CADI: RESTORING OUTPUT DIVERSITY IN LARGE LANGUAGE MODELS VIA CALIBRATION AND SELF-DISTILLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Large language models (LLMs) often concentrate their outputs on a small set of canonical responses, causing repeated sampling to explore only a narrow subset of valid outputs. Inference-time methods can improve diversity, but their gains depend on persistent prompting or decoding strategies rather than being internalized in model parameters, while training-based methods alter the model itself but typically require carefully designed diversity-specific rewards. In this paper, we propose CADI, a two-stage framework that restores access to valid alternatives suppressed by concentrated model preferences. First, diversity calibration uses simple candidate-selection tasks to reduce overly concentrated model preferences while preserving the model's original preference structure. Second, online guided self-distillation contrasts plain and guidance-conditioned response sets generated by the current model, and distills the guided behavior only when it achieves a Pareto improvement over unguided generation. Experiments on NoveltyBench and Artificial Hivemind show that CADI consistently outperforms strong training-based baselines in semantic diversity and utility across two LLM backbones, while largely preserving response quality and general capabilities. Our code is available at `https://anonymous.4open.science/r/CADI-D4B3`.

1 INTRODUCTION

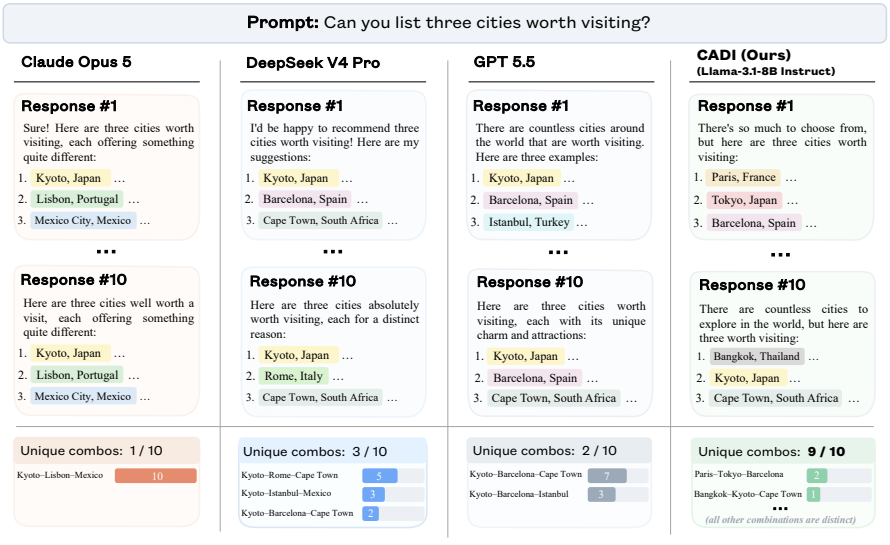


Figure 1: Across 10 samples for the same prompt, Claude Opus 5, DeepSeek V4 Pro, and GPT-5.5 produce only a few unique combinations. In contrast, CADI produces 9 unique combinations, illustrating greater output diversity. See Section 4.1 for experimental details.

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.

Large language models (LLMs) are increasingly used in applications that demand a broad range of useful outputs rather than a single high-quality response. For example, synthetic data generation requires broad coverage (Yu et al., 2023), scientific discovery and creative writing benefit from novel alternatives (Wang et al., 2024a; Lee et al., 2022), open-ended problem solving often calls for exploring multiple plausible solutions. Output diversity is therefore an important dimension of LLM utility, provided that it is achieved without compromising output quality.

However, as illustrated in Figure 1, even state-of-the-art LLMs often concentrate their outputs on a small set of frequent or canonical responses, producing highly similar generations under repeated sampling (Zhang et al., 2025b; Kirk et al., 2024; Anschel et al., 2025). This concentration can arise during both pre-training and post-training. Although pre-training corpora span broad domains, their contents are highly imbalanced, with common concepts and expressions appearing far more frequently than valid long-tail alternatives. Likelihood-based pre-training inherits these frequency imbalances, giving frequent patterns stronger generative priors (Kandpal et al., 2023). Post-training can further sharpen these preferences: supervised fine-tuning (SFT) increases the likelihood of demonstrated responses, while reinforcement learning (RL) can concentrate probability mass on a narrow set of high-reward modes, reducing policy entropy and output diversity (Kirk et al., 2024; Tajwar et al., 2024; Cui et al., 2025). Consequently, LLMs may possess broad knowledge and capabilities, yet expose only a narrow subset through their default generation behavior.

Recent work improves LLM output diversity through either inference-time or post-training interventions. At inference time, typical decoding strategies (Holtzman et al., 2019; Hewitt et al., 2022; Nguyen et al., 2025) seek to improve output diversity by reshaping the token-level sampling distribution of a fixed model. Prompt-based methods instead steer a fixed model toward a broader range of valid responses through diversity-oriented instructions or alternative conditioning contexts (Zhang et al., 2025a; Wang et al., 2025; Ruan et al., 2025). These methods offer flexible control over output diversity at inference time, but leave the model's underlying distribution unchanged, rather than internalizing the diversity gain into its parameters. Training-based approaches instead incorporate diversity into post-training through group-aware rewards (Anschel et al., 2025) or diversity-aware training objectives (Chen et al., 2026; Chung et al., 2025). Although effective, these methods generally rely on multiple responses per prompt and explicit diversity signals, optimizing specialized multi-response objectives or reward designs for the target training tasks rather than learning a general diversity preference that transfers across tasks. Together, these limitations raise a central question:

> *Can we restore access to valid but under-expressed outputs in LLMs without persistent prompting or diversity-specific reward design, while preserving their original capabilities?*

To answer this question, we propose CADI (**CA**libration and Self-**DI**stillation), a two-stage framework that internalizes diverse generation into LLM parameters. In Stage 1, *diversity calibration*, we calibrate model probabilities on simple candidate-selection tasks toward a mixture of the base-model and uniform distributions, reducing excessive concentration while preserving the base model's relative preference structure. This calibrated preference transfers to open-ended generation. In Stage 2, *online guided self-distillation*, the current model produces plain and guidance-conditioned response sets, and only guided behaviors satisfying a margin-constrained Pareto criterion are distilled into an unguided student. The updated student is then used to refresh the training data in the next round. Together, the two stages improve output diversity without persistent inference-time guidance or diversity-specific rewards, while largely preserving the model's original capabilities.

We summarize our main contributions as follows:

- We formalize the relationship between probability concentration and finite-sample output diversity, showing that more concentrated semantic response distributions induce more repeated generations and lower expected coverage.
- We propose CADI, a two-stage framework that restores access to valid alternatives. It combines diversity calibration to reduce excessive concentration with guided self-distillation to selectively internalize verified diversity improvements.
- Experiments on NoveltyBench and Artificial Hivemind across two LLM backbones show that CADI consistently outperforms strong training-based baselines in semantic diversity and utility, while largely preserving response quality and general capabilities.

