GIST: GIBBS-TARGETED HIERARCHICAL DATA SELECTION FOR OFFLINE RLVR

Anonymous authors
Paper under double-blind review

ABSTRACT

Offline reinforcement learning with verifiable rewards (RLVR) trains on a fixed pool of sampled responses and typically supervises every token of every response, although responses and tokens are not equally informative. We study which ones to supervise through the Gibbs-optimal policy of KL-regularized RLVR, which offline training targets, and measure the mismatch between a policy and this target with a bidirectional KL divergence. At the reference policy, this mismatch factorizes exactly and separates the two outcomes: correct responses contribute mainly through the forward term, concentrated on their low-probability tokens, and incorrect responses through the reverse term, concentrated on the responses the policy generates often and on their high-probability tokens. Based on this analysis, we propose **GIST (Gibbs-Targeted Hierarchical Data Selection)**, a two-level selection method that approximates these principles with signals available before training. GIST uses the sampling temperature to select correct responses on-policy and incorrect responses near the policy's mode, and it uses the initial policy's token probabilities to supervise only the tokens where the mismatch lies. Applied to two offline training objectives, GIST raises the average accuracy over GSM8K, MATH-500, and MBPP for all three models under both objectives. In a controlled comparison on Qwen3-0.6B/GSM8K, it outperforms seven selection baselines that choose from the same candidates while supervising 66.6% fewer tokens than training on all candidates of the same prompts.

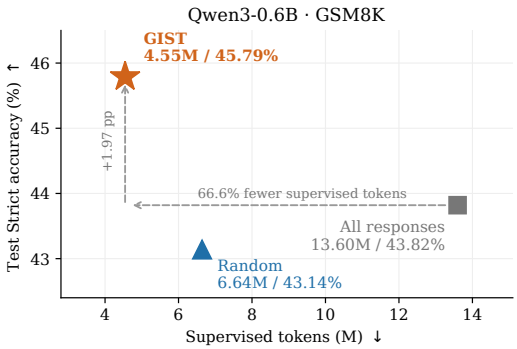

Figure 1: Test Strict accuracy versus supervised tokens on Qwen3-0.6B/GSM8K. All methods use BOLT on the same candidate pool. GIST reaches the highest accuracy while supervising 66.6% fewer tokens than training on all candidates and 31.5% fewer than random selection.

1 INTRODUCTION

Offline reinforcement learning with verifiable rewards (RLVR) trains a language model on a fixed pool of responses whose outcomes are checked automatically (Lambert et al., 2024; Shao et al., 2024; DeepSeek-AI, 2025; Wang et al., 2024b). Common approaches supervise every token of the responses they retain, although responses and tokens are not equally informative. Deciding which responses to keep and which of their tokens to supervise is therefore a central design choice in offline RLVR.

Existing methods address parts of this choice: rejection sampling keeps verified correct responses (Zelikman et al., 2022; Yuan et al., 2023), reward-weighted objectives such as BOLT weight responses by their reward (Peng et al., 2019; Shu et al., 2026), and other methods subsample rollouts (Li et al.,

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.

2026; Xu et al., 2025) or select tokens by entropy or probability (Wang et al., 2025a; Liu et al., 2026). They leave open how to combine response selection with outcome-dependent token supervision, although whether a token deserves supervision depends on both the outcome and how the response was sampled. A low-probability token in a high-temperature sample may be noise, but in a verified correct response it is more likely a valid step that the policy under-weights, while failures sampled at low temperature reveal errors near the policy's mode.

We study selection through the Gibbs-optimal policy of KL-regularized RLVR, the target that offline training fits, and measure the mismatch between the policy and this target with a bidirectional KL divergence and a KL-curvature score that bounds it from above and below. The forward term governs correct responses, whose informative instances are under-represented and whose informative tokens have low probability. The reverse term governs incorrect responses, whose informative instances are over-represented and whose informative tokens have high probability. At the reference policy, where offline selection takes place, the mismatch factorizes exactly.

We then propose **GIST** (Gibbs-Targeted Hierarchical Data Selection), a two-level selection method that approximates these principles with three quantities available at data-construction time: the sampling temperature, the verifier outcome, and the initial policy's token probabilities. At the response level, GIST takes correct responses from on-policy samples and incorrect responses from near-mode samples, and it keeps the prompts that supply both. At the token level, it supervises the low-probability tokens of correct responses and the high-probability tokens of incorrect ones.

Paired with two offline objectives, BOLT and SFT, GIST improves accuracy on GSM8K, MATH-500, and MBPP in nearly all settings (29 of 30) across three models from two families, and it raises the average accuracy in every case. On Qwen3-0.6B/GSM8K, where all methods choose from the same candidates, GIST outperforms seven selection baselines while supervising 66.6% fewer tokens and taking 49.9% fewer optimizer steps than training on all candidates of the same prompts (Figure 1).

Our contributions are as follows. (i) We derive selection principles for offline RLVR from the bidirectional mismatch to the Gibbs-optimal target, separating the roles of correct and incorrect responses at the response and token levels (Section 3). (ii) We propose GIST, a two-level selection method that approximates these principles from signals observed during data construction (Section 4). (iii) Across three models and two offline objectives, GIST improves nearly all benchmark results, and in a controlled comparison it outperforms seven selectors while supervising 66.6% fewer tokens (Section 5).

2 PRELIMINARIES

KL-regularized RLVR. For a prompt x and response y, a verifier assigns a binary reward $r(x, y) \in \{0, 1\}$, with $r = 1$ for correct and $r = 0$ for incorrect responses. Throughout, the reference policy π_{ref} is the initial policy π_0, and KL-regularized RLVR (Ouyang et al., 2022; Lambert et al., 2024) maximizes

$$J_x(\pi) = \mathbb{E}_{y\sim\pi(\cdot|x)}\big[r(x, y)\big] - \beta\,\mathrm{KL}\big(\pi(\cdot \mid x) \,\|\, \pi_{\text{ref}}(\cdot \mid x)\big), \tag{1}$$

where $\beta > 0$ controls the strength of regularization.

Proposition 2.1 (Gibbs target)**.** *For a fixed prompt x, the unique maximizer of Eq.* (1) *is*

$$\pi^\star(y \mid x) = \frac{\pi_{\text{ref}}(y \mid x)\, e^{r(x,y)/\beta}}{Z(x)}, \qquad Z(x) = \sum_{y'} \pi_{\text{ref}}(y' \mid x)\, e^{r(x,y')/\beta}, \tag{2}$$

and for any policy π with finite KL divergence to π_{ref},

$$J_x(\pi^\star) - J_x(\pi) = \beta\,\mathrm{KL}\big(\pi(\cdot \mid x) \,\|\, \pi^\star(\cdot \mid x)\big). \tag{3}$$

The target reweights the reference policy by the exponentiated reward (Peters et al., 2010; Korbak et al., 2022; Rafailov et al., 2023), and Eq. (3) identifies the objective gap with the reverse KL to $\pi^\star$.

Offline fitting of the Gibbs target. Offline training fits $\pi^\star$ using a fixed pool of responses drawn from a rollout distribution μ, with $\mu(y \mid x) > 0$ wherever $\pi^\star(y \mid x) > 0$. Rewriting the forward KL

as an expectation under μ gives

$$\mathbb{E}_x\big[\mathrm{KL}(\pi^\star\|\pi_\theta)\big] = \mathbb{E}_x\,\mathbb{E}_{y\sim\mu}\big[-\rho(x,y)\log\pi_\theta(y\mid x)\big] + C, \qquad \rho(x,y) = \frac{\pi_{\mathrm{ref}}(y\mid x)}{\mu(y\mid x)}\,\frac{e^{r(x,y)/\beta}}{Z(x)}, \tag{4}$$

where C does not depend on θ. When $\mu = \pi_{\mathrm{ref}}$, the weight reduces to $e^{r(x,y)/\beta}/Z(x)$, and Eq. (4) becomes the reference-sampled weighted-likelihood objective of BOLT (Shu et al., 2026). This objective fits the target in the forward direction, whereas the objective gap in Eq. (3) is a reverse KL. Its weights also separate the two outcome classes: an incorrect response receives $e^{-1/\beta}$ times the weight of a correct response to the same prompt. Section 3 uses both directions to locate the mismatch, and Section 4.3 describes how the weights are computed in training.

