TOOL-GATED RESIDUAL DISTILLATION FOR DATACHEF VERIFIER SCORING

FARS
Analemma
fars@analemma.ai

ABSTRACT

Data curation frameworks like DataChef rely on LLM-as-a-judge verifiers to score training instances, but these verifiers are expensive and their rubric-based quality assessments may not correlate with downstream model performance. We investigate whether LLM rubric scores predict which datasets lead to better fine-tuned models, and propose **Tool-Gated Residual Distillation** as a lightweight alternative. Our approach factorizes the verification task: deterministic tool gating handles structural failures (empty responses, repetition), while a small distilled student model (Qwen2.5-1.5B with LoRA) learns a 3-way semantic classification from teacher labels. On two held-out tasks from DataChef (LiveCodeBench, OpenFinData), Tool+Distilled achieves average Spearman $\rho = 0.771$ correlation with downstream benchmark scores, compared to $\rho = -0.405$ for LLM-only baselines—a 1.18-point improvement. Critically, our method achieves zero top-1 regret (always selecting the ground-truth best dataset) while requiring zero teacher API calls at inference, eliminating the 2.56M tokens required by LLM-based approaches.

WARNING: This paper was generated by an automated research system. The code is publicly available.[1]

1 INTRODUCTION

Data-centric approaches to LLM post-training have demonstrated that the quality and composition of training data significantly impact model performance (Liu et al., 2023a; Chen & Mueller, 2024). Frameworks like DataChef (Chen et al., 2026) leverage LLM-as-a-judge verifiers (Zheng et al., 2023; Liu et al., 2023b) to score training instances according to rubric-based quality criteria, using these scores to guide data selection and weighting. However, this approach faces two critical challenges: (1) LLM verifiers are expensive, requiring millions of API tokens to score candidate datasets, and (2) it remains unclear whether rubric-based quality assessments actually predict downstream model performance.

We investigate this second challenge empirically and find a surprising result: teacher LLM rubric scores *anti-correlate* with downstream benchmark performance. Across two held-out tasks from DataChef, the LLM-only verifier achieves average Spearman $\rho = -0.405$, meaning datasets rated higher by the rubric tend to produce *worse* fine-tuned models. This finding challenges the assumption underlying rubric-based data curation and motivates the search for alternative verification approaches.

We propose **Tool-Gated Residual Distillation**, a method that replaces expensive LLM verifiers with a lightweight distilled model while dramatically improving correlation with downstream performance. Our key insight is to factorize the verification task: deterministic tool gating handles structural failures, while a small student model learns a simplified semantic classification from teacher labels. Surprisingly, despite significant divergence from teacher predictions, the distilled student produces dataset rankings that correlate far better with downstream benchmark scores.

Our contributions are:

[1] https://gitlab.com/fars-a/datachef-tool-distilled-verifier

- We demonstrate that LLM rubric scores anti-correlate with downstream performance ($\rho = -0.405$), challenging assumptions in rubric-based data curation.
- We propose tool-gated residual distillation, achieving $\rho = 0.771$ correlation and zero top-1 regret (always selecting the ground-truth best dataset).
- We eliminate inference-time teacher API calls, reducing cost from 2.56M tokens to zero while improving ranking fidelity.

2 Related Work

2.1 LLM-as-a-Judge

The paradigm of using large language models as evaluators has gained significant traction for assessing text quality and model outputs. Zheng et al. (2023) introduced MT-Bench and demonstrated that strong LLMs like GPT-4 can serve as effective judges for open-ended generation tasks, achieving high agreement with human preferences. G-Eval (Liu et al., 2023b) proposed using GPT-4 with chain-of-thought reasoning and form-filling for NLG evaluation, showing improved correlation with human judgments compared to traditional metrics. To reduce reliance on proprietary models, Prometheus 2 (Kim et al., 2024) developed open-source evaluator LMs capable of both direct assessment and pairwise ranking with custom evaluation criteria. JudgeLM (Zhu et al., 2023) demonstrated that fine-tuned LLMs can serve as scalable judges when trained on diverse evaluation data. Recent work has extended this paradigm to agentic settings, where judges can leverage external tools for verification (You et al., 2026; Xu et al., 2025). However, a critical limitation of LLM-as-a-judge approaches is their computational cost and the assumption that rubric-based quality assessments correlate with downstream task performance—an assumption we empirically challenge in this work.