2 PRELIMINARIES

2.1 PROBLEM FORMULATION

Let x denote an input prompt and $y = (y_1, \ldots, y_T)$ a generated response of length T. An autoregressive language model p_θ factorizes its sequence probability as:

$$p_\theta(y \mid x) = \prod_{t=1}^{T} p_\theta(y_t \mid x, y_{<t}).$$

For theoretical analysis, we abstract the plausible, quality-satisfying responses to x into K semantic equivalence classes $\mathcal{R}_1, \ldots, \mathcal{R}_K$, grouped according to task-level semantic equivalence, and let $\mathcal{R} = \bigcup_{i=1}^{K} \mathcal{R}_i$. The model induces a conditional semantic distribution $\mathbf{p} = (p_1, \ldots, p_K) \in \Delta^{K-1}$ over this response space, where:

$$p_i = \Pr_\theta(y \in \mathcal{R}_i \mid x,\, y \in \mathcal{R}) = \frac{\sum_{y \in \mathcal{R}_i} p_\theta(y \mid x)}{\sum_{y \in \mathcal{R}} p_\theta(y \mid x)}.$$

This finite semantic response space is an analytical abstraction rather than the full natural-language output space. Given N independent samples from $\mathbf{p}$, let n_i denote the number of occurrences of class $\mathcal{R}_i$. We define the finite-sample coverage and its expectation as:

$$C_N(\mathbf{p}) = \sum_{i=1}^{K} \mathbb{I}[n_i > 0], \qquad \mathbb{E}[C_N(\mathbf{p})] = \sum_{i=1}^{K} \left[1 - (1 - p_i)^N\right].$$

For a fixed sampling budget N, our diversity objective is to increase expected coverage $\mathbb{E}[C_N(\mathbf{p})]$ by reducing excessive concentration in $\mathbf{p}$. This analysis is conditional on plausible, quality-satisfying responses, while response quality itself is treated as a separate desideratum.

2.2 PROBABILITY CONCENTRATION LIMITS OUTPUT DIVERSITY

We characterize how concentration in the conditional semantic distribution $\mathbf{p}$ affects diversity under a finite sampling budget. Let $K \geq 2$ and $\mathbf{p}, \mathbf{q} \in \Delta^{K-1}$, and let $\mathbf{p}^\downarrow$ and $\mathbf{q}^\downarrow$ denote $\mathbf{p}$ and $\mathbf{q}$ with their entries sorted in non-increasing order. We say that $\mathbf{p}$ majorizes $\mathbf{q}$, written $\mathbf{p} \succ \mathbf{q}$, when:

$$\sum_{i=1}^{m} p_i^\downarrow \geq \sum_{i=1}^{m} q_i^\downarrow, \qquad m = 1, \ldots, K-1.$$

Thus, $\mathbf{p} \succ \mathbf{q}$ means that $\mathbf{p}$ is at least as concentrated as $\mathbf{q}$. To build intuition, we first examine a simple consequence of concentration: semantic collisions between two independent samples.

Proposition 2.1 (Concentration increases semantic collisions)**.** *The probability that two independent samples from* $\mathbf{p}$ *belong to the same semantic response class is:*

$$P_{\text{coll}}(\mathbf{p}) = \sum_{i=1}^{K} p_i^2.$$

If $\mathbf{p} \succ \mathbf{q}$*, then* $P_{\text{coll}}(\mathbf{p}) \geq P_{\text{coll}}(\mathbf{q})$*, with strict inequality unless the distributions differ only by permutation. Consequently, the expected number of colliding pairs among* N *samples is* $\binom{N}{2} P_{\text{coll}}(\mathbf{p})$.

Proposition 2.1 establishes that concentration increases pairwise redundancy. Pairwise collisions, however, capture redundancy only at the pair level and do not directly quantify how many distinct semantic responses are observed across N samples. We therefore turn to expected coverage.

Proposition 2.2 (Concentration reduces expected coverage)**.** *For every integer* $N \geq 2$*, the expected coverage* $\mathbb{E}[C_N(\mathbf{p})]$ *is strictly Schur-concave in* $\mathbf{p}$*. Therefore:*

$$\mathbf{p} \succ \mathbf{q} \quad \Longrightarrow \quad \mathbb{E}[C_N(\mathbf{p})] \leq \mathbb{E}[C_N(\mathbf{q})],$$

with strict inequality unless the distributions differ only by permutation.

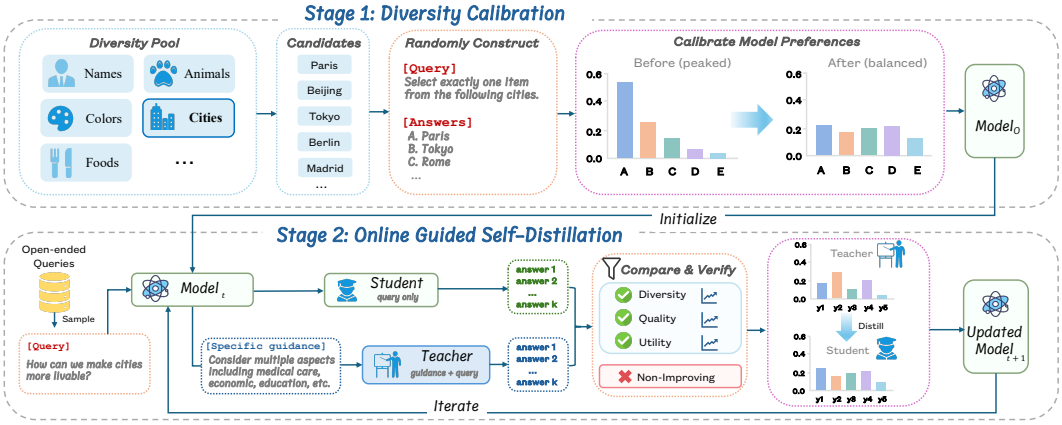


Figure 2: Overview of CADI. Stage 1 performs diversity calibration on controlled candidate-selection tasks, reducing overly concentrated model preferences. Stage 2 compares plain and guidance-conditioned responses from the same frozen model snapshot and distills only verified guided behaviors into an unguided student.

We next characterize the unconstrained extrema of expected coverage by considering the most and least concentrated distributions on the probability simplex.

Corollary 2.3 (Unconstrained coverage extrema)**.** *For any* $\mathbf{p} \in \Delta^{K-1}$ *and* $N \geq 2$*:*

$$1 \leq \mathbb{E}[C_N(\mathbf{p})] \leq K \left[1 - \left(1 - \frac{1}{K}\right)^N\right].$$

The lower bound is attained by a point-mass distribution, whereas the upper bound is attained by the uniform distribution.

Uniformity is only an unconstrained reference for maximal coverage, not a requirement on the model distribution. More importantly, high concentration can leave valid alternatives unobserved under finite sampling despite their presence in the model's support. If $p_i > 0$ for all i, then:

$$\lim_{N \to \infty} \Pr(C_N(\mathbf{p}) = K) = 1.$$

which implies that all supported classes are eventually observed almost surely. However, low-probability classes may remain effectively unobserved at practical sampling budgets. Proofs of the two propositions and the corollary are provided in Appendix B.

3 Methodology

3.1 Stage 1: Diversity Calibration

Motivated by the analysis in Section 2.2, we calibrate overly concentrated model preferences using simple candidate-selection tasks with enumerable valid outputs. Each prompt x is paired with K valid candidates $\mathcal{A}_x = \{a_1, \ldots, a_K\}$, drawn from categories such as animals, cities, and colors, and asks the model to select one. Since all candidates are valid, these tasks isolate the model's relative preference among alternative responses. Writing $a_i = (a_{i,1}, \ldots, a_{i,T_i})$ for the token sequence of candidate a_i with length T_i, and $a_{i,<t}$ for its prefix before position t, we quantify this preference using length-normalized sequence log-likelihoods:

$$s_\theta(a_i \mid x) = \frac{1}{T_i} \sum_{t=1}^{T_i} \log p_\theta(a_{i,t} \mid x, a_{i,<t}), \qquad q_{\theta,i}(x, \mathcal{A}_x) = \frac{\exp(s_\theta(a_i \mid x))}{\sum_{j=1}^{K} \exp(s_\theta(a_j \mid x))}.$$

We denote the resulting candidate-level preference distribution as $\mathbf{q}_\theta(x, \mathcal{A}_x) = (q_{\theta,1}, \ldots, q_{\theta,K})$, with normalization by T_i preventing a systematic preference for shorter candidates.

