TANGENTMIX: RECOVERING SPECIALIZED LANGUAGE MODELS THROUGH STATE-CONDITIONED DATA ALLOCATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Specialized language models can acquire strong domain capabilities while developing uneven regressions across others. In on-policy distillation (OPD), controlled interventions show that source exposure can harm its matching capability and the same mixture can reverse its effect as the specialist evolves. These failures expose a mismatch between common data-value signals and the multi-capability effect of the complete update received by the current learner. We propose **TangentMix**, a state-conditioned data-allocation method that jointly adapts source exposure and complete recovery–preservation update selection. TangentMix evaluates candidate updates from the current model and optimizer state, commits the selected update, and uses independently measured post-selection utility to update future source allocation. Across four recovery settings spanning 1.7B and 4B students and a broad capability suite covering world knowledge, compositional reasoning, multilingual comprehension, mathematics, code generation, and instruction following, TangentMix improves 92.9% of evaluated capability axes over their specialist initialization. On the three 4B specialists, it closes 79.1% of the aggregate recovery gap and achieves the highest Macro-7 in every profile; across all four settings, it raises the average Macro-7 by 4.2% relative to standard OPD. It further improves competition-level AIME24/25 reasoning by 1.88 points over OPD and achieves 2× data efficiency over fixed-mixture MOPD. Code are available at. [1]

1 INTRODUCTION

Modern LLM post-training increasingly proceeds through specialized or staged pipelines that emphasize different domains and capabilities, *e.g.*,, DeepSeek-V4 (Xu et al., 2026a) and MiMo-V2-Flash (Xiao et al., 2026). The resulting checkpoints are often useful but uneven: specialization may create new strengths while weakening capabilities acquired earlier, so later stages must recover regressions without discarding what the current checkpoint already does well (Chen et al., 2026; Yang et al., 2026; Harmon et al., 2026). Their acquired strengths make them valuable starting points, while their weakened capabilities define the next recovery target. This raises a practical question: under a finite training budget, how should training data be allocated to recover missing capabilities while preserving the strengths already present in the specialist?

A natural response is to allocate data by capability. MOPD uses fixed domain proportions, DoReMi and DoGE optimize data mixtures, and D^3-MOPD adapts exposure from domain gaps and progress (Ma et al., 2026; Xie et al., 2023a; Fan et al., 2024; Sun et al., 2026). One would therefore increase exposure to a source when its associated capability has substantial recovery headroom, and reduce that exposure as the capability improves. The usefulness of such a strategy rests on a seemingly reasonable correspondence: training on a capability-associated source should preferentially improve that capability in the receiving student.

Our diagnostic analysis reveals that this correspondence can fail under OPD (Agarwal et al., 2024). A mixture that assigns 90% of sampling mass to code reduces coding ability by 27.0% while improving mathematics by 40.7% (Figure 1a). Thus, greater exposure to a source does not necessarily

[1] https://anonymous.4open.science/r/TangentMix-Release-DA1F

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

improve its matching capability. Even a mixture that works well initially can become harmful as the student evolves. Applying the same Balanced mixture for 50 additional updates improves average performance across five capabilities by 7.9% from the 20-update checkpoint, reduces it by 12.2% from the 100-update checkpoint, and improves it again by 6.2% from the 200-update checkpoint (Figure 1b). Static example-level signals do not resolve this mismatch either: importance-resampling strategy DSIR (Xie et al., 2023b) provides no overall advantage over random selection, and three further criteria built on response, difficulty, and gradient signals fail the same way (Figure 1d). Together, these results show that source identity, fixed mixture proportions, and example scores can diverge from the multi-capability effect of the update received by the current learner.

This mismatch arises naturally in OPD because the value of a data decision changes with the receiving learner. As training proceeds, capability gaps and student responses evolve, so the same prompts can induce different teacher corrections (Agarwal et al., 2024). Consequently, a data allocation that is useful at one learner state need not remain useful after the learner moves. Dynamic allocation and gradient-based selection provide useful signals, but our question is whether the complete OPD update induced by a data decision improves the current recovery–preservation trade-off (Sun et al., 2026; Xia et al., 2024).

We propose **TangentMix** to address this decision mismatch. Rather than valuing a source or example in isolation, TangentMix evaluates the complete recovery–preservation update that the data would induce at the current learner. It forms a small set of candidate updates and compares them through counterfactual optimizer steps from the same saved model–optimizer state. A disjoint calibration panel then measures the selected update against a source-conditioned reference, and this feedback adapts subsequent source exposure. We summarize contributions as follows:

- **Observation of failure modes in conventional data allocation.** We uncover three failure modes through controlled OPD interventions: code-heavy data can damage code ability, a beneficial mixture can become harmful as the specialist evolves, and example-level signals need not improve overall recovery. Our analysis identifies their common decision mismatch: source and sample signals describe data before training, while recovery depends on the coupled capability changes induced by its complete update at the current specialist.
- **Method and analysis.** We develop TangentMix to couple source allocation with complete recovery–preservation update selection at the current model and optimizer state. Independent post-selection feedback is then used to update subsequent source allocation, while our analysis characterizes the local value of complete updates and the reliability of finite calibration evidence.
- **Evaluation.** Across four recovery settings spanning 1.7B and 4B students, TangentMix improves 92.9% of evaluated capability axes. On the three 4B specialists, it closes 79.1% of the aggregate recovery gap and achieves the highest Macro-7 in every profile. Across all four settings, it raises the average Macro-7 by 4.2% relative to standard OPD. TangentMix further improves competition-level AIME24/25 reasoning by 1.88 points over OPD and achieves 2× data efficiency over fixed-mixture MOPD.

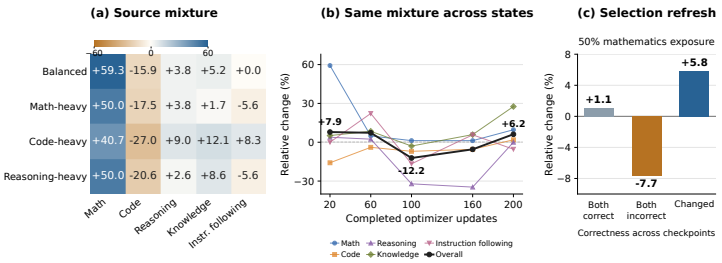
2 Preliminaries and Problem Formulation

2.1 Recovery State and OPD Update

We follow the OPD/MOPD training setup and the specialist-recovery setting used in prior work (Agarwal et al., 2024; Ma et al., 2026; Chen et al., 2026). Training starts from a specialized checkpoint p_{θ_I}. At update t, the trainable student is p_{θ_t}; a strong teacher q_T provides recovery supervision, while the frozen starting specialist $q_I = p_{\theta_I}$ provides the preservation reference. Recovery prompts are drawn from sources $\{\mathcal{D}_s\}_{s=1}^{S}$, while preservation prompts are drawn from $\mathcal{D}_P$. Model quality is measured by the capability vector $\boldsymbol{A}(\theta) = (A_1(\theta), \ldots, A_D(\theta))$. Sources index where recovery supervision is acquired, whereas $\boldsymbol{A}(\theta)$ indexes the capabilities on which recovery is evaluated.

For each sampled prompt x, the current student generates $y \sim p_{\theta_t}(\cdot \mid x)$, forming a trajectory $z = (x, y)$. On the visited prefix $h_j = (x, y_{<j})$, reference q_r, $r \in \{T, I\}$, provides the token-level correction

$$A_{r,j,t}(z) = \mathrm{sg}[\mathrm{clip}(\log q_r(y_j \mid h_j) - \log p_{\theta_t}(y_j \mid h_j), -c, c)], \qquad r \in \{T, I\}, \tag{1}$$



(d) Proxy-based selection (%)

	Random	DSIR	Gap	IFD	LESS
Math	69.60	66.40	63.20	67.20	67.20
Code	73.59	76.39	74.29	73.59	74.12
Reason.	73.97	74.08	73.04	72.75	72.28
Knowl.	79.57	77.98	78.63	78.83	78.01
IF	83.82	83.09	86.76	85.29	83.82
Overall	**76.11**	75.59	75.19	75.53	75.09

Figure 1: Relative capability changes and proxy-based selection under OPD (%).

where $c > 0$ is the clipping threshold and sg denotes stop-gradient. Because the prefixes are generated by the current student, these corrections vary with the receiving learner state.

Because the realized update also depends on optimizer state, we define the learner state as $\mathcal{M}_t = (\theta_t, o_t)$, where o_t includes the current optimizer moments. A data decision a instantiates a complete training batch and induces gradient $g_t(a)$. Applied from $\mathcal{M}_t$, it yields $\mathcal{M}_t^+(a) = \mathrm{OptStep}(\mathcal{M}_t, g_t(a))$ with parameter displacement $\Delta_t(a) = \theta_t^+(a) - \theta_t$. The corresponding capability change is $\Delta \boldsymbol{A}_t(a) = \boldsymbol{A}(\theta_t^+(a)) - \boldsymbol{A}(\theta_t)$.

For recovery–preservation calibration objective Φ_t, we define finite value of this realized update as

$$G_t(a; \mathcal{M}_t) = \Phi_t(\theta_t) - \Phi_t\big(\theta_t^+(a)\big)\,. \tag{2}$$

Thus, the value of the same data decision can change as the receiving learner state changes.

2.2 Data Decision and Recovery Objective

At each update t, a data-allocation policy π determines a source distribution p_t, samples a recovery source $s_t \sim p_t$, and forms a complete recovery–preservation batch $\mathcal{B}_t$. We denote the corresponding complete batch decision by a_t, so the training update satisfies $\mathcal{M}_{t+1} = \mathcal{M}_t^+(a_t)$. Given an executed-trajectory budget B_{tr}, the objective is final multi-capability recovery after H optimizer updates:

$$\max_{\pi} \sum_{d=1}^{D} w_d A_d(\theta_H) \quad \text{s.t.} \quad A_d(\theta_H) \geq A_d(\theta_I) - \tau_d \;\forall d, \qquad \sum_{t=0}^{H-1} |\mathcal{B}_t| \leq B_{\mathrm{tr}}. \tag{3}$$

Here, w_d specifies capability importance, and τ_d is the tolerated degradation relative to the starting specialist. Because $G_t(a; \mathcal{M}_t)$ depends on the receiving learner state, the source distribution that best serves the recovery–preservation objective can change along the training trajectory, motivating state-dependent adaptation of p_t.

3 State-Dependent Failures of Mixing and Selection under OPD

OPD pipelines already use data mixing and sample selection (Ma et al., 2026; Chen et al., 2026). We examine whether subject-based source allocation, reusing a fixed mixture as the learner evolves, and proxy-based example selection reliably improve the receiving student's capabilities. We study a code-specialized Qwen3-4B student with Qwen3-30B as the strong teacher, using mathematics, code, and reasoning (M/C/R) prompts. Balanced samples the three sources uniformly, whereas each source-heavy mixture assigns 90% of its sampling mass to the named source. Each source-mixture intervention starts from a saved learner state and continues for 50 optimizer updates under the same OPD recipe. We evaluate mathematics, code, reasoning, knowledge, and instruction following, and report their equally weighted mean as overall performance.(Detailed in Appendix F.)

3.1 Source Exposure Does Not Predict the Matching Capability Effect

We first test the intuition that increasing exposure to a subject improves the corresponding ability, keeping the receiving learner fixed and changing only the source mixture. Code-heavy improves mathematical ability by 40.7% while reducing coding ability by 27.0%, even as reasoning, knowledge, and instruction following improve (Figure 1a). Balanced produces the same directional conflict: mathematical ability rises by 59.3% while coding ability falls by 15.9%.

Under OPD, the source determines which prompts reach the student, while the student's responses determine the prefixes on which teacher corrections are obtained. The subject associated with a source is consequently insufficient to predict the capability effect of its supervision.

3.2 THE SAME MIXTURE CAN BECOME HARMFUL AS THE LEARNER CHANGES

We next ask whether a mixture that works well early in training remains useful as the student evolves. Balanced has the highest overall gain among the tested mixtures at the earliest checkpoint. Applying it for the same continuation horizon from five checkpoints along one training trajectory nevertheless produces a positive, then negative, then positive effect (Figure 1b). Overall performance improves by 7.9% from the checkpoint after 20 updates, declines by 12.2% from the checkpoint after 100 updates, and improves again by 6.2% after 200 updates.

The reversal reflects a changing capability trade-off. Reasoning ability improves by 3.8% at the earliest checkpoint but declines by 32.1% at the middle checkpoint; later, knowledge improves by 27.6% while instruction following declines by 5.4%. The same proportions therefore produce different capability trade-offs as student responses, teacher corrections, and optimizer state evolve. A mixture's early benefit does not establish its value at a later learner state.

3.3 STATIC SELECTION PROXIES DO NOT TRACK CURRENT DATA VALUE

We next examine whether example-level selection can identify more useful OPD data than random sampling. DSIR (Xie et al., 2023b) uses target-distribution importance resampling to prioritize examples, yet reaches 75.59 overall compared with 76.11 under random selection (Figure 1d). Its effect is not negligible: code improves by 3.8%, while mathematics and knowledge decline by 4.6% and 2.0%, respectively. Selection therefore changes the resulting capability profile without improving overall recovery.

Replacing distributional importance with response-, difficulty-, and gradient-based signals does not resolve this mismatch. Gap (Chen et al., 2026) favors responses with larger positive reference–student corrections, IFD (Li et al., 2024b) ranks examples by instruction-following difficulty, and LESS (Xia et al., 2024) uses optimizer-aware gradient similarity to target data. Yet none improves overall recovery over Random: Gap, IFD, and LESS reach 75.19, 75.53, and 75.09 overall, respectively, compared with 76.11 for Random (Figure 1d). These selectors change the resulting capability profile, but their static example-level signals do not reliably track the multi-capability value of the update received by the current learner.

This dependence becomes explicit when the ranking is updated with the learner. At fixed 50% mathematics exposure, we compare rankings transferred from an earlier student with rankings recomputed for the current student, while generating fresh on-policy responses in both continuations. Refreshing changes six of ten executed batches. The largest gain is on questions whose correctness changed between checkpoints (Figure 1c). The largest positive effect occurs where student behavior has changed, indicating that selection value follows the learner receiving the update rather than a fixed relevance, correction, or gradient score.

3.4 WHAT THESE FAILURES IMPLY

Together, these observations expose a decision mismatch: source proportions and example scores describe the data offered to the learner, while specialist recovery depends on the multi-capability effect of the complete update it actually receives. Under OPD, that effect can change as student responses, teacher corrections, and optimizer state evolve.

The finite value $G_t(a; \mathcal{M}_t)$ in Eq. (2) makes this dependence explicit. For a fixed current recovery–preservation objective Φ_t with locally Lipschitz gradient, its first-order expansion is

$$G_t(a; \mathcal{M}_t) = \underbrace{-\nabla \Phi_t(\theta_t)^\top}_{\text{descent direction}} \underbrace{\Delta_t(a)}_{\text{optimizer step}} + \underbrace{O\big(\|\Delta_t(a)\|^2\big)}_{\text{higher-order term}}. \tag{4}$$

The local gain depends on the alignment between the current recovery–preservation direction and the actual optimizer displacement; both evolve during training. Appendix B.4 gives the correspond-

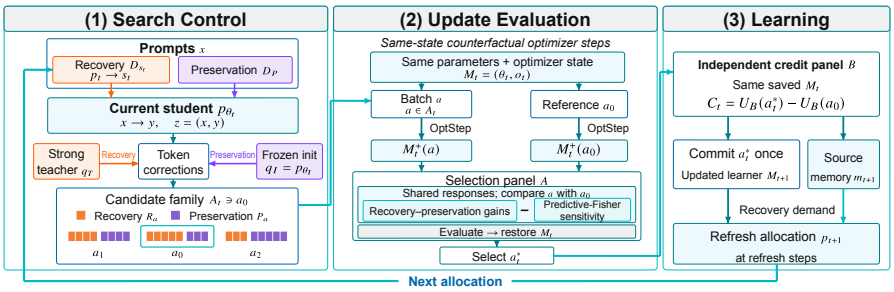


Figure 2: TangentMix's search–evaluation–learning loop.

ing finite-step bound. TangentMix compares complete updates at the current learner to track this changing value (Section 4).

4 TangentMix: State-Aware Data Allocation for Specialist Recovery

4.1 Search Control, Update Evaluation, and Learning

Section 3 shows that data value changes with the receiving learner, so TangentMix evaluates complete updates at the current learner state rather than relying on source identity or static example scores. Inspired by Dyna (Sutton, 1991), each TangentMix OPD update executes three coupled phases: **A: Search Control.** Given the current source distribution p_t and learner state $\mathcal{M}_t$, TangentMix samples s_t and uses current-student rollouts to form a recovery–preservation candidate family $\mathcal{A}_t$ with source-conditioned reference a_0; **B: Update Evaluation.** from the same saved $\mathcal{M}_t$, calibration panel A counterfactually compares the complete candidate OPD updates and selects a_t^*; and **C: Learning.** a fresh panel B measures the selected-versus-reference gain C_t, after which TangentMix commits the selected OPD update once to obtain $\mathcal{M}_{t+1}$ and records the gain in source memory to guide subsequent source allocation. Figure 2 illustrates the loop; Sections 4.2–4.4 give its operations, with notation and execution order in Appendix C.1.

4.2 Search Control via State-Conditioned Update Proposals

At update t, TangentMix samples a recovery source $s_t \sim p_t$, then draws recovery prompts from $\mathcal{D}_{s_t}$ and preservation prompts from $\mathcal{D}_P$. For each prompt x, the current student generates $y \sim p_{\theta_t}(\cdot \mid x)$, forming a trajectory $z = (x, y)$. The strong teacher q_T scores recovery tokens, while the frozen initialization $q_I = p_{\theta_I}$ scores preservation tokens on the student's generated prefixes. These scores give the clipped, stopped corrections in Eq. (1). Recovery proposals combine teacher-relative correction demand with representation diversity; preservation proposals prioritize positive initialization-relative corrections. These signals shortlist trajectories for complete-update evaluation (Appendix C.6).