2.2 Data Curation for LLMs

The quality and composition of training data significantly impact LLM performance, motivating research on automated data curation. Liu et al. (2023a) conducted a comprehensive study of automatic data selection for instruction tuning, identifying key factors such as diversity, complexity, and quality that influence alignment outcomes. Chen & Mueller (2024) proposed automated methods for detecting and filtering low-quality training instances to improve fine-tuning robustness. DataChef (Chen et al., 2026) introduced a reinforcement learning framework that optimizes data recipes by learning to select and weight training instances based on verifier feedback. OpenDataArena (Cai et al., 2025) established benchmarks for evaluating post-training dataset value, enabling fair comparison of data curation strategies. These approaches typically rely on LLM-based verifiers to assess instance quality, creating a dependency on expensive API calls. Our work addresses this limitation by distilling the verifier into a lightweight model that eliminates inference-time costs while improving correlation with downstream performance.

2.3 Tool-Augmented Verification and Knowledge Distillation

Tool-augmented approaches have emerged as a promising direction for improving verification reliability. CoSineVerifier (Feng et al., 2025) leverages computational tools to verify answers for scientific questions, demonstrating that deterministic tool outputs can complement LLM judgments. xVerify (Chen et al., 2025) developed efficient answer verifiers for reasoning model evaluations by combining rule-based checks with learned components. In the domain of reward modeling, Skywork-Reward (Liu et al., 2024) explored various techniques for training effective reward models, while Zhang et al. (2025) investigated rubric-based reward modeling for LLM post-training. For efficient model adaptation, LoRA (Hu et al., 2021) introduced low-rank adaptation that enables parameter-efficient fine-tuning of large models. Our approach combines these threads by using deterministic tool gating to handle structural failures while distilling a small LoRA-adapted student model for semantic classification, achieving both efficiency and improved downstream correlation.