Base-Anchored Calibration Target. Prior work formulates diffuse generation as matching the model's distribution to a prescribed target over enumerable outputs (Zhang et al., 2024). While a uniform target effectively reduces concentration, directly matching it removes the relative preference structure inherited from the base model and imposes a more aggressive reshaping of the model distribution, which may in turn disrupt its original capabilities (as shown in Section 4.2).

We therefore moderate the strength of this reshaping by retaining the base candidate distribution in the target. Let θ_0 denote the initial model parameters, and let $\mathbf{q}_0 = \mathbf{q}_{\theta_0}(x, \mathcal{A}_x)$ denote the corresponding frozen candidate distribution. We define:

$$\mathbf{q}^\star_\alpha(x, \mathcal{A}_x) = \alpha \mathbf{q}_0 + (1-\alpha)\mathbf{u}_K, \qquad \alpha \in [0, 1],$$

where $\mathbf{u}_K = (1/K, \ldots, 1/K)$ is the uniform distribution. Here, α controls the strength of candidate-level reshaping. For any $\alpha > 0$, the target preserves the weak ordering of the base candidates while shrinking every pairwise probability gap:

$$q^\star_{\alpha,i} - q^\star_{\alpha,j} = \alpha \left(q_{0,i} - q_{0,j}\right).$$

This contraction also implies that $\mathbf{q}_0 \succ \mathbf{q}^\star_\alpha$, so the target is no more concentrated than the base distribution (proof in Appendix B.4).

Calibration Objective. We train the candidate-level distribution to match the controlled target $\mathbf{q}^\star_\alpha$ using the forward KL divergence:

$$\mathcal{L}_{\text{cal}}(\theta) = \mathbb{E}_{(x,\mathcal{A}_x)\sim\mathcal{D}_{\text{cal}}}\left[D_{\text{KL}}(\mathbf{q}^\star_\alpha(x, \mathcal{A}_x) \,\|\, \mathbf{q}_\theta(x, \mathcal{A}_x))\right].$$

Because this candidate-level objective does not directly constrain broader changes in the model's token-level behavior, we additionally regularize the calibrated model toward the original frozen base model at each answer-token position. Specifically, we define:

$$\mathcal{L}_{\text{tok}}(\theta) = \mathbb{E}_{(x,\mathcal{A}_x)\sim\mathcal{D}_{\text{cal}}}\left[\frac{1}{\sum_{i=1}^{K} T_i}\sum_{i=1}^{K}\sum_{t=1}^{T_i} D_{\text{KL}}(p_{\theta_0}(\cdot \mid x, a_{i,<t}) \,\|\, p_\theta(\cdot \mid x, a_{i,<t}))\right].$$

The KL divergence compares the full next-token distributions of the two models under the same prefix and is averaged over all candidate answer tokens. The overall Stage 1 objective is:

$$\mathcal{L}_{\text{Stage1}}(\theta) = \mathcal{L}_{\text{cal}}(\theta) + \beta\mathcal{L}_{\text{tok}}(\theta),$$

Together, the two objectives reduce excessive preference concentration while limiting broader behavioral drift, yielding the calibrated parameters θ_{cal} used to initialize Stage 2.

3.2 Stage 2: Online Guided Self-Distillation

Stage 1 reduces excessive preference concentration, exposing a broader range of responses under finite sampling. Building on this richer distribution, Stage 2 identifies query-specific Pareto improvements and selectively distills them, removing the need for guidance at inference time.

Let θ_{cal} denote the Stage 1 parameters and θ_t the model at Stage 2 round t, initialized as $\theta_1 = \theta_{\text{cal}}$. At the beginning of each round, we freeze the current snapshot $\bar{\theta}_t$. For each prompt x, an external generator produces a query-specific guidance $g = G(x)$, and the frozen snapshot generates:

$$Y_t^{\text{plain}} \sim \pi_{\bar{\theta}_t}(\cdot \mid x), \qquad Y_t^{\text{guided}} \sim \pi_{\bar{\theta}_t}(\cdot \mid x, g).$$

Let $D(Y)$, $U(Y)$, and $Q(Y)$ denote the diversity, utility, and quality of a response set Y, respectively. For $Z \in \{D, U, Q\}$, we define the guidance-induced change as $\Delta Z(x) = Z(Y_t^{\text{guided}}) - Z(Y_t^{\text{plain}})$. We retain only guidance satisfying the margin-constrained Pareto criterion:

$$\bigwedge_{Z\in\{D,U,Q\}} \Delta Z(x) \geq \tau_Z.$$

where τ_Z denotes the acceptance margin for metric Z. Thus, only guidance that satisfies the prescribed improvement margin on all three metrics is retained for distillation. Definitions of these metrics and details of their computation are provided in Section 4.1.

For each guidance g that satisfies the Pareto criterion for prompt x, we treat the guidance-conditioned frozen snapshot as the *guided teacher* and initialize an unguided *student* from θ_t. We partition Y_t^{guided} into m semantic clusters and retain one generated response from each cluster as a teacher trajectory $\mathcal{T}(x) = \{y_1, \ldots, y_m\}$. For token k along trajectory y_j, we define:

$$p^G_{t,k} = \pi_{\bar{\theta}_t}(\cdot \mid x, g, y_{j,<k}), \qquad p^S_k = \pi_\theta(\cdot \mid x, y_{j,<k}),$$

corresponding to the guided teacher and unguided student. The student is optimized with:

$$\mathcal{L}_{\text{Stage2}}(x) = \frac{1}{m} \sum_{j=1}^{m} \frac{1}{|y_j|} \sum_{k=1}^{|y_j|} D_{\text{KL}}\left(p^G_{t,k} \,\|\, p^S_k\right).$$

This objective distills the verified guidance-induced behavior into the unguided student. After each round, the updated model becomes θ_{t+1}, which is frozen to regenerate and reselect data for the next round. At inference time, the final model responds to x without guidance.

4 Experiments

4.1 Experimental Settings

Models and Baselines. We use `Llama-3.1-8B-Instruct` (Grattafiori et al., 2024) and `Qwen2.5-7B-Instruct` (Yang et al., 2024) as our open-weight backbone models. We additionally evaluate three frontier models: `DeepSeek-V4-Pro` (Xu et al., 2026), `GPT-5.5` (OpenAI, 2026), and `Claude Opus 5` (Anthropic, 2026) for reference. We compare against the following training-based diversity baselines: 1) **GAPO** (Anschel et al., 2025), which extends GRPO with group-aware rewards based on the empirical frequency of responses within each rollout group. 2) **DivPO** (Lanchantin et al., 2025), which constructs preference pairs that favor rare, high-quality responses over common, low-quality ones. 3) **DARLING** (Li et al., 2025a), which incorporates a learned diversity reward into on-policy reinforcement learning over self-generated response sets. 4) **DQO** (Chen et al., 2026), which jointly optimizes response quality and semantic diversity using a determinantal point process over groups of response embeddings.

Datasets and Benchmarks. For Stage 1, we construct a synthetic candidate-selection dataset spanning everyday semantic categories, such as animals, cities, colors, and foods, using `GPT-5.5`, followed by manual review and supplementation to ensure candidate validity and category coverage. For Stage 2, we train on 10k WildChat prompts (Zhao et al., 2024) after filtering evaluation-set overlaps, with an additional 1k prompts held out for validation.