Token-level view. Let $s_t = (x, y_{<t})$ denote the prefix at step t, and let $Z(s)$ be the expectation of $e^{r/\beta}$ over reference continuations of s, so that $Z(x)$ is the normalizer in Eq. (2) and $Z(s) = e^{V(s)/\beta}$ for the soft value V (Ziebart, 2010; Haarnoja et al., 2017; Rafailov et al., 2024). The Gibbs target then factorizes over tokens as

$$\pi^\star(a\mid s) = \pi_{\mathrm{ref}}(a\mid s)\,\frac{Z(sa)}{Z(s)}, \tag{5}$$

so a token is more likely under the target than under the reference policy exactly when it raises the soft value. For two distributions p and q over responses with conditionals $p_t = p(\cdot\mid s_t)$ and $q_t = q(\cdot\mid s_t)$, the chain rule for KL divergence (Cover & Thomas, 2006) gives

$$\mathrm{KL}(p\|q) = \sum_t \mathbb{E}_{s_t\sim p}\big[\mathrm{KL}(p_t\|q_t)\big]. \tag{6}$$

Throughout, $\varphi(u) = u\log u - u + 1 \geq 0$, so that $\mathrm{KL}(p\|q) = \sum_y q_y\,\varphi(p_y/q_y)$.

3 A KL-Curvature View of Data Selection

To decide which responses and tokens to supervise, we ask where the policy differs most from the Gibbs target of Section 2, first for whole responses and then for individual tokens (proofs in Appendix A).

3.1 Bidirectional Mismatch and Curvature Score

Fix a prompt x, and write $p_y = \pi^\star(y\mid x)$ and $q_y = \pi_\theta(y\mid x)$. For $\lambda \geq 0$, define

$$D_\lambda(p,q) = \mathrm{KL}(p\|q) + \lambda\,\mathrm{KL}(q\|p), \qquad U_y^\lambda = (p_y - q_y)^2\left(\frac{1}{q_y} + \frac{\lambda}{p_y}\right). \tag{7}$$

The forward term is the target-fitting loss of Section 2; it measures how much the policy under-represents target behavior (Minka, 2005). The reverse term, scaled by β, is the RLVR objective gap (Proposition 2.1); it measures how much policy mass lies on responses that the target down-weights. The score U_y^λ weights the squared probability gap by the local curvature of both KL terms (Csiszár, 1967), so a gap of a given size counts more when it occurs at small probabilities.

Theorem 3.1 (KL-curvature bounds)**.** *Let p, q be positive on a common support with $\gamma \leq p_y/q_y \leq \Gamma$, where $0 < \gamma \leq 1 \leq \Gamma$. Then $c_- U^\lambda \leq D_\lambda \leq c_+ U^\lambda$, where $U^\lambda = \sum_y U_y^\lambda$, $c_- = \frac{1}{2}\min\{\Gamma^{-1}, \gamma\}$, and $c_+ = \frac{1}{2}\max\{\gamma^{-1}, \Gamma\}$. As the density ratios converge uniformly to one, $D_\lambda = \frac{1}{2}U^\lambda + o(U^\lambda)$.*

Up to constants, the score thus bounds the mismatch from above and below, and it is exact to second order near the target. The curvature factor also explains why a verifier label alone does not determine a response's value: the same probability gap counts more where the policy or the target places little mass. The coefficient λ weights the two directions and is used only in the analysis.

3.2 Budgeted Response Selection

To select under a budget, we attribute the mismatch to individual responses through the contribution $K_y^\lambda = q_y\varphi(p_y/q_y) + \lambda p_y\varphi(q_y/p_y)$, which is nonnegative and satisfies $D_\lambda = \sum_y K_y^\lambda$ (Section 2).

Corollary 3.2 (Budgeted selection guarantee)**.** *Let S_U contain the B candidates with the largest U_y^λ, and let $S^\star$ maximize $\sum_{y\in S} K_y^\lambda$ over subsets of size B. Under the conditions of Theorem 3.1, $\sum_{y\in S_U} K_y^\lambda \geq (c_-/c_+) \sum_{y\in S^\star} K_y^\lambda$.*

Selecting by the curvature score therefore captures a constant fraction of the largest attainable mismatch, similar to coreset guarantees (Sener & Savarese, 2018; Mirzasoleiman et al., 2020), and this fraction approaches one as the policy approaches the target. Note that the guarantee is about the mismatch captured at a fixed policy; it does not concern the outcome of training on the selected set.

With binary rewards, the target is $p_y = \pi_{\mathrm{ref}}(y \mid x)e^{r(x,y)/\beta}/Z(x)$, and the score behaves differently on the two classes.

Corollary 3.3 (Informative correct and incorrect responses)**.** *For fixed $\lambda > 0$: (i) if $r = 1$ and $q_y \ll p_y$, then $U_y^\lambda \sim p_y^2/q_y$ and the forward term dominates; (ii) if $r = 0$ and $q_y \gg p_y$, then $U_y^\lambda \sim \lambda q_y^2/p_y$ and the reverse term dominates. At comparable reference probabilities, (i) favors under-represented correct responses and (ii) favors over-represented incorrect responses.*

Under-represented correct responses drive the forward (coverage) error, and over-represented incorrect ones drive the reverse error, which Proposition 2.1 identifies with the RLVR objective gap. The two directions describe the same mismatch: mass moved onto under-represented correct responses must come from other responses, and the reverse term measures how much of it sits on over-represented errors. Offline selection evaluates the mismatch at $q = \pi_0 = \pi_{\mathrm{ref}}$, where both classes have an exact form.

Proposition 3.4 (Exact mismatch at the reference policy)**.** *At $q = \pi_{\mathrm{ref}}$, $p_y/q_y = \kappa_r(x) := e^{r/\beta}/Z(x)$ and $K_y^\lambda = q_y\, h_\lambda(\kappa_r(x))$, where $h_\lambda(\kappa) = \kappa\log\kappa - \kappa + 1 + \lambda(\kappa - 1 - \log\kappa)$. With $P(x)$ the reference success probability, $\kappa_1 \approx 1/P(x) > 1$ and $\kappa_0 \approx e^{-1/\beta}/P(x) < 1$.*

Hence every correct response is under-represented and every incorrect response is over-represented, as in Corollary 3.3. Within a class, $K_y^\lambda \propto q_y$, so frequently generated incorrect responses contribute most. For correct responses, the under-representation is uniform at the response level and localized at the token level (Section 3.3).

Remark 3.5 (The curvature score at the reference policy). At $q = \pi_{\mathrm{ref}}$, Eq. (7) gives

$$U_y^\lambda = q_y\,(\kappa_r - 1)^2\Big(1 + \frac{\lambda}{\kappa_r}\Big), \tag{8}$$

so K_y^λ/U_y^λ is constant within a class and Corollary 3.2 is exact there; across classes, Proposition 3.4 gives the exact weights (Appendix A.3).

3.3 Token-Level Mismatch

Applying the chain rule of Eq. (6) to both terms of D_λ decomposes the mismatch over prefixes, with the forward part weighted by $p(s_t)$ and the reverse part by $q(s_t)$. For an observed token, we define its conditional score U_t^λ and contribution K_t^λ as in Eq. (7) and Section 3.2, with p_t, q_t evaluated at that token. Theorem 3.1 and Corollary 3.2 also hold for these conditional quantities at each prefix, and the two regimes again separate.

Corollary 3.6 (Informative tokens)**.** *For fixed $\lambda > 0$ and comparable target conditionals across positions: (i) in a correct response, low-probability tokens ($q_t \ll p_t$) carry the forward mismatch, with $U_t^\lambda \sim p_t^2/q_t$; (ii) in an incorrect response, high-probability tokens ($q_t \gg p_t$) carry the reverse mismatch, with $U_t^\lambda \sim \lambda q_t^2/p_t$.*

Proposition 3.7 (Conditional token-level factorization)**.** *At $\pi_\theta = \pi_{\mathrm{ref}}$, with $q_t = \pi_{\mathrm{ref}}(y_t \mid s_t)$ and $\rho_t = Z(s_t y_t)/Z(s_t)$, we have $K_t^\lambda = q_t\, h_\lambda(\rho_t)$, $\prod_t \rho_t = \kappa_{r(x,y)}(x)$, and $\rho_t \leq 1/q_t$.*

By Eq. (5), ρ_t is the token's soft-value ratio; GIST ranks positions by these conditional quantities and does not estimate the prefix weights (Appendix A). The bound $\rho_t \leq 1/q_t$ supports Corollary 3.6(i) at the reference policy even without the comparability condition: a large soft-value gain requires a low q_t, although a low probability alone does not imply a large gain. For an incorrect response, $\log Z$ decreases by $\log(1/\kappa_0) \approx 1/\beta + \log P(x)$ in total, and for a given decrease, K_t^λ grows with q_t, as in

Table 1: Selection principles implied by the bidirectional mismatch.