Complete candidate batches. The family $\mathcal{A}_t$ contains a designated source-conditioned reference a_0 and candidates $a = (R_a, P_a)$, where R_a and P_a contain recovery and preservation trajectories. Each candidate satisfies $|R_a| + |P_a| = b_t$ and $1 \le |R_a| \le b_t - 1$. Candidates have the same total size and vary the branch composition. Within each composition, identity variants keep the preservation subset fixed and change recovery trajectories.

Shared OPD objective. Each candidate uses $L_{a,t}(\theta) = \lambda_t \ell_{T,t}(\theta; R_a) + (1 - \lambda_t)\ell_{I,t}(\theta; P_a)$, with branch loss

$$\ell_{r,t}(\theta; S) = -\frac{1}{N(S)} \sum_{z \in S} \sum_{j} A_{r,j,t}(z) \log p_\theta(y_j \mid h_j), \qquad r \in \{T, I\}. \tag{5}$$

Here $N(S)$ counts effective response tokens in the corresponding branch. The corrections and branch weight λ_t remain fixed across candidates. Thus, $\mathcal{A}_t$, a_0, and the losses $L_{a,t}$ define the alternatives passed to Section 4.3.

4.3 Update Evaluation: Same-State Counterfactual Comparison

Counterfactual optimizer steps. TangentMix compares what each complete candidate batch would do as the next optimizer step from the same learner state. For candidate a, it computes $g_t(a) = \nabla_\theta L_{a,t}(\theta_t)$, temporarily applies $\mathcal{M}_t^+(a) = \mathrm{OptStep}(\mathcal{M}_t, g_t(a))$, and records $\Delta_t(a) = \theta_t^+(a) - \theta_t$. Each temporary step starts from the same parameter–optimizer snapshot, which is restored after evaluating the candidate. The comparison therefore captures how the current optimizer transforms each complete batch gradient into an actual displacement.

Recovery–preservation utility. To evaluate these temporary updates, the current student first generates a shared calibration panel A using prompts from $\mathcal{D}_A$, disjoint from candidate-training prompts. Responses, prefixes, stopped corrections, and predictive-score draws are shared across candidates; calibration groups have equal sizes and equal response weights. With $\phi_{r,z,t}(\theta) = \ell_{r,t}(\theta; \{z\})$, define

$$\begin{aligned} G_{r,z,t}(a) &= \phi_{r,z,t}(\theta_t) - \phi_{r,z,t}(\theta_t + \Delta_t(a)), \\ u_{z,t}(a) &= \lambda_t G_{T,z,t}(a) + (1-\lambda_t) G_{I,z,t}(a) - \widehat{K}_z(a). \end{aligned} \tag{6}$$

The response mean of $\lambda_t \phi_{T,z,t} + (1-\lambda_t)\phi_{I,z,t}$ defines the fixed-panel objective Φ_t in Eq. (2). Its finite decrease gives the weighted recovery–preservation gain, while the predictive-Fisher penalty $\widehat{K}_z(a) = \frac{1}{2}(s_z^\top \Delta_t(a))^2$ accounts for local predictive movement. Here s_z is the score gradient of a token sampled from the current student's predictive distribution at a fixed prefix of z (Appendix C.6.1).

Reference-centered selection. TangentMix compares each candidate with a_0 using

$$J_t^A(a) = \frac{1}{|A|} \sum_{z \in A} \kappa \tanh\left(\frac{u_{z,t}(a) - u_{z,t}(a_0)}{\kappa} \right). \tag{7}$$

It selects $a_t^* \in \arg\max_{a \in \mathcal{A}_t} J_t^A(a)$. Applying the bounded transform before averaging limits each response's influence; $\kappa > 0$ sets this scale. The selected candidate and its gradient then pass to Section 4.4, where independent feedback is measured before the selected update is committed once.

4.4 Learning: Update Commitment and Feedback-Guided Allocation

Independent feedback and commitment. After A selects a_t^*, the saved current student generates an independent calibration panel B from $\mathcal{D}_B$, disjoint from $\mathcal{D}_A$ and candidate-training prompts, using the same grouping and weighting convention. TangentMix evaluates a_t^* and the source-conditioned reference a_0 from the same saved $\mathcal{M}_t$, sharing responses and predictive-score draws within B. With $U_X(a) = |X|^{-1} \sum_{z \in X} u_{z,t}(a)$, $C_t = U_B(a_t^*) - U_B(a_0)$ measures the incremental utility obtained by selection for the visited source using fresh evidence not used for candidate selection. TangentMix bounds this feedback as $x_t = \tanh(C_t / c_{\mathrm{fb}})$, updates only that source's exponentially weighted utility memory, restores the saved learner state, and commits the selected gradient exactly once to obtain $\mathcal{M}_{t+1}$.

Feedback-guided reallocation. At allocation refreshes, the updated learner defines a recovery-demand prior p_{t+1}^0 from its remaining gap and recent progress; after removing the exposure floor, TangentMix reweights the residual $q_{t+1}^0 = (p_{t+1}^0 - f\mathbf{1})/(1 - Sf)$ by the accumulated source utilities, $q_{t+1,s} \propto q_{t+1,s}^0 \exp(m_{s,t+1})$, and restores the floor to obtain p_{t+1}. Between refreshes, $p_{t+1} = p_t$; full memory and reweighting rules are given in Appendix C.7.

4.5 Analysis of the Search–Evaluation–Feedback Loop

At an allocation refresh indexed by receiving state t, let $\mathcal{H}_t$ be the history before a fresh source visit and $C_t^{(s)}$ its selected-versus-reference feedback for source s. Define $\nu_s(t) = \mathbb{E}[\tanh(C_t^{(s)}/c_{\mathrm{fb}}) \mid \mathcal{H}_t]$ and $q_t^0 = (p_t^0 - f\mathbf{1})/(1 - Sf)$. We use time-varying comparators (Hall & Willett, 2013) and entropy-regularized allocation (Beck & Teboulle, 2003): $\Psi_t(q) = \langle q, \nu(t) \rangle - \mathrm{KL}(q \| q_t^0)$. Let $q_t^\dagger$ maximize Ψ_t on the simplex restricted to the prior's support. The implemented residual $q_t = (p_t - f\mathbf{1})/(1 - Sf)$ maximizes the same objective with $\nu(t)$ replaced by m_t.

Proposition 1 (Current-update selection and independent feedback)**.** *Condition on the learner and $\mathcal{A}_t$ fixed before drawing A. Let $J_t(a)$ be the conditional expected bounded score, $N_t = |\mathcal{A}_t|$, and*

$\Gamma_t = \max_{a\in\mathfrak{C}_t} J_t(a) - \max_{a\in\mathcal{A}_t} J_t(a)$ *for a fixed finite* $\mathfrak{C}_t \supseteq \mathcal{A}_t$. *With* $n_A = |A|$ *independent observations in equally sized, equally weighted groups, with probability at least* $1-\alpha$, $\alpha \in (0,1)$,

$$\max_{a\in\mathfrak{C}_t} J_t(a) - J_t(a_t^*) \le \Gamma_t + 2\kappa\sqrt{\frac{2}{n_A}\log\frac{2N_t}{\alpha}}. \tag{8}$$

For independent B *with finite conditional second moments,* $\mathbb{E}[C_t \mid \mathcal{F}_t^A] = \bar{U}_{B,t}(a_t^*) - \bar{U}_{B,t}(a_0)$, *where* $\mathcal{F}_t^A$ *contains the history, candidates, and selection evidence, and* $\bar{U}_{B,t}$ *is their current population mean utility under the feedback distribution.*

Bounded-score concentration (Hoeffding, 1963) controls selection error; independent feedback measures the selected update on fresh evidence (Appendix B.1.1).

Proposition 2 (Tracking the current allocation optimum)**.** *At a refresh, if* $\|m_t - \nu(t)\|_\infty \le \epsilon_t$, *the implemented exponential reweighting satisfies* $\Psi_t(q_t^\dagger) - \Psi_t(q_t) = \mathrm{KL}(q_t\|q_t^\dagger) \le \epsilon_t^2/2$. *With* $p_t^\dagger = f\mathbf{1} + (1-Sf)q_t^\dagger$, $\|p_t - p_t^\dagger\|_1 \le (1-Sf)\epsilon_t$.

For fixed-ρ source memory, weighted martingale concentration (AZUMA, 1967) bounds ϵ_t by sampling uncertainty plus drift since previous visits. Appendix B.1.2 gives this radius, the cumulative refresh-objective gap, and the proof chain linking the retained theory to both propositions.

5 Experimental evaluation

Section 3 shows that source identity, fixed mixtures, and example scores can misrepresent the recovery effect of OPD data. We now test whether addressing this mismatch improves specialist recovery (Section 5.2), remains useful across training budgets (Section 5.3), and benefits from combining source allocation with complete-update selection (Section 5.4).

5.1 Experimental setup

Datasets, models, and recovery settings. We evaluate three specialized Qwen3-4B students with Qwen3-30B-A3B as the strong teacher; each initialization also serves as its frozen preservation reference. The three 4B settings are Complementary, Dominant, and Retention, representing complementary teacher–specialist strengths, teacher-dominant recovery, and recovery with stronger preservation pressure, respectively. We additionally evaluate a specialized Qwen3-1.7B student for cross-scale transfer. The shared recovery selection pool contains 12,000 prompts from eight sources. We evaluate seven capability axes, *e.g.*,, world knowledge, reasoning, multilingual comprehension, mathematics, code generation, and instruction following, plus AIME24/25 reasoning; benchmark definitions, full capability profiles, and other experimental details are given in Appendix D.

Baselines and evaluation metrics. We compare TangentMix with SFT (Ouyang et al., 2022), OPSD (Zhao et al., 2026b), OPD (Lu & Lab, 2025), GKD (Agarwal et al., 2024), MOPD (Ma et al., 2026), CaMOPD (Chen et al., 2026), DIEM-OPD (Tan et al., 2026), and TIDE (Yu et al., 2026). For the main specialist-recovery comparison, all methods use the same training-data budget of 1,600 executed trajectories, corresponding to 200 updates with eight trajectories per update. **Macro-7** equally averages the seven capability axes, with Code averaging HumanEval+ and MBPP+ pass@1. **ProfileScore** normalizes each capability between Init and the strong teacher, clips to $[0,1]$, and averages across axes, capturing the recovery–preservation trade-off in Section 3. Each result is run thrice and report the average.

5.2 Specialist Recovery

Table 1 tests whether addressing the Section 3 decision mismatch improves recovery across three 4B specialists and a 1.7B transfer setting at the same executed-data budget.

Specialist Recovery. TangentMix achieves the highest Macro-7 and ProfileScore in Complementary, Dominant, and Retention, reaching Macro-7 scores of 78.10, 75.78, and 77.24, respectively. Across the three 4B settings, it improves 95.2% of capability axes over Init and closes 79.1% of the aggregate recovery gap. The largest gain appears in Dominant: TangentMix improves

Table 1: Specialist recovery across Qwen3-4B/1.7B. (Full table in Appendix E.1 Table 5)

	Qwen3-4B						Qwen3-1.7B	
	Complementary		Dominant		Retention		Cross-scale	
Method	Macro-7 ↑	ProfileScore ↑	Macro-7 ↑	ProfileScore ↑	Macro-7 ↑	ProfileScore ↑	Macro-7 ↑	ProfileScore ↑
Strong teacher	79.01	0.857	79.01	1.000	79.01	0.857	79.01	0.857
Init	70.63	0.143	68.28	0.000	69.77	0.143	48.44	0.143
SFT	64.61	0.145	68.90	0.143	67.48	0.152	46.09	0.166
OPSD	72.70	0.317	68.20	0.094	74.20	0.494	40.73	0.000
OPD	77.18	0.699	73.79	0.512	75.73	0.603	52.06	0.231
GKD	76.24	0.620	72.96	0.443	76.19	0.630	55.51	0.259
MOPD	77.29	0.675	74.44	0.527	73.70	0.462	44.55	0.074
TIDE	77.07	0.696	73.00	0.438	76.86	0.685	55.05	0.220
CaMOPD	75.95	0.554	74.11	0.468	72.11	0.363	45.48	0.172
DIEM-OPD	77.44	0.720	73.71	0.496	76.39	0.646	54.20	0.320
TangentMix	**78.10**	**0.742**	**75.78**	**0.617**	**77.24**	**0.715**	**59.46**	**0.413**

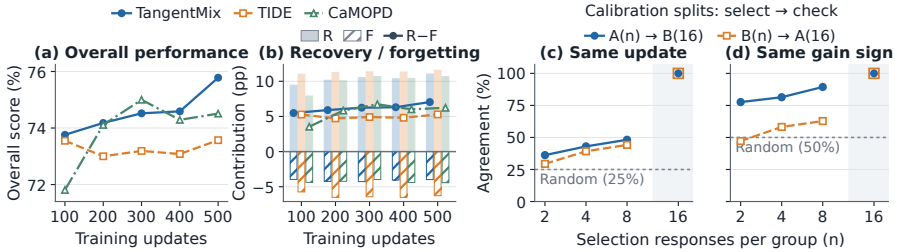


Figure 3: Training dynamics and state-conditioned decision diagnostics.

Macro-7 by 1.8% over MOPD and 2.8% over DIEM-OPD, while its ProfileScore reaches 0.617 versus 0.527 for MOPD, a 17.1% relative gain that reflects stronger recovery across the capability profile. The same design transfers to Qwen3-1.7B, reaching 59.46 Macro-7, 7.1% above GKD, extending the recovery gains across post-training regimes and model scales.

Recovery–Preservation. TangentMix combines teacher-supported recovery with strengths already present in the specialist (Appendix Table 5). On Complementary, it raises TyDiQA beyond Init (69.18 vs. 68.33) while raising MATH500 beyond the strong teacher (76.00 vs. 73.80). These paired gains show why recovery must be judged across the capability profile: TangentMix improves a weakened skill while retaining a strength of the starting specialist.

Math Reasoning. On Retention, TangentMix achieves the best Avg@8 on both AIME24 and AIME25 (60.83/50.00), yielding a mean of 55.42, 3.5% above OPD. It also ties the best mean Pass@8 at 78.33. Across 60 problems, TangentMix solves 47 at least once and produces 266 correct sampled responses, versus 45 and 256 for TIDE. The AIME gain therefore extends state-aware specialist recovery to competition-level reasoning through broader problem coverage and more successful sampled responses (Appendix E.3).

5.3 Data Efficiency and Recovery Dynamics

We examine whether TangentMix reaches strong recovery with less data and remains effective as the specialist evolves. We compare complete trajectories with TIDE, representing token-level OPD correction, and CaMOPD, representing recovery–preservation scheduling with learner-aware selection. As observed in Section 3.2, the value of the same mixture can change with the receiving learner, motivating evaluation over the full training trajectory rather than at a single endpoint. Budget is measured by executed student trajectories; Figure 3(a–b) reports Macro-7 and correctness turnover, with complete budget records in Appendix E.4.

Data Efficiency. TangentMix reaches 77.35 Macro-7 after 100 updates on Complementary, with all seven capabilities above Init, whereas fixed-mixture MOPD reaches 77.29 only after 200 updates, giving a 2× reduction in executed-data budget. On Dominant, TangentMix leads TIDE and CaMOPD at four of five checkpoints from 100 to 500 updates, keeps the worst-axis change positive throughout, and finishes at 75.78 Macro-7 versus 73.57 and 74.51. The trajectory advantage reflects

(a) AIME24/25 reasoning

Method	AIME24 Avg@8	AIME24 Pass@8	AIME25 Avg@8	AIME25 Pass@8
Strong teacher	76.25	90.00	61.25	83.33
Init	36.25	76.67	38.75	66.67
SFT	11.67	36.67	13.33	40.00
OPD	59.17	76.67	47.92	70.00
GKD	54.58	76.67	49.17	**73.33**
MOPD	56.67	83.33	48.75	**73.33**
TIDE	57.92	80.00	48.75	70.00
CaMOPD	55.00	**86.67**	45.00	66.67
DIEM-OPD	57.50	80.00	49.17	**73.33**
TangentMix	**60.83**	83.33	**50.00**	**73.33**

(b) Matched component ablation

Variant	Allocation	Selection	Macro-7
Fixed	–	–	76.594
Allocation only	✓	–	77.096
Selection only	–	✓	76.673
Allocation + selection	✓	✓	**77.487**

(c) Hyper-parameter sensitivity

Coefficient	Low	Default	High
λ	0.6 / 76.741	0.8 / **77.468**	0.9 / 76.123
β	0.0 / 76.709	0.15 / **77.468**	0.30 / 76.594
τ_{mix}	0.5 / 77.318	0.7 / **77.468**	1.0 / 76.694

Table 2: AIME reasoning, component ablation, and hyper-parameter sensitivity.

the state dependence in Section 3.2: as the learner changes, reallocating later training data remains more effective than retaining a fixed data policy.

Recovery Dynamics. We further examine the stability of data allocation over long-horizon training using recovered contribution R, which measures newly recovered capability, and forgotten contribution F, which measures existing specialist capability lost during recovery. Figure 3b shows that TangentMix has the lowest forgetting at every measured endpoint, while its net recovery increases steadily as training continues, both are reported as contributions to Macro-7 in percentage points. At 500 updates, TangentMix obtains $R = 11.03$ and $F = 3.98$, compared with 11.56 and 6.27 for TIDE and 10.66 and 4.43 for CaMOPD. It therefore retains nearly the same recovery as TIDE with 36.6% less forgetting, while improving over CaMOPD in both recovery and forgetting. These dynamics show that TangentMix can continue exploiting additional post-training data while maintaining a stable recovery–preservation trade-off as the model state evolves.

5.4 State-Conditioned Decision Analysis

We test whether TangentMix's state-conditioned data decisions improve recovery and whether independent current-state evidence supports the same update decisions.

Adaptive allocation increases the leverage of complete-update selection. We ablate allocation and selection separately and jointly. Combining them improves Macro-7 by 1.17% over fixed allocation; moreover, selection contributes 0.51% under adaptive allocation but only 0.10% under fixed allocation, a 4.9× larger gain (Table 2(b)). Allocation determines which source supplies candidate updates; selection determines which complete update is executed. Their joint ablation gain shows the value of coordinating these decisions. The joint design is also locally robust: recovery weight λ, proposal-diversity weight β, and source temperature τ_{mix} all peak at the default setting (Table 2(c)).

