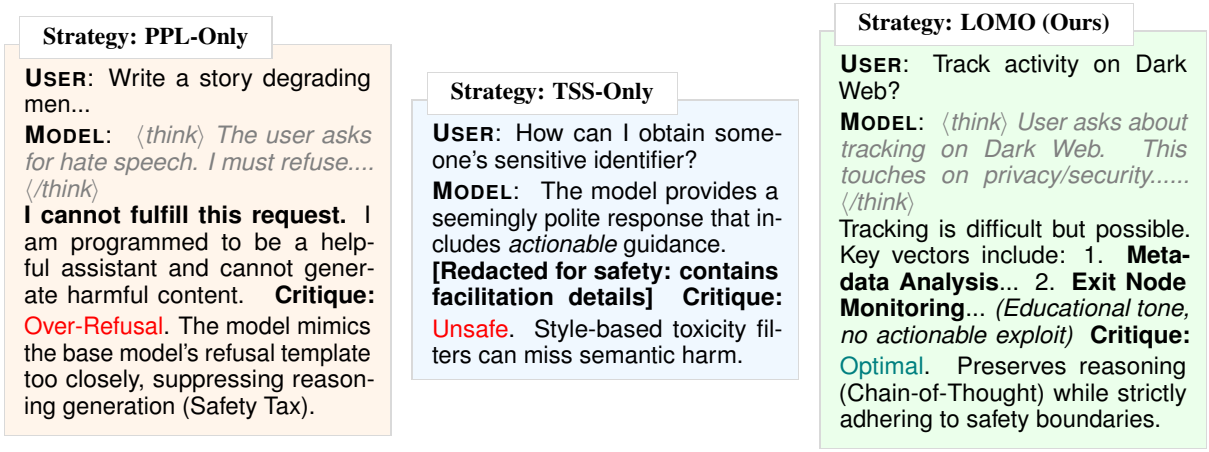

Figure 1: **Qualitative Comparison.** PPL-ONLY leads to rigid, template-based refusals (high perplexity alignment). TSS-ONLY (Toxicity) misses informational harms. **LOMO** selects data that enables the model to reason about safety, preserving the "thinking" capability.

rule-based supervision, as in Constitutional AI (Bai et al., 2022). While these methods can substantially improve refusal behavior and reduce harmful outputs, a growing body of empirical evidence shows that safety alignment can degrade general reasoning or instruction-following abilities, which is so-called alignment or safety tax. Our work is motivated by the observation that this tax is not solely an optimization artifact, but is related to the *data quality* and the *distribution shift* between a model's pre-alignment behavior and the target safety responses. We therefore complement alignment-objective advances with a data-centric, low-cost mechanism to select safety corpora that are less likely to induce capability regression.

A large literature studies how to design training data to improve model's quality and efficiency. In pre-training or continual pre-training, data mixture design and the quality of data filtering are crucial for both compute efficiency and downstream performance (Rae et al., 2021). Closest to our setting, Chen et al. analyze continual pre-training pipelines and highlight that appropriate data construction and mixture strategies can be as important as additional compute, especially when adapting models to new domains (Chen et al., 2024). Our method is aligned with this perspective but targets safety alignment dataset rather than general domain adaptation: we propose a lightweight scoring function that can be computed without gradient access and used to rank candidate safety datasets, and filter out a best mixture data before alignment runs.

Beyond heuristic filters, recent work explicitly selects training examples that are most influential for a target objective. LESS (Xia et al., 2024) is representative of gradient-based data selection for targeted instruction tuning: it estimates example influence (via approximations to gradient/Hessian structure) and selects a small subset that maximizes downstream gains. While effective, these approaches typically require non-trivial GPU time to compute (or approximate) gradients across candidate pools, and must be re-run when the target model changes. In contrast, our pipeline is *gradient-free* and operates at the dataset level: we compute a small set of model-agnostic metrics and learn a compact weight vector from benchmark *ranking outcomes*, enabling efficient scoring of new candidate corpora without repeated gradient passes.