Class	Dominant term	Response level	Token level
Correct	forward (coverage)	under-represented	low-probability tokens
Incorrect	reverse (over-commitment)	over-represented	high-probability tokens

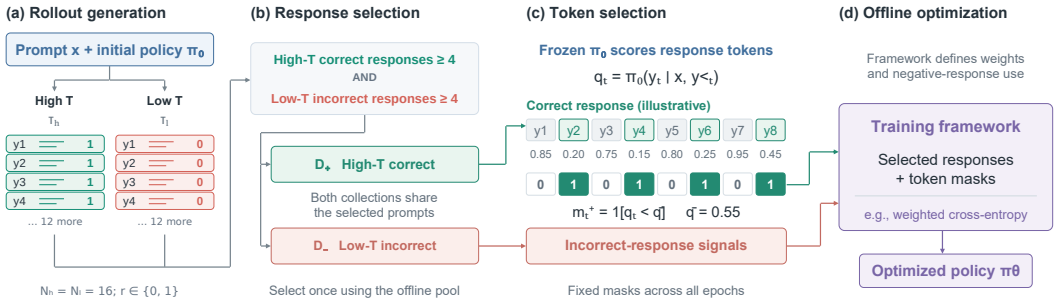

Figure 2: Overview of GIST (schematic). (a) Rollouts on-policy ($\tau_h = 1$) and near the mode (τ_l). (b) Correct responses are taken from τ_h and incorrect ones from τ_l. (c) The frozen initial policy scores tokens to mask both classes. (d) Selected responses and masks are passed to an offline training objective.

(ii). Because h_λ is increasing on $[1, \infty)$, the largest conditional contribution attainable at a position with $\rho_t \geq 1$ is

$$q_t\, h_\lambda(1/q_t) = \log \tfrac{1}{q_t} - 1 + q_t + \lambda\big(1 - q_t - q_t \log \tfrac{1}{q_t}\big), \tag{9}$$

which decreases in q_t. The high-probability tokens of a correct response therefore account for little of its mismatch.

Remark 3.8 (Where β enters selection). The within-class ranking by q_y, the bound $\rho_t \leq 1/q_t$, and the signs of the soft-value totals hold for every $\beta > 0$, so the four principles in Table 1 do not depend on β. Only the balance between the two classes depends on β and λ, through the ratio of their total contributions,

$$R_\lambda(x) = \frac{\big(1 - P(x)\big)\, h_\lambda(\kappa_0)}{P(x)\, h_\lambda(\kappa_1)}. \tag{10}$$

At $P(x) = 0.5$ and $\beta = 0.1$, R_λ is 2.6 for $\lambda = 0$ and 13.4 for $\lambda = 1$; at $\beta = 0.5$, it is 1.8 and 2.5. Since the analysis does not fix the class balance, GIST sets it through the number of retained responses in each class.

Table 1 summarizes the four principles, each of which follows from the curvature of the two KL terms.

4 GIST: Gibbs-Targeted Hierarchical Data Selection

The principles in Table 1 depend on target probabilities and soft values, which are not available during data construction. Instead of estimating these quantities, GIST uses three observable signals: the sampling temperature, the verifier outcome, and the initial-policy token probability. With these signals, it selects responses and then tokens within them, and it keeps the prompts that supply both outcomes (Figure 2). All selections are made once, before training.

4.1 Prompt and Response Selection

For each prompt x, the initial policy $\pi_0 = \pi_{\text{ref}}$ generates $N_h = 16$ responses at temperature $\tau_h = 1.0$ and $N_l = 16$ at $\tau_l = 0.3$, without top-p (Holtzman et al., 2020) or top-k truncation. A verifier assigns $r(x, y) = 1$ to correct and $r(x, y) = 0$ to incorrect responses. GIST takes correct responses from the on-policy pool and incorrect responses from the near-mode pool; we call this assignment *temperature pairing*:

$$\mathcal{Y}_+(x) = \{y \in \mathcal{Y}_h(x) : r(x, y) = 1\}, \qquad \mathcal{Y}_-(x) = \{y \in \mathcal{Y}_l(x) : r(x, y) = 0\}. \tag{11}$$

Within each class, the contribution is proportional to q_y (Proposition 3.4), so sampling at τ_l concentrates on the incorrect responses that contribute most. For correct responses, we choose $\tau_h = 1$ with the token level in mind: on-policy responses keep the low-probability steps that the mask selects (Section 5.4), and they need no temperature correction in Eq. (4). The subsequent selection and masking, however, make the training loss an approximation to that objective. A prompt is *eligible* if $|\mathcal{Y}_+(x)| \geq B_+$ and $|\mathcal{Y}_-(x)| \geq B_-$. The quotas are a practical choice and do not follow from the analysis. They guarantee samples of both classes and bound the weights of correct responses; the cost is that the hardest prompts are excluded. For each eligible prompt, GIST samples $B_+ = 4$ correct and $B_- = 4$ incorrect responses uniformly without replacement, treating duplicates as distinct instances. The analysis does not fix the balance between the classes (Remark 3.8), and we use equal quotas.

4.2 Token Selection

For a retained response y, let $q_t = \pi_0(y_t \mid x, y_{<t})$ and let $\bar{q}$ be its mean over the response positions, excluding EOS. The masks are

$$m_t^+ = \mathbf{1}\big[q_t < \bar{q}\big] \text{ for correct responses}, \qquad m_t^- = \mathbf{1}\big[q_t > \bar{q}\big] \text{ for incorrect responses}. \tag{12}$$

These masks implement the token-level principles, with the response mean as the threshold between low and high probability. Correct responses keep their low-probability tokens, where the underrepresented steps lie (Corollary 3.6(i), Proposition 3.7), and incorrect responses keep their high-probability tokens, the confident errors of Corollary 3.6(ii). Both masks are applied in every run that uses GIST's token selection. They depend only on the verifier outcome and the initial policy, not on the training objective, and are computed once with the frozen initial policy and stay fixed during training.

4.3 Training with the Selected Data

GIST outputs the selected responses, their rewards, and the token masks, so it can be combined with any offline objective that trains on weighted responses; we use two. BOLT (Shu et al., 2026), the weighted-likelihood objective of Section 2, weights each response by $e^{r/\beta}$, rescaled so that the weights of the prompt's 32 candidates average to one. The weights are computed on the full candidate pool before selection and are not renormalized afterward. A correct response therefore receives a weight of about $32/n_1$, where n_1 is the number of correct candidates, so prompts that the policy solves less often count more, and an incorrect response receives $e^{-1/\beta}$ times that weight. SFT instead gives every retained response unit weight. Both objectives average, over the minibatch, each response's weight times its mean negative log-likelihood on the masked positions; unmasked positions remain in the context but receive no loss.

5 Experiments

5.1 Main Results

Setup. We train Qwen3-0.6B, Qwen3.5-2B, and Gemma-4-E2B-it (Yang et al., 2025; Qwen Team, 2026; Gemma Team, 2026) and evaluate on GSM8K, MATH-500, and MBPP-full (Cobbe et al., 2021; Hendrycks et al., 2021; Lightman et al., 2024; Austin et al., 2021). For math, we report Strict accuracy, which requires the prescribed answer format, and Flexible accuracy (Flex.), which also credits correctly extracted answers; for code, we report pass@1. We pair GIST with two objectives: BOLT ($\beta = 0.1$) and SFT, which imitates every training response with unit weight. All trained methods share 32 candidates per prompt (16 per temperature) and validation-based checkpoint selection; each objective is trained either on all candidates or on GIST's selected subset (+ GIST; details in Appendix B).

Results. Averaged over the three benchmarks (Avg. in Table 2), GIST improves both objectives on every model, by 1.26–2.55 points over BOLT and 1.11–4.41 points over SFT, and BOLT + GIST has the highest average for all three models. GIST improves over BOLT on both mathematical benchmarks for all three models under both metrics (Table 2). On GSM8K, the gains reach 2.50 points on Strict and 3.26 on Flexible. On Flexible, BOLT falls below the initial checkpoint for two models, whereas BOLT + GIST improves on it. For Qwen3-0.6B, paired tests over the 1,319 GSM8K

Table 2: Test results with BOLT and SFT (%). Avg.: mean of GSM8K Flex., MATH-500 Flex., and MBPP-full pass@1. Bold marks the best score within each model and column. Parentheses: change vs. Initial.