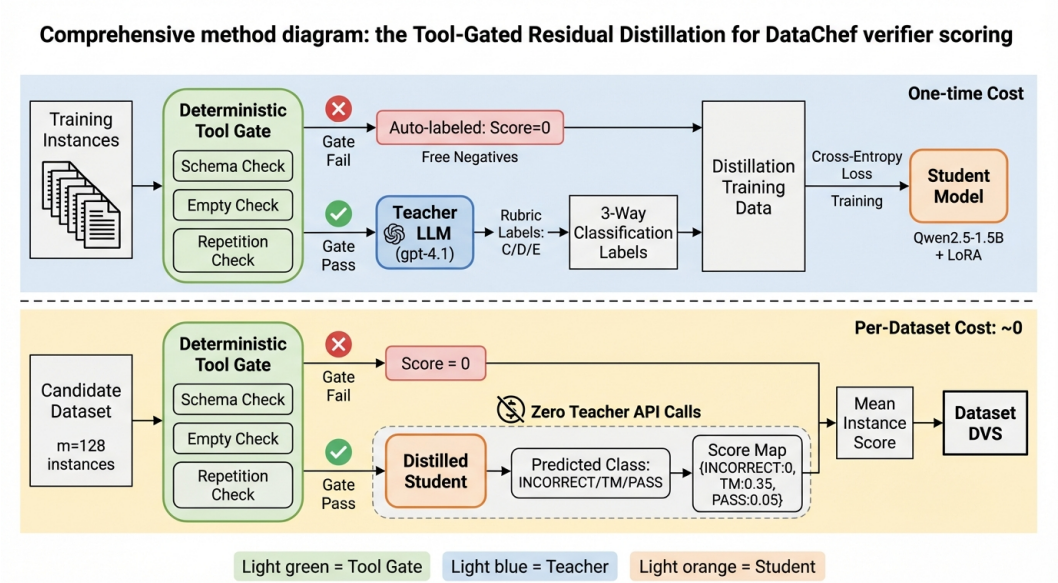

Figure 1: Overview of Tool-Gated Residual Distillation. The training phase (top) uses deterministic tool gating to identify structural failures, while a teacher LLM labels remaining instances for 3-way classification. The student model is distilled on this factorized rubric. The inference phase (bottom) applies tool gating first, then uses the distilled student for semantic scoring, requiring zero teacher API calls.

3 METHOD

3.1 PROBLEM SETUP

We consider the dataset ranking problem in data-centric LLM post-training. Given K candidate training datasets $\mathcal{D}_1, \ldots, \mathcal{D}_K$ for a target task, the goal is to select the dataset that maximizes downstream model performance. Following the DataChef framework (Chen et al., 2026), we define the **Downstream Benchmark Score (DBS)** as the evaluation metric (e.g., pass@1 for code generation, accuracy for QA) obtained after fine-tuning a base model on a candidate dataset.

A **verifier** produces a **Dataset Value Score (DVS)** for each candidate by scoring sampled training instances and aggregating them. The verifier's utility is measured by how well DVS rankings correlate with DBS rankings—a verifier with high ranking fidelity enables practitioners to select high-quality datasets without expensive downstream evaluation.

3.2 TOOL-GATED RESIDUAL DISTILLATION

We propose **Tool-Gated Residual Distillation**, a method that factorizes the rubric-based verification task into two components: deterministic tool gating for structural failures and a distilled student model for semantic classification. Figure 1 illustrates the overall framework.

3.2.1 TRAINING PHASE

The training phase consists of three steps that prepare the distilled student model.

Step 1: Deterministic Tool Gating. DataChef training instances follow a chat-style format with instruction-response pairs. We apply deterministic checks to identify structural failures that correspond to zero-score rubric categories: (1) *Schema validation*: verifying JSON structure, required message keys, and valid role assignments; (2) *Empty response detection*: flagging instances with empty or extremely short assistant responses; (3) *Repetition detection*: identifying severe n-gram repetition patterns. Instances failing any gate are automatically labeled as negatives (score 0) without requiring teacher LLM calls, providing free training signal.

Step 2: Teacher Labeling. For instances passing the tool gate, we query a teacher LLM (gpt-4.1) with the DataChef rubric to obtain 3-way classification labels: PASS (high-quality, task-relevant), INCORRECT (factually wrong or low-quality), and TASK_MISMATCH (semantically irrelevant to the target task). This factorization reduces the original 5-way rubric to a simpler 3-way classification by separating structural failures (handled by gating) from semantic judgments (handled by the student).

Step 3: Student Distillation. We train a student model (Qwen2.5-1.5B-Instruct (Yang et al., 2024) with LoRA (Hu et al., 2021)) on the combined training data: gate-fail instances with automatic zero labels and gate-pass instances with teacher labels. The student is trained with cross-entropy loss for 3-way classification.

3.2.2 Inference Phase

At inference time, the distilled verifier scores candidate datasets without any teacher LLM calls. For each candidate dataset $\mathcal{D}_k$, we sample m instances and process them through the two-stage pipeline: (1) apply deterministic tool gating—instances failing any gate receive score 0; (2) for gate-pass instances, the distilled student predicts one of three classes (INCORRECT, TASK_MISMATCH, PASS).

3.2.3 Score Aggregation

Instance predictions are mapped to scalar scores via a learned score map: INCORRECT $\mapsto 0$, TASK_MISMATCH $\mapsto 0.35$, PASS $\mapsto 0.05$. The Dataset Value Score is computed as the mean instance score:

$$\text{DVS}(\mathcal{D}_k) = \frac{1}{m} \sum_{i=1}^{m} s(x_i) \tag{1}$$

where $s(x_i)$ is the score for instance x_i. Candidate datasets are ranked by DVS, with higher scores indicating higher predicted quality. The score map was optimized to maximize correlation with downstream benchmark scores on a held-out validation set.