We evaluate both open-ended generation diversity and general model capabilities. For open-ended generation diversity, we evaluate on two benchmarks: 1) **NoveltyBench** (Zhang et al., 2025b) measures a model's ability to produce multiple distinct, high-quality responses to the same open-ended prompt. We conduct evaluation on NoveltyBench-Curated. 2) **Artificial Hivemind** (Jiang et al., 2026) evaluates output homogeneity on real-world open-ended queries that admit many plausible responses. We use the human-verified Infinity-Chat100 subset to measure intra-model diversity under repeated sampling. To examine whether improved diversity comes at the cost of capabilities inherited from the base model, we additionally evaluate **HumanEval** (Chen et al., 2021), **MMLU-Pro** (Wang et al., 2024b), and **GSM8K** (Cobbe et al., 2021), covering code generation, knowledge-intensive reasoning, and mathematical reasoning, respectively. Detailed evaluation protocols for the diversity and general-capability benchmarks are provided in Appendices C.2 and C.3, respectively.

Evaluation Metrics. For NoveltyBench, we report Distinct_{10} for the number of semantic response classes covered by ten generations, Quality_{10} for the mean quality across observed classes, and Utility_{10} for a position-discounted measure that rewards early discovery of high-quality new classes. For Artificial Hivemind, we report $\text{Heterogeneity} = 100 \times (1 - \text{Homogeneity})$, where larger values indicate greater semantic variation. We additionally report Div-BLEU, EAD, and Sent-BERT diversity, which measure n-gram-overlap diversity, length-adjusted lexical diversity, and embedding-based semantic diversity, respectively. Metric definitions are provided in Appendix C.2.

Experiment Details. For Stage 1, we use $\alpha = 0.6$ and set the candidate set size to $K = 4$. We use backbone-specific β values, setting $\beta = 0.02$ for Llama-3.1-8B-Instruct and $\beta = 0.005$ for Qwen2.5-7B-Instruct, as the two backbones exhibit different sensitivities to this objective. Both

Table 1: Performance on NoveltyBench and Artificial Hivemind. Div-BLEU, EAD, and Sent-BERT are macro-averaged across both benchmarks. Bold and underlined values denote the best and second-best results, light and dark blue indicate Stage 1 and the full CADI.

	General Diversity Metrics			NoveltyBench			Artificial Hivemind
Model / Method	**Div-BLEU**	**EAD**	**Sent-BERT**	**Distinct$_{10}$**	**Quality$_{10}$**	**Utility$_{10}$**	**Heterogeneity**
DeepSeek-V4-Pro	46.64	70.41	31.50	5.02	9.06	5.56	20.11
GPT-5.5	31.21	41.06	19.94	2.60	9.10	3.53	13.38
Claude Opus 5	36.97	63.46	26.12	3.46	8.76	3.95	14.65
Llama-3.1-8B-Instruct	45.56	66.10	29.01	5.74	7.78	4.92	19.80
w. GAPO	47.80	67.69	30.25	5.87	7.75	5.19	20.50
w. DivPO	50.62	73.17	30.50	5.94	7.65	5.24	20.70
w. DARLING	56.41	**78.56**	31.17	6.00	8.06	5.53	21.40
w. DQO	44.48	69.45	27.61	5.54	**8.16**	5.15	19.30
w. CADI (Stage 1)	57.74	74.62	34.69	7.03	7.43	5.60	23.10
w. CADI	**58.33**	76.73	**35.74**	**7.21**	7.39	**5.73**	**24.40**
Qwen2.5-7B-Instruct	39.92	59.55	23.99	4.56	7.58	4.05	17.20
w. GAPO	42.96	63.20	25.46	4.97	7.62	4.39	18.00
w. DivPO	46.29	67.17	25.27	5.28	7.08	4.46	17.80
w. DARLING	**51.97**	**72.26**	26.78	5.43	7.12	4.52	18.90
w. DQO	44.68	63.51	25.13	5.06	7.49	4.41	18.20
w. CADI (Stage 1)	45.92	64.70	26.59	5.41	7.66	4.71	19.10
w. CADI	46.49	65.24	**27.38**	**5.56**	**7.68**	**4.89**	**19.40**

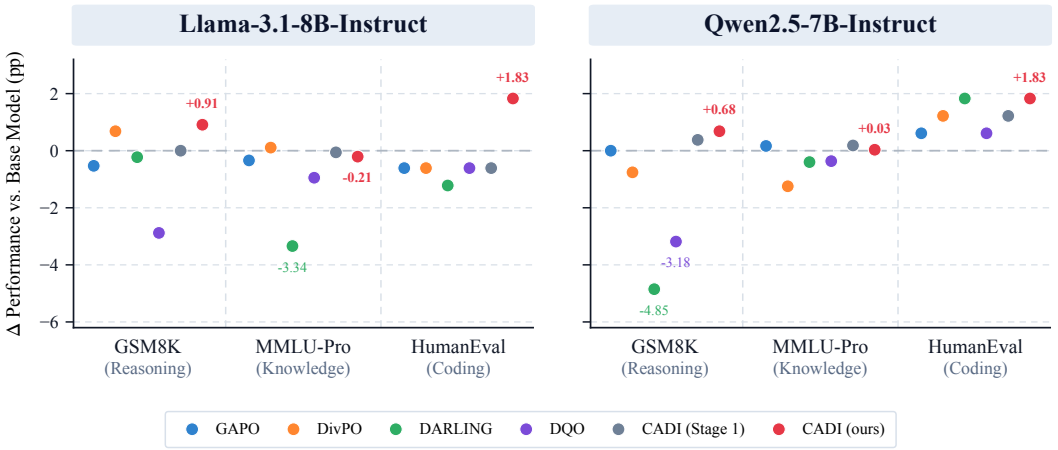


Figure 3: General capability changes after diversity training. Each point shows the percentage-point change relative to the corresponding base model on GSM8K, MMLU-Pro, and HumanEval.

models are trained for 800 optimization steps with an initial learning rate of 5×10^{-5}, using 50 warmup steps followed by a cosine learning-rate schedule (Loshchilov & Hutter, 2016).

For Stage 2, we use `GPT-5.5` to generate concise query-specific guidance for each prompt. We use a constant learning rate of 5×10^{-6}. We select the final Stage 2 checkpoint using only the held-out validation set and report test-set results only for the selected checkpoint. Complete training hyperparameters and implementation details are provided in Appendix C.1.

For `Claude Opus 5`, `GPT-5.5` and `DeepSeek-V4-Pro`, we access the models through OpenRouter[1], with reasoning disabled. All learned open-weight variants are trained with LoRA (Hu et al., 2021). For baseline reproducibility, we use the official implementations for DARLING and DQO, and reimplement DivPO and GAPO following their original papers and hyperparameter configurations as no official code was available. We use `lm-evaluation-harness` (Gao et al., 2021) to

[1] `https://openrouter.ai/`

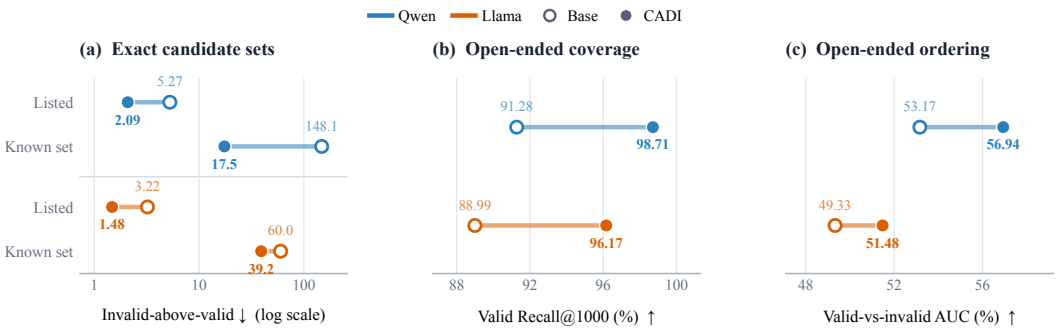


Figure 4: CADI improves the model's underlying ordering of valid continuations. (a) On controlled tasks with exact valid sets, fewer invalid tokens rank above each valid token after calibration. (b–c) On open-ended story continuations, CADI increases both valid-candidate recall within the top 1,000 tokens and valid-versus-invalid ordering AUC.

evaluate GSM8K, MMLU-Pro and HumanEval. Unless otherwise specified, we use sampling with temperature $= 1.0$, top-$p = 1.0$ and top-$k = 10$.