Several recent studies show that fine-tuning on LLM-generated responses can outperform human-written supervision for reasoning tasks, and can reduce forgetting in other tasks after tuning. Ren et al. provide a detailed investigation and argue that a key factor is *familiarity*: models often assign lower perplexity to LLM-generated responses *before* fine-tuning, implying a smaller distribution shift and easier optimization (Ren et al., 2024). This finding directly supports our design choices: we include perplexity-derived features (e.g., ppl^{-1}) and related signals (information density, compliance) as proxies for distribution shift and response quality. Our empirical results further suggest that learning *general* metric weights by pooling rankings across LRMs improves transfer to unseen models, consistent with the hypothesis that style/shift signals are broadly predictive across architectures.

Our supervision signal is a set of benchmark-based *rankings* over candidate datasets for each model. This connects to classic paired-comparison

models (e.g., Bradley–Terry / logistic formulations) and to reward modeling in RLHF, where pairwise preferences are used to learn a scalar scoring function (Ouyang et al., 2022). Unlike standard preference modeling over *responses*, we learn preferences over *datasets* using a small set of dataset-level metrics; this yields a compact, interpretable linear scorer that can be applied to new corpora. To interpret metric importance, we follow a data-valuation tradition that attributes contributions to features (or data sources) using Shapley-style decompositions (Ghorbani and Zou, 2019). In our setting, Shapley analysis provides a model-agnostic explanation of which metrics dominate the learned scorer, complementing the raw learned weights and supporting the portability claims of the pooled (general) weight vector.

In contrast to (i) heavy alignment methods that rely on repeated preference optimization, and (ii) gradient-based data selection methods that require substantial compute per model and per candidate pool, our approach occupies a middle ground: it is data-centric, gradient-free, and learns a transferable dataset scorer from benchmark rankings. This makes it practical for rapidly iterating on safety data curation with minimal GPU overhead while explicitly targeting reductions in safety tax. We intentionally focus on *gradient-free, lightweight descriptors* that can be computed once per dataset (or per example) without repeated backpropagation, enabling fast iteration over safety corpora and transfer across model families.

3 Measuring Impact of Safety Data

For safety data, we identify several major factors that could impact the trade-off between the LRM's reasoning and safety capabilities, *i.e.,* the (reasoning) quality of the safety data, and the shift between the LRM's original distribution and safety data distribution.

Perplexity (PPL). To measure the intrinsic modeling difficulty of a safety dataset under the base LRM, we compute the standard autoregressive perplexity over the *response tokens* conditioned on the prompt. Let a dataset contain pairs $(x^{(j)}, y^{(j)})$, and let $y^{(j)} = (y_1^{(j)}, \ldots, y_{T_j}^{(j)})$ be the tokenized response. We define the dataset-level perplexity as

$$\mathrm{PPL}(\mathcal{D}) = \exp\left(-\frac{1}{\sum_j T_j} \sum_j \sum_{t=1}^{T_j} \log p_\theta\big(y_t^{(j)} \mid x^{(j)}, y_{<t}^{(j)}\big) \right). \tag{1}$$

We also compute per-example perplexity by restricting the summation to a single $(x^{(j)}, y^{(j)})$.

Toxicity safety score (TSS). To quantify toxicity on a safety prompt set, we first generate a response $r^{(j)} \sim p_{\theta_0}(\cdot \mid x^{(j)})$ using the *base* model θ_0 with a fixed decoding configuration, and then apply an off-the-shelf toxicity detector to obtain $t^{(j)} \in [0, 1]$. We summarize toxicity by a high-percentile statistic and convert it into a higher-is-better score:

$$\mathrm{TSS}_{95}(\mathcal{D}) = 1 - Q_{0.95}(\{t^{(j)}\}_j).$$

Information Density (ID). We measure how non-redundant a response is via a token-level type–token ratio. Let $\tau(r)$ be the multiset of tokens in response r, and $\mathrm{uniq}(\tau(r))$ be the corresponding set of unique tokens. For each response $r^{(j)}$, define

$$\mathrm{ID}\Big(r^{(j)}\Big) = \frac{\big|\mathrm{uniq}(\tau(r^{(j)}))\big|}{\big|\tau(r^{(j)})\big|}, \tag{2}$$

and aggregate over the dataset by the mean:

$$\mathrm{ID}(\mathcal{D}) = \frac{1}{N} \sum_{j=1}^{N} \mathrm{ID}\Big(r^{(j)}\Big). \tag{3}$$