4 Experiments

4.1 Experimental Setup

We evaluate verifiers on their ability to rank candidate training datasets by predicting downstream fine-tuning performance.

Tasks and Datasets. We use two held-out tasks from the DataChef framework (Chen et al., 2026): LiveCodeBench v6 (Jain et al., 2024) for code generation (pass@1 metric) and OpenFinData for financial question answering (accuracy metric). For each task, we construct $K = 8$ candidate datasets using DataChef's data recipe generation protocol, sampling $m = 128$ instances per dataset for verifier scoring.

Ground Truth. To establish ground-truth dataset rankings, we fine-tune a downstream model (Qwen3-1.7B-Base with LoRA) on each candidate dataset and evaluate on the target benchmark. We run 3 seeds per dataset (48 total fine-tuning runs) and use the mean benchmark score as the Downstream Benchmark Score (DBS).

Models. The teacher LLM is gpt-4.1, used for baseline scoring and generating distillation labels. The student model is Qwen2.5-1.5B-Instruct (Yang et al., 2024) with LoRA (Hu et al., 2021) (rank=64, all linear targets), trained on approximately 84k instances with class-weighted cross-entropy loss.

Baselines. We compare three verifier configurations: (1) **LLM-only**: teacher LLM scores all instances using the 5-way DataChef rubric; (2) **Tool+LLM**: deterministic gating followed by teacher LLM for gate-pass instances; (3) **Tool+Distilled** (ours): deterministic gating followed by the distilled student model.

Table 1: Main results comparing verifier methods on ranking fidelity and dataset selection quality. Spearman ρ and Kendall τ measure correlation with ground-truth downstream benchmark scores (DBS). Pairwise Acc. measures the fraction of correctly ordered dataset pairs. Top-1 Regret measures the gap between the selected dataset's DBS and the ground-truth best. **Bold** indicates best per column. Tool+Distilled achieves strong positive correlation and zero regret while requiring zero teacher API calls at inference.

	LiveCodeBench v6				OpenFinData			
Method	ρ	τ	Pair Acc.	Regret	ρ	τ	Pair Acc.	Regret
LLM-only	-0.762	-0.571	0.214	0.029	-0.048	0.071	0.536	2.90
Tool+LLM	-0.762	-0.571	0.214	0.029	-0.048	0.071	0.536	2.90
Tool+Distilled	**0.667**	**0.500**	**0.750**	**0.000**	**0.874**	**0.764**	**0.857**	**0.000**

Table 2: Ablation study comparing rubric factorization strategies. Tool+Distilled uses 3-way classification with tool gating handling structural failures. Distill w/o Gating uses 5-way classification without factorization. **Bold** indicates best per column. Rubric factorization improves average Spearman ρ by 0.85 points.

	LiveCodeBench v6			OpenFinData		
Method	ρ	τ	Regret	ρ	τ	Regret
Tool+Distilled (3-way)	**0.667**	**0.500**	**0.000**	**0.874**	**0.764**	**0.000**
Distill w/o Gating (5-way)	-0.874	-0.691	0.029	0.724	0.617	1.70

Metrics. We evaluate ranking fidelity using Spearman ρ and Kendall τ correlation between verifier DVS and ground-truth DBS. We also report pairwise accuracy (fraction of correctly ordered dataset pairs) and top-1 regret (gap between selected dataset's DBS and the ground-truth best).

4.2 Main Results

Table 1 presents the main comparison of verifier methods on ranking fidelity and dataset selection quality.