4.2 Results

Main Results. Table 1 shows that CADI consistently improves semantic coverage and utility across both backbones. Its advantage is particularly clear on semantic and cluster-based metrics: although DARLING attains stronger lexical diversity on Qwen, CADI achieves higher Sent-BERT diversity, Distinct_{10}, Utility_{10}, and Heterogeneity. This gap suggests that increased surface-form variation does not necessarily translate into broader semantic coverage or more useful response sets.

The API models exhibit a similar distinction. All three achieve markedly higher response quality than the open-weight backbones, while DeepSeek-V4-Pro shows substantially stronger diversity and utility than GPT-5.5 and Claude Opus 5. This indicates that output diversity remains highly model-dependent even among frontier models, rather than simply tracking response quality.

Notably, much of CADI's gain already emerges after Stage 1, despite calibration using only simple candidate-selection tasks. The gains transfer to both open-ended benchmarks and semantic metrics not directly optimized during calibration, supporting the hypothesis that reshaping concentrated preferences generalizes beyond the calibration tasks. Stage 2 provides further improvements by selectively internalizing query-specific guided behaviors that satisfy the Pareto criterion. CADI also remains effective at larger model scales, as shown in Appendix C.4.

Preference-Order Analysis. To examine whether CADI restores access to valid alternatives rather than merely flattening the training candidates, we analyze how calibration changes the ordering of valid continuations. As shown in Figure 4, CADI reduces the number of invalid tokens ranked above valid candidates on controlled tasks with known valid sets. More importantly, this effect transfers to open-ended continuations, where calibration improves both valid-candidate recall among top-ranked tokens and valid-versus-invalid ordering AUC on both backbones.

These results suggest that diversity calibration restores the accessibility of valid alternatives that were previously suppressed by concentrated model preferences. This provides evidence that the gains from simple candidate-selection tasks reflect a broader change in the model's preference ordering, rather than task-specific uniformization. Additional experimental details and per-prefix diagnostics are provided in Appendix C.5.

4.3 Ablation and Sensitivity Analysis

Ablation Study. Figure 5 highlights the complementary roles of CADI's core components. Removing Stage 1 causes the largest drop in semantic coverage and utility, underscoring calibration as the primary source of transferable diversity gains. Removing either the base-aware target or the

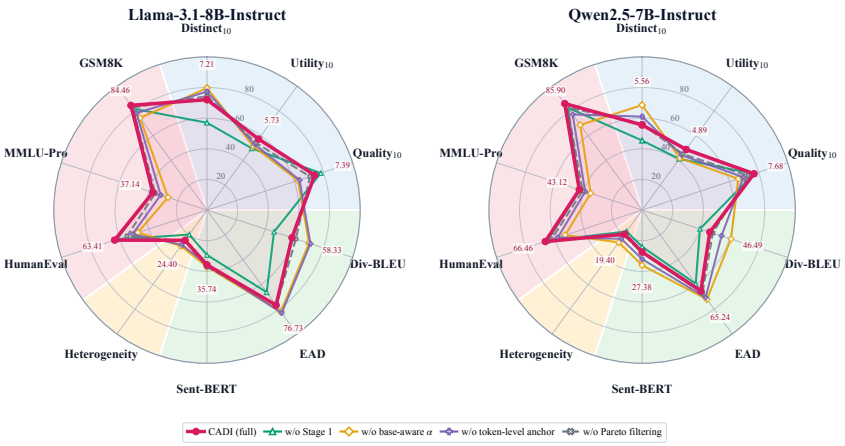


Figure 5: Ablation of key CADI components on Llama-3.1-8B-Instruct and Qwen2.5-7B-Instruct. Shaded sectors group metrics by evaluation category. General-capability deviations from the full CADI model are magnified by 2.5× for visibility. Numeric annotations report the original scores.

token-level anchor can improve some diversity metrics, but consistently degrades response quality and general capabilities, showing that these components are essential for controlling distributional drift. Similarly, removing Pareto filtering yields occasional diversity gains at the expense of quality and utility. Overall, the full method achieves the most consistent performance across diversity, quality, and capability preservation.

Sensitivity to Calibration Hyperparameters. To isolate the effect of each hyperparameter, we sweep α with $\beta = 0$ and β with $\alpha = 0$. At the unconstrained point $(\alpha, \beta) = (0, 0)$, the two backbones exhibit markedly different behavior, indicating different sensitivities to aggressive calibration. Despite this difference, both show a consistent overall trend: introducing moderate regularization improves the diversity–quality balance, whereas stronger retention progressively shifts the model toward higher quality but smaller diversity gains. These results support the need for backbone-specific calibration strengths rather than a universally fixed setting.

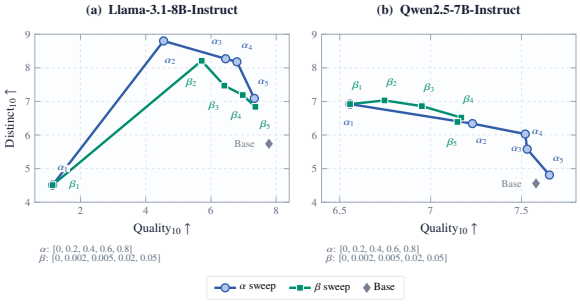


Figure 6: Sensitivity of CADI to hyperparameters.

5 Conclusion

We study output diversity in LLMs from the perspective of learned preference concentration. We first show that more concentrated semantic response distributions lead to greater repetition and lower expected coverage under finite sampling. Building on this analysis, we introduce CADI, a two-stage framework that combines controlled diversity calibration with online guided self-distillation. Stage 1 reduces excessive preference concentration while limiting distributional drift, and Stage 2 selectively internalizes guidance-induced behaviors that yield verified improvements. Across NoveltyBench and Artificial Hivemind, CADI consistently improves semantic diversity and utility over strong training-based baselines on two LLM backbones, while largely preserving general capabilities. These results suggest that output diversity can be restored by recalibrating concentrated model preferences and selectively internalizing verified guided behaviors, rather than relying solely on inference-time interventions or task-specific diversity rewards.