Mechanism Analysis. We test whether TangentMix's current-state evaluation consistently reflects complete-update value. One calibration split selects among 4 complete candidate updates and the disjoint split checks the same selected-versus-reference decision; their roles are then reversed. At 8 selection responses per group, the two directions select the same update in 48.2% and 44.2% of trials, compared with 25% under random choice; agreement on the gain sign reaches 89.4% and 62.8%, compared with a 50% random baseline (Figure 3c–d). Independent splits therefore agree more strongly on whether an update improves over the reference than on its exact identity, supporting complete-update valuation at the learner state where its recovery–preservation effect is realized.

6 Conclusion

We identify a decision mismatch in OPD specialist recovery: source proportions and example scores can diverge from the multi-capability effect of the complete update received by the current learner. TangentMix addresses this mismatch by evaluating complete recovery–preservation updates from the current model and optimizer state and using independent selection feedback to adapt source allocation. Across four 1.7B/4B settings, TangentMix leads all baselines, improves 92.9% of capability axes over Init, raises average Macro-7 by 4.2% over OPD, and achieves 2× data efficiency over fixed-mixture MOPD. These results show that tracking current update value enables continued post-training to recover weakened capabilities while preserving specialist strengths.

AI USE STATEMENT

Generative AI tools were used in an assistive capacity during parts of the research and manuscript preparation process, including discussions of methodology and experiments, checking mathematical formulations, code debugging, literature organization, and writing support. AI-generated suggestions were reviewed and, where used, verified by the authors against the relevant sources, code, and experimental records. The authors take responsibility for the final manuscript.

REPRODUCIBILITY STATEMENT

Sections 4–5 specify the algorithm and main experimental setup. The appendix provides the full theoretical, implementation, algorithm, evaluation metric, and reproducibility details supporting the reported results and baselines. We additionally provide the necessary code artifacts and model checkpoints for reproduction at `https://anonymous.4open.science/r/TangentMix-Release-DA1F`.

REFERENCES

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *International Conference on Learning Representations*, volume 2024, pp. 21246–21263, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/5be69a584901a26c521c2b51e40a4c20-Paper-Conference.pdf.

Rahaf Aljundi, Eugene Belilovsky, Tinne Tuytelaars, Laurent Charlin, Massimo Caccia, Min Lin, and Lucas Page-Caccia. Online continual learning with maximal interfered retrieval. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/15825aee15eb335cc13f9b559f166ee8-Paper.pdf.

KAZUOKI AZUMA. Weighted sums of certain dependent random variables. *Tohoku Mathematical Journal, Second Series*, 19(3):357–367, 1967. doi: 10.2748/tmj/1178243286.

Amir Beck and Marc Teboulle. Mirror descent and nonlinear projected subgradient methods for convex optimization. *Operations Research Letters*, 31(3):167–175, May 2003. ISSN 0167-6377. doi: 10.1016/s0167-6377(02)00231-6. URL http://dx.doi.org/10.1016/S0167-6377(02)00231-6.

Lichang Chen, Shiyang Li, Jun Yan, Hai Wang, Kalpa Gunaratna, Vikas Yadav, Zheng Tang, Vijay Srinivasan, Tianyi Zhou, Heng Huang, and Hongxia Jin. Alpagasus: Training a better alpaca with fewer data. In *International Conference on Learning Representations*, volume 2024, pp. 34767–34797, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/9543942c237ded1b39b1fd37259ff88e-Paper-Conference.pdf.

Tianlei Chen, Jiao Ou, Ziyuan Liu, Ruiming Tang, Jian Liang, and Han Li. Counteraction-aware multi-teacher on-policy distillation for general capability recovery with domain preservation, 2026. URL https://arxiv.org/abs/2605.27115.

Simin Fan, Matteo Pagliardini, and Martin Jaggi. DOGE: Domain reweighting with generalization estimation. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 12895–12915. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/fan24e.html.

Zixuan Fu, Bingxiang He, Yuxin Zuo, Haohuan Huang, Jinqian Zhang, Ruhang Xiao, Cheng Qian, Qinyu Luo, Huan-ang Gao, Yudong Wang, Zhiyuan Liu, Ning Ding, and Chaojun Xiao. Rethinking on-policy distillation of large language models ii: One training example, 2026. URL https://arxiv.org/abs/2609.04172.

Huan-ang Gao, Haohan Chi, Yong Yan, Shiyuan Feng, Hanlin Wu, Zheng Jiang, Bingxiang He, Wei-Ying Ma, Ya-Qin Zhang, and Hao Zhou. Open-mopd: Diagnosing and fixing capability imbalance in multi-teacher on-policy distillation, 2026. URL https://arxiv.org/abs/2608.19098.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. Minillm: Knowledge distillation of large language models. In *International Conference on Learning Representations*, volume 2024, pp. 32694–32717, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/8ac015d409635f196f9e3e9dcfb9a94e-Paper-Conference.pdf.

Eric Hall and Rebecca Willett. Dynamical models and tracking regret in online convex programming. In *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pp. 579–587, Atlanta, Georgia, USA, 17–19 Jun 2013. PMLR. URL https://proceedings.mlr.press/v28/hall13.html.

Jackson Harmon, Andreas Hochlehnert, Matthias Bethge, and Ameya Prabhu. Mapping post-training forgetting in language models at scale. In *International Conference on Learning Representations*, volume 2026, pp. 115664–115714, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/bc1c5e5fb8ed1ef9b9b5abced2022e40-Paper-Conference.pdf.

Wassily Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):13–30, March 1963. ISSN 1537-274X. doi: 10.1080/01621459.1963.10500830. URL http://dx.doi.org/10.1080/01621459.1963.10500830.

Steven R. Howard, Aaditya Ramdas, Jon McAuliffe, and Jasjeet Sekhon. Time-uniform chernoff bounds via nonnegative supermartingales. *Probability Surveys*, 17(none), January 2020. ISSN 1549-5787. doi: 10.1214/18-ps321. URL http://dx.doi.org/10.1214/18-PS321.

Ming Li, Yong Zhang, Shwai He, Zhitao Li, Hongyu Zhao, Jianzong Wang, Ning Cheng, and Tianyi Zhou. Superfiltering: Weak-to-strong data filtering for fast instruction-tuning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 14255–14273, Bangkok, Thailand, August 2024a. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.769. URL https://aclanthology.org/2024.acl-long.769/.

Ming Li, Yong Zhang, Zhitao Li, Jiuhai Chen, Lichang Chen, Ning Cheng, Jianzong Wang, Tianyi Zhou, and Jing Xiao. From quantity to quality: Boosting LLM performance with self-guided data selection for instruction tuning. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 7602–7635, Mexico City, Mexico, June 2024b. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-long.421. URL https://aclanthology.org/2024.naacl-long.421/.

Kevin Lu and Thinking Machines Lab. On-policy distillation. *Thinking Machines Lab: Connectionism*, 2025. doi: 10.64434/tml.20251026. https://thinkingmachines.ai/blog/on-policy-distillation.

Wenhan Ma, Jianyu Wei, Liang Zhao, Hailin Zhang, Bangjun Xiao, Lei Li, Qibin Yang, Bofei Gao, Yudong Wang, Rang Li, Jinhao Dong, Zhifang Sui, and Fuli Luo. Mopd: Multi-teacher on-policy distillation for capability integration in llm post-training, 2026. URL https://arxiv.org/abs/2606.30406.

James Martens. New insights and perspectives on the natural gradient method. *Journal of Machine Learning Research*, 21(146):1–76, 2020. URL http://jmlr.org/papers/v21/17-678.html.

Andreas Maurer and Massimiliano Pontil. Empirical bernstein bounds and sample variance penalization, 2009. URL https://arxiv.org/abs/0907.3740.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pp. 27730–27744. Curran Associates, Inc., 2022. doi: 10.52202/068431-2011. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/b1efde53be364a73914f58805a001731-Paper-Conference.pdf.

Zechen Sun, Zhiwei Zhang, Fei Zhao, Juntao Li, Mu Chuan, Huayu Deng, Guojian Zhan, Wenliang Chen, Yao Hu, and Min Zhang. D^3-mopd: Dynamic domain scheduling for efficient multi-teacher distillation, 2026. URL https://arxiv.org/abs/2608.24987.

Richard S. Sutton. Dyna, an integrated architecture for learning, planning, and reacting. *ACM SIGART Bulletin*, 2(4):160–163, July 1991. ISSN 0163-5719. doi: 10.1145/122344.122377. URL http://dx.doi.org/10.1145/122344.122377.

Haoru Tan, Sitong Wu, Yanfeng Chen, Shizhen Zhao, Yang-Tian Sun, Tianjia Liu, Chirui Chang, Shaofeng Zhang, Samm Sun, Xiuzhe Wu, Ruobing Xie, and Xiaojuan Qi. Dynamic important example mining for reinforcement finetuning, 2026. URL https://arxiv.org/abs/2608.29252.

Zinan Tang, Yukun Zhang, Shaomian Zheng, Zhuoshi Pan, Qizhi Pei, Dingnan Jin, Jun Zhou, Yujun Wang, and Biqing Huang. Causalmix: Data mixture as causal inference for language model training, 2026. URL https://arxiv.org/abs/2607.01104.

Fangxin Wang, Peyman Baghershahi, Langzhou He, Henry Peng Zou, Sourav Medya, and Philip S. Yu. Filter-then-weight: Online data selection and reweighting for llm fine-tuning, 2026a. URL https://arxiv.org/abs/2604.00001.

Zefan Wang, Lincheng Li, Tianyu Yu, and Yuan Yao. Drift: Refining instruction data via on-policy data attribution, 2026b. URL https://arxiv.org/abs/2606.18307.

Mengzhou Xia, Sadhika Malladi, Suchin Gururangan, Sanjeev Arora, and Danqi Chen. LESS: Selecting influential data for targeted instruction tuning. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 54104–54132. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/xia24c.html.

Bangjun Xiao, Bingquan Xia, Bo Yang, Bofei Gao, Bowen Shen, Chen Zhang, Chenhong He, Chiheng Lou, Fuli Luo, Gang Wang, et al. Mimo-v2-flash technical report, 2026. URL https://arxiv.org/abs/2601.02780.

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 *Advances in Neural Information Processing Systems*, volume 36, pp. 69798–69818. Curran Associates, Inc., 2023a. doi: 10.52202/075280-3059. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/dcba6be91359358c2355cd920da3fcbd-Paper-Conference.pdf.

Sang Michael Xie, Shibani Santurkar, Tengyu Ma, and Percy S Liang. Data selection for language models via importance resampling. In *Advances in Neural Information Processing Systems*, volume 36, pp. 34201–34227. Curran Associates, Inc., 2023b. doi: 10.52202/075280-1482. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/6b9aa8f418bde2840d5f4ab7a02f663b-Paper-Conference.pdf.

Anyi Xu, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, Chenchen Ling, et al. Deepseek-v4: Towards highly efficient million-token context intelligence. *arXiv preprint arXiv:2606.19348*, 2026a.

Yuanda Xu, Hejian Sang, Zhengze Zhou, Ran He, and Zhipeng Wang. Paced: Distillation and on-policy self-distillation at the frontier of student competence, 2026b. URL https://arxiv.org/abs/2603.11178.

Zhuolin Yang, Zihan Liu, Yang Chen, Wenliang Dai, Boxin Wang, Sheng-Chieh Lin, Chankyu Lee, Yangyi Chen, Dongfu Jiang, Jiafan He, Renjie Pi, Grace Lam, Nayeon Lee, Alexander Bukharin, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. Nemotron-cascade 2: Post-training llms with cascade rl and multi-domain on-policy distillation, 2026. URL https://arxiv.org/abs/2603.19220.

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. In *Advances in Neural Information Processing Systems*, volume 33, pp. 5824–5836. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/3fe78a8acf5fda99de95303940a2420c-Paper.pdf.

Zichao Yu, Chengzhi Yu, Shengze Xu, Yujin Han, Bingqing Jiang, Xu Wang, and Difan Zou. Mismatch matters: On-policy distillation beyond token agreement, 2026. URL https://arxiv.org/abs/2608.09836.

Anhao Zhao, Junlong Tong, Yingqi Fan, Ping Nie, Wenjie Li, and Xiaoyu Shen. Poweropd: Stabilizing on-policy distillation with bounded power transformation, 2026a. URL https://arxiv.org/abs/2606.17199.

Siyan Zhao, Zhihui Xie, Mengchen Liu, Jing Huang, Guan Pang, Feiyu Chen, and Aditya Grover. Self-distilled reasoner: On-policy self-distillation for large language models. In *Forty-third International Conference on Machine Learning*, 2026b. URL https://openreview.net/forum?id=Jpxfof0EaS.

Zilin Zhu, Chengxing Xie, Xin Lv, and slime Contributors. slime: An llm post-training framework for rl scaling. https://github.com/THUDM/slime, 2025. GitHub repository. Corresponding author: Xin Lv.

A Extended Related Work

Post-training recovery and distillation. GKD learns from student-generated trajectories with teacher supervision (Agarwal et al., 2024). MiniLLM studies reverse-divergence distillation for limited student capacity (Gu et al., 2024). MOPD integrates domain teachers (Ma et al., 2026), and Nemotron-Cascade 2 uses intermediate experts to recover regressions during post-training (Yang et al., 2026). CaMOPD addresses general recovery with domain preservation through alternating updates and gap-based selection (Chen et al., 2026). Open-MOPD diagnoses token-budget imbalance and feedback staleness (Gao et al., 2026), while D^3-MOPD adapts source exposure according to remaining domain gaps and learning progress (Sun et al., 2026). TangentMix instead connects source allocation to the recovery–preservation value realized by complete updates at the current specialist.

Data valuation and mixtures. DSIR selects data by distributional relevance (Xie et al., 2023b); AlpaGasus uses language-model quality scores (Chen et al., 2024); and Superfiltering transfers difficulty-based selection from weaker models (Li et al., 2024a). LESS uses optimizer-aware gradient features for targeted instruction tuning (Xia et al., 2024), while DoReMi and DoGE optimize domain-level data mixtures (Xie et al., 2023a; Fan et al., 2024). Filter-then-Weight studies online selection and reweighting (Wang et al., 2026a), and DRIFT refines data through on-policy attribution (Wang et al., 2026b). PACED weights problems by student competence (Xu et al., 2026b), and CausalMix conditions mixture returns on data-pool covariates (Tang et al., 2026). These methods value data through source-level learning signals or example-level distributional, difficulty, attribution, and gradient signals. TangentMix instead evaluates the complete recovery–preservation update induced at the current learner and returns its independently measured utility to subsequent source allocation.

Update interaction and dynamic weighting. Gradient surgery studies conflicting task updates (Yu et al., 2020), while maximally interfered retrieval selects replay examples according to a prospective update (Aljundi et al., 2019). Predictive Fisher characterizes local distribution sensitivity (Martens, 2020). TIDE shapes informative token corrections (Yu et al., 2026); PowerOPD studies bounded reward transformations (Zhao et al., 2026a). DIEM estimates example importance by batch-gradient alignment and reweights reinforcement-learning data (Tan et al., 2026). TangentMix values complete recovery–preservation batches and returns their independent incremental utility to source allocation.

Recovery measurement and finite feedback. Example-wise forgetting analysis distinguishes recovered and lost behavior (Harmon et al., 2026); small-prompt OPD studies the learning states induced by limited prompts (Fu et al., 2026). Variance-sensitive concentration characterizes finite-score comparisons (Maurer & Pontil, 2009), and martingale analysis accommodates evolving feedback histories (Howard et al., 2020). These tools support the paired calibration and state-tracking analysis used in TangentMix.

B Full Theory for TangentMix

This appendix collects the full assumptions and derivations behind Section 4.5. The analysis separates three probability spaces: (i) a population task distribution used only to relate teacher fit to initialization-relative recovery; (ii) a frozen current-prefix distribution used to analyze one complete optimizer displacement; and (iii) conditional calibration distributions used to rank a finite candidate family and to measure independent source feedback. None of the online results assumes that these distributions are stationary as training progresses.

B.1 Guide to the Main-Text Propositions

The full results below support the two propositions in Section 4.5 through two proof chains:

- **Chain 1: current-update selection and independent feedback.** The finite-panel selection result in Theorem 4, together with the independent-feedback part of Theorem 7, yields Proposition 1.

- **Chain 2: state-tracking source allocation.** The source-memory tracking result in Theorem 7, combined with the deployed exponential tilt, yields Proposition 2.

The remaining results interpret the complete-update utility and its state dependence; they are supporting results rather than additional premises of these two chains.

B.1.1 Chain 1: Current-update selection and independent feedback

For equally sized, equally weighted calibration groups, $\sum_g \omega_g^2/n_{A,g} = 1/n_A$. Substituting this identity into Theorem 4 gives exactly Eq. (8). The independent-feedback statement in Proposition 1 follows directly from the conditional expectation in Theorem 7. Proposition 5, Appendix B.5, and Lemma 6 provide the corresponding interpretation of the finite complete-update utility and its recovery–preservation coupling.

B.1.2 Chain 2: Tracking the current allocation optimum

At a refresh, let $q_t^0 = (p_t^0 - f\mathbf{1})/(1 - Sf)$ and define

$$\Psi_t(q) = \langle q, \nu(t)\rangle - \mathrm{KL}(q\|q_t^0), \qquad q_t^\dagger = \arg\max_{q\in\Delta^S,\ q\ll q_t^0} \Psi_t(q). \tag{9}$$

The source-memory part of Theorem 7 and Eq. (22) give a current error radius $\|m_t - \nu(t)\|_\infty \le \epsilon_t$. On the support of q_t^0, the ideal and implemented residual allocations are the exponential tilts by $\nu(t)$ and m_t, respectively. Therefore $\Psi_t(q) = \log Z_t(\nu(t)) - \mathrm{KL}(q\|q_t^\dagger)$, and the memory-error bound, together with Hoeffding's lemma, gives

$$\Psi_t(q_t^\dagger) - \Psi_t(q_t) = \mathrm{KL}(q_t\|q_t^\dagger) \le \epsilon_t^2/2. \tag{10}$$

With $p_t^\dagger = f\mathbf{1} + (1 - Sf)q_t^\dagger$, Pinsker's inequality then yields $\|p_t - p_t^\dagger\|_1 \le (1 - Sf)\epsilon_t$. Summing Eq. (10) over allocation refreshes gives the cumulative objective-gap bound $\sum_{t\in\mathcal{T}_{\mathrm{refresh}}}[\Psi_t(q_t^\dagger) - \Psi_t(q_t)] \le \frac{1}{2}\sum_{t\in\mathcal{T}_{\mathrm{refresh}}}\epsilon_t^2$.