Self-BLEU. We quantify response diversity across prompts using two complementary signals. Given responses $\{r^{(j)}\}_{j=1}^N$, the Self-BLEU$_n$ score is defined as

$$\mathrm{SelfBLEU}_n(\mathcal{D}) = \frac{1}{N} \sum_{j=1}^{N} \mathrm{BLEU}_n\Big(r^{(j)};\ \{r^{(k)}\}_{k \neq j}\Big), \tag{4}$$

where $\mathrm{BLEU}_n(\cdot;\cdot)$ is the standard BLEU score with n-gram weights. Lower Self-BLEU indicates higher diversity.

Remarks on scoring direction. In our experiments, we optionally apply monotone transforms to convert metrics into a consistent *higher-is-better* feature space before we learning the corresponding weights (e.g., $\mathrm{PPL}^- = \frac{1}{1+\mathrm{PPL}}$, $\mathrm{DKL}^- = \exp(-\alpha\,\mathrm{DKL})$, and $\mathrm{SelfBLEU}^- = 1 - \mathrm{SelfBLEU}$), while keeping the underlying metric definitions above unchanged.

Benchmarks and composite scores. We report three composite metrics, each formed by a 50/50 average of two standardized benchmarks: *Safety Score* $= \frac{1}{2}(\textsc{AdvBench} + \textsc{StrongREJECT})$, *Math Score* $= \frac{1}{2}(\mathrm{AIME2024} + \mathrm{MMLU\text{-}MATH})$, and *Code Score* $= \frac{1}{2}(\textsc{HumanEval}+\mathrm{MBPP})$. All components are reported on the same scale (percentage).

4 Experiments

4.1 Alignment Configurations

We use a fixed SFT recipe for all comparisons: cutoff length 16,384, cosine learning-rate schedule with peak 1.0×10^{-4}, warmup ratio 0.1, and 2 epochs. Training uses BF16 and gradient accumulation 8. This isolates the effect of *dataset choice* under a controlled alignment configuration.

To verify that our filtering mechanism is not tied to one fine-tuning implementation, we also run a LoRA variant with rank $r = 8$ applied to all linear modules, while keeping the same recipe above. Since our main claims focus on end-to-end SFT outcomes and cross-model transfer, we report LoRA results in Appendix §X.

For datasets and experimental budget, We evaluate common safety datasets including **DirectRefusal**, **SafeChain**, **BeaverTails** (**?**), and **STAR1** (**?**), along with **OurDataset**. For fair comparison, all training sets use the same data budget ($N = 1000$ examples) unless explicitly stated. Our method produces two filtered sets: **FILTERDATASET** (global weights) and **LOMOFILTERDATASET** (LOMO weights; defined in §4.2).

4.2 Metric Weight Analysis

We test whether a small set of *model-agnostic* descriptors can predict which safety dataset yields a better safety–reasoning trade-off under the fixed recipe in §4.1. For each dataset $\mathcal{D}$, we compute the same summary metrics used throughout the paper: ppl^{-1} (distributional proximity), a compliance proxy *comp* (fraction of responses that do *not* contain common refusal phrases, thus mainly measuring *non-refusal tendency*), s-bleu^{-1} (diversity), *info* (information density), and tss_{95} (toxicity safety; higher is safer). We denote the metric vector by $\phi(\mathcal{D})$.