AI USE STATEMENT

In this work, we used generative AI tools to aid and polish writing, to support retrieval and discovery of related work, and to generate synthetic training materials. Specifically, `GPT-5.5` was used to generate the candidate-selection data for Stage 1 and the query-specific guidance for Stage 2, as described in Sections 3.1 and 3.2. All AI-assisted text was reviewed and revised by the authors, suggested references were verified against the original sources, the Stage 1 data were inspected before use, and the Stage 2 guidance was retained only through the Pareto verification procedure described in Section 3.2. We have not used generative AI tools for other tasks requiring disclosure, and the remaining disclosure categories are not applicable to this work. We take responsibility for the final content of this work, including text, claims, citations, and artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

The two stages of CADI are described in Sections 3.1 and 3.2, with their end-to-end procedures given in Algorithms 1 and 2 in Appendix D. Section 4.1 specifies the models, data, baselines, and evaluation settings, while Appendix C.1 reports the core training hyperparameters and implementation details. Complete evaluation protocols for the diversity and general-capability benchmarks are provided in Appendices C.2 and C.3. The training and evaluation code, configuration files, synthetic Stage 1 data, and Stage 2 training and validation data are available in our anonymous repository at `https://anonymous.4open.science/r/CADI-D4B3`.

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.

Oron Anschel, Alon Shoshan, Adam Botach, Shunit Haviv Hakimi, Asaf Gendler, Emanuel Ben Baruch, Nadav Bhonker, Igor Kviatkovsky, Manoj Aggarwal, and Gerard Medioni. Group-aware reinforcement learning for output diversity in large language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 32382–32403, 2025.

Anthropic. Introducing Claude Opus 5, July 2026. URL `https://www.anthropic.com/news/claude-opus-5`.

Amanda Askell, Yuntao Bai, Anna Chen, Dawn Drain, Deep Ganguli, Tom Henighan, Andy Jones, Nicholas Joseph, Ben Mann, Nova DasSarma, et al. A general language assistant as a laboratory for alignment. *arXiv preprint arXiv:2112.00861*, 2021.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021.

Yilei Chen, Souradip Chakraborty, Lorenz Wolf, Ioannis Paschalidis, and Aldo Pacchiano. Post-training large language models for diverse high-quality responses. In *International Conference on Learning Representations*, volume 2026, pp. 127946–127975, 2026.

John Joon Young Chung, Vishakh Padmakumar, Melissa Roemmele, Yuqian Sun, and Max Kreminski. Modifying large language model post-training for diverse creative writing. *arXiv preprint arXiv:2503.17126*, 2025.

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. *arXiv preprint arXiv:2110.14168*, 2021.

Ganqu Cui, Yuchen Zhang, Jiacheng Chen, Lifan Yuan, Zhi Wang, Yuxin Zuo, Haozhan Li, Yuchen Fan, Huayu Chen, Weize Chen, et al. The entropy mechanism of reinforcement learning for reasoning language models. *arXiv preprint arXiv:2505.22617*, 2025.

Zhiyuan Fan, Guanqiao Chen, Yanyi Huang, Mingkuan Zhao, Dadi Guo, and Yi R Fung. Learning diverse responses with prefix-conditioned supervised fine-tuning. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 247–276, 2026.

Leo Gao, Jonathan Tow, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Kyle McDonell, Niklas Muennighoff, et al. A framework for few-shot language model evaluation. *Zenodo*, 2021.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

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.

John Hewitt, Christopher D Manning, and Percy Liang. Truncation sampling as language model desmoothing. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pp. 3414–3427, 2022.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. The curious case of neural text degeneration. *arXiv preprint arXiv:1904.09751*, 2019.

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

Liwei Jiang, Yuanjun Chai, Margaret Li, Mickel Liu, Raymond Fok, Nouha Dziri, Yulia Tsvetkov, Maarten Sap, and Yejin Choi. Artificial hivemind: The open-ended homogeneity of language models (and beyond). *Advances in Neural Information Processing Systems*, 38, 2026.

Renren Jin, Pengzhi Gao, Yuqi Ren, Zhuowen Han, Tongxuan Zhang, Wuwei Huang, Wei Liu, Jian Luan, and Deyi Xiong. Revisiting entropy in reinforcement learning for large reasoning models. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 25300–25322, 2026.

Nikhil Kandpal, Haikang Deng, Adam Roberts, Eric Wallace, and Colin Raffel. Large language models struggle to learn long-tail knowledge. In *International conference on machine learning*, pp. 15696–15707. PMLR, 2023.

Yoon Kim and Alexander M Rush. Sequence-level knowledge distillation. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pp. 1317–1327, 2016.

Robert Kirk, Ishita Mediratta, Christoforos Nalmpantis, Jelena Luketina, Eric Hambro, Edward Grefenstette, and Roberta Raileanu. Understanding the effects of rlhf on llm generalisation and diversity. In *International Conference on Learning Representations*, volume 2024, pp. 20620–20653, 2024.

Oh Joon Kwon, Daiki E Matsunaga, and Kee-Eung Kim. Gdpo: Learning to directly align language models with diversity using gflownets. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 17120–17139, 2024.

Jack Lanchantin, Angelica Chen, Shehzaad Dhuliawala, Ping Yu, Jason Weston, Sainbayar Sukhbaatar, and Ilia Kulikov. Diverse preference optimization. *arXiv preprint arXiv:2501.18101*, 2025.

Mina Lee, Percy Liang, and Qian Yang. Coauthor: Designing a human-ai collaborative writing dataset for exploring language model capabilities. In *Proceedings of the 2022 CHI conference on human factors in computing systems*, pp. 1–19, 2022.

Tianjian Li, Yiming Zhang, Ping Yu, Swarnadeep Saha, Daniel Khashabi, Jason Weston, Jack Lanchantin, and Tianlu Wang. Jointly reinforcing diversity and quality in language model generations. *arXiv preprint arXiv:2509.02534*, 2025a.

Ziniu Li, Congliang Chen, Tian Xu, Zeyu Qin, Jiancong Xiao, Zhi-Quan Luo, and Ruoyu Sun. Preserving diversity in supervised fine-tuning of large language models. In *International Conference on Learning Representations*, volume 2025, pp. 66127–66154, 2025b.

Siyang Liu, Sahand Sabour, Yinhe Zheng, Pei Ke, Xiaoyan Zhu, and Minlie Huang. Rethinking and refining the distinct metric. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 762–770. Association for Computational Linguistics, 2022. doi: 10.18653/v1/2022.acl-short.86. URL https://aclanthology.org/2022.acl-short.86/.

Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016.

Kou Misaki and Takuya Akiba. String seed of thought: Prompting llms for distribution-faithful and diverse generation. In *International Conference on Learning Representations*, volume 2026, pp. 89078–89136, 2026.

Minh Nguyen, Andrew Baker, Clement Neo, Allen Roush, Andreas Kirsch, and Ravid Shwartz-Ziv. Turning up the heat: Min-p sampling for creative and coherent llm outputs. In *International Conference on Learning Representations*, volume 2025, pp. 70333–70366, 2025.

OpenAI. GPT-5.5 System Card, April 2026. URL https://openai.com/index/gpt-5-5-system-card/.

Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence embeddings using siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pp. 3982–3992, 2019. doi: 10.18653/v1/D19-1410.

Zhiwen Ruan, Yixia Li, Yefeng Liu, Yun Chen, Weihua Luo, Peng Li, Yang Liu, and Guanhua Chen. G2: Guided generation for enhanced output diversity in llms. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 14127–14145, 2025.

Idan Shenfeld, Mehul Damani, Jonas Hübotter, and Pulkit Agrawal. Self-distillation enables continual learning. *arXiv preprint arXiv:2601.19897*, 2026.

Ruiqi Shi and Sinno Jialin Pan. Diverse text decoding via iterative reweighting. In *The Fourteenth International Conference on Learning Representations*, 2026.

Stewart Slocum, Asher Parker-Sartori, and Dylan Hadfield-Menell. Diverse preference learning for capabilities and alignment. In *International Conference on Learning Representations*, volume 2025, pp. 24760–24790, 2025.