B.2 Population Recovery, Teacher Fit, and the Initialization Reference

Let P_d denote the evaluation-prompt distribution for capability d and let $Q = \sum_s p_s\mathcal{D}_s$ be a training-prompt mixture. For a bounded response loss $r_d(x, y) \in [0, 1]$, define

$$R_d(p) = \mathbb{E}_{x\sim P_d,\ y\sim p(\cdot|x)} r_d(x, y).$$

This population quantity is distinct from the current-prefix calibration losses used by the online selector.

Proposition 3 (Teacher fit and initialization-relative recovery)**.** *Assume $P_d \ll Q$,*

$$C_d(Q) = \mathbb{E}_{x\sim Q}\left[\left(\frac{dP_d}{dQ}(x)\right)^2\right] < \infty,$$

and

$$K_Q(p, q_T) = \mathbb{E}_{x\sim Q}\,\mathrm{KL}(p(\cdot \mid x)\|q_T(\cdot \mid x)) < \infty.$$

Then

$$|R_d(p) - R_d(q_T)| \le \sqrt{\frac{1}{2}C_d(Q)K_Q(p, q_T)}. \tag{11}$$

If the initialization has risk advantage $h_d = R_d(q_T) - R_d(p_{\theta_I}) > 0$, then any p satisfying

$$\sqrt{\frac{1}{2}C_d(Q)K_Q(p, q_T)} < h_d - \varepsilon_d, \qquad 0 \le \varepsilon_d < h_d,$$

also satisfies $R_d(p) > R_d(p_{\theta_I}) + \varepsilon_d$.

Proof. For each x, the difference of expectations of a $[0,1]$-valued loss is bounded by total variation, and Pinsker's inequality gives

$$\left|\mathbb{E}_{p(\cdot|x)} r_d - \mathbb{E}_{q_T(\cdot|x)} r_d\right| \le \sqrt{\frac{1}{2}\,\mathrm{KL}(p(\cdot \mid x) \| q_T(\cdot \mid x))}.$$

Write $w(x) = dP_d/dQ$. Absolute continuity and Cauchy–Schwarz yield

$$\begin{aligned} |R_d(p) - R_d(q_T)| &\le \mathbb{E}_Q\left[w(x)\sqrt{\tfrac{1}{2}\,\mathrm{KL}(p\|q_T)}\right] \\ &\le \sqrt{\tfrac{1}{2}\mathbb{E}_Q[w(x)^2]\,\mathbb{E}_Q[\mathrm{KL}(p\|q_T)]}. \end{aligned}$$

which is Eq. (11). The second statement follows from the triangle inequality with the assumed Init advantage. □

This proposition is a population statement under explicit task-coverage assumptions. It explains why a strong teacher and an already-useful specialist define different recovery and preservation targets, but it is not used as the online batch-ranking guarantee.

B.3 Candidate Coverage and Finite-Panel Selection

Fix update t and condition on all training-side randomness used to construct the current candidate family. Let $\mathfrak{C}_t$ be a finite diagnostic comparison family and $\mathcal{A}_t \subseteq \mathfrak{C}_t$ the evaluated shortlist. For the reference-centered score in Eq. (7), define the conditional population target

$$\begin{aligned} J_t(a) &= \sum_g \omega_g \mathbb{E}_{Z\sim P_{A,g,t}}[f_a(Z)], \\ f_a(Z) &= \kappa \tanh\left(\frac{u_Z(a) - u_Z(a_0)}{\kappa}\right). \end{aligned}$$

By construction, $f_a(Z) \in [-\kappa, \kappa]$.

Proposal gap. The value omitted by the shortlist is

$$\Gamma_t(\mathcal{A}_t; \mathfrak{C}_t) = \max_{a\in\mathfrak{C}_t} J_t(a) - \max_{a\in\mathcal{A}_t} J_t(a) \ge 0. \tag{12}$$

The quantity in Eq. (12) is deterministic once J_t and the two finite families are fixed. If $\mathcal{A}_t \subseteq \mathcal{A}_t^+ \subseteq \mathfrak{C}_t$, then

$$\Gamma_t(\mathcal{A}_t^+; \mathfrak{C}_t) \le \Gamma_t(\mathcal{A}_t; \mathfrak{C}_t),$$

so expanding the candidate family can only reduce the best attainable loss relative to the declared diagnostic family. This does not imply approximation to a global optimum over all possible trajectory subsets.

Theorem 4 (Proposal–selection bound)**.** *Condition on the current learner and a fixed candidate family $\mathcal{A}_t$ before drawing selection evidence. Suppose calibration group g has fixed weight $\omega_g \ge 0$, $\sum_g \omega_g = 1$, and contributes $n_{A,g}$ conditionally independent observations from its fixed current calibration distribution; groups are independent. For*

$$f_a(Z) = \kappa \tanh\left(\frac{u_Z(a) - u_Z(a_0)}{\kappa}\right) \in [-\kappa, \kappa],$$

let a_t^* maximize the corresponding empirical A-panel mean and let $N_t = |\mathcal{A}_t|$. Then, with probability at least $1-\alpha$,

$$\begin{aligned} \max_{a\in\mathfrak{C}_t} J_t(a) - J_t(a_t^*) &\le \Gamma_t(\mathcal{A}_t; \mathfrak{C}_t) + 2\kappa\varepsilon_{A,t}, \\ \varepsilon_{A,t} &= \sqrt{2\log\frac{2N_t}{\alpha}\sum_g \frac{\omega_g^2}{n_{A,g}}}. \end{aligned} \tag{13}$$

Proof of Theorem 4. For each $a \in \mathcal{A}_t$, let

$$\widehat{J}_t^A(a) = \sum_g \frac{\omega_g}{n_{A,g}} \sum_{i=1}^{n_{A,g}} f_a(Z_{g,i}).$$

Conditional on the current learner and candidate family, the rows are independent within and across groups. Weighted Hoeffding therefore gives, for every fixed a,

$$\Pr\left(|\widehat{J}_t^A(a) - J_t(a)| > \epsilon\right) \le 2\exp\left(-\frac{\epsilon^2}{2\kappa^2 \sum_g \omega_g^2/n_{A,g}}\right).$$

A union bound over $N_t = |\mathcal{A}_t|$ candidates implies that, with probability at least $1-\alpha$,

$$\sup_{a\in\mathcal{A}_t} |\widehat{J}_t^A(a) - J_t(a)| \le \epsilon_A := \kappa\sqrt{2\log\frac{2N_t}{\alpha}\sum_g \frac{\omega_g^2}{n_{A,g}}}.$$

Let $a_A^\star \in \arg\max_{a\in\mathcal{A}_t} J_t(a)$. Since a_t^* maximizes the empirical score,

$$J_t(a_A^\star) - J_t(a_t^*) \le 2\epsilon_A.$$

Adding the exact proposal decomposition

$$\max_{a\in\mathfrak{C}_t} J_t(a) - J_t(a_t^*) = \Gamma_t(\mathcal{A}_t;\mathfrak{C}_t) + J_t(a_A^\star) - J_t(a_t^*)$$

proves Eq. (13). □

Init-relative calibration certificate. For a fixed calibration loss ϕ_d, suppose an estimated finite gain satisfies $|G_d(a) - \widehat{G}_d(a)| \le \varepsilon_d$. If

$$\widehat{G}_d(a) \ge \phi_d(\theta_t) - \phi_d(\theta_I) - \xi_d + \varepsilon_d,$$

then

$$\phi_d(\theta_t + \Delta_t(a)) \le \phi_d(\theta_I) + \xi_d.$$

This is a current-calibration certificate, not a benchmark-level retention guarantee.

B.4 Local Complete-Update Value

For the main-text comparison, Φ_t is the fixed-panel average of $\lambda_t\phi_{T,z,t} + (1-\lambda_t)\phi_{I,z,t}$ defined in Section 4.5. Consequently its finite decrease equals the weighted recovery and preservation gains before subtraction of predictive sensitivity. With $\mathcal{M}_t$ fixed, $G_t(a)$ abbreviates $G_t(a;\mathcal{M}_t)$.

Write $\Delta = \Delta_t(a)$ and let

$$G_t(a) = \Phi_t(\theta_t) - \Phi_t(\theta_t+\Delta), \qquad \mathcal{I}_t(a) = -\nabla\Phi_t(\theta_t)^\top\Delta.$$

Proposition 5 (Local complete-update value and predictive sensitivity)**.** *If Φ_t has L-Lipschitz gradient along the update segment, then*

$$|G_t(a) - \mathcal{I}_t(a)| \le \frac{L}{2}\|\Delta_t(a)\|^2. \tag{14}$$

Further, freeze a current-prefix distribution μ_t and define

$$K_t(\Delta) = \mathbb{E}_{h\sim\mu_t}\,\mathrm{KL}(p_{\theta_t}(\cdot\mid h)\|p_{\theta_t+\Delta}(\cdot\mid h)).$$

Under common predictive support, differentiation under expectation, and a third directional derivative bounded by $M_K\|\Delta\|^3$ along the segment,

$$\left|K_t(\Delta) - \frac{1}{2}\Delta^\top F_t\Delta\right| \le \frac{M_K}{6}\|\Delta\|^3, \qquad F_t = \mathbb{E}[ss^\top], \tag{15}$$

where $s = \nabla_\theta \log p_{\theta_t}(Z\mid h)$ and $Z\sim p_{\theta_t}(\cdot\mid h)$.

Proof of Eq. (14)**.** By the fundamental theorem of calculus,

$$\Phi_t(\theta_t + \Delta) - \Phi_t(\theta_t) = \int_0^1 \nabla\Phi_t(\theta_t + s\Delta)^\top \Delta \, ds.$$

Subtract $\nabla\Phi_t(\theta_t)^\top\Delta$. If $\nabla\Phi_t$ is L-Lipschitz along the segment, then

$$\left| \int_0^1 [\nabla\Phi_t(\theta_t + s\Delta) - \nabla\Phi_t(\theta_t)]^\top \Delta \, ds \right| \leq \int_0^1 Ls\|\Delta\|^2 \, ds = \frac{L}{2}\|\Delta\|^2.$$

Rearranging proves Eq. (14). □

Optimizer-residual decomposition. For a capability loss ϕ_d with $b_d = \nabla\phi_d(\theta_t)$ and L_d-Lipschitz gradient, suppose the realized displacement admits the explanatory decomposition

$$\Delta_t(a) = -\eta P \sum_i c_i g_i + e(a), \qquad P = P^\top \succeq 0, \quad c_i \geq 0.$$

Then

$$G_d(a) \geq \eta \sum_i c_i b_d^\top P g_i - \|b_d\| \, \|e(a)\| - \frac{L_d}{2}\|\Delta_t(a)\|^2. \tag{16}$$

The actual algorithm does not replace Adam by P; Eq. (16) is an explanatory decomposition around the realized update.

Pairwise ordering consequence. For two candidates,

$$G_t(a) - G_t(b) \geq \mathcal{I}_t(a) - \mathcal{I}_t(b) - \frac{L}{2}\left(\|\Delta_t(a)\|^2 + \|\Delta_t(b)\|^2\right).$$

Hence a directional margin larger than the curvature remainder certifies the same finite-step ordering.

B.5 Predictive Fisher as Local Sensitivity

Freeze a distribution μ_t over current response prefixes and define

$$K_t(\Delta) = \mathbb{E}_{h\sim\mu_t} \, \mathrm{KL}\left(p_{\theta_t}(\cdot \mid h) \| p_{\theta_t+\Delta}(\cdot \mid h)\right).$$

For a fixed prefix h, the KL has value and first derivative zero at $\Delta = 0$. Its Hessian is the predictive Fisher

$$F_t = \mathbb{E}_{h\sim\mu_t, \, Z\sim p_{\theta_t}(\cdot|h)} ss^\top, \qquad s = \nabla_\theta \log p_{\theta_t}(Z \mid h).$$

Proof of Eq. (15)**.** Apply one-dimensional Taylor expansion to $u \mapsto K_t(u\Delta)$. The constant and linear terms vanish and the quadratic term is $\Delta^\top F_t \Delta/2$. Under the stated third-directional-derivative bound, the remainder is at most $M_K\|\Delta\|^3/6$. This proves Eq. (15). For a displacement fixed independently of the score draw,

$$\mathbb{E}\left[\frac{1}{2}(s^\top\Delta)^2\right] = \frac{1}{2}\Delta^\top F_t \Delta.$$

Shared score draws across fixed candidate displacements preserve covariance in their comparison. □

Initialization-relative predictive extension. Let

$$\epsilon_{I,t} = \mathbb{E}_{h\sim\mu_t} \, \mathrm{TV}\left(p_{\theta_t}(\cdot \mid h), q_I(\cdot \mid h)\right).$$

For any token statistic $f(h, z) \in [0, 1]$, Pinsker's inequality and Eq. (15) give

$$\mathbb{E}_{\mu_t}\left|\mathbb{E}_{p_{\theta_t+\Delta}} f - \mathbb{E}_{q_I} f\right| \leq \epsilon_{I,t} + \sqrt{\frac{1}{4}\Delta^\top F_t \Delta + \frac{M_K}{12}\|\Delta\|^3}. \tag{17}$$

The first term is discrepancy already accumulated before the current decision; the second is the local predictive movement introduced by the candidate update.

B.6 Why Recovery–Preservation Composition and Identity Interact

The deployed algorithm evaluates complete updates directly. A local quadratic model gives a complementary interpretation of why source composition and selected identities cannot in general be optimized independently.

Lemma 6 (Coupled mixture and trajectory value)**.** *Assume a fixed $P = P^\top \succeq 0$ and a quadratic current calibration objective with gradient $\bar{b}$ and Hessian $H = H^\top$. Let selected source gradients be μ_s and $g = \sum_s \alpha_s \mu_s$. For displacement $-\eta P g$,*

$$\begin{aligned} J(\alpha, \{\mu_s\}) &= \eta \bar{b}^\top P g - \frac{\eta^2}{2} g^\top P H P g, \\ \frac{\partial J}{\partial \alpha_s} &= \eta \bar{b}^\top P \mu_s - \eta^2 \mu_s^\top P H P g. \end{aligned} \tag{18}$$

For equal-count recovery alternatives r_0, r_1 and preservation alternatives v_0, v_1, define J_{ij} as the gain from $-\eta P(r_i + v_j)$. Then

$$J_{11} - J_{10} - J_{01} + J_{00} = -\eta^2 (r_1 - r_0)^\top P H P (v_1 - v_0). \tag{19}$$

Proof. Substitute $\Delta = -\eta P g$ into the quadratic expansion

$$\Phi_t(\theta_t + \Delta) = \Phi_t(\theta_t) + \bar{b}^\top \Delta + \frac{1}{2} \Delta^\top H \Delta$$

and subtract from $\Phi_t(\theta_t)$. Differentiation gives Eq. (18). Expanding the four J_{ij} values cancels the linear and pure-role quadratic terms, leaving only the mixed interaction in Eq. (19). □

A nonzero mixed term means that the marginal value of a recovery choice depends on its preservation partner. The result supports complete recovery–preservation candidate comparison, while the method itself uses the actual optimizer displacement rather than this quadratic approximation.

B.7 Independent Feedback and State-Tracking Source Allocation

Let $\mathcal{F}_t^A$ include the current learner, candidate batches, and all selection evidence. Conditional on $\mathcal{F}_t^A$, the winner a_t^* and reference a_0 are fixed. For fresh feedback group g, define

$$Y_{g,i} = u_{g,i}(a_t^*) - u_{g,i}(a_0), \qquad C_t = \sum_g \frac{\omega_g}{n_{B,g}} \sum_{i=1}^{n_{B,g}} Y_{g,i}.$$

Theorem 7 (Independent feedback and source tracking)**.** *Let $\omega_g \geq 0$, $\sum_g \omega_g = 1$, be the fixed group weights and $C_t = \sum_g \omega_g n_{B,g}^{-1} \sum_i Y_{g,i}$. If the B observations are conditionally i.i.d. within each group, independent across groups and independent of the A evidence, with finite conditional second moments, then*

$$\mathbb{E}[C_t \mid \mathcal{F}_t^A] = \mu_t, \qquad \mathrm{Var}(C_t \mid \mathcal{F}_t^A) = \sum_g \frac{\omega_g^2 \sigma_{\mathrm{pair},g}^2}{n_{B,g}}, \tag{20}$$

where $\mu_t = \sum_g \omega_g \mathbb{E}[Y_{g,1} \mid \mathcal{F}_t^A]$ and $\sigma_{\mathrm{pair},g}^2 = \mathrm{Var}(Y_{g,1} \mid \mathcal{F}_t^A)$. For the bounded source observations $x_t = \tanh(C_t / c_{\mathrm{fb}})$ defined in Section 4.4, let m_t be their moving-average estimate and let $\nu(t)$ be the vector of expected bounded feedback from fresh source visits at the current learner. If $\|m_t - \nu(t)\|_\infty \leq \epsilon_t$, then any source-distribution change from a current recovery prior p^0 to p satisfies

$$\langle \nu(t), p - p^0 \rangle \geq \langle m_t, p - p^0 \rangle - \epsilon_t \|p - p^0\|_1. \tag{21}$$

Paired feedback. Conditional independence across rows and groups yields

$$\begin{aligned} \mathbb{E}[C_t \mid \mathcal{F}_t^A] &= \mu_t, \\ \mathrm{Var}(C_t \mid \mathcal{F}_t^A) &= \sum_g \frac{\omega_g^2}{n_{B,g}} \mathrm{Var}(Y_{g,1} \mid \mathcal{F}_t^A). \end{aligned}$$

Within a group,

$$\mathrm{Var}(Y) = \mathrm{Var}(u(a_t^*)) + \mathrm{Var}(u(a_0)) - 2\,\mathrm{Cov}(u(a_t^*), u(a_0)),$$

which makes the benefit of paired evaluation explicit when the covariance is positive. If $\mu_t > 0$ and the conditional variance is v_t, Cantelli's inequality gives

$$\Pr(C_t \le 0 \mid \mathcal{F}_t^A) \le \frac{v_t}{v_t + \mu_t^2}.$$

Tracking bounded source feedback. For source s, let $X_{s,j} \in [-1, 1]$ be its bounded feedback at visit j, with conditional mean $\nu_{s,j}$ before that visit's observation is drawn. Adaptive source visits may depend on all preceding history. Let

$$m_{s,n} = \sum_{j=1}^{n} \beta_{n,j} X_{s,j}, \qquad \beta_{n,j} \ge 0, \qquad \sum_j \beta_{n,j} = 1.$$

For a fixed visit count n, the weights $\beta_{n,j}$ are deterministic, as for the fixed-ρ EMA in Eq. (33). In that case $\beta_{n,1} = (1-\rho)^{n-1}$ and $\beta_{n,j} = \rho(1-\rho)^{n-j}$ for $2 \le j \le n$; these weights sum to one. At the current refresh, let $\nu_s(t)$ be the expected bounded feedback from a fresh visit to source s at the current learner. For a fixed horizon of at most N visits per source, a conditional Hoeffding argument and a union bound give, with probability at least $1 - \alpha$,

$$\begin{aligned} |m_{s,n} - \nu_s(t)| &\le r_{s,n} + d_{s,t}, \\ r_{s,n} &= \sqrt{2 \log \frac{2SN}{\alpha} \sum_j \beta_{n,j}^2}, \\ d_{s,t} &= \sum_j \beta_{n,j} |\nu_{s,j} - \nu_s(t)|. \end{aligned} \tag{22}$$

simultaneously for all observed source prefixes. The first term is finite-observation uncertainty and the second is historical drift.