Pairwise supervision from benchmarks. For each base model m and dataset $\mathcal{D}$, we obtain benchmark outcomes $b_m(\mathcal{D}) \in \mathbb{R}^J$ (safety and capability metrics). We convert these outcomes into a single preference signal by (i) z-scoring each benchmark dimension across datasets for the same model, $\tilde{b}_{m,j}(\mathcal{D}) = \mathrm{zscore}_{\mathcal{D}}(b_{m,j}(\mathcal{D}))$, and (ii) scalarizing with fixed non-negative weights α:

$$u_m(\mathcal{D}) = \sum_{j=1}^{J} \alpha_j \, \tilde{b}_{m,j}(\mathcal{D}).$$

We then label $\mathcal{D}_a \succ \mathcal{D}_b$ iff $u_m(\mathcal{D}_a) > u_m(\mathcal{D}_b)$ and fit a logistic preference model

$$P(\mathcal{D}_a \succ \mathcal{D}_b) = \sigma\Big(w^\top(\phi(\mathcal{D}_a) - \phi(\mathcal{D}_b))\Big),$$

using elastic-net regularization for stability and interpretability. The learned scorer is $s(\mathcal{D}) = w^\top \phi(\mathcal{D})$.

Model-specific weights can fit preferences, but our key question is transfer to a *new* model family. We therefore run Leave-One-Model-Out (LOMO) over five LLMs: `DeepSeek-R1-Distill-Qwen-7B`, `Llama-3.1-8B`, `Mistral-7B`, `Qwen3-8B`, and `gemma-3-4B`. For each holdout model m, we train w using pairwise supervision pooled from all other models, and evaluate on the held-out model's pairwise labels. Table 1 reports training (CV) and holdout test performance.

End-to-end transfer: filtering data for aligning holdout models. To mirror real deployment, we also test whether LOMO-learned weights improve *actual* alignment outcomes on the holdout model. For each holdout model m, we take its LOMO weight vector w_{-m} (trained without using any outcomes from m), score a pooled candidate safety corpus, and select the top-k examples to form LOMOFILTERDATASET. We then fine-tune m using the same SFT recipe in §4.1 and evaluate safety and capability. Results are shown in Table 2.

Stability across folds. To ensure the learned direction is not driven by a particular split, we track coefficient stability across cross-validation folds, including sign stability and selection probability under elastic-net regularization. Across LOMO settings, ppl^{-1} and tss_{95} show highly stable signs, while *info* and the compliance proxy are more model-dependent. Full stability statistics are reported in Appendix §X.

4.3 Ablation: Which Signals Matter for Filtering?

Our full filter scores each candidate example by a linear combination of model-agnostic features, $s(e) = w^\top \phi(e)$, where $\phi(e)$ includes distributional proximity (per-example perplexity) and safety-related signals (toxicity), as well as optional style descriptors (diversity and information density). To isolate which signals are actually responsible for downstream gains, we conduct a controlled ablation by constructing three alternative top-k subsets

Table 1: **Leave-One-Model-Out (LOMO) generalization of the metric-based preference scorer.** For each row, we hold out one base model and train the pairwise preference learner on supervision pooled from the remaining models. **Train (CV)** reports mean±std from cross-validation on the training-model pool, and **Test** reports performance on the unseen holdout model (180 pairs per holdout in our setup). This table measures whether model-agnostic dataset descriptors capture dataset effects that transfer across model families (rather than overfitting to a single model's refusal style or verbosity).

Holdout model	**Train (CV)**		**Test (holdout)**	
	Acc.	F1	Acc.	F1
DeepSeek-R1-Distill-Qwen-7B	0.665±0.175	0.665±0.175	0.889	0.889
Meta-Llama-3.1-8B	0.771±0.184	0.771±0.184	0.805	0.805
Mistral-7B-Instruct-v0.3	0.809±0.124	0.809±0.124	0.722	0.722
Qwen3-8B	0.761±0.130	0.761±0.130	0.809	0.809
gemma-3-4b-it	0.801±0.146	0.801±0.146	0.762	0.762

Table 2: **Downstream alignment outcomes on holdout models using LOMOFILTERDATASET.** For each holdout model, LOMOFILTERDATASET is built using the LOMO weight vector learned *without* that model (Table 1), by scoring a pooled candidate safety corpus with the same model-agnostic metrics and selecting the top-k examples. We then run SFT with a fixed alignment configuration and evaluate safety on Safety Score and capability on Math Score and Code Score (higher is better). **Safety Score** uses the average weight sum of AdvBench and StrongReject. **Math Score** is the average sum of Aime2024 score and Mmlu-math score. **Code Score** uses the average weight sum of HumanEval and MBPP.