Charlie Snell, Dan Klein, and Ruiqi Zhong. Learning by distilling context. *arXiv preprint arXiv:2209.15189*, 2022.

Fahim Tajwar, Anikait Singh, Archit Sharma, Rafael Rafailov, Jeff Schneider, Tengyang Xie, Stefano Ermon, Chelsea Finn, and Aviral Kumar. Preference fine-tuning of llms should leverage suboptimal, on-policy data. *arXiv preprint arXiv:2404.14367*, 2024.

Qihan Wang, Shidong Pan, Tal Linzen, and Emily Black. Multilingual prompting for improving llm generation diversity. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 6378–6400, 2025.

Qingyun Wang, Doug Downey, Heng Ji, and Tom Hope. Scimon: Scientific inspiration machines optimized for novelty. In *Proceedings of the 62nd annual meeting of the association for computational linguistics (volume 1: long papers)*, pp. 279–299, 2024a.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, et al. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark. *Advances in Neural Information Processing Systems*, 37:95266–95290, 2024b.

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*, 2026.

Shaoyang Xu and Wenxuan Zhang. Language of thought shapes output diversity in large language models. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 13802–13816, 2026.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, et al. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

Tianzhu Ye, Li Dong, Xun Wu, Shaohan Huang, and Furu Wei. On-policy context distillation for language models. *arXiv preprint arXiv:2602.12275*, 2026.

Yue Yu, Yuchen Zhuang, Jieyu Zhang, Yu Meng, Alexander J Ratner, Ranjay Krishna, Jiaming Shen, and Chao Zhang. Large language model as attributed training data generator: A tale of diversity and bias. *Advances in neural information processing systems*, 36:55734–55784, 2023.

Longfei Yun, Chenyang An, Zilong Wang, Letian Peng, and Jingbo Shang. The price of format: Diversity collapse in llms. In *EMNLP (Findings)*, pp. 15454–15468, 2025.

Yongcheng Zeng, Guoqing Liu, Weiyu Ma, Ning Yang, Haifeng Zhang, and Jun Wang. Token-level direct preference optimization. *arXiv preprint arXiv:2404.11999*, 2024.

Jiayi Zhang, Simon Yu, Derek Chong, Anthony Sicilia, Michael R Tomz, Christopher D Manning, and Weiyan Shi. Verbalized sampling: How to mitigate mode collapse and unlock llm diversity. *arXiv preprint arXiv:2510.01171*, 2025a.

Yiming Zhang, Avi Schwarzschild, Nicholas Carlini, Zico Kolter, and Daphne Ippolito. Forcing diffuse distributions out of language models, 2024.

Yiming Zhang, Harshita Diddee, Susan Holm, Hanchen Liu, Xinyue Liu, Vinay Samuel, Barry Wang, and Daphne Ippolito. Noveltybench: Evaluating language models for humanlike diversity. *arXiv preprint arXiv:2504.05228*, 2025b.

Wenting Zhao, Xiang Ren, Jack Hessel, Claire Cardie, Yejin Choi, and Yuntian Deng. Wildchat: 1m chatgpt interaction logs in the wild. In *International Conference on Learning Representations*, volume 2024, pp. 34590–34605, 2024.

Yaoming Zhu, Sidi Lu, Lei Zheng, Jiaxian Guo, Weinan Zhang, Jun Wang, and Yong Yu. Texygen: A benchmarking platform for text generation models. In *The 41st international ACM SIGIR conference on research & development in information retrieval*, pp. 1097–1100, 2018.

A Related Work

Diversity Collapse during LLM Training. Recent work attributes diversity collapse to the objectives and optimization dynamics of LLM training. During supervised fine-tuning (SFT), cross-entropy training can narrow the output distribution by concentrating probability mass on demonstrated responses, while instruction-tuning conventions such as structured templates and special tokens may further restrict possible generations (Li et al., 2025b; Yun et al., 2025; Fan et al., 2026). Beyond SFT, preference-based post-training can further concentrate the model distribution toward preferred or high-reward responses (Kirk et al., 2024; Kwon et al., 2024; Zeng et al., 2024; Slocum et al., 2025). This tendency also emerges in reinforcement learning for reasoning as *entropy collapse*, where policy optimization progressively reduces entropy and response diversity (Jin et al., 2026). These findings suggest that diversity collapse is rooted in model training rather than decoding alone.

Diverse Generation in Large Language Models. Existing methods for improving LLM output diversity can be organized by their point of intervention. At inference time, sampling methods reshape or truncate the next-token distribution, such as nucleus sampling (Holtzman et al., 2019), η-sampling (Hewitt et al., 2022), and min-p sampling (Nguyen et al., 2025), while response-aware decoding suppresses semantic patterns repeated across prior generations (Shi & Pan, 2026). Other inference-time methods elicit alternative modes through verbalized response distributions (Zhang et al., 2025a), auxiliary guidance (Ruan et al., 2025), randomized intermediate strings (Misaki & Akiba, 2026), or multilingual reasoning (Xu & Zhang, 2026). In particular, verbalized sampling jointly generates multiple responses together with their probabilities, whereas G2 conditions its guide modules on previously generated responses. Both therefore exploit cross-response information at inference time, requiring persistent conditioning or additional generation-time computation. The resulting overhead can grow with the number and length of sampled responses, making these approaches increasingly costly for long-form generation. Training-based approaches instead reshape the learned policy through diversity-preserving SFT (Li et al., 2025b; Fan et al., 2026), group-aware rewards (Anschel et al., 2025), diversity-aware preference optimization (Chung et al., 2025), or set-level semantic objectives (Chen et al., 2026). In contrast, CADI learns and internalizes a transferable diversity preference, rather than explicitly optimizing diversity over sampled responses for specific training tasks, allowing it to generalize beyond the tasks used for training.

Context Distillation and Self-Distillation. Knowledge distillation for autoregressive generation has evolved from sequence-level supervision (Kim & Rush, 2016) to objectives better suited to generative LLMs, including reverse-KL distillation and training on student-generated trajectories (Gu et al., 2024; Agarwal et al., 2024). A related line of work, context distillation, internalizes behaviors or knowledge elicited by additional context so that the context can be removed at inference time (Askell et al., 2021; Snell et al., 2022). More recent methods construct self-teachers through asymmetric conditioning, using demonstrations or richer contextual information to supervise the same underlying model (Shenfeld et al., 2026; Ye et al., 2026). CADI differs by generating query-specific diversity guidance, retaining only guidance that yields verified improvements, and distilling the resulting guided behaviors into an unguided student.