		GSM8K ↑		MATH-500 ↑		MBPP-full ↑	
Model	Method	Strict	Flex.	Strict	Flex.	pass@1	Avg. ↑
Qwen3-0.6B	Initial	40.86	46.10	44.20	49.80	22.80	39.57
	SFT	42.08 (+1.22)	47.38 (+1.28)	44.40 (+0.20)	50.00 (+0.20)	24.20 (+1.40)	40.53 (+0.96)
	+ GIST	43.82 (+2.96)	48.14 (+2.04)	44.80 (+0.60)	50.40 (+0.60)	28.80 (+6.00)	42.45 (+2.88)
	BOLT	43.29 (+2.43)	47.46 (+1.36)	44.60 (+0.40)	50.40 (+0.60)	26.60 (+3.80)	41.49 (+1.92)
	+ GIST	**45.79** (+4.93)	**49.73** (+3.63)	**45.40** (+1.20)	**51.00** (+1.20)	**31.40** (+8.60)	**44.04** (+4.47)
Qwen3.5-2B	Initial	54.51	55.19	46.00	55.00	37.40	49.20
	SFT	55.04 (+0.53)	55.80 (+0.61)	46.20 (+0.20)	55.00 (+0.00)	39.60 (+2.20)	50.13 (+0.93)
	+ GIST	56.18 (+1.67)	56.71 (+1.52)	46.80 (+0.80)	55.80 (+0.80)	41.20 (+3.80)	51.24 (+2.04)
	BOLT	54.13 (−0.38)	54.51 (−0.68)	46.40 (+0.40)	54.40 (−0.60)	**42.40** (+5.00)	50.44 (+1.24)
	+ GIST	**56.63** (+2.12)	**57.09** (+1.90)	**47.40** (+1.40)	**56.20** (+1.20)	41.80 (+4.40)	**51.70** (+2.50)
Gemma-4-E2B-it	Initial	12.28	26.99	2.60	38.60	55.00	40.20
	SFT	12.81 (+0.53)	27.82 (+0.83)	12.40 (+9.80)	42.00 (+3.40)	55.80 (+0.80)	41.87 (+1.67)
	+ GIST	13.87 (+1.59)	29.04 (+2.05)	41.80 (+39.20)	53.00 (+14.40)	56.80 (+1.80)	46.28 (+6.08)
	BOLT	12.89 (+0.61)	26.31 (−0.68)	40.80 (+38.20)	52.80 (+14.20)	56.40 (+1.40)	45.17 (+4.97)
	+ GIST	**14.10** (+1.82)	**29.57** (+2.58)	**42.20** (+39.60)	**53.20** (+14.60)	**57.00** (+2.00)	**46.59** (+6.39)

Table 3: Selection methods with BOLT on a shared candidate pool (Qwen3-0.6B/GSM8K, %).

	Random	PODS	RFT-style	RSI-S	ETS	ProFit	HES	GIST
Strict ↑	43.14	43.52	44.05	43.21	43.75	42.46	44.20	**45.79**
Flex. ↑	48.60	48.22	48.14	47.84	47.54	46.47	47.31	**49.73**

questions support both gains (Appendix B.5). BOLT also trains on prompts whose candidates are all incorrect (Appendix B), so this compares the two pipelines end to end; Sections 5.2 and 5.3 isolate the selection components. Gains on MATH-500 are smaller (0.40–1.80 points). On MBPP-full, GIST gains 4.80 points on Qwen3-0.6B and 0.60 on Gemma-4-E2B-it, and trails BOLT by 0.60 on Qwen3.5-2B.

GIST also improves over SFT for every model, benchmark, and metric (Table 2). The gains are 0.76–1.74 points on GSM8K, 0.40–0.80 on MATH-500 for the two Qwen models, and 1.00–4.60 on MBPP-full. On Gemma-4-E2B-it, MATH-500 Strict rises from 12.40% to 41.80% while Flexible rises by 11.00 points, so most of this gain comes from following the prescribed answer format. On GSM8K, the gains from GIST are similar under Flexible and Strict accuracy for both objectives, which suggests that they do not come only from answer formatting.

5.2 Comparison with Selection Baselines

All selectors share the candidate pool, the 1,593 eligible prompts, and the BOLT objective (Table 3). Random samples four correct and four incorrect responses uniformly and supervises all tokens, RFT-style keeps all correct responses, and the others adapt HES, PODS, ProFit, ETS, and RSI-S (Li et al., 2026; Xu et al., 2025; Liu et al., 2026; Wang et al., 2025a; Lv et al., 2026b) (Appendix C).

GIST achieves the highest accuracy on both metrics, 1.59–3.33 points above all seven baselines on Strict and 1.13–3.26 points on Flexible. It exceeds Random by 2.65 points on Strict and 1.13 points on Flexible while supervising 31.5% fewer tokens (Section 5.4), and it exceeds training on all candidates by 1.97 and 3.03 points. Appendix B.5 reports paired significance tests for these comparisons.

5.3 Ablation Study

We ablate GIST on the shared pool, changing one design choice at a time (Tables 4 and 5).

Removing pairing (row c), the mask (row d), or the Boltzmann weights (row e, the SFT + GIST entry of Table 2) from GIST lowers Strict/Flexible accuracy by 2.20/0.98, 2.04/2.12, and 1.97/1.59 points, respectively, and removing all selection (row a) costs 1.97/3.03 (Table 4). Replacing GIST's token rule also lowers both metrics (Table 5): supervising all tokens costs 2.04 and 2.12 points, a random

Table 4: Component ablation, Qwen3-0.6B on GSM8K (%). *Sub.*: 4+4 responses instead of all 32; *Pair.*: correct from τ_h, incorrect from τ_l; *Mask*: m^+ of Eq. (12); *Wt.*: Boltzmann vs. unit. Parentheses: change vs. (f).

	Sub.	Pair.	Mask	Wt.	Strict ↑	Flex. ↑
(a)	×	×	×	✓	43.82 (−1.97)	46.70 (−3.03)
(b)	✓	×	×	✓	43.14 (−2.65)	48.60 (−1.13)
(c)	✓	×	✓	✓	43.59 (−2.20)	48.75 (−0.98)
(d)	✓	✓	×	✓	43.75 (−2.04)	47.61 (−2.12)
(e)	✓	✓	✓	×	43.82 (−1.97)	48.14 (−1.59)
(f)	✓	✓	✓	✓	**45.79**	**49.73**

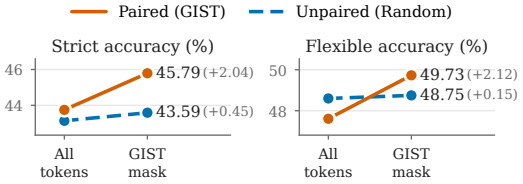

Figure 3: Pairing and masking on Qwen3-0.6B/GSM8K. Unpaired responses are the Random selector; the mask helps mainly with paired responses.

Table 5: Token rule for correct responses, Qwen3-0.6B on GSM8K (%). *Kept*: approximate share of positions supervised per correct response (%). †Same number of positions per response as GIST. Parentheses: change vs. GIST.

Positions	Kept	Strict ↑	Flex. ↑
All tokens	100	43.75 (−2.04)	47.61 (−2.12)
Random†	31	44.81 (−0.98)	45.64 (−4.09)
Highest-prob.†	31	44.05 (−1.74)	47.23 (−2.50)
Below-mean (GIST)	31	**45.79**	**49.73**

Table 6: Supervised tokens up to each method's selected checkpoint (Qwen3-0.6B/GSM8K). *Reduction* is relative to all candidates.

	All	Random	GIST
Tokens	13.60M	6.64M	4.55M
Reduction	–	51.2%	**66.6%**
Strict ↑	43.82	43.14	**45.79**
Flex. ↑	46.70	48.60	**49.73**

subset of the same size costs 0.98 and 4.09, and keeping the highest-probability tokens of correct responses costs 1.74 and 2.50.

Rows (b), (c), (d), and (f) of Table 4 form a 2×2 design of pairing and masking (Figure 3). The mask adds 2.04 and 2.12 points with pairing but only 0.45 and 0.15 without it, an interaction of 1.59 and 1.97 points.

5.4 Discussion

Token and response selection. The analysis in Section 3 holds at the policy that selects the data, so we compare its consequences with ablations that change one part of the data at a time. The token-rule ablation keeps GIST's responses fixed and changes only which positions are supervised, which tests Corollary 3.6(i). By Proposition 3.7, a token can gain much soft value only if q_t is low ($\rho_t \leq 1/q_t$), so the high-probability tokens of a correct response should carry little of its mismatch. Keeping them instead of the low-probability ones loses most of the mask's benefit (Table 5), and the ProFit adaptation, which also keeps high-probability tokens, is the weakest selector (Table 3). Below-mean tokens make up about 31% of correct responses at both temperatures, but their mean q_t is 0.41 at τ_h and 0.58 at τ_l (Appendix B.5); since the attainable contribution in Eq. (9) decreases in q_t, the mask should find more informative tokens in on-policy responses. Without the mask, a whole correct response contributes in proportion to q_y (Proposition 3.4), which favors near-mode responses. Together, these results predict that pairing and masking help mainly in combination, which matches the interaction in Section 5.3.

Supervision and training cost. On the shared pool, GIST keeps 12,744 of the 50,976 candidates and supervises 66.6% fewer tokens than training on all candidates and 31.5% fewer than Random, while scoring higher than both (Table 6, Figure 1, Appendix B.6). Masked positions still enter the forward and backward passes, so the compute saving over training on all candidates comes from keeping a quarter of the responses. GIST reaches its selected checkpoint in 2,394 optimizer steps, about half of the 4,779 needed when training on all candidates, although its selected checkpoint comes after twice as many epochs. Selection reuses BOLT's candidate pool and adds one forward pass of the frozen initial policy to compute q_t.