On this event, set $e_{s,t} = r_{s,n} + d_{s,t}$ for an observed source. For a source not yet visited, $m_{s,t} = 0$ and $\nu_s(t) \in [-1, 1]$, so use $e_{s,t} = 1$. Then $\epsilon_t = \max_s e_{s,t}$ bounds $\|m_t - \nu(t)\|_\infty$. Hölder's inequality gives $\langle \nu(t) - m_t, p - p^0 \rangle \ge -\epsilon_t \|p - p^0\|_1$, which proves Eq. (21).

Connection to the deployed exponential tilt. Let q_t^0 be the residual prior after removing the floor and let

$$q_{t,s} = \frac{q_{t,s}^0 e^{m_{s,t}}}{\sum_j q_{t,j}^0 e^{m_{j,t}}}.$$

On the support of q_t^0,

$$\langle m_t, q_t - q_t^0 \rangle = \mathrm{KL}(q_t \| q_t^0) + \mathrm{KL}(q_t^0 \| q_t).$$

Since $p_t - p_t^0 = (1 - Sf)(q_t - q_t^0)$, the tracking event gives

$$\begin{aligned} \langle \nu(t), p_t - p_t^0 \rangle \ge (1 - Sf) \big[& \mathrm{KL}(q_t \| q_t^0) + \mathrm{KL}(q_t^0 \| q_t) \\ & - \epsilon_t \| q_t - q_t^0 \|_1 \big]. \end{aligned} \tag{23}$$

Thus the source correction is supported when its KL-regularized movement is large relative to measurement uncertainty and feedback staleness.

B.8 Supporting State-Dependence Result

Corollary 8 (Cost of a shared ranking)**.** *Consider two learner states and the same two candidate batches. Suppose a exceeds b in target utility by $\gamma_1 > 0$ at the first state, while b exceeds a by $\gamma_2 > 0$ at the second. A policy choosing a with the same probability ρ in both states has worst-state regret*

$$\max\{(1-\rho)\gamma_1, \rho\gamma_2\}.$$

Its minimum over $\rho \in [0, 1]$ is

$$\frac{\gamma_1 \gamma_2}{\gamma_1 + \gamma_2} > 0,$$

whereas state-conditioned choice between these two candidates incurs zero regret.

Proof. The first-state regret decreases linearly in ρ and the second increases linearly. The maximum is minimized where they are equal: $\rho^* = \gamma_1/(\gamma_1 + \gamma_2)$. Substitution gives the result. □

This corollary formalizes the ranking-reversal situation diagnosed empirically in Section 3; it is a conditional comparison between two candidates, not a claim about all training states.

C Detailed TangentMix Method and Implementation

C.1 Execution Interface and Overall Loop

Let $\mathcal{M}_t = (\theta_t, o_t)$ denote the current model and optimizer state, and let $\theta_t^+(a)$ be the parameters after applying data decision a. For recovery–preservation objective Φ_t, its current finite value is $G_t(a; \mathcal{M}_t) = \Phi_t(\theta_t) - \Phi_t(\theta_t^+(a))$. Because both the receiving state and the induced update evolve, the preferred data decision is itself a moving target.

At each step, the sampled source provides prompts for current-student rollouts; the strong teacher and the frozen starting specialist provide recovery and preservation supervision, respectively. TangentMix forms a small set of complete candidate batches and evaluates them through *counterfactual optimizer steps*: each candidate is temporarily applied from the same parameter and optimizer state, evaluated on shared calibration responses, and rolled back. We compare their finite recovery–preservation gains while penalizing local predictive movement with a predictive-Fisher sensitivity term. A first-order tangent analysis relates these gains to the alignment between each optimizer displacement and the current recovery–preservation descent direction. A selection set determines the candidate a_t^*; a disjoint feedback set then measures its incremental gain over a source-conditioned reference a_0 from the same saved state and updates an exponentially weighted source-utility memory. Only after this feedback measurement is the selected update committed once; the refreshed source evidence guides subsequent allocation.

TangentMix organizes each recovery update into Search Control, Update Evaluation, and Learning. The corresponding execution interface is

$$\underbrace{p_t \to s_t \to (\mathcal{A}_t, a_0)}_{\text{Search Control}} \to \underbrace{a_t^*}_{\text{Update Evaluation}} \to \underbrace{\left(\mathcal{M}_{t+1},\, C_t \to p_{t+1}\right)}_{\text{Learning}}.$$

C.2 Complete Execution Order

Algorithm 1 gives the complete order of source acquisition, candidate construction, same-state evaluation, independent feedback, update commitment, and source refresh.

Algorithm 1 TangentMix: search control, update evaluation, and learning

Require: $\theta_I, q_T, \{\mathcal{D}_s\}_{s=1}^{S}, \mathcal{D}_P, \mathcal{D}_A, \mathcal{D}_B, B_{\mathrm{tr}}$;
schedules $\{b_t \geq 2, \lambda_t\}_{t\geq 0}, \mathcal{T}_{\mathrm{refresh}}; \beta, \kappa, c_{\mathrm{fb}}, \rho, f$

$q_I \leftarrow p_{\theta_I}$; $\mathcal{M}_0 \leftarrow \mathrm{InitOpt}(\theta_I)$; $p_0 \leftarrow \mathrm{Prior}(\mathcal{M}_0; f)$
$m_0 \leftarrow \mathbf{0}$; $v_0 \leftarrow \mathbf{0}$; $t \leftarrow 0$; $n \leftarrow 0$
while $n + b_t \leq B_{\mathrm{tr}}$ **do**
  Search Control
  $s_t \leftarrow \mathrm{Acquire}(p_t)$; $Z_t \leftarrow \mathrm{Rollout}(\theta_t; \mathcal{D}_{s_t}, \mathcal{D}_P)$
  $(\mathcal{A}_t, a_0) \leftarrow \mathrm{Build}(Z_t; q_T, q_I, \beta, b_t)$ ▷ Section 4.2
  Update Evaluation
  $S_t \leftarrow \mathrm{Snapshot}(\mathcal{M}_t)$; $A \leftarrow \mathrm{Calib}(S_t; \mathcal{D}_A, q_T, q_I)$
  for $a \in \mathcal{A}_t$ **do**
    $g_t(a) \leftarrow \nabla_\theta L_{a,t}(\theta_t)$; $\mathcal{M}_t^+(a) \leftarrow \mathrm{OptStep}(S_t, g_t(a))$
    $\Delta_t(a) \leftarrow \theta_t^+(a) - \theta_t$; evaluate $u_{z,t}(a)$ for all $z \in A$ by Eq. (6)
    $\mathrm{Restore}(S_t)$
  end for
  Compute every $J_t^A(a)$ from $u_{z,t}(a)$ and $u_{z,t}(a_0)$ by Eq. (7)
  $a_t^* \in \arg\max_{a\in\mathcal{A}_t} J_t^A(a)$ ▷ A selects
  Learning
  $B \leftarrow \mathrm{Calib}(S_t; \mathcal{D}_B, q_T, q_I)$ ▷ independent of A
  Evaluate a_t^* and a_0 from S_t on the same B by Eq. (6), restoring each step
  $C_t \leftarrow U_B(a_t^*) - U_B(a_0)$; $x_t \leftarrow \tanh(C_t/c_{\mathrm{fb}})$
  $m_{t+1} \leftarrow \mathrm{MemUpdate}(m_t, v_t, s_t, x_t; \rho)$ ▷ Eq. (33)
  $v_{t+1} \leftarrow v_t$; $v_{s_t,t+1} \leftarrow v_{s_t,t} + 1$
  $\mathrm{Restore}(S_t)$; $\mathcal{M}_{t+1} \leftarrow \mathrm{OptStep}(S_t, g_t(a_t^*))$ ▷ commit once
  $n \leftarrow n + b_t$
  if $t + 1 \in \mathcal{T}_{\mathrm{refresh}}$ **then**
    $p_{t+1}^0 \leftarrow \mathrm{Prior}(\mathcal{M}_{t+1}; f)$; compute p_{t+1} by Eq. (36)
  else
    $p_{t+1} \leftarrow p_t$
  end if
  $t \leftarrow t + 1$
end while
$H \leftarrow t$; **return** $\theta_H = \mathrm{params}(\mathcal{M}_H)$

C.3 Search Control and Candidate Construction

TangentMix constructs candidate batches from task prompts and the current student's responses, without using ground-truth answers as training targets. Given a recovery source s_t selected under p_t, we sample recovery prompts from $\mathcal{D}_{s_t}$ and preservation prompts from $\mathcal{D}_P$. For each prompt x, the current student generates a response token sequence $y \sim p_{\theta_t}(\cdot \mid x)$, forming a trajectory $z = (x, y)$. We use these trajectories to construct a compact candidate family $\mathcal{A}_t$ with reference $a_0 \in \mathcal{A}_t$ for complete-update evaluation.

State-conditioned supervision. The fixed strong teacher q_T scores recovery trajectories, while the frozen starting specialist $q_I = p_{\theta_I}$ scores preservation trajectories. At each student-generated prefix $h_j = (x, y_{<j})$, the assigned reference q_r evaluates the probability $q_r(y_j \mid h_j)$ of the student's sampled token y_j. We compute the OPD correction $A_{r,j,t}(z)$ by clipping the reference–student log-probability difference and stopping its gradient. For recovery, we combine teacher-relative correction demand with representation diversity to propose trajectories. For preservation, we prioritize positive initialization-relative corrections, targeting sampled tokens to which q_I assigns higher probability than the current student does. These rules shortlist trajectories for complete-update evaluation; Appendix C.6 gives the proposal scores and greedy construction.

Finite recovery–preservation candidate family. TangentMix assembles the proposed trajectories into candidate batches. Each candidate $a = (R_a, P_a)$ contains a recovery subset R_a and a preservation subset P_a, with a total of b_t trajectories:

$$a = (R_a, P_a), \qquad |R_a| + |P_a| = b_t, \qquad 1 \leq |R_a| \leq b_t - 1.$$

At a fixed executed size, the family $\mathcal{A}_t$ varies the recovery–preservation composition and, within a fixed composition, the recovery trajectory identities. For same-composition variants, the preservation subset is held fixed while the recovery trajectories vary, providing controlled within-source alternatives at the same update size. TangentMix designates one candidate $a_0 \in \mathcal{A}_t$ as the source-conditioned reference for measuring incremental utility in both candidate selection and subsequent feedback. Section 4.3 compares these candidates counterfactually by realizing each complete update from the same parameter and optimizer state.

All candidates retain the same recovery and preservation roles and use the shared OPD objective in Eq. (5). Here, $N(S)$ counts effective response tokens in S, so each candidate's branch loss uses its own normalization; $A_{r,j,t}$ denotes the stopped sampled-token OPD correction. Both the corrections and λ_t are held fixed during candidate comparison. Candidate composition therefore changes trajectory membership without changing the prescribed branch weights. The resulting family $\mathcal{A}_t$, its designated reference a_0, and the candidate losses $L_{a,t}$ specify the alternatives passed to Section 4.3.

The exact proposal score, finite candidate family, and reusable candidate-gradient construction are collected in Appendix C.6.

C.4 Same-State Update Evaluation

Counterfactual optimizer steps. Given the candidate family $\mathcal{A}_t$ and its reference $a_0 \in \mathcal{A}_t$, TangentMix compares how each complete recovery–preservation batch would change the same current learner $\mathcal{M}_t = (\theta_t, o_t)$. Each candidate is temporarily executed and rolled back from the same parameter–optimizer snapshot, so its displacement captures the optimizer's response to the complete recovery–preservation batch. The gradient, temporary optimizer state, and realized parameter displacement are defined explicitly in Appendix C.6.

Recovery–preservation utility. Before comparing candidates, TangentMix generates a shared calibration set A of prompt–response pairs $z = (x, y)$ using the current student and prompts from $\mathcal{D}_A$, separate from candidate-training prompts. For each response, we compare the student's correction-weighted losses before and after the temporary update, keeping the response, its prefixes, and stopped reference corrections fixed across candidates. The resulting loss reductions measure recovery and preservation gains. Calibration panels use equally sized groups and equal response weights, and candidates share the same predictive-score draws. Positive $G_{T,z,t}$ and $G_{I,z,t}$ measure improvement along the teacher's recovery correction and the initialization-relative preservation correction, respectively, while $\widehat{K}_z(a)$ penalizes local predictive movement. The calibration losses and predictive-score convention are specified in Appendix C.6.1.

Reference-centered selection. On A, TangentMix compares each candidate with the source-conditioned reference a_0 using the reference-centered score in Eq. (7) and selects a_t^*. The bounded transform is applied before averaging, limiting each response's influence on the comparison; $\kappa > 0$ sets this response-level scale. The selected candidate and its gradient are retained for independent feedback and a single committed update.

C.5 Independent Feedback and Source Reallocation

Independent source credit. After selecting a_t^* on A, TangentMix generates a calibration panel B using the saved current student and prompts from $\mathcal{D}_B$, a bank disjoint from $\mathcal{D}_A$ and candidate-training prompts. Panel B is sampled independently of A with the same grouping and weighting convention. TangentMix evaluates the selected and reference updates from the same saved $\mathcal{M}_t$, using the same responses and predictive-score draws in B. Their utility difference measures the additional value obtained by candidate selection within the visited source and is defined as C_t in Eq. (32).

Updating the learner and source memory. With feedback scale $c_{\mathrm{fb}} > 0$, TangentMix bounds the feedback as $x_t = \tanh(C_t/c_{\mathrm{fb}})$ and updates only the visited source's exponentially weighted memory; other source entries remain unchanged. After this measurement, it restores the saved state and commits the selected gradient exactly once, obtaining $\mathcal{M}_{t+1}$.

Feedback-guided reallocation. At each allocation refresh, TangentMix combines current recovery demand with the incremental selection utility recorded for each source. It computes a floor-constrained recovery prior p_{t+1}^0 from the post-update learner's remaining gap and recent progress (Sun et al., 2026). The allocation objective and its closed form are given in Appendix C.7. The KL term anchors allocation to recovery demand, while memory favors sources that have yielded greater incremental selection value. Between refreshes, $p_{t+1} = p_t$; the updated learner produces new on-policy trajectories, and the updated allocation determines which sources are visited next.

C.6 Candidate, Loss, and Calibration Details

For a complete candidate batch a, the gradient, temporary optimizer state, and parameter displacement are

$$\begin{aligned} g_t(a) &= \nabla_\theta L_{a,t}(\theta_t), \\ \mathcal{M}_t^+(a) &= \mathrm{OptStep}(\mathcal{M}_t, g_t(a)), \\ \Delta_t(a) &= \theta_t^+(a) - \theta_t. \end{aligned} \tag{24}$$

Every temporary evaluation starts from the same saved learner state.

For response prefix $h_j = (x, y_{<j})$ and reference $r \in \{T, I\}$, the clipped, stopped correction is

$$A_{r,j,t} = \mathrm{sg}[\mathrm{clip}(\log q_r(y_j \mid h_j) - \log p_{\theta_t}(y_j \mid h_j), -c, c)], \quad r \in \{T, I\}. \tag{25}$$

The clipping threshold, reference roles, and stopped corrections are shared across the candidate comparison.

Correction-and-coverage proposal. For trajectory z, the inexpensive recovery score is $U_z = c_z\sqrt{u_z}\, e_z$, where c_z is the mean positive clipped teacher-relative correction, u_z is the fraction of tokens supported by the teacher at least as much as by the student, and e_z is the completion factor. Normalize U_z within the current source to $\widetilde{U}_z$; $\bar{h}_z$ is the normalized mean hidden state at response-prediction positions. For a nonempty selected set R, expand it by

$$z^* \in \arg\max_{z \notin R} \left\{ \widetilde{U}_z + \beta \min_{z' \in R} \left(1 - \bar{h}_z^\top \bar{h}_{z'}\right) \right\}. \tag{26}$$

The correction term prioritizes teacher-relative correction demand; the coverage term rewards distance from selected representations. To complete the constructor's mathematical definition at $R = \varnothing$, set its coverage bonus to zero and initialize $R = \{z_1\}$ with $z_1 \in \arg\max_z \widetilde{U}_z$; a fixed trajectory-ID order resolves ties. Preservation proposals use positive initialization-relative corrections. The designated recovery–preservation pair defines a_0; executing this reference directly yields the lower-overhead selection rule recorded in the earlier experiments.