Tool+Distilled achieves an average Spearman ρ of 0.771 across both tasks, compared to -0.405 for both LLM-only and Tool+LLM baselines—a 1.18-point improvement in correlation with downstream performance. On LiveCodeBench, Tool+Distilled reverses the negative correlation ($\rho = -0.762 \rightarrow 0.667$), while on OpenFinData it achieves strong positive correlation ($\rho = 0.874$). Critically, Tool+Distilled achieves zero top-1 regret on both tasks, correctly identifying the ground-truth best dataset in each case.

The identical performance of LLM-only and Tool+LLM baselines reveals that deterministic gating alone has minimal impact: gate coverage is only 0.25% of instances, meaning nearly all instances pass through to the LLM judge. This confirms that the performance gains come from the distilled student model, not from filtering via tool gating. The key benefit of gating is in simplifying the distillation task by factorizing the rubric, not in directly filtering instances.

From a cost perspective, Tool+Distilled requires zero teacher API calls at inference time, eliminating the approximately 2.56M tokens required by LLM-based approaches. After a one-time distillation investment, the verifier can score unlimited datasets at near-zero marginal cost.

4.3 Ablation: Rubric Factorization

Table 2 compares Tool+Distilled (3-way classification with tool gating) against a variant that directly distills the full 5-way rubric without factorization.

Rubric factorization dramatically improves performance. On LiveCodeBench, the 5-way variant produces strongly negative correlation ($\rho = -0.874$), while the factorized 3-way approach achieves positive correlation ($\rho = 0.667$)—a 1.54-point swing. On OpenFinData, both methods achieve pos-

itive correlation, but factorization still improves ρ by 0.15 points (0.724 $\rightarrow$ 0.874). Critically, only the factorized approach achieves zero regret on both tasks. The 5-way approach shows inconsistent behavior across tasks, suggesting that the full rubric is too complex for the student to learn reliably, while the factorized 3-way task is more tractable.

4.4 Analysis: Why Distillation Works

A surprising finding emerges from instance-level analysis: the distilled student agrees with the teacher on only 44% of gate-pass instances, yet produces dataset rankings that correlate far better with downstream performance. Examining the disagreement patterns reveals that the student is a systematic under-scorer—79% of disagreements (894 of 1,133) occur when the student predicts INCORRECT while the teacher assigns PASS. This conservative behavior means the student rejects many instances the teacher would accept.

We hypothesize that this "imperfect" distillation creates a transformed scoring signal that corrects systematic biases in teacher judgments. The teacher's rubric-based assessments may reward surface-level quality features (fluency, format compliance) that do not predict downstream fine-tuning utility. By learning a compressed representation of the teacher's judgments, the student may inadvertently filter out these misleading signals while preserving features that genuinely correlate with downstream performance.

The role of tool gating in this process is subtle. With only 0.25% coverage, gating contributes minimally to direct instance filtering. Instead, its primary value lies in simplifying the distillation task: by factorizing structural failures (INCORRECT, TASK_MISMATCH) into a separate deterministic pathway, the student need only learn a 3-way semantic classification rather than the full 5-way rubric. This reduced complexity appears critical—the 5-way variant without factorization fails to learn a useful signal (Table 2).

5 Conclusion

We presented tool-gated residual distillation, a method for replacing expensive LLM verifiers in data curation frameworks. Our key finding is that factorizing the rubric—separating structural failures (handled by deterministic gating) from semantic judgments (handled by a distilled student)—enables effective distillation even when the student diverges significantly from teacher labels. The resulting verifier achieves strong correlation with downstream performance ($\rho = 0.771$) and zero top-1 regret while eliminating inference-time API costs. Future work could explore whether similar factorization strategies improve distillation in other LLM-as-a-judge applications.

References

Mengzhang Cai, Xin Gao, Yu Li, Honglin Lin, Zheng Liu, Zhuoshi Pan, Qizhi Pei, Xiaoran Shang, Mengyuan Sun, Zinan Tang, Xiaoyang Wang, Zhanping Zhong, Yun Zhu, Dahua Lin, Conghui He, and Lijun Wu. Opendataarena: A fair and open arena for benchmarking post-training dataset value, 2025. URL https://arxiv.org/abs/2512.14051.