Generality. GIST selects data using only the sampling temperature, the verifier outcome, and the initial policy's token probabilities, with the same settings for every model and benchmark (Appendix B). Because the selection does not depend on the training objective, it can be paired with other offline methods. SFT imitates every retained response with unit weight, including the incorrect ones, and SFT + GIST still matches or exceeds BOLT trained on all candidates in 14 of the 15 comparisons. The benefit of the selection therefore does not come only from BOLT's reward weighting, although with GIST's data the weights add a further 0.20–2.60 points in every comparison.

6 Related Work

Offline RLVR and policy optimization. Reinforcement learning with verifiable rewards (RLVR) improves reasoning using automatically checkable outcomes (Lambert et al., 2024; Shao et al., 2024; DeepSeek-AI, 2025; Yu et al., 2025). Most RLVR methods optimize online with PPO-style updates (Schulman et al., 2017; Liu et al., 2025; Yue et al., 2025), whereas offline approaches (Levine et al., 2020) such as ILQL (Snell et al., 2023) and OREO (Wang et al., 2024b) learn from previously collected rollouts. BOLT (Shu et al., 2026) shows that Boltzmann-weighted likelihood, in the spirit of reward-weighted regression (Peters & Schaal, 2007; Peng et al., 2019), on reference-policy samples fits the Gibbs-optimal target of KL-regularized RLVR. These methods define the training objective; GIST instead decides which responses and tokens of a finite rollout pool to train on, and can be paired with BOLT or other offline objectives. Work on alignment and distillation chooses which KL direction or f-divergence the model should minimize (Go et al., 2023; Wang et al., 2024a; Gu et al., 2024), whereas we use both directions to select the training data.

Data selection for reasoning post-training. Selection for post-training (Zhou et al., 2023; Xia et al., 2024) has been studied at two granularities. At the response level, rejection-sampling fine-tuning and self-training (Zelikman et al., 2022; Yuan et al., 2023; Dong et al., 2023; Gulcehre et al., 2023; Singh et al., 2024) keep verified correct trajectories, HES (Li et al., 2026) ranks trajectories by the entropy of a few high-entropy tokens, PODS (Xu et al., 2025) subsamples rollouts for policy updates, and dynamic sampling in DAPO (Yu et al., 2025) discards prompts whose rollouts are all correct or all incorrect; other methods filter RL prompts by difficulty or learning impact (Bae et al., 2025; Li et al., 2025; Wang et al., 2025b). At the token level, Rho-1 (Lin et al., 2024a) and Token Cleaning (Pang et al., 2025) select tokens by excess loss or update influence, and critical-token estimation (Lin et al., 2024b) contrasts models trained on correct and incorrect trajectories. In RLVR, policy entropy governs exploration (Cui et al., 2025) and high-entropy forking tokens concentrate effective updates, which motivates entropy-based token selection (ETS) (Wang et al., 2025a); RSI-S (Lv et al., 2026b) and GMTS (Lv et al., 2026a) use surprisal- and gradient-based token scores, and ProFit (Liu et al., 2026) keeps high-probability tokens during SFT. Each of these methods selects at a single level with its own criterion. GIST derives both from the same bidirectional mismatch and applies them hierarchically, keeping, unlike ProFit, the low-probability tokens of correct responses.

7 Conclusion

We studied data selection for offline RLVR through the bidirectional mismatch between the policy and the Gibbs-optimal target. In the regimes we identify, the forward term governs correct responses and the reverse term governs incorrect ones. GIST turns this view into response- and token-level selection rules that use only the sampling temperature, the verifier outcome, and the initial policy's token probabilities. Across three models and two offline objectives, it improves nearly all benchmark results, and it outperforms seven selectors on a shared candidate pool while supervising far fewer tokens than training on all candidates. More broadly, the analysis suggests that which responses and which tokens to train on can be read from the policy itself, and we expect this view to apply to other offline objectives and to settings in which the data are re-selected as the policy changes.

Ethics Statement

This work uses publicly available models and benchmarks and involves no human subjects or personal data. We do not foresee ethical concerns beyond those of the underlying models.

Reproducibility Statement

Proofs of all theoretical results are given in Appendix A. Appendix B specifies the data construction and eligibility rule (Appendix B.1), the token masks (Appendix B.2), the training objectives and hyperparameters (Appendix B.3), the evaluation protocol (Appendix B.4), and the paired tests (Appendix B.5); Appendix C describes every selection baseline and ablation control. All models and benchmarks are publicly available.

AI Use Statement

In this work, we used generative AI tools to edit the paper for readability, to polish the formatting of tables and figures, and to check that the numbers reported in the text match the tables. We have not used generative AI tools to develop the theoretical framework, formulate or prove the mathematical claims, design or implement the method and experiments, generate or process data, or interpret the results. We have reviewed all AI-assisted edits, checked every reported number against our experimental logs, and take full responsibility for the content of the paper.

References

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, and Charles Sutton. Program synthesis with large language models, 2021. URL https://arxiv.org/abs/2108.07732.

Sanghwan Bae et al. Online difficulty filtering for reasoning oriented reinforcement learning, 2025. URL https://arxiv.org/abs/2504.03380.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems, 2021. URL https://arxiv.org/abs/2110.14168.

Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley, second edition, 2006.

Imre Csiszár. Information-type measures of difference of probability distributions and indirect observations. *Studia Scientiarum Mathematicarum Hungarica*, 2:299–318, 1967.

Ganqu Cui et al. The entropy mechanism of reinforcement learning for reasoning language models, 2025. URL https://arxiv.org/abs/2505.22617.

DeepSeek-AI. DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning, 2025. URL https://arxiv.org/abs/2501.12948.

Hanze Dong, Wei Xiong, Deepanshu Goyal, Yihan Zhang, Winnie Chow, Rui Pan, Shizhe Diao, Jipeng Zhang, Kashun Shum, and Tong Zhang. RAFT: Reward ranked finetuning for generative foundation model alignment. *Transactions on Machine Learning Research*, 2023. URL https://arxiv.org/abs/2304.06767.

Rotem Dror, Gili Baumer, Segev Shlomov, and Roi Reichart. The hitchhiker's guide to testing statistical significance in natural language processing. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, 2018. URL https://aclanthology.org/P18-1128/.

Bradley Efron and Robert J. Tibshirani. *An Introduction to the Bootstrap*. Chapman & Hall, 1993.

Leo Gao et al. A framework for few-shot language model evaluation. https://github.com/EleutherAI/lm-evaluation-harness, 2024.

Gemma Team. Gemma 4 technical report, 2026. URL https://arxiv.org/abs/2607.02770.

Dongyoung Go, Tomasz Korbak, Germán Kruszewski, Jos Rozen, Nahyeon Ryu, and Marc Dymetman. Aligning language models with preferences through f-divergence minimization. In *International Conference on Machine Learning*, 2023. URL https://arxiv.org/abs/2302.08215.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. MiniLLM: Knowledge distillation of large language models. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2306.08543.

Caglar Gulcehre et al. Reinforced self-training (ReST) for language modeling, 2023. URL https://arxiv.org/abs/2308.08998.

Tuomas Haarnoja, Haoran Tang, Pieter Abbeel, and Sergey Levine. Reinforcement learning with deep energy-based policies. In *International Conference on Machine Learning*, 2017. URL https://arxiv.org/abs/1702.08165.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH dataset. In *Advances in Neural Information Processing Systems Datasets and Benchmarks Track*, 2021. URL https://arxiv.org/abs/2103.03874.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. The curious case of neural text degeneration. In *International Conference on Learning Representations*, 2020. URL https://arxiv.org/abs/1904.09751.

Tomasz Korbak, Ethan Perez, and Christopher L. Buckley. RL with KL penalties is better viewed as Bayesian inference. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, 2022. URL https://arxiv.org/abs/2205.11275.

Nathan Lambert et al. Tülu 3: Pushing frontiers in open language model post-training, 2024. URL https://arxiv.org/abs/2411.15124.

Sergey Levine, Aviral Kumar, George Tucker, and Justin Fu. Offline reinforcement learning: Tutorial, review, and perspectives on open problems, 2020. URL https://arxiv.org/abs/2005.01643.

Xiaoyuan Li, Yubo Ma, Chengpeng Li, Fengbin Zhu, Yiyao Yu, Keqin Bao, Wenjie Wang, Fuli Feng, and Dayiheng Liu. Unified data selection for LLM reasoning, 2026. URL https://arxiv.org/abs/2605.22389.

Xuefeng Li, Haoyang Zou, and Pengfei Liu. LIMR: Less is more for RL scaling, 2025. URL https://arxiv.org/abs/2502.11886.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2305.20050.

Zhenghao Lin, Zhibin Gou, Yeyun Gong, Xiao Liu, Yelong Shen, Ruochen Xu, Chen Lin, Yujiu Yang, Jian Jiao, Nan Duan, and Weizhu Chen. Rho-1: Not all tokens are what you need, 2024a. URL https://arxiv.org/abs/2404.07965.