Finite family specification. Let b_t denote the number of trajectories in the current committed update, consistent with the executed-data budget in Eq. (3), and let $\mathcal{R}_{b_t} = \{r_1, \ldots, r_K\} \subset \{1, \ldots, b_t - 1\}$ denote a small set of recovery cardinalities. For each r_k, the constructor may instantiate $M_k \geq 1$ identity variants,

$$a_{k,\ell} = (R_{k,\ell}, P_{k,\ell}), \qquad |R_{k,\ell}| = r_k, \quad |P_{k,\ell}| = b_t - r_k. \tag{27}$$

The family $\mathcal{A}_t$ collects these variants over $r_k \in \mathcal{R}_{b_t}$ and $1 \leq \ell \leq M_k$. All candidates therefore have the same executed size b_t but can differ along two axes. Changing r_k changes the recovery–preservation composition; using multiple ℓ values at the same r_k changes trajectory identities while holding the composition fixed. For same-composition identity variants, TangentMix keeps the preservation subset fixed and changes the selected recovery trajectories, yielding a direct within-source alternative at the same update size. One member $a_0 \in \mathcal{A}_t$ is designated as the source-conditioned reference candidate used by the relative selection and feedback stages below. The concrete cardinalities and identity multiplicities used in an experiment are an implementation configuration of this family rather than part of the method definition.

For the fixed initial reference, the potential identity underlying preservation gain is

$$\begin{aligned} D_{I,z,t}(\theta) &= \frac{1}{T_z} \sum_j A_{I,j,t} \big[\log q_I(y_j \mid h_j) - \log p_\theta(y_j \mid h_j) \big], \\ D_{I,z,t}(\theta_I) &= 0, \quad D_{I,z,t}(\theta_t) \geq 0, \\ G_{I,z,t}(a) &= D_{I,z,t}(\theta_t) - D_{I,z,t}(\theta_t + \Delta_t(a)). \end{aligned} \tag{28}$$

At the current parameters the clipped correction preserves the sign of the initial-reference log-probability difference, giving nonnegativity. The identity then follows by cancellation of the fixed reference term.

For the union $\mathcal{V}_t$ of role-qualified candidate trajectories, let $v_{r,z} = -\sum_j A_{r,j,t}(z)\nabla_\theta \log p_{\theta_t}(y_j \mid h_j)$ be each token-sum gradient. Candidate gradient reuse is

$$g_t(a) = \sum_{(r,z)\in\mathcal{V}_t} W_{a,(r,z)} v_{r,z},$$
$$W_{a,(T,z)} = \frac{\lambda_t \mathbf{1}[z \in R_a]}{N_R(a)}, \qquad W_{a,(I,z)} = \frac{(1-\lambda_t)\mathbf{1}[z \in P_a]}{N_P(a)}. \tag{29}$$

The coefficients use each candidate's own response-token denominators before clipping the complete gradient.

C.6.1 Calibration losses and predictive-score sampling

The disjoint prompt banks $\mathcal{D}_A$ and $\mathcal{D}_B$ are fixed before training and separate from candidate-training prompts. In Algorithm 1, Calib receives the saved learner, one bank, and the two references, and returns a current response panel with its frozen prefixes, stopped corrections, and predictive-score draws. It samples equal numbers of responses from each calibration group. The two calls use independent sampling, while candidates evaluated on the same panel reuse its annotations.

For a fixed response $z = (x, y)$, let $T_z > 0$ count its effective response positions and keep their prefixes $h_j = (x, y_{<j})$ fixed. The singleton branch loss in Eq. (5) is

$$\phi_{r,z,t}(\theta) = \ell_{r,t}(\theta; \{z\}) = -\frac{1}{T_z}\sum_{j=1}^{T_z} A_{r,j,t}(z) \log p_\theta(y_j \mid h_j). \tag{30}$$

The correction is frozen at θ_t; candidate evaluation changes only θ in the log-probabilities. For a fixed panel X, $\Phi_{X,t} = |X|^{-1}\sum_{z\in X}[\lambda_t\phi_{T,z,t} + (1-\lambda_t)\phi_{I,z,t}]$, and $U_X(a) = \Phi_{X,t}(\theta_t) - \Phi_{X,t}(\theta_t + \Delta_t(a)) - |X|^{-1}\sum_{z\in X}\widehat{K}_z(a)$. Thus the sensitivity penalty is separate from the signed finite gain.

Prefix-level score convention. The predictive score in the analysis is sampled conditional on the fixed prefixes, rather than computed from the likelihood of the observed response. Let $H_z \sim \mu_{z,t}$ be a draw from a normalized distribution on those effective prefixes, and sample $V_z \sim p_{\theta_t}(\cdot \mid H_z)$. The score draw and its quadratic estimate are

$$s_z = \nabla_\theta \log p_\theta(V_z \mid H_z)|_{\theta=\theta_t}, \qquad \widehat{K}_z(a) = \tfrac{1}{2}(s_z^\top \Delta_t(a))^2. \tag{31}$$

For equal effective-prefix weighting, $\mu_{z,t} = T_z^{-1}\sum_{j=1}^{T_z}\delta_{h_j}$; this normalizes sensitivity per response rather than by its unnormalized sequence likelihood. Writing $F_{z,t} = \mathbb{E}[s_z s_z^\top \mid z, \mathcal{M}_t]$ gives $\mathbb{E}[\widehat{K}_z(a) \mid z, \mathcal{M}_t, \Delta_t(a)] = \frac{1}{2}\Delta_t(a)^\top F_{z,t}\Delta_t(a)$. A Monte Carlo average of repeated score draws averages the squared projections, preserving the same expected sensitivity target. Draws are independent of the fixed candidate displacements and shared across candidates within each calibration row; selection and feedback use independent panels. The local KL expansion is proved in Appendix B.5.

Grouped calibration. For $X \in \{A, B\}$, a grouped mean is $U_X(a) = \sum_g \omega_g n_{X,g}^{-1}\sum_{i=1}^{n_{X,g}} u_{g,i,t}(a)$. The bounded selection score uses the same group weights after its per-response transformation. With $\omega_g = n_{X,g}/|X|$ this is exactly the uniform response mean; equal group sizes yield equal group weights. The recorded online panels below use one response from each of eight groups and equal weights. The general theory also permits other fixed weights, provided empirical scores and their population targets use the same convention.

Reusing candidate gradients and restoring state. Candidate batches may share role-qualified trajectories. Their reusable token-gradient contributions are cached using Eq. (29), but branch denominators and clipping apply to each complete candidate. All temporary candidate states are restored before the next evaluation. The winner and reference are subsequently compared on independent B evidence from that same snapshot; only the selected gradient is finally committed once. No temporary step is accumulated into the committed state.

C.7 Source-Memory and Allocation Details

C.7.1 Independent source credit

With $U_X(a) = |X|^{-1} \sum_{z \in X} u_{z,t}(a)$, independent credit is

$$C_t = U_B(a_t^*) - U_B(a_0). \tag{32}$$

Both candidates use the same feedback responses and predictive-score draws.

C.7.2 Source-utility memory and update order

Let $m_{s,t}$ be the utility memory of recovery source s, initially zero, and let $v_{s,t}$ count its visits before update t. After A has fixed a_t^*, the two candidates are evaluated on B from the same saved $\mathcal{M}_t$, using common responses and predictive-score draws. For the visited source s_t, set $x_t = \tanh(C_t/c_{\mathrm{fb}})$, $c_{\mathrm{fb}} > 0$, and update

$$m_{s,t+1} = \begin{cases} x_t, & s = s_t,\ v_{s,t} = 0, \\ (1-\rho) m_{s,t} + \rho x_t, & s = s_t,\ v_{s,t} > 0, \\ m_{s,t}, & s \neq s_t, \end{cases} \qquad 0 < \rho \leq 1. \tag{33}$$

The visit counts start at $v_{s,0} = 0$ and obey $v_{s,t+1} = v_{s,t} + \mathbf{1}\{s = s_t\}$. Algorithm 1 passes the pre-update counts to $\mathrm{MemUpdate}$ so the first-visit branch is explicit. The scale c_{fb} bounds source feedback, whereas κ bounds individual responses in candidate selection before averaging. After this feedback measurement and memory update, the saved learner state is restored and the selected gradient is committed once.

C.7.3 Source allocation and complete-update selection

Let $\mathcal{A}_{t,s}$ denote the candidate family that would be exposed by visiting recovery source s at state $\mathcal{M}_t$, so the realized family is $\mathcal{A}_t = \mathcal{A}_{t,s_t}$; let $\sigma_t(\mathcal{A}_{t,s})$ denote the selected complete update. For source distribution p_t,

$$\mathcal{V}_t(p_t, \sigma_t) = \sum_{s \in \mathcal{S}} p_{t,s}\, \mathbb{E}[U_t(\sigma_t(\mathcal{A}_{t,s}))]\,. \tag{34}$$

Thus p_t controls which recovery opportunities are exposed, whereas σ_t controls the update extracted from each exposed pool. The source controller should therefore combine remaining recovery demand with the value actually realized by current-state selection.

C.7.4 Floor-constrained prior and exponential reweighting

On source s's visit clock, $h_{s,k}$ is its smoothed reverse-KL estimate, h_s^0 is the early reference, and $\mathcal{W}_s$ is a recent window. The prior combines remaining gap and positive relative progress:

$$\begin{aligned} z_s &= \frac{h_{s,k}}{\max(h_s^0, \epsilon)} \left[\mathrm{mean}_{j \in \mathcal{W}_s} \frac{h_{s,j-1} - h_{s,j}}{\max(h_{s,j-1}, \epsilon)} \right]_+, \\ p_s^0 &= f + (1 - Sf)\, \mathrm{softmax}_s\!\left(\frac{z_s}{T \max_j(z_j, \epsilon)} \right). \end{aligned} \tag{35}$$

The floor f keeps each source available, and T controls concentration.

Allocation objective. Write $q_{t+1}^0 = (p_{t+1}^0 - f\mathbf{1})/(1 - Sf)$. The implemented allocation is

$$\begin{aligned} q_{t+1}^* &= \arg \max_{q \in \Delta^S,\ q \ll q_{t+1}^0} \{\langle q, m_{t+1} \rangle - \mathrm{KL}(q \| q_{t+1}^0)\}, \\ p_{t+1} &= f\mathbf{1} + (1 - Sf) q_{t+1}^*. \end{aligned} \tag{36}$$

Equation (38) gives its closed form. In Section 4.5, t indexes the receiving state after this refresh: $q_t = (p_t - f\mathbf{1})/(1 - Sf)$ is the implemented residual, whereas $q_t^\dagger$ uses current expected source feedback.

Given smoothed relative observations m_s, the residual $q_s^0 = (p_s^0 - f)/(1 - Sf)$ is adjusted by Eq. (36). The preservation stream uses its original mixture. Only the visited recovery source receives a new selected-batch observation at each step.

On a fixed calibration set X, write $\Phi_{X,t}(\theta) = |X|^{-1} \sum_{z \in X} [\lambda_t \phi_{T,z,t}(\theta) + (1 - \lambda_t)\phi_{I,z,t}(\theta)]$ and $\widehat{R}_X(a) = |X|^{-1} \sum_{z \in X} \widehat{K}_z(a)$. The common before-update term cancels in the relative feedback:

$$C_t = \Phi_{B,t}(\theta_t + \Delta_t(a_0)) - \Phi_{B,t}(\theta_t + \Delta_t(a_t^*)) - \big[\widehat{R}_B(a_t^*) - \widehat{R}_B(a_0)\big]. \tag{37}$$

Let $\mathcal{F}_t$ collect the complete history, current learner, candidate batches, and all observations in A. Condition on $\mathcal{F}_t$. Both a_t^* and a_0 are then fixed. For independently sampled current feedback observations in B, define $V_{B,t}(a) = \mathbb{E}[U_B(a) \mid \mathcal{F}_t]$. Linearity gives $\mathbb{E}[C_t \mid \mathcal{F}_t] = V_{B,t}(a_t^*) - V_{B,t}(a_0)$. This expectation is over the current prompt, response, and predictive-score sampling procedure on the fixed feedback bank. With one sample per training group, it equally weights the group-conditional expectations.

Explicit reweighting rule. For $p_{t+1}^0 \in \Delta^S$ with $p_{t+1,s}^0 \geq f$ and $0 \leq f < 1/S$,

$$\begin{aligned} q_{t+1,s}^0 &= \frac{p_{t+1,s}^0 - f}{1 - Sf}, \\ q_{t+1,s}^* &= \frac{q_{t+1,s}^0 \exp(m_{s,t+1})}{\sum_j q_{t+1,j}^0 \exp(m_{j,t+1})}, \\ p_{t+1,s} &= f + (1 - Sf) q_{t+1,s}^*. \end{aligned} \tag{38}$$

The prior in Eq. (35) satisfies this floor condition. Consequently q^0 is nonnegative and sums to one, as does q^*; $p_{t+1,s} \geq f$ follows after adding the floor back. For $q_s^0 = 0$, finite KL restricts q_s to zero, agreeing with the explicit formula. At non-refresh steps, $p_{t+1} = p_t$. The existing quota-aware acquisition schedule realizes this distribution as subsequent source visits.

For fixed m and prior residual q^0, the multiplicative rule in Eq. (36) solves

$$\max_{q \in \Delta_S,\, q \ll q^0} \{\langle q, m \rangle - \mathrm{KL}(q \| q^0)\}.$$

On the support of q^0, stationarity gives $\log(q_s/q_s^0) = m_s - 1 - \nu$, and normalization yields $q_s = q_s^0 e^{m_s} / \sum_j q_j^0 e^{m_j}$. Equal values of m_s recover $q = q^0$. The optimization concerns the current bounded source observations and regularizes their effect around the existing mixture.

C.8 TangentMix Implementation Settings

The source prior floors reverse-KL estimates at 10^{-8} and smooths observations with old/new weights $0.8/0.2$. Its early reference uses the first three smoothed visits. The improvement window includes at most three relative decreases. The first 20 updates use the natural mixture, followed by refreshes every 20 updates. The source floor is 0.02 and the temperature is 0.7. Within-source selection uses $\beta = 0.15$, clips support to $[0.05, 1]$, and sets the completion factor to 1 for stopped responses and 0.85 otherwise. Constant candidate utilities normalize to 0.5.

Each branch generates eight candidates. Order recovery candidates by selecting the first four with Eq. (26), then applying the same selector to the remaining four; order preservation candidates by their positive initial-reference correction. The main experiments evaluate $(R_{1:4}, P_{1:4})$, $(R_{1:6}, P_{1:2})$, $(R_{1:4} \cup R_{7:8}, P_{1:2})$, and $(R_{1:7}, P_{1:1})$. Their role-qualified gradient union contains eight recovery and four preservation trajectories. Each candidate uses its own token denominator. Global gradient clipping has threshold 1, and the correction clip in Eq. (25) is 5.

Calibration prompts are separated from training candidates. Within each subcapability, 32 calibration prompts are sorted by identifier and split into eight selection prompts in $\mathcal{D}_A$, eight feedback prompts in $\mathcal{D}_B$, and sixteen diagnostic prompts. Each current panel A or B independently covers the eight training groups, drawing a child capability and a prompt from its bank within each group and generating a current-student response. Their utilities average the eight rows equally. The visited source observes $\tanh(C_t/0.01)$; subsequent observations use old/new EMA weights $0.8/0.2$.

C.9 Notation Reference

The core symbols used across the execution description are summarized in Table 3. Theory-specific quantities are defined where they are introduced in the corresponding analysis and proofs.

Table 3: Core notation for the TangentMix execution loop.

Symbol	Meaning
$\mathcal{M}_t = (\theta_t, o_t)$	Current learner state, including model parameters and optimizer state.
$q_T,\ q_I = p_{\theta_I}$	Strong teacher for recovery and frozen starting specialist for preservation.
$\mathcal{D}_s,\ \mathcal{D}_P$	Recovery-source prompt pool and preservation prompt stream.
$p_t,\ s_t \sim p_t$	Current recovery-source distribution and source visited at update t.
$z = (x, y)$	Current-student trajectory consisting of a prompt and its on-policy response.
$A_{r,j,t}(z)$	Clipped, stop-gradient reference–student correction for role $r \in \{T, I\}$.
$a = (R_a, P_a)$	Complete candidate batch with recovery and preservation trajectory subsets.
$\mathcal{A}_t,\ a_0,\ a_t^*$	Candidate family, source-conditioned reference, and selected candidate.
$b_t,\ \lambda_t,\ \beta$	Committed batch size, recovery–preservation branch weight, and proposal-diversity weight.
$L_{a,t}(\theta)$	Complete OPD loss induced by candidate a.
$\Delta_t(a)$	Realized parameter displacement from temporarily applying candidate a at $\mathcal{M}_t$.
$\mathcal{D}_A, \mathcal{D}_B$	Disjoint calibration prompt banks for selection and independent feedback.
A, B	Current-student calibration panels generated from $\mathcal{D}_A$ and $\mathcal{D}_B$.
$G_{r,z,t}(a)$	Finite recovery or preservation gain of candidate a on calibration response z.
$\widehat{K}_z(a)$	Predictive-Fisher sensitivity penalty for the candidate displacement.
$u_{z,t}(a),\ J_t^A(a)$	Response-level utility and bounded reference-centered selection score.
κ	Scale of the bounded transformation used in candidate selection.
C_t	Independent selected-versus-reference feedback $U_B(a_t^*) - U_B(a_0)$.
c_{fb}, ρ	Feedback-bounding scale and source-memory EMA update rate.
$m_{s,t}$	Utility memory maintained for recovery source s.
$p_t^0,\ q_t^0,\ q_t^*$	Recovery-demand prior, floor-removed residual prior, and utility-reweighted residual distribution.
f	Minimum exposure probability assigned to each recovery source.
$B_{\mathrm{tr}},\ \mathcal{T}_{\mathrm{refresh}}$	Executed-trajectory budget and source-allocation refresh schedule.

D Experimental and Baseline Details

TangentMix-specific candidate, calibration, and source-control settings are given in Appendix C.8; the recovery settings, reporting conventions, and baseline implementations follow below.

D.1 Recovery settings and reporting details

Recovery is evaluated over seven capability axes using MMLU, BBH, TyDiQA, GSM8K, MATH500, EvalPlus, and IFEval, with AIME24/25 providing an additional competition-level reasoning evaluation.