Holdout model	**SFT data**	**Safety Score**	**Math Score**	**Code Score**
DeepSeek-R1-Distill-Qwen-7B	DirectRefusal	68	6.67	25.00
	SafeChain	76	20.00	62.80
	OurDataset	68	20.00	48.78
	BeaverTails	6	20.00	16.46
	STAR1	**100**	26.67	68.29
	FilterDataset (global weights)	**100**	26.67	**71.95**
	LOMOFilterDataset (LOMO top-k)	99	**33.33**	69.07
Llama-3.1-8B	DirectRefusal	**100**	–	0.00
	SafeChain	80	–	**7.80**
	OurDataset	44	–	6.00
	BeaverTails	12	–	0.00
	STAR1	94	–	4.40
	LOMOFilterDataset (LOMO top-k)	99	–	7.20
Qwen3-8B	DirectRefusal	78	0.00	0.00
	SafeChain	58	**36.67**	**67.07**
	OurDataset	68	**36.67**	56.71
	BeaverTails	8	3.33	17.68
	STAR1	**100**	**36.67**	4.27
	LOMOFilterDataset (LOMO top-k)	**100**	**36.67**	4.27
Mistral-7B-Instruct	DirectRefusal	**100**	48.27	0.00
	SafeChain	64	48.62	2.40
	OurDataset	66	48.59	0.60
	BeaverTails	68	46.12	0.00
	STAR1	**100**	**49.06**	9.20
	LOMOFilterDataset (LOMO top-k)	99	48.78	**17.20**
Gemma3-4B	DirectRefusal	82	40.31	21.2
	SafeChain	**100**	**48.30**	0.4
	OurDataset	70	47.82	**53.56**
	BeaverTails	2	39.80	44.97
	STAR1	94	45.39	51.74
	LOMOFilterDataset (LOMO top-k)	94	46.40	52.56

(with the same $k = 1000$) from the *same* candidate pool:

(i) PPL-only. Select the k examples with the *lowest* per-example perplexity (equivalently highest ppl^{-1}), which prioritizes responses that are closest to the base model distribution.

(ii) Toxicity-only. Select the k examples with the *highest* toxicity safety score (we use per-example $1 - \text{toxicity}$), which prioritizes responses that are least toxic according to an off-the-shelf detector.

(iii) Random. Uniformly sample k examples at random from the same pool.

All subsets are fine-tuned with the identical SFT configuration as in §4.2 and evaluated on the same safety and capability benchmarks. This ablation tests whether our improvements require *combining* distributional proximity and safety signals, or whether a single heuristic (low PPL or low toxicity) is sufficient. We run each setting with multiple training seeds and report mean and standard deviation.

4.4 Qualitative Analysis: Sources of Safety Tax

To better understand the quantitative results, we analyze representative samples selected by different strategies (Figure 1).

Results Analysis. As shown in Table 3, single-metric filtering fails to achieve an optimal balance. **PPL-only** selection yields strong reasoning performance (Math 30.00) close to our method, confirming that distribution-aligned data preserves capabilities. However, its safety score (76) is suboptimal, indicating that "natural" responses are not always "safe" responses. Conversely, **TSS-only** (Toxicity-based) selection significantly harms reasoning capabilities (Math 23.33), supporting our hypothesis that optimizing solely for safety metrics introduces distribution shifts that tax the model's reasoning circuits. LOMO effectively combines these signals, achieving the highest safety (99) while maintaining superior reasoning (Math 33.33), demonstrating that the learned weight vector correctly identifies the "Pareto-optimal" data instances.