Zicheng Lin et al. Critical tokens matter: Token-level contrastive estimation enhances LLM's reasoning capability, 2024b. URL https://arxiv.org/abs/2411.19943.

Tao Liu, Taiqiang Wu, Runming Yang, Shaoning Sun, Junjie Wang, and Yujiu Yang. ProFit: Leveraging high-value signals in SFT via probability-guided token selection. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 15383–15401, San Diego, California, United States, July 2026. Association for Computational Linguistics. doi: 10.18653/v1/2026.findings-acl.755. URL https://aclanthology.org/2026.findings-acl.755/.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding R1-Zero-like training: A critical perspective, 2025. URL https://arxiv.org/abs/2503.20783.

Outongyi Lv, Yuanwei Zhang, and Xiaoqun Zhang. GMTS: Gradient magnitude-based token selection improves RLVR training for LLM reasoning, 2026a. URL https://arxiv.org/abs/2608.30632.

Outongyi Lv, Yanzhao Zheng, Yuanwei Zhang, Zhenghao Huang, Xingjun Wang, Baohua Dong, Hangcheng Zhu, and Yingda Chen. Which tokens matter? Adaptive token selection for RLVR with the relative surprisal index, 2026b. URL https://arxiv.org/abs/2606.31575.

Quinn McNemar. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157, 1947.

Tom Minka. Divergence measures and message passing. Technical Report MSR-TR-2005-173, Microsoft Research, 2005.

Baharan Mirzasoleiman, Jeff Bilmes, and Jure Leskovec. Coresets for data-efficient training of machine learning models. In *International Conference on Machine Learning*, 2020. URL https://arxiv.org/abs/1906.01827.

Long Ouyang et al. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, 2022. URL https://arxiv.org/abs/2203.02155.

Jinlong Pang, Na Di, Zhaowei Zhu, Jiaheng Wei, Hao Cheng, Chen Qian, and Yang Liu. Token cleaning: Fine-grained data selection for LLM supervised fine-tuning. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 47837–47858. PMLR, 2025. URL https://proceedings.mlr.press/v267/pang25a.html.

Xue Bin Peng, Aviral Kumar, Grace Zhang, and Sergey Levine. Advantage-weighted regression: Simple and scalable off-policy reinforcement learning, 2019. URL https://arxiv.org/abs/1910.00177.

Jan Peters and Stefan Schaal. Reinforcement learning by reward-weighted regression for operational space control. In *International Conference on Machine Learning*, 2007.

Jan Peters, Katharina Mülling, and Yasemin Altun. Relative entropy policy search. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2010.

Qwen Team. Qwen3.5: Towards native multimodal agents. https://qwen.ai/blog?id=qwen3.5, 2026.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems*, volume 36, pp. 53728–53741, 2023. URL https://arxiv.org/abs/2305.18290.

Rafael Rafailov, Joey Hejna, Ryan Park, and Chelsea Finn. From r to q^*: Your language model is secretly a Q-function. In *Conference on Language Modeling*, 2024. URL https://arxiv.org/abs/2404.12358.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms, 2017. URL https://arxiv.org/abs/1707.06347.

Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. In *International Conference on Learning Representations*, 2018. URL https://arxiv.org/abs/1708.00489.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models, 2024. URL https://arxiv.org/abs/2402.03300.

Yao Shu, Chenxing Wei, Hongbin Lin, Shuang Qiu, and Hui Xiong. Reference-sampled Boltzmann projection for KL-Regularized RLVR: Target-matched weighted SFT, finite one-shot gaps, and policy mirror descent, 2026. URL https://arxiv.org/abs/2605.02469.

Avi Singh et al. Beyond human data: Scaling self-training for problem-solving with language models. *Transactions on Machine Learning Research*, 2024. URL https://arxiv.org/abs/2312.06585.

Charlie Victor Snell, Ilya Kostrikov, Yi Su, Sherry Yang, and Sergey Levine. Offline RL for natural language generation with implicit language Q learning. In *International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2206.11871.

Chaoqi Wang, Yibo Jiang, Chenghao Yang, Han Liu, and Yuxin Chen. Beyond reverse KL: Generalizing direct preference optimization with diverse divergence constraints. In *International Conference on Learning Representations*, 2024a. URL https://arxiv.org/abs/2309.16240.

Huaijie Wang, Shibo Hao, Hanze Dong, Shenao Zhang, Yilin Bao, Ziran Yang, and Yi Wu. Offline reinforcement learning for LLM multi-step reasoning, 2024b. URL https://arxiv.org/abs/2412.16145.

Shenzhi Wang, Le Yu, Chang Gao, Chujie Zheng, Shixuan Liu, Rui Lu, Kai Dang, Xionghui Chen, Jianxin Yang, Zhenru Zhang, Yuqiong Liu, An Yang, Andrew Zhao, Yang Yue, Shiji Song, Bowen Yu, Gao Huang, and Junyang Lin. Beyond the 80/20 rule: High-entropy minority tokens drive effective reinforcement learning for LLM reasoning. In *Advances in Neural Information Processing Systems*, 2025a. URL https://arxiv.org/abs/2506.01939.

Yiping Wang et al. Reinforcement learning for reasoning in large language models with one training example, 2025b. URL https://arxiv.org/abs/2504.20571.

Mengzhou Xia, Sadhika Malladi, Suchin Gururangan, Sanjeev Arora, and Danqi Chen. LESS: Selecting influential data for targeted instruction tuning. In *International Conference on Machine Learning*, 2024. URL https://arxiv.org/abs/2402.04333.

Yixuan Even Xu, Yash Savani, Fei Fang, and J. Zico Kolter. Not all rollouts are useful: Down-sampling rollouts in LLM reinforcement learning, 2025. URL https://arxiv.org/abs/2504.13818.

An Yang et al. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Hongli Yu, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Lin Yan, Mu Qiao, Yonghui Wu, and Mingxuan Wang. DAPO: An open-source LLM reinforcement learning system at scale. In *Advances in Neural Information Processing Systems*, 2025. URL https://arxiv.org/abs/2503.14476.

Zheng Yuan, Hongyi Yuan, Chengpeng Li, Guanting Dong, Keming Lu, Chuanqi Tan, Chang Zhou, and Jingren Zhou. Scaling relationship on learning mathematical reasoning with large language models, 2023. URL https://arxiv.org/abs/2308.01825.

Yang Yue et al. Does reinforcement learning really incentivize reasoning capacity in LLMs beyond the base model? In *Advances in Neural Information Processing Systems*, 2025. URL https://arxiv.org/abs/2504.13837.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah D. Goodman. STaR: Bootstrapping reasoning with reasoning. In *Advances in Neural Information Processing Systems*, 2022. URL https://arxiv.org/abs/2203.14465.

Chunting Zhou et al. LIMA: Less is more for alignment. In *Advances in Neural Information Processing Systems*, 2023. URL https://arxiv.org/abs/2305.11206.

Brian D. Ziebart. *Modeling Purposeful Adaptive Behavior with the Principle of Maximum Causal Entropy*. PhD thesis, Carnegie Mellon University, 2010.

A PROOFS

Throughout, $\varphi(u) = u \log u - u + 1$, which satisfies $\varphi \geq 0$, $\varphi(1) = \varphi'(1) = 0$, and $\varphi''(u) = 1/u$.

A.1 GIBBS TARGET AND OFFLINE FITTING

Proof of Proposition 2.1. Fix x and write $p = \pi^\star(\cdot \mid x)$. Since $r = \beta \log(p/\pi_{\mathrm{ref}}) + \beta \log Z(x)$, for any π with finite KL divergence to π_{ref},

$$J_x(\pi) = \mathbb{E}_\pi\Big[\beta \log \frac{p}{\pi_{\mathrm{ref}}}\Big] + \beta \log Z(x) - \beta\, \mathbb{E}_\pi\Big[\log \frac{\pi}{\pi_{\mathrm{ref}}}\Big] = \beta \log Z(x) - \beta\, \mathrm{KL}(\pi \| p).$$

Nonnegativity of KL shows that p is the unique maximizer with $J_x(p) = \beta \log Z(x)$, and subtracting gives Eq. (3). □

Derivation of Eq. (4). $\mathrm{KL}(p\|\pi_\theta) = \sum_y p_y \log p_y - \sum_y p_y \log \pi_\theta(y \mid x)$. The first term does not depend on θ. If $\mu(y \mid x) > 0$ wherever $p_y > 0$, the second term equals $\mathbb{E}_{y\sim\mu}[-(p_y/\mu_y) \log \pi_\theta(y \mid x)]$, and $p_y/\mu_y = \rho(x, y)$ by Eq. (2).