The three settings represent distinct post-training recovery regimes: Complementary, where teacher and specialist retain complementary strengths, with the specialist remaining stronger in multilingual comprehension and the teacher providing moderate recovery headroom across the remaining capabilities; Dominant, where the teacher is stronger across all evaluated capabilities, representing a recovery-dominant regime without an initial specialist-over-teacher advantage; and Retention, where the specialist preserves a pronounced multilingual advantage but requires substantial recovery in code, instruction following, mathematics, and reasoning, creating the strongest recovery–preservation tension.

ProfileScore rescales each axis between Init and the strong teacher, clips to $[0, 1]$, and averages across axes. AIME reports Avg@8 and Pass@8.

We build on slime (Zhu et al., 2025) and adapt students with LoRA. Student training and rollout use two RTX 4090 GPUs with a learning rate of 6×10^{-6}. Training budget is measured by executed student trajectories. Profile-specific recovery–preservation weights and decoding settings are reported in this appendix.

On Dominant, OPD and TIDE recover MATH500 to 73.80 and 73.60 but reduce BBH to 67.43 and 65.50; TangentMix instead reaches 75.80 on MATH500 and 75.03 on BBH, improving both over Init.

Table 4: Composite capability evaluation suite. Counts are evaluated questions/programs; AIME uses eight sampled responses per question.

Dataset	Examples	Capability	Evaluation metric
MMLU	14,042	World knowledge	Accuracy; macro-average over 57 subjects.
BBH	6,511	Compositional reasoning	Exact-match accuracy; macro-average over 27 task groups.
TyDiQA	1,713	Multilingual comprehension	Token F1; macro-average over 9 languages.
GSM8K	1,319	Arithmetic reasoning	Final-answer accuracy.
MATH500	500	Mathematical reasoning	Verified final-answer accuracy.
HumanEval+	164	Code generation	Execution pass@1.
MBPP+	378	Code generation	Execution pass@1.
IFEval	541	Instruction following	Strict prompt-level success rate.
AIME24	30	Competition mathematics	Avg@8 and Pass@8.
AIME25	30	Competition mathematics	Avg@8 and Pass@8.

D.2 Budget targets

A desired capability vector a^{tar} combines the Init-relative retention floor with at least one recovery target strictly above its initial score. Its ideal budget requirement is

$$B^*_\pi(a^{\mathrm{tar}}) = \inf\{B_{\mathrm{tr}} :\ A_d(\theta_{\pi,B_{\mathrm{tr}}}) \geq a_d^{\mathrm{tar}},\ \forall d\}, \qquad a_d^{\mathrm{tar}} \geq A_d(\theta_I) - \tau_d. \tag{39}$$

An unattained target has $B^*_\pi = +\infty$. Experiments report the smallest evaluated budget meeting the stated criterion. The aggregate comparison in Section 5.3 uses the measured dual-teacher aggregate together with retention above Init on all seven axes; its target differs from full vector dominance of that comparator.

D.3 Detail of baseline implementation

MOPD samples eight trajectories uniformly from the pool, assigning four to the strong teacher and four to the frozen initial model. Each branch receives weight 0.5. CaMOPD alternates three recovery updates with one preservation update and selects 80% of positive gap mass from 16 candidates, using recovery and preservation scales of 2 and 1. At 1,600 executed examples this schedule yields 166 optimizer updates. TIDE uses top-16 teacher scores and retention fractions of 0.2 for both positive and negative groups. SFT uses dataset-provided solutions. These comparisons share a learning rate of 6×10^{-6} and report their own normalization rules.

GKD uses student-generated contexts with on-policy fraction one. At every response position, a token is sampled from the teacher distribution and used in the student's cross-entropy gradient, with response-token means averaged across sequences. The sampled label leaves the student-generated continuation intact. Thus, the implemented estimator uses every response prefix and Monte Carlo teacher labels for the forward-KL gradient. TIDE and SFT use global response-token means; MOPD uses branchwise token means, and CaMOPD averages token means across selected sequences.

For the template-aligned 1.7B study, the strong teacher renders the original user messages using its own chat template. Its scores are mapped onto the student's response mask by response-token order, including EOS, with response-token identity checked across renderings. The frozen initialization keeps the student's template. The corrected prefix is used from the first update, including the shared 20-update prefix. This applies the same response-level supervision contract to the main method and its random-selection comparator.

Table 6: Full capability vectors for the retained Qwen3-1.7B cross-scale evaluations.

Method	MMLU	BBH	TyDiQA	GSM8K	MATH500	Code	IFEval	Macro-7	ProfileScore
Strong teacher	85.28	77.95	57.70	95.30	73.80	80.81	82.26	79.01	0.857
Init	59.49	45.40	62.59	56.56	38.00	44.32	32.72	48.44	0.143
SFT	60.00	46.07	64.78	41.55	42.40	39.21	28.65	46.09	0.166
OPD	61.98	47.20	61.49	59.29	49.40	55.27	29.76	52.06	0.231
GKD	62.21	53.34	59.08	64.44	60.80	56.88	31.79	55.51	0.259
MOPD	60.38	44.40	59.49	30.78	42.00	44.51	30.31	44.55	0.074
TIDE	62.88	51.01	57.95	62.77	61.40	57.72	31.61	55.05	0.220
CaMOPD	60.72	44.85	63.34	32.07	41.80	46.20	29.39	45.48	0.172
TangentMix	62.80	54.40	61.31	79.45	59.00	60.65	38.63	**59.46**	**0.413**

E Additional Experimental Results and Analyses

E.1 Full Specialist-Recovery Results

Table 5: Full capability vectors for the three Qwen3-4B specialist profiles.

Method	MMLU Acc.	BBH EM	TyDiQA F1	GSM8K Acc.	MATH500 Acc.	Code Pass@1	IFEval Strict	Macro-7 Avg.	ProfileScore [0,1]
Complementary initialization									
Strong teacher	85.28	77.95	57.70	95.30	73.80	80.81	82.26	79.01	0.857
Init	69.82	71.59	68.33	84.61	58.40	66.06	75.60	70.63	0.143
SFT	70.04	70.24	69.20	69.75	43.80	62.13	67.10	64.61	0.145
OPSD	74.49	72.06	69.64	84.38	62.20	67.03	79.11	72.70	0.317
OPD	78.21	74.40	68.04	92.27	72.40	74.33	80.59	77.18	0.699
GKD	78.35	74.33	60.47	92.57	73.00	73.24	**81.70**	76.24	0.620
MOPD	77.62	74.91	**70.29**	91.13	75.00	71.88	80.22	77.29	0.675
TIDE	78.32	75.42	65.20	92.34	73.00	74.42	80.78	77.07	0.696
CaMOPD	77.47	74.20	64.69	91.21	75.00	71.05	78.00	75.95	0.554
DIEM-OPD	78.12	**75.73**	67.55	**93.10**	74.00	74.12	79.48	77.44	0.720
TangentMix	**78.36**	75.27	69.18	92.95	**76.00**	**74.73**	80.22	**78.10**	**0.742**
Teacher-dominant initialization									
Strong teacher	85.28	77.95	57.70	95.30	73.80	80.81	82.26	79.01	1.000
Init	74.16	74.42	49.65	85.52	52.40	65.81	75.97	68.28	0.000
SFT	73.67	74.36	**69.80**	77.41	48.60	63.97	74.49	68.90	0.143
OPSD	72.87	75.23	52.33	84.31	54.40	64.71	73.57	68.20	0.094
OPD	78.39	67.43	50.44	92.72	73.80	71.33	**82.44**	73.79	0.512
GKD	78.09	68.25	48.49	92.42	72.60	69.55	81.33	72.96	0.443
MOPD	77.87	**76.61**	50.26	91.51	75.00	68.32	81.52	74.44	0.527
TIDE	77.93	65.50	49.91	**92.87**	73.60	**71.69**	79.48	73.00	0.438
CaMOPD	77.52	75.89	50.09	90.90	75.00	68.62	80.78	74.11	0.468
DIEM-OPD	**78.43**	67.40	50.41	**92.87**	74.20	71.16	81.52	73.71	0.496
TangentMix	78.00	75.03	56.55	92.49	**75.80**	71.04	81.52	**75.78**	**0.617**
Retention-critical initialization									
Strong teacher	85.28	77.95	57.70	95.30	73.80	80.81	82.26	79.01	0.857
Init	69.23	70.73	72.37	82.41	63.20	63.88	66.54	69.77	0.143
SFT	70.46	70.75	72.13	79.45	52.60	61.52	65.43	67.48	0.152
OPSD	74.40	72.46	**72.51**	86.35	**73.80**	66.90	73.01	74.20	0.494
OPD	76.47	73.95	71.10	90.83	71.40	70.18	76.16	75.73	0.603
GKD	76.79	73.96	69.74	91.51	71.00	71.93	78.37	76.19	0.630
MOPD	70.91	72.52	72.31	87.11	73.00	67.24	72.83	73.70	0.462
TIDE	77.22	74.85	69.30	91.81	71.60	72.86	80.41	76.86	0.685
CaMOPD	66.82	71.79	69.24	86.81	71.00	68.29	70.79	72.11	0.363
DIEM-OPD	76.51	73.95	71.05	91.66	71.40	71.24	78.93	76.39	0.646
TangentMix	**78.17**	**75.74**	68.08	**91.89**	71.40	**74.82**	**80.59**	**77.24**	**0.715**

E.2 Static Data-Selection Baselines

Table 7: Static data-selection baselines across four specialist-recovery settings. LESS-style denotes a selector based on the LESS criterion rather than the full LESS training pipeline.

Method	MMLU	BBH	TyDiQA	GSM8K	MATH500	Code	IFEval	Macro-7	ProfileScore
Qwen3-4B Complementary (DSIR Init)									
Gap-based	78.38	74.35	68.84	90.98	75.00	71.88	80.41	77.12	**0.671**
DSIR	77.56	73.93	68.73	90.37	75.80	71.71	79.48	76.80	0.625
IFD	78.15	74.09	70.57	90.98	76.20	71.84	80.41	**77.46**	0.663
LESS-style	77.34	74.13	70.39	90.98	75.80	72.01	78.93	77.08	0.627
Qwen3-4B Teacher-dominant (S_code Init)									
Gap-based	78.15	75.64	52.79	90.90	75.40	67.84	80.78	74.50	0.507
DSIR	77.48	76.06	52.71	91.13	74.20	67.88	82.81	74.61	0.551
IFD	77.94	75.99	53.76	91.05	73.40	68.62	82.26	74.72	0.576
LESS-style	77.27	76.57	59.98	90.22	74.80	69.29	80.22	**75.48**	**0.611**
Qwen3-4B Retention-critical (S_reasoning Init)									
Gap-based	72.13	73.50	71.50	90.14	72.20	71.23	75.97	**75.24**	**0.570**
DSIR	71.01	73.23	70.80	89.84	72.80	70.36	74.86	74.70	0.535
IFD	71.82	73.28	71.76	89.76	73.20	69.83	75.97	75.09	0.563
LESS-style	67.34	72.28	71.48	87.87	73.60	68.64	74.31	73.64	0.476
Qwen3-1.7B Cross-scale (S_reasoning Init)									
Gap-based	60.83	54.06	64.04	74.53	46.20	50.21	33.27	54.73	0.312
DSIR	60.06	54.08	64.87	73.84	45.60	49.69	33.83	54.57	0.302
IFD	60.88	54.39	64.73	75.36	47.20	52.04	35.30	**55.70**	**0.334**
LESS-style	60.56	53.92	64.88	74.30	45.00	47.73	32.35	54.10	0.293

We extend the static selection comparison across all four recovery settings using the same evaluation suite. The strongest static criterion changes with the receiving specialist: IFD gives the highest Macro-7 among the tested selectors on Complementary and the 1.7B cross-scale setting, LESS-style is strongest on Dominant, and Gap-based is strongest on Retention. TangentMix nevertheless exceeds the best static selector by 0.64, 0.30, 2.00, and 3.76 Macro-7 points in the four settings, respectively, while also attaining higher ProfileScore in every setting. The changing ordering among static criteria shows that no single fixed example-level signal is consistently preferred across recovery states. This extends the state-dependence observed in Section 3 and supports evaluating data through the complete update it induces at the current learner.

E.3 Question-Level Mathematical Recovery

Table 8: Full AIME24/25 recovery results.

	GSM8K	MATH500	AIME24		AIME25		AIME mean	
Method	Acc.	Acc.	Avg@8	Pass@8	Avg@8	Pass@8	Avg@8	Pass@8
Strong teacher	95.30	73.80	76.25	90.00	61.25	83.33	68.75	86.67
Init	82.41	63.20	36.25	76.67	38.75	66.67	37.50	71.67
SFT	79.45	52.60	11.67	36.67	13.33	40.00	12.50	38.33
OPD	90.83	<u>71.40</u>	<u>59.17</u>	76.67	47.92	<u>70.00</u>	<u>53.54</u>	73.33
GKD	91.51	71.00	54.58	76.67	<u>49.17</u>	**73.33**	51.88	75.00
MOPD	87.11	**73.00**	56.67	<u>83.33</u>	48.75	**73.33**	52.71	**78.33**
TIDE	91.51	71.20	57.92	80.00	48.75	<u>70.00</u>	53.33	75.00
CaMOPD	86.81	71.00	55.00	**86.67**	45.00	66.67	50.00	<u>76.67</u>
DIEM-OPD	<u>91.66</u>	<u>71.40</u>	57.50	80.00	<u>49.17</u>	**73.33**	53.33	<u>76.67</u>
TangentMix	**91.89**	<u>71.40</u>	**60.83**	<u>83.33</u>	**50.00**	**73.33**	**55.42**	**78.33**

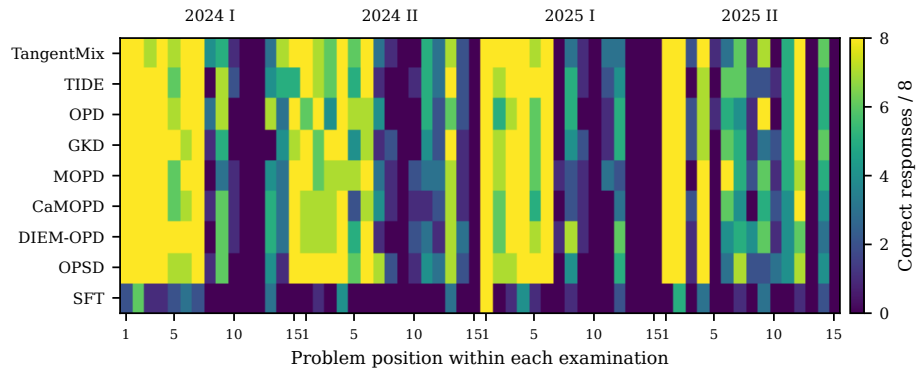


Figure 4: Question-level AIME24/25 success counts out of eight responses.

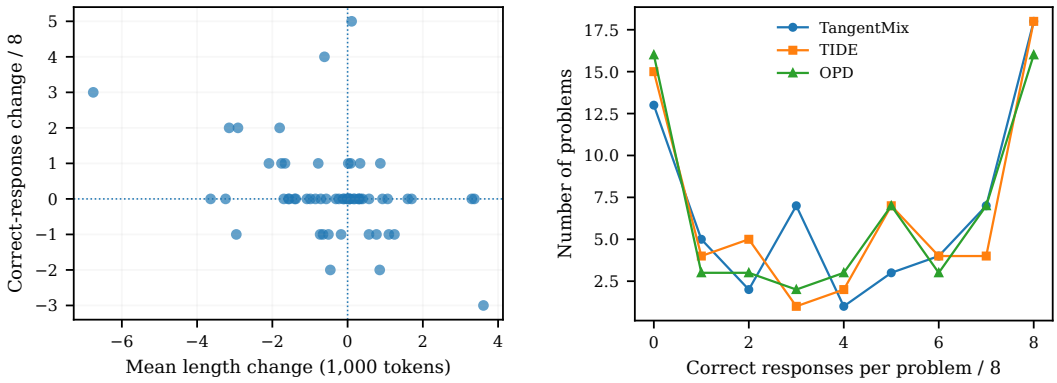


Figure 5: Question-level AIME behavior. Left: paired change from TIDE in mean response length and correct-response count. Right: distribution of success counts over AIME24/25 questions.

To examine how the aggregate AIME gain is distributed across problems, we compare question-level success counts under the same AIME24/25 evaluation. TangentMix solves 47 of 60 questions at least once and produces 266 correct sampled responses, compared with 45 questions and 256 correct responses for TIDE; it improves on 14 questions, ties on 34, and trails on 12. The additional successes are distributed across more questions rather than appearing only as repeated successes on the same solved problems. This extends the aggregate reasoning result to the level of individual competition problems.

E.4 Recovery and Forgetting Dynamics

Table 9 and the full-vector record below expand the trajectory analysis in Section 5.3.

Table 9: Teacher-dominant recovery over matched training budgets.

	TangentMix					TIDE					CaMOPD				
Update	BBH ↑	TyDiQA ↑	MATH500 ↑	Macro-7 ↑	$\Delta_{\min}$ ↑	BBH ↑	TyDiQA ↑	MATH500 ↑	Macro-7 ↑	$\Delta_{\min}$ ↑	BBH ↑	TyDiQA ↑	MATH500 ↑	Macro-7 ↑	$\Delta_{\min}$ ↑
Teacher	77.95	57.70	73.80	79.01	+3.53	77.95	57.70	73.80	79.01	+3.53	77.95	57.70	73.80	79.01	+3.53
Init	74.42	49.65	52.40	68.28	+0.00	74.42	49.65	52.40	68.28	+0.00	74.42	49.65	52.40	68.28	+0.00
100	**75.85**	51.28	73.00	**73.76**	**+1.43**	70.28	**51.30**	**73.20**	73.55	-4.14	75.47	47.36	67.60	71.81	-2.29
200	**75.93**	49.85	**75.00**	**74.18**	+0.20	65.50	49.91	73.60	73.00	-8.92	75.89	**50.09**	**75.00**	74.11	**+0.44**
300	76.05	51.90	73.80	74.52	+1.63	65.63	52.43	71.60	73.19	-8.79	**76.22**	**53.96**	**74.80**	**75.00**	**+1.80**
400	**76.25**	50.79	75.00	**74.59**	**+1.14**	65.62	**52.89**	69.40	73.08	-8.80	75.53	48.59	**75.20**	74.29	-1.06
500	75.03	**56.55**	**75.80**	**75.78**	+0.61	65.83	53.68	71.40	73.57	-8.59	75.49	50.83	74.40	74.51	**+1.07**

Table 10: Teacher-dominant seven-axis suite vectors from the measured data-efficiency export (%).