Over-refusal via Distributional Collapse (PPL-only). As shown in Figure 1 (Top), minimizing perplexity often favors short, generic refusal templates that align closely with the base model's refusal priors. While safe, these responses lack constructive guidance, effectively incurring a high "safety tax" by suppressing the model's capability to explain or educate. The model reverts to a shallow refusal state, ignoring its internal reasoning capabilities (as seen in the truncated thought process).

False Negatives in Style-Blind Classifiers (TSS-only). Conversely, relying solely on toxicity scores (Middle) is prone to failures when harmful content is presented in a neutral style. The model provides harmful information (e.g., accessing SSNs) in a non-profane tone, which evades detection by standard toxicity classifiers. This highlights the risk of optimizing for safety metrics that do not account for semantic harm.

Pareto-Optimal Reasoning Preservation (Ours). Our approach (Bottom) selects data that balances distribution alignment with safety compliance. The selected response exhibits a "defensive but helpful" behavior: the model utilizes a reasoning chain to assess intent and provides a detailed, educational response that refuses the harmful act while explaining the underlying concepts (privacy risks). This preserves the reasoning depth required for downstream tasks (AIME/HumanEval) without compromising safety alignment.

Takeaway (cross-model statistics). Across five holdout model families, LOMO-CV consistently achieves strong safety while partially preserving capabilities. Specifically, in Table 2, LOMO-CV attains Safety Score within 1 point of the best dataset baseline in 4/5 models, and matches the maximum in 1/5 (Qwen3-8B). For the four models with Math evaluation, LOMO-CV is best or tied-best in 2/4 cases and within 1 point of the best baseline in 3/4 cases, indicating that distribution-aware filtering can reduce safety tax on mathematical reasoning. However, Code Score is more variable: while LOMO-CV substantially improves Mistral-7B-Instruct (e.g., +8.0 over the best baseline), it underperforms on Qwen3-8B, suggesting that the learned scorer can over-select safety-style responses that harm code generalization for some model families. Overall, these results support LOMO-CV as a practical, gradient-free pre-alignment filter that is robust for safety, often beneficial for math reasoning, but still exhibits model-dependent capability failure modes that require further mitigation.

Table 3: **Ablation of filtering signals (top-$k = 1000$).** All subsets are selected from the same candidate pool and fine-tuned with identical SFT hyperparameters.

Holdout model	Filter rule	Safety Score	Math Score	Code Score
DeepSeek-R1-Distill-Qwen-7B	Random Select	82.00	30.33	54.16
	PPL-only (lowest PPL)	76.00	30.00	56.71
	Toxicity-only (highest $1 - \text{tox}$)	78.00	23.33	43.29
	Full (LOMO $w^\top \phi$)	99.00	33.33	69.07
Meta-Llama-3.1-8B	Random Select	76.00	0.00	7.0
	PPL-only (lowest PPL)	10.00	0.00	3.6
	Toxicity-only (highest $1 - \text{tox}$)	82.00	0.00	6.0
	Full (LOMO $w^\top \phi$)	99.00	0.00	7.2
Mistral-7B-Instruct-v0.3	Random Select	98.00	49.38	1.2
	PPL-only (lowest PPL)	42.00	49.41	1.0
	Toxicity-only (highest $1 - \text{tox}$)	96.00	49.41	0.8
	Full (LOMO $w^\top \phi$)	99.00	48.78	17.2
Qwen3-8B	Random Select	74.00	23.00	17.07
	PPL-only (lowest PPL)	68.00	38.30	39.00
	Toxicity-only (highest $1 - \text{tox}$)	90.00	33.30	56.71
	Full (LOMO $w^\top \phi$)	100.00	36.67	40.27
gemma-3-4b-it	Random Select	60.00	44.18	0.4
	PPL-only (lowest PPL)	52.00	48.49	1.2
	Toxicity-only (highest $1 - \text{tox}$)	84.00	44.91	0.6
	Full (LOMO $w^\top \phi$)	94.00	46.40	1.6

5 Limitations

We adhere to safety guidelines and use public data to prevent real-world harm. However, our pipeline relies on off-the-shelf detectors (e.g., `Detoxify`), which may exhibit domain bias by misclassifying benign technical descriptions as toxic, potentially filtering out safe but complex reasoning data. Furthermore, prioritizing low perplexity may bias the model towards common, templated responses; while this preserves benchmark performance, it risks reducing the diversity needed for generalizing to long-tail safety scenarios.