A.2 KL-CURVATURE BOUNDS AND BUDGETED SELECTION

Proof of Theorem 3.1. Because $\sum_y (p_y - q_y) = 0$, we have $\mathrm{KL}(p\|q) = \sum_y q_y \varphi(p_y/q_y)$ and $\mathrm{KL}(q\|p) = \sum_y p_y \varphi(q_y/p_y)$. For u in an interval $[m, M]$ with $0 < m \leq 1 \leq M$, Taylor's theorem around $u = 1$ gives $(u-1)^2/(2M) \leq \varphi(u) \leq (u-1)^2/(2m)$. With $u = p_y/q_y \in [\gamma, \Gamma]$,

$$\frac{(p_y - q_y)^2}{2\Gamma q_y} \leq q_y \varphi\Big(\frac{p_y}{q_y}\Big) \leq \frac{(p_y - q_y)^2}{2\gamma q_y},$$

and with $u = q_y/p_y \in [\Gamma^{-1}, \gamma^{-1}]$,

$$\frac{\gamma(p_y - q_y)^2}{2p_y} \leq p_y \varphi\Big(\frac{q_y}{p_y}\Big) \leq \frac{\Gamma(p_y - q_y)^2}{2p_y}.$$

Multiplying the second pair by λ and adding gives $c_- U_y^\lambda \leq K_y^\lambda \leq c_+ U_y^\lambda$ for each y, and summing over y gives the theorem. For the local statement, $\varphi(1+h) = \frac{1}{2}h^2 + O(|h|^3)$ applied to both ratios gives $D_\lambda = \frac{1}{2}U^\lambda + O(\delta\, U^\lambda)$ with $\delta = \sup_y |p_y/q_y - 1| \to 0$. □

Proof of Corollary 3.2. Using the per-response bounds and the optimality of S_U for U^λ, $\sum_{S_U} K_y^\lambda \geq c_- \sum_{S_U} U_y^\lambda \geq c_- \sum_{S^\star} U_y^\lambda \geq (c_-/c_+) \sum_{S^\star} K_y^\lambda$. □

Proof of Corollary 3.3. Write $U_y^\lambda = (p_y - q_y)^2/q_y + \lambda(p_y - q_y)^2/p_y$. (i) With $a = q_y/p_y \to 0$, the first term is $p_y^2(1-a)^2/q_y$ and the second is $\lambda p_y(1-a)^2$; their ratio is $1/(\lambda a) \to \infty$, so $U_y^\lambda \sim p_y^2/q_y$. (ii) With $b = p_y/q_y \to 0$, the second term is $\lambda q_y^2(1-b)^2/p_y$ and the first is $q_y(1-b)^2$; their ratio is $\lambda/b \to \infty$, so $U_y^\lambda \sim \lambda q_y^2/p_y$. At fixed target probability, the first expression decreases and the second increases in q_y. □

A.3 SELECTION AT THE REFERENCE POLICY

Proof of Proposition 3.4. By Eq. (2), $p_y = q_y e^{r/\beta}/Z(x) = \kappa_r q_y$ when $q = \pi_{\mathrm{ref}}$. Substituting into $K_y^\lambda = q_y \varphi(p_y/q_y) + \lambda p_y \varphi(q_y/p_y)$ gives $q_y \varphi(\kappa_r) + \lambda \kappa_r q_y \varphi(1/\kappa_r)$, and $\kappa\varphi(1/\kappa) = \kappa - 1 - \log\kappa$ yields $K_y^\lambda = q_y h_\lambda(\kappa_r)$. With $r \in \{0, 1\}$, $Z(x) = e^{1/\beta}P(x) + 1 - P(x) = e^{1/\beta}P(x)\big(1 + O(e^{-1/\beta}/P(x))\big)$, so $\kappa_1 = e^{1/\beta}/Z(x) \approx 1/P(x)$ and $\kappa_0 = 1/Z(x) \approx e^{-1/\beta}/P(x)$. □

Between-class constants (Remark 3.5). At $q = \pi_{\mathrm{ref}}$, the density ratios range from $\gamma = \kappa_0$ to $\Gamma = \kappa_1$, so $c_-/c_+ = \gamma^2 \approx e^{-2/\beta}/P(x)^2$, below 10^{-7} at $\beta = 0.1$ for $P(x) \geq 0.25$. For $\lambda = 1$ and $P(x) \in [0.25, 0.75]$, Eq. (8) gives an incorrect-to-correct score ratio between about 5×10^2 and 8.5×10^4 for responses of equal probability, whereas the exact ratio $h_\lambda(\kappa_0)/h_\lambda(\kappa_1)$ is between 2 and 101.

A.4 Token-Level Results

Chain rule and target conditional. With a finite horizon and terminated sequences padded by an absorbing state, $\log(p(y \mid x)/q(y \mid x)) = \sum_t \log(p_t(y_t \mid s_t)/q_t(y_t \mid s_t))$. Taking expectations under p and conditioning each term on its prefix gives the forward part of Eq. (6); exchanging p and q gives the reverse part. For the conditional, summing the Gibbs target over completions of a prefix gives $p(s) = \pi_{\mathrm{ref}}(s)Z(s)/Z(x)$, and the ratio of $p(sa)$ to $p(s)$ is Eq. (5). Applying the proof of Theorem 3.1 to $(p_t(y_t \mid s_t), q_t(y_t \mid s_t))$ gives $c_- U_t^\lambda \le K_t^\lambda \le c_+ U_t^\lambda$, and the proofs of Corollaries 3.2 and 3.3 apply verbatim to tokens, which proves Corollary 3.6.

Proof of Proposition 3.7. At $\pi_\theta = \pi_{\mathrm{ref}}$, Eq. (5) gives $p_t/q_t = \rho_t$, and the computation in the proof of Proposition 3.4 gives $K_t^\lambda = q_t h_\lambda(\rho_t)$. Since $Z(s_1) = Z(x)$ and $Z(s_{T+1}) = e^{r(x,y)/\beta}$, the product $\prod_t \rho_t$ telescopes to $\kappa_{r(x,y)}(x)$. Finally, $Z(s_t) = \sum_a \pi_{\mathrm{ref}}(a \mid s_t)Z(s_t a) \ge q_t Z(s_t y_t)$, so $\rho_t \le 1/q_t$. For the prefix-weighted share, the forward part $p(s_t)\, q_t \varphi(\rho_t)$ uses $p(s_t) = q(s_t)Z(s_t)/Z(x)$, and the reverse part $\lambda q(s_t)\, p_t \varphi(q_t/p_t)$ uses $\rho\, \varphi(1/\rho) = \rho - 1 - \log \rho$ with $p_t = \rho_t q_t$. □

Eq. (9)**.** $h'_\lambda(\kappa) = \log \kappa + \lambda(1 - 1/\kappa) \ge 0$ for $\kappa \ge 1$, so $K_t^\lambda = q_t h_\lambda(\rho_t) \le q_t h_\lambda(1/q_t)$ whenever $1 \le \rho_t \le 1/q_t$. Expanding gives the stated expression, whose derivative in q_t is $(1-1/q_t)+\lambda \log q_t < 0$ on $(0, 1)$. Along a correct response $\sum_t \log \rho_t = \log \kappa_1 \approx \log(1/P(x))$, and along an incorrect response $\sum_t \log \rho_t = \log \kappa_0 \approx -1/\beta - \log P(x)$.

Eq. (10)**.** By Proposition 3.4, the total contribution of class r is $h_\lambda(\kappa_r) \sum_{y:r(x,y)=r} q_y$, which equals $P(x)h_\lambda(\kappa_1)$ for $r = 1$ and $(1 - P(x))h_\lambda(\kappa_0)$ for $r = 0$. The values quoted in Remark 3.8 follow by direct evaluation.

B Implementation Details

B.1 Data Construction

For each training prompt, the initial policy π_0 generates $N_l = 16$ responses at $\tau_l = 0.3$ and $N_h = 16$ at $\tau_h = 1.0$, without top-p or top-k truncation. Rollouts use the chat template with thinking disabled, seed 42, and a stop at the closing `</answer>` tag, with at most 1,024 new tokens for GSM8K and MATH and 2,048 for MBPP. Validation sets are 500 held-out training questions for GSM8K and for MATH, and MBPP task IDs 511–600. The verifier assigns $r = 1$ to correct and $r = 0$ to incorrect responses: final-answer matching for GSM8K and MATH-500, and the unit tests for MBPP-full. Malformed or truncated responses are kept and verified normally; a response from which no answer can be extracted or verified receives $r = 0$. Duplicate responses are kept as separate instances. A prompt is eligible if it has at least $B_+ = 4$ correct τ_h responses and at least $B_- = 4$ incorrect τ_l responses, and GIST samples four of each uniformly without replacement. Table 7 reports the eligible prompts; on Qwen3-0.6B/GSM8K, GIST retains 12,744 of the 50,976 candidates of the 1,593 eligible prompts. In Table 2, BOLT and SFT train on all 32 candidates of every training prompt, and each objective with + GIST trains on GIST's selected subset. The controlled comparisons of Sections 5.2–5.4 use only the eligible prompts, and "all candidates" there refers to the 32 candidates of each eligible prompt.