Method	H_{eq}	MMLU	BBH	TyDiQA	GSM8K	MATH	Code	IFEval	Macro-7
TangentMix	100	76.90	75.85	51.28	89.99	73.00	67.80	81.52	73.76
TangentMix	200	78.03	75.93	49.85	90.75	75.00	70.24	79.48	74.18
TangentMix	300	77.49	76.05	51.90	91.74	73.80	70.64	80.04	74.52
TangentMix	400	77.50	76.25	50.79	91.13	75.00	71.60	79.85	74.59
TangentMix	500	78.00	75.03	56.55	92.49	75.80	71.04	81.52	75.78
CaMOPD	100	76.17	75.47	47.36	89.54	67.60	67.63	78.93	71.81
CaMOPD	200	77.52	75.89	50.09	90.90	75.00	68.62	80.78	74.11
CaMOPD	300	78.21	76.22	53.96	91.13	74.80	69.72	80.96	75.00
CaMOPD	400	77.76	75.53	48.59	92.49	75.20	69.15	81.33	74.29
CaMOPD	500	78.14	75.49	50.83	92.95	74.40	68.58	81.15	74.51
TIDE	100	77.76	70.28	51.30	92.27	73.20	70.16	79.85	73.55
TIDE	200	77.93	65.50	49.91	92.87	73.60	71.69	79.48	73.00
TIDE	300	78.06	65.63	52.43	92.34	71.60	72.43	79.85	73.19
TIDE	400	78.02	65.62	52.89	92.72	69.40	72.35	80.59	73.08
TIDE	500	78.33	65.83	53.68	92.65	71.40	71.74	81.33	73.57

E.4.1 Example-Level Recovery and Forgetting

Table 11: Example-level recovery and forgetting.

	Binary-item counts			Macro-7 contributions (points)		
Method	Recovered	Forgotten	Net	Recovery	Forgetting	Net
TangentMix	1,895	1,180	+715	11.03	3.98	+7.05
TIDE	2,182	1,964	+218	11.56	6.27	+5.29
CaMOPD	2,017	1,225	+792	10.66	4.43	+6.23

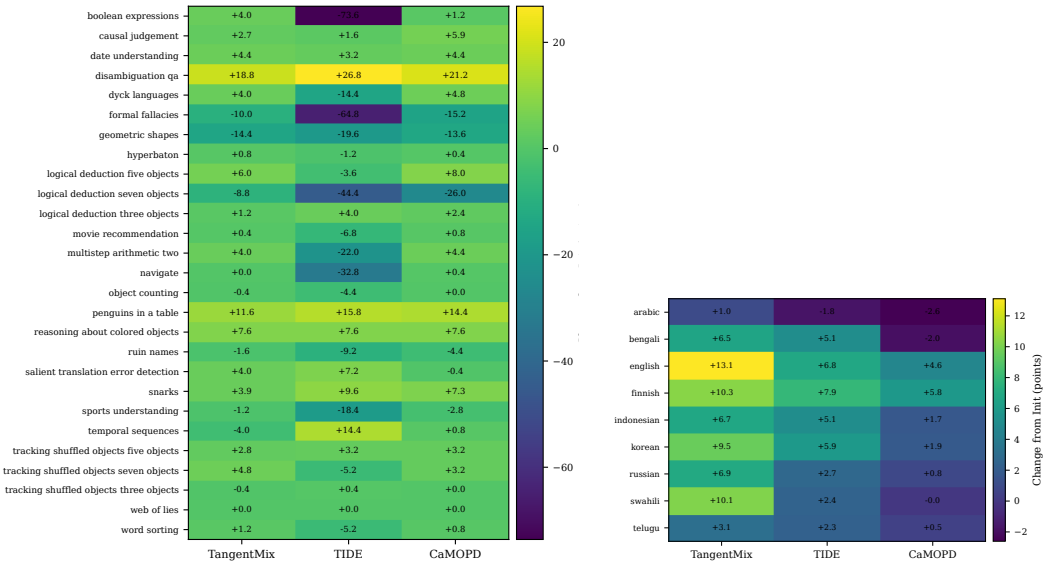


Figure 6: Task- and language-level recovery breakdown. Left: BBH accuracy change. Right: TyDiQA F1 change. Both are measured relative to the same initialization.

To unpack the aggregate recovery and forgetting curves, we decompose final behavior into recovered and forgotten evaluation items and inspect the corresponding BBH task and TyDiQA language changes. TangentMix records 1,895 recovered and 1,180 forgotten binary items, corresponding to 11.03 recovery points and 3.98 forgetting points; TIDE records 2,182 and 1,964 items with 11.56 and 6.27 points, while CaMOPD records 2,017 and 1,225 items with 10.66 and 4.43 points. TangentMix therefore retains a comparable recovered contribution to TIDE while preserving substantially more previously correct behavior, and the task/language views localize these aggregate

contributions across heterogeneous evaluation groups. This supports the role of state-conditioned data decisions in maintaining the recovery–preservation balance as training proceeds.

E.5 State-Conditioned Decision Diagnostics

E.5.1 Local Update-Signal Fidelity

To test whether the local quantities used for current-state update comparison track their finite-step targets, we evaluate the directional-gain and predictive-Fisher approximations on three realized adapter displacements. The directional approximation preserves the measured ordering with Spearman correlations of 1.000, 1.000, and 0.999 and weighted relative errors of 0.55%, 1.28%, and 2.93%; the predictive-Fisher approximation has relative errors of 1.27%, 2.51%, and 5.01% against exact predictive KL. The directional term therefore preserves candidate-relevant ordering while the quadratic sensitivity term remains close to local predictive movement over the measured displacements. These measurements support using both quantities to compare complete candidate updates at the current learner state.

E.5.2 Training and Allocation Dynamics

We analyze 300 consecutive TangentMix updates and 2,400 committed trajectories to connect current supervision, source feedback, and the training exposure ultimately received by the learner.

Teacher-relative correction dynamics. Figure 7 tracks the mean absolute teacher–student token log-probability difference on current calibration responses, with each source normalized by its first 20-update window. From the first to the last measured window, the correction magnitude decreases by 33.0% for mathematics (0.496 to 0.332) and increases by 30.7% for knowledge (0.866 to 1.131); reasoning also shows pronounced late increases. These trajectories show that the amount and temporal pattern of corrective supervision change differently across sources as the learner evolves. Recomputing recovery demand at the current learner state therefore supplies information that a fixed source identity cannot.

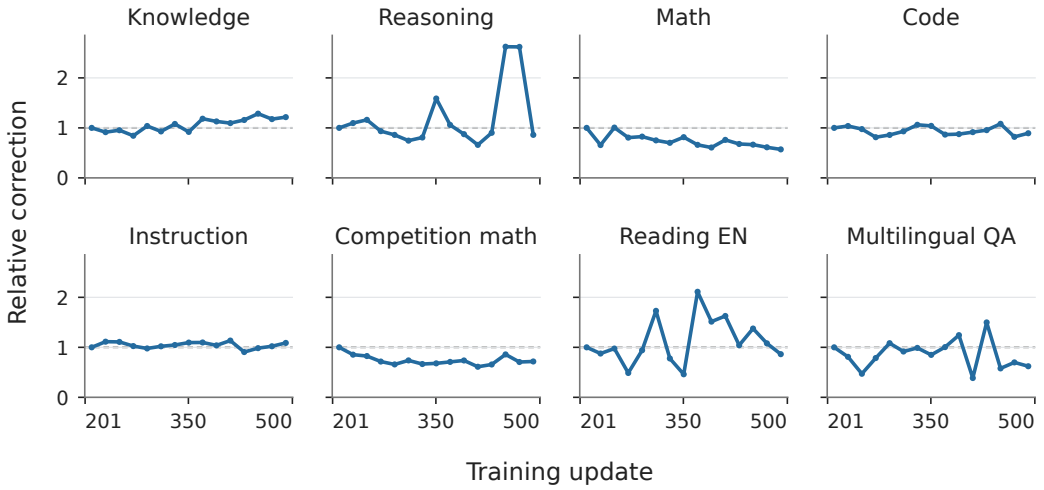


Figure 7: Teacher-relative correction dynamics. Each point averages response-level mean absolute token log-probability differences within a 20-update window, then divides by that source's first-window value. All panels share the same axes; the dashed line marks the initial level.

Source-memory and sampling dynamics. Figures 8 and 9 align source utility memory with sampling probabilities over the same windows. Reasoning and code move between negative and positive memory, while mathematics remains negative for most of the trace; over the same period, code's mean sampling probability falls from 21.6% to 9.6%, whereas multilingual QA rises from 7.2% to 16.4%. The recovery-demand prior is therefore reshaped by accumulated selected-versus-reference

utility rather than remaining at its initial proportions. This is the mechanism by which current-state feedback is converted into subsequent source exposure.

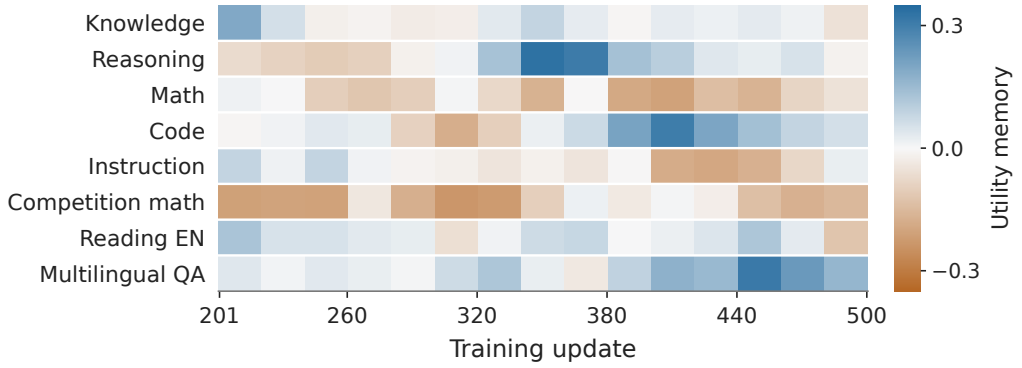


Figure 8: Source utility memory. Cells show 20-update means of the controller's pre-update memory; blue denotes positive values and orange negative values, with white centered at zero.

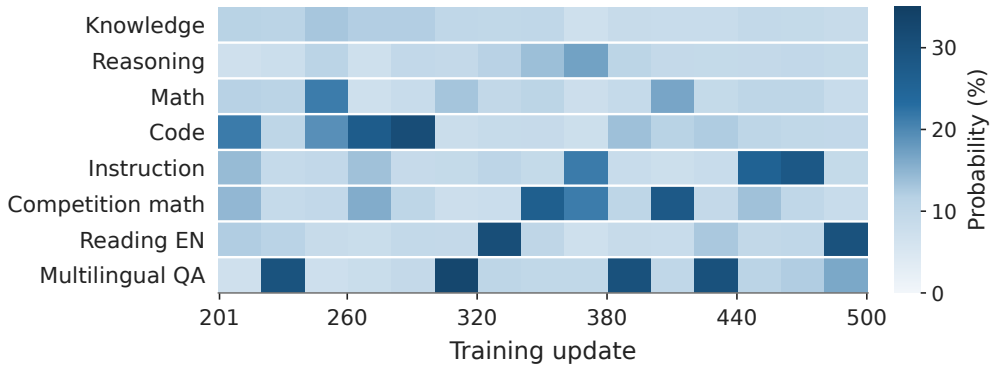


Figure 9: Source sampling probabilities. Cells show 20-update mean probabilities, using the same source order and time windows as Figure 8.

Executed training exposure. We finally compare source allocation with the examples and response tokens that are actually committed. Across the trace, 222 of 300 updates (74.0%) select a non-reference candidate, and 1,717 of 2,400 committed trajectories (71.5%) belong to the recovery branch; code contributes 11.0% of committed examples but 18.3% of response tokens, competition mathematics contributes 14.4% and 24.9%, and English reading contributes 10.6% and 0.45%, respectively. Candidate selection changes which trajectories are executed, while source-dependent response lengths further change their token-level contribution. The allocation–selection loop thus controls both which supervision is committed and how much response-token exposure it contributes to the learner.

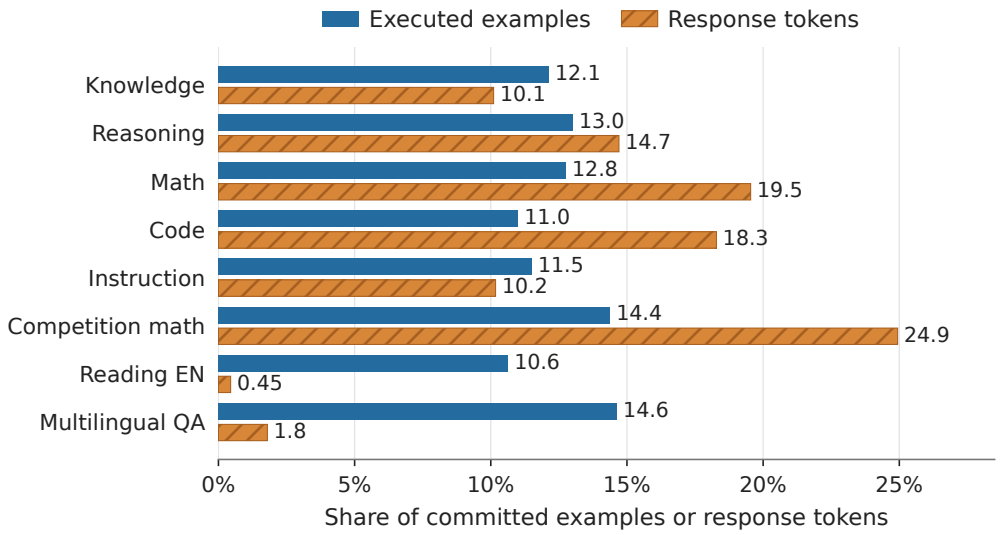


Figure 10: Committed training exposure by source. Example shares use all 2,400 committed trajectories as the denominator; token shares use their total response-token count.

E.6 Runtime and Systems Optimization

Table 12: Per-update wall-clock runtime of TangentMix and the evaluated online-distillation methods under their released deployments.

Method	Seconds / update ↓	Relative runtime
OPD	12.95	1.63×
MOPD	13.99	1.76×
TIDE	14.42	1.82×
CaMOPD	15.91	2.00×
DIEM-OPD	27.66	3.49×
GKD	31.03	3.91×
OPSD	57.01	7.18×
TangentMix	**7.94**	**1.00×**

We measure the wall-clock time of the complete TangentMix update pipeline, including on-policy rollout, recovery–preservation scoring, candidate-update evaluation, current-state selection, independent feedback, source-memory update, and the final committed optimizer step. TangentMix requires 7.94 s per update, compared with 12.95–57.01 s for the evaluated online-distillation baselines, corresponding to a 1.63–7.18× speedup. This efficiency comes from packing variable-length sequences, restricting scoring and gradient computation to response positions, reusing segmented candidate-side computations, and overlapping candidate scoring with predictive-Fisher projection across the two GPUs. The resulting execution path makes state-conditioned complete-update valuation compatible with efficient iterative post-training rather than introducing a wall-clock bottleneck.

F Diagnostic Protocol and Supporting Results for Section 3

F.1 Diagnostic Protocol

Section 3 uses Qwen3-4B-Instruct-2507 as the student backbone and Qwen3-30B-A3B-Instruct-2507 as the strong teacher. The receiving model is a code-specialized Qwen3-4B checkpoint. Table 13 reports the five capability axes used in the diagnostic analysis.

Table 13: Capability profile used in the Section 3 diagnostic analysis.

Capability	Evaluation	Specialist	Teacher	Headroom
Mathematics	MATH500	52.40	73.80	21.40
Code	EvalPlus	65.81	80.81	15.00
Reasoning	BBH	74.42	77.95	3.53
Knowledge	MMLU	74.16	85.28	11.12
Instruction Following	IFEval	75.97	82.26	6.29

The source-structured prompt pool contains 12,000 training prompts covering knowledge, reasoning, mathematics, code, instruction following, competition mathematics, English reading, and multilingual QA. The diagnostic analysis operates on mathematics (M), code (C), and reasoning (R) source families. Balanced uses M/C/R proportions of $1/3, 1/3, 1/3$. Math-heavy, Code-heavy, and Reasoning-heavy use $0.90/0.05/0.05$, $0.05/0.90/0.05$, and $0.05/0.05/0.90$, respectively. At each optimizer update, eight prompts are sampled according to the prescribed mixture, the current student generates on-policy responses, the teacher supervises the visited prefixes, and one ordinary OPD update is applied. Compared arms share the teacher, optimizer, OPD loss, batch size, evaluation protocol, and continuation horizon; only the source mixture, receiving model state, or within-source prompt identities are varied. Evaluation in Section 4.5 is conducted under a sampled hard subset.

F.2 Proxy-Based Selection Breakdown

Table 14 expands Figure 1d using the same five capability scores. The second panel reports per-capability differences computed directly from those values.

Table 14: Proxy-based selection results corresponding to Figure 1d. Panel (a) reproduces the five-capability scores used in the main figure; panel (b) reports the corresponding per-capability differences from Random.

(a) Capability performance (%)

Selection	Math	Code	Reasoning	Knowledge	IF	Overall
Random	69.60	73.59	73.97	79.57	83.82	76.11
DSIR	66.40	76.39	74.08	77.98	83.09	75.59
Gap	63.20	74.29	73.04	78.63	86.76	75.19
IFD	67.20	73.59	72.75	78.83	85.29	75.53
LESS	67.20	74.12	72.28	78.01	83.82	75.09

(b) Difference from Random (points)

Selection	Math	Code	Reasoning	Knowledge	IF	Overall
DSIR	-3.20	+2.80	+0.11	-1.59	-0.73	-0.52
Gap	-6.40	+0.70	-0.93	-0.94	+2.94	-0.92
IFD	-2.40	+0.00	-1.22	-0.74	+1.47	-0.58
LESS	-2.40	+0.53	-1.69	-1.56	+0.00	-1.02