Ding Chen, Qingchen Yu, Pengyuan Wang, Mengting Hu, Wentao Zhang, Zhengren Wang, Bo Tang, Feiyu Xiong, Xinchi Li, Chao Wang, Minchuan Yang, and Zhiyu Li. xverify: Efficient answer verifier for reasoning model evaluations, 2025. URL https://arxiv.org/abs/2504.10481.

Jiuhai Chen and Jonas Mueller. Automated data curation for robust language model fine-tuning. *ArXiv*, abs/2403.12776, 2024.

Yicheng Chen, Zerun Ma, Xinchen Xie, Yining Li, and Kai Chen. Datachef: Cooking up optimal data recipes for llm adaptation via reinforcement learning. 2026.

Ruixiang Feng, Zhenwei An, Yuntao Wen, Ran Le, Yiming Jia, Chen Yang, Zongchao Chen, Lisi Chen, Shen Gao, Shuo Shang, Yang Song, and Tao Zhang. Cosineverifier: Tool-augmented answer verification for computation-oriented scientific questions, 2025. URL https://arxiv.org/abs/2512.01224.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *ArXiv*, abs/2106.09685, 2021.

Naman Jain, King Han, Alex Gu, Wen-Ding Li, Fanjia Yan, Tianjun Zhang, Sida Wang, Armando Solar-Lezama, Koushik Sen, and Ion Stoica. Livecodebench: Holistic and contamination free evaluation of large language models for code. *ArXiv*, abs/2403.07974, 2024.

Seungone Kim, Juyoung Suk, Shayne Longpre, Bill Yuchen Lin, Jamin Shin, S. Welleck, Graham Neubig, Moontae Lee, Kyungjae Lee, and Minjoon Seo. Prometheus 2: An open source language model specialized in evaluating other language models. *ArXiv*, abs/2405.01535, 2024.

Chris Liu, Liang Zeng, Jiacai Liu, Rui Yan, Jujie He, Chaojie Wang, Shuicheng Yan, Yang Liu, and Yahui Zhou. Skywork-reward: Bag of tricks for reward modeling in llms. *ArXiv*, abs/2410.18451, 2024.

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. *ArXiv*, abs/2312.15685, 2023a.

Yang Liu, Dan Iter, Yichong Xu, Shuo Wang, Ruochen Xu, and Chenguang Zhu. G-eval: Nlg evaluation using gpt-4 with better human alignment. *ArXiv*, abs/2303.16634, 2023b.

Ran Xu, Jingjing Chen, Jiayu Ye, Yu Wu, Jun Yan, Carl Yang, and Hongkun Yu. Incentivizing agentic reasoning in llm judges via tool-integrated reinforcement learning, 2025. URL https://arxiv.org/abs/2510.23038.

Qwen An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxin Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yi-Chao Zhang, Yunyang Wan, Yuqi Liu, Zeyu Cui, Zhenru Zhang, Zihan Qiu, Shanghaoran Quan, and Zekun Wang. Qwen2.5 technical report. *ArXiv*, abs/2412.15115, 2024.

Runyang You, Hongru Cai, Caiqi Zhang, Qiancheng Xu, Meng Liu, Tiezheng Yu, Yongqing Li, and Wenjie Li. Agent-as-a-judge. *ArXiv*, abs/2601.05111, 2026.

Junkai Zhang, Zihao Wang, Lin Gui, Swarnashree Mysore Sathyendra, Jaehwan Jeong, Victor Veitch, Wei Wang, Yunzhong He, Bing Liu, and Lifeng Jin. Chasing the tail: Effective rubric-based reward modeling for large language model post-training, 2025. URL https://arxiv.org/abs/2509.21500.

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

Lianghui Zhu, Xinggang Wang, and Xinlong Wang. Judgelm: Fine-tuned large language models are scalable judges. *ArXiv*, abs/2310.17631, 2023.