Table 7: Eligible prompts over training prompts.

Model	GSM8K	MATH	MBPP
Qwen3-0.6B	1,593 / 6,973	1,619 / 7,000	66 / 374
Qwen3.5-2B	2,574 / 6,973	1,100 / 7,000	65 / 374
Gemma-4-E2B-it	1,373 / 6,973	1,646 / 7,000	58 / 374

B.2 Token Masks

Token probabilities $q_t = \pi_0(y_t \mid x, y_{<t})$ are computed once with the frozen initial policy and without temperature scaling, and the masks stay fixed during training. The mean $\bar{q}$ is taken over the eligible

response positions, which exclude prompt, padding, and EOS positions, so EOS never enters $\bar{q}$ or the mask. Comparisons are strict, so positions equal to the mean are not selected. Whenever GIST's token selection is applied, correct responses use $m_t^+ = \mathbf{1}[q_t < \bar{q}]$ and incorrect responses use $m_t^- = \mathbf{1}[q_t > \bar{q}]$; the baselines keep their own token rules (Appendix C). Unselected positions remain in the context but receive no loss.

B.3 Training Objective

BOLT weights are computed within each prompt's full pool of 32 candidates before selection, as described in Section 4.3, with $r \in \{0, 1\}$ and $\beta = 0.1$, and are not renormalized after selection. An incorrect response therefore has weight $e^{-10} \approx 4.5 \times 10^{-5}$ times that of a correct response to the same prompt. No off-policy correction is applied to responses from the τ_l pool, so $32/n_1$ estimates the inverse success rate of the mixed-temperature pool rather than $1/P(x)$; the BOLT baseline uses the same mixed pool. For a prompt whose 32 candidates are all incorrect, every weight equals one, so BOLT imitates these responses; GIST excludes such prompts through eligibility. The loss multiplies each response's mean masked token negative log-likelihood by its weight and averages over the minibatch. SFT uses the same loss with $w = 1$ for every response. SFT + GIST, which is also row (e) of Table 4, applies GIST's selection and masks, so the positions of incorrect responses that m^- selects are imitated with the same weight as the positions of correct ones. All other settings match BOLT. We fine-tune LoRA adapters (rank 8, $\alpha = 64$, dropout 0.05) with fused AdamW ($\beta_1 = 0.9$, $\beta_2 = 0.95$, weight decay 0.1), a constant schedule with 10% warmup, BF16, and gradient checkpointing. Learning rates are 5×10^{-6} for GSM8K and 10^{-5} for MATH and MBPP; the global batch size is 32 for GSM8K and 128 for MATH and MBPP, and the maximum training length is 1,024 and 4,096 tokens, respectively. Training runs for up to 12 epochs with early stopping (patience 2) after at least 5 epochs, and the checkpoint with the best validation Strict accuracy (pass@1 for MBPP) is evaluated on the test set.

B.4 Evaluation

GSM8K is evaluated with lm-evaluation-harness (Gao et al., 2024) using 8-shot Q:/A: prompts without the chat template and at most 256 new tokens. GSM8K Strict requires "The answer is ..." followed by an exact match, and MATH Strict requires an exact match of the normalized last boxed answer; Flexible accuracy extracts the final answer from other formats. MBPP-full reports pass@1, executing the benchmark's test cases in a bubblewrap sandbox with a 5-second timeout and CPU, memory, and file-size limits. Each entry is a single training run. The binomial standard error of a single test accuracy is about 1.4 points on GSM8K ($n = 1{,}319$) and 2.2 points on MATH-500 and MBPP-full ($n = 500$); Appendix B.5 reports paired tests for the main GSM8K comparisons.

B.5 Paired Tests and Token Statistics

Table 8 reports paired tests on the 1,319 GSM8K test questions (Qwen3-0.6B), using per-question correctness of the two compared checkpoints: a paired bootstrap 95% interval for the accuracy difference (Efron & Tibshirani, 1993) and McNemar's test (McNemar, 1947), following Dror et al. (2018).

Table 8: Paired tests on GSM8K (Qwen3-0.6B, 1,319 questions). Differences are computed from per-question counts and may differ by 0.01 from differences of the rounded accuracies in the main text.

Comparison	Metric	Difference (pp)	Bootstrap 95% CI	McNemar p
GIST − BOLT	Strict	$+2.50$	$[+0.30, +4.62]$	0.027
	Flex.	$+2.27$	$[+0.15, +4.40]$	0.044
GIST − Random	Strict	$+2.65$	$[+0.61, +4.70]$	0.014
	Flex.	$+1.14$	$[-0.83, +3.11]$	0.293
GIST − All candidates	Strict	$+1.97$	$[-0.08, +4.02]$	0.072
	Flex.	$+3.03$	$[+0.99, +5.08]$	0.005

Below-mean tokens by sampling temperature. We measure the fraction of positions with $q_t < \bar{q}$ in correct responses, excluding EOS. Pooled over tokens, it is 31.57% at τ_l and 31.93% at τ_h over all training prompts, and 31.38% and 31.75% over the 1,593 eligible prompts. With each response weighted equally, it is 31.39% and 30.69% over all prompts, and 31.22% and 30.56% over eligible prompts. Comparing correct responses within the same prompt, the high-temperature fraction is lower by 0.36 points over all prompts (95% CI $[-0.50, -0.21]$) and by 0.57 points over eligible prompts ($[-0.74, -0.41]$). Both temperatures thus contain a substantial fraction of below-mean tokens, and the sign of the small difference depends on aggregation, reflecting differences in response length. The probability of the selected tokens differs more: on the eligible prompts, below-mean tokens of correct responses have mean (median) q_t of 0.583 (0.603) at τ_l and 0.411 (0.415) at τ_h. The above-mean tokens that m^- selects in incorrect responses have mean (median) q_t of 0.974 (0.998) at τ_l and 0.956 (0.993) at τ_h.

B.6 Retained Data and Supervision

Table 9 extends Table 6 with the retained responses, selected epochs, and optimizer steps. Supervised tokens count active loss positions through the selected checkpoint, including repeated use across epochs and the positions that m^- selects in incorrect responses. Relative to training on all candidates, Random supervises 51.2% fewer tokens and GIST 66.6% fewer; GIST supervises 31.5% fewer than Random. Random and GIST retain the same number of responses but not the same responses, so the per-epoch reduction from masking also reflects response-length differences.

Table 9: Retained data and supervision on the shared 1,593-prompt pool (Qwen3-0.6B/GSM8K, BOLT).

Quantity	All candidates	Random	GIST
Retained responses	50,976	12,744	12,744
Selected epoch	3	5	6
Supervised tokens (M)	13.596	6.641	4.546
Supervised tokens per epoch (M)	4.53	1.33	0.76
Optimizer steps	4,779	1,995	2,394
Test Strict / Flex. (%)	43.82 / 46.70	43.14 / 48.60	45.79 / 49.73

C Baselines and Ablation Controls

All selectors and controls use the shared candidate pool, the 1,593 eligible prompts, and the BOLT objective of Appendix B, unless stated otherwise.

Selection baselines. *Random* samples four correct and four incorrect responses per prompt uniformly from the combined temperature pools and supervises all eligible tokens. *RFT-style* retains all verifier-correct responses. The HES, PODS, ProFit, ETS, and RSI-S adaptations apply each method's selection rule within this pipeline instead of reproducing its full training algorithm, and each retains eight responses per prompt. HES scores a response by the summed predictive entropy of its top 0.5% highest-entropy positions and keeps the four highest-scoring correct responses and four uniformly chosen incorrect ones. PODS keeps the eight responses that maximize the variance of their binary rewards, which gives four of each class. Both supervise all eligible tokens. ProFit, ETS, and RSI-S keep GIST's responses and select tokens in both classes with one shared rule. ProFit keeps tokens whose frozen-policy probability is at least 0.1 (96.9% of tokens), ETS keeps the 20% of tokens with the highest predictive entropy, and RSI-S keeps tokens whose score $1 + \log p_t / H_t$ lies in $[-6, 0.95]$ (99.3% of tokens).

Ablation controls (Tables 4 and 5). The Mask column of Table 4 and the token rules of Table 5 vary the treatment of correct responses, on which the BOLT weight concentrates. In Table 4, row (a) trains on all 32 candidates of each eligible prompt with all eligible tokens, and row (b) is the Random selector. Row (c) applies GIST's masks to the same responses as Random. Row (d) keeps GIST's responses, supervises all eligible tokens of correct responses, and uses m^- for incorrect ones; it is the All tokens entry of Table 5. Row (e) keeps GIST's responses and masks and sets $w = 1$ for every

response, which is SFT + GIST. In the token-rule ablation, *Random* keeps, in each correct response, as many uniformly chosen positions as GIST selects, and *Highest-probability* keeps the same number of highest-probability positions; incorrect responses use m^- in all variants.