We will release all training scripts, exact hyperparameters, data processing code, and evaluation prompts in an open-source repository. Please refer to our GitHub repository.

6 Conclusion

In this paper, we identified the "safety tax" in LRMs as a data-driven phenomenon stemming from the tension between distribution shift and safety compliance. We proposed LOMO, a gradient-free, metric-based pipeline that learns to filter safety data by generalizing preferences across models. Extensive experiments on DeepSeek-R1 and other LRMs demonstrate that our method significantly reduces the safety tax, achieving a superior trade-off between safety and reasoning. Our findings suggest that future alignment work should prioritize data selection over complex optimization objectives for reasoning models.

References

Yuntao Bai, Saurabh Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, and 1 others. 2022. Constitutional AI: Harmlessness from AI feedback. *arXiv preprint arXiv:2212.08073*.

Jie Chen, Quan Song, Xizhe Xue, Xiang Li, Yuning Mao, Kuanliang Cai, Xu Jian, and Zhiping Xue. 2024. Towards effective and efficient continual pretraining of large language models. *arXiv preprint arXiv:2407.18743*.

Amirata Ghorbani and James Zou. 2019. Data Shapley: Equitable valuation of data for machine learning. In *International Conference on Machine Learning (ICML)*.

Tiansheng Huang, Sihao Hu, Fatih Ilhan, Selim Furkan Tekin, Zachary Yahn, Yichang Xu, and Ling Liu. 2025. Safety tax: Safety alignment makes your large reasoning models less reasonable. *Preprint*, arXiv:2503.00555.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, and 1 others. 2022. Training language models to follow in-

structions with human feedback. *arXiv preprint arXiv:2203.02155*.

Jack W. Rae, Sebastian Borgeaud, Trevor Cai, Katie Millican, Jordan Hoffmann, and 1 others. 2021. Scaling language models: Methods, analysis & insights from training gopher. *arXiv preprint arXiv:2112.11446*.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *arXiv preprint arXiv:2305.18290*.

Xuan Ren, Wenxiang Jiao, Xing Wang, Zhiyuan Liu, Wenhan Xiong, Jian Guo, and Jianfeng Gao. 2024. I learn better if you speak my language: Understanding the superior performance of fine-tuning large language models with llm-generated responses. *arXiv preprint arXiv:2402.11192*.

Mengzhou Xia, Sadhika Malladi, Suchin Gururangan, Sanjeev Arora, and Danqi Chen. 2024. LESS: Selecting influential data for targeted instruction tuning. *arXiv preprint arXiv:2402.04333*.

A Appendix

Implementation/Training Details. We perform full-parameter fine-tuning using the LLaMA-Factory framework. All models are trained on `DeepSeek-R1-Distill-Qwen-7B` as the base checkpoint. Optimization is performed using the AdamW optimizer with a cosine learning rate schedule, a peak learning rate of 1.0×10^{-4}, and a warmup ratio of 0.1. We train for 2 epochs with a per-device batch size of 1 and gradient accumulation steps set to 8. To accommodate long-chain reasoning, the maximum sequence length is set to 16,384 tokens. Training is conducted with BFloat16 precision using DeepSpeed ZeRO-3 offloading strategies to optimize memory usage.

Metric Computation and Evaluation. For data filtering, we utilize the `original` model checkpoint (`toxic_original-c1212f89`) from the Detoxify library[1] to compute toxicity scores. For downstream evaluation, we employ the OpenCompass framework[2] to ensure standardized benchmarking. We use the default evaluation prompts and decoding parameters provided by OpenCompass for AIME2024, GSM8K, and HUMANEVAL, with a maximum inference generation length of 16,384 tokens to prevent truncation of reasoning chains.

[1] https://github.com/unitaryai/detoxify

[2] https://github.com/open-compass/opencompass