B PROOFS

B.1 PROOF OF PROPOSITION 2.1

Proof. Let Z_1 and Z_2 denote the semantic-class indices of two independent samples from $\mathbf{p}$. By independence, the collision probability is:

$$\Pr(Z_1 = Z_2) = \sum_{i=1}^{K} \Pr(Z_1 = i) \Pr(Z_2 = i) = \sum_{i=1}^{K} p_i^2 = P_{\text{coll}}(\mathbf{p}).$$

Since $u \mapsto u^2$ is strictly convex, the symmetric function $\sum_{i=1}^{K} p_i^2$ is strictly Schur-convex. Therefore, if $\mathbf{p} \succ \mathbf{q}$, we have:

$$P_{\text{coll}}(\mathbf{p}) \geq P_{\text{coll}}(\mathbf{q}),$$

with equality only when $\mathbf{p}$ and $\mathbf{q}$ differ by a permutation. To obtain the expected number of colliding pairs among N samples, let I_{ab} indicate whether samples a and b belong to the same semantic class. For every $a < b$, its expectation is:

$$\mathbb{E}[I_{ab}] = \Pr(Z_a = Z_b) = P_{\text{coll}}(\mathbf{p}).$$

Applying linearity of expectation then gives:

$$\mathbb{E}\left[\sum_{a<b} I_{ab}\right] = \sum_{a<b} \mathbb{E}[I_{ab}] = \binom{N}{2} P_{\text{coll}}(\mathbf{p}).$$

□

B.2 PROOF OF PROPOSITION 2.2

Proof. Let $I_i = \mathbb{I}[n_i > 0]$ indicate whether semantic class i appears at least once among the N samples. Since each sample misses class i with probability $1 - p_i$, independence gives:

$$\Pr(n_i = 0) = (1 - p_i)^N.$$

The probability that class i is observed at least once is therefore:

$$\mathbb{E}[I_i] = \Pr(n_i > 0) = 1 - (1 - p_i)^N.$$

By linearity of expectation, the expected coverage is:

$$\mathbb{E}[C_N(\mathbf{p})] = \sum_{i=1}^{K} \mathbb{E}[I_i] = \sum_{i=1}^{K} \left[1 - (1 - p_i)^N\right].$$

Define the scalar function:

$$f_N(u) = 1 - (1 - u)^N.$$

For $N \geq 2$, its derivative is:

$$f_N'(u) = N(1 - u)^{N-1},$$

which is strictly decreasing on $[0, 1]$. Hence, f_N is strictly concave. Because $\mathbb{E}[C_N(\mathbf{p})] = \sum_i f_N(p_i)$ is a symmetric sum of a strictly concave function, it is strictly Schur-concave. Consequently:

$$\mathbf{p} \succ \mathbf{q} \quad \Longrightarrow \quad \mathbb{E}[C_N(\mathbf{p})] \leq \mathbb{E}[C_N(\mathbf{q})],$$

with equality only when the two distributions differ by a permutation. □

B.3 Proof of Corollary 2.3

Proof. Consider the point-mass and uniform distributions:

$$\mathbf{e}_1 = (1, 0, \ldots, 0), \qquad \mathbf{u} = \left(\frac{1}{K}, \ldots, \frac{1}{K}\right).$$

The point mass is maximally concentrated on the probability simplex, whereas the uniform distribution is minimally concentrated. Thus, for every $\mathbf{p} \in \Delta^{K-1}$:

$$\mathbf{e}_1 \succ \mathbf{p} \succ \mathbf{u}.$$

Applying Proposition 2.2 yields:

$$\mathbb{E}[C_N(\mathbf{e}_1)] \leq \mathbb{E}[C_N(\mathbf{p})] \leq \mathbb{E}[C_N(\mathbf{u})].$$

Evaluating the two endpoints gives:

$$\mathbb{E}[C_N(\mathbf{e}_1)] = 1, \qquad \mathbb{E}[C_N(\mathbf{u})] = K\left[1 - \left(1 - \frac{1}{K}\right)^N\right].$$

Therefore, the lower bound is attained by any point mass up to permutation, whereas the upper bound is attained by the uniform distribution. □

B.4 Proof of the Base-Preserving Flattening Property

Proof. Let $q_{0,1}^{\downarrow} \geq \cdots \geq q_{0,K}^{\downarrow}$ denote the entries of $\mathbf{q}_0$ sorted in non-increasing order. The calibration target is defined as:

$$\mathbf{q}_\alpha^\star = \alpha \mathbf{q}_0 + (1 - \alpha)\mathbf{u}_K.$$

Since multiplying by $\alpha \geq 0$ and adding the same constant to every coordinate preserves the ordering, the sorted target satisfies:

$$q_{\alpha,i}^{\star\downarrow} = \alpha q_{0,i}^{\downarrow} + \frac{1 - \alpha}{K}.$$

For every $m = 1, \ldots, K - 1$, define the cumulative mass of the largest m entries as:

$$S_m = \sum_{i=1}^{m} q_{0,i}^{\downarrow}.$$

Because the average of the largest m entries cannot be smaller than the overall average of a probability vector, we have:

$$S_m \geq \frac{m}{K}.$$

The corresponding cumulative mass under the calibration target is therefore:

$$\begin{aligned}
\sum_{i=1}^{m} q_{\alpha,i}^{\star\downarrow} &= \alpha S_m + (1 - \alpha)\frac{m}{K} \\
&= S_m - (1 - \alpha)\left(S_m - \frac{m}{K}\right) \\
&\leq S_m \\
&= \sum_{i=1}^{m} q_{0,i}^{\downarrow}.
\end{aligned}$$

Since both distributions sum to one, the defining inequalities of majorization hold for every $m < K$. Hence:

$$\mathbf{q}_0 \succ \mathbf{q}_\alpha^\star.$$

Moreover, when $\alpha < 1$ and $\mathbf{q}_0$ is non-uniform, at least one of these inequalities is strict, so the target is strictly less concentrated than the base distribution. □

C Experiment Details

C.1 Implementation Details and Hyperparameters

The core hyperparameter configurations used to train CADI are listed below.

Table 2: Core training hyperparameters for CADI.

Hyperparameter	**Value**
Stage 1: Diversity Calibration	
Candidate-set size K	4
Base-mixture coefficient α	0.6
Token-level KL weight β	0.005 ($0.02^{\dagger}$)
Optimizer updates	800
Calibration prompts per process	8
Learning rate	5×10^{-5}
Learning-rate schedule	cosine
Warmup steps	50
Stage 2: Online Guided Self-Distillation	
$\lvert Y_t^{\text{plain}}\rvert, \lvert Y_t^{\text{guided}}\rvert$	8, 8
Pareto margins (τ_D, τ_U, τ_Q)	$(0, 0, 1)$
Global prompt batch size	64
Learning rate	5×10^{-6}
Learning-rate schedule	constant
Warmup steps	0
Maximum prompt / completion length	1,200 / 2,048
Shared Optimization Settings	
Optimizer	AdamW
Maximum gradient norm	1.0
LoRA rank / scaling factor / dropout	16 / 32 / 0.05
LoRA target modules	q, k, v, o, gate, up, down projections
Training precision	bfloat16

† Llama-3.1-8B-Instruct uses $\beta = 0.02$.

Code resources. Our training and evaluation code, configuration files, synthetic Stage 1 data, and Stage 2 training and validation data are available at `https://anonymous.4open.science/r/CADI-D4B3`. We use the authors' official implementations of DARLING and DQO, available at `https://github.com/facebookresearch/darling` and `https://github.com/fairytale9/diversity-quality-optimization`, respectively.

C.2 Evaluation Protocols

NoveltyBench. We evaluate all 100 prompts in NoveltyBench-Curated (Zhang et al., 2025b), sampling 10 responses per prompt. We follow the released implementation for greedy semantic clustering using the `microsoft/deberta-v3-large` tokenizer[2] with the `yimingzhang/deberta-v3-large-generation-similarity` classifier[3], and response quality scoring using `Skywork/Skywork-Reward-Gemma-2-27B-v0.2`[4], retaining the official clustering settings and reward-to-quality mapping.

Given k sampled responses, let c_i denote the semantic equivalence class of the i-th response and u_i its quality score. Following NoveltyBench, we define

$$\text{Distinct}_k = \left|\{c_i : i \in [k]\}\right|, \qquad \text{Utility}_k = \frac{1-\rho}{1-\rho^k} \sum_{i=1}^{k} \rho^{i-1} \mathbb{I}[c_i \notin \{c_1, \ldots, c_{i-1}\}]\, u_i,$$

[2] `https://huggingface.co/microsoft/deberta-v3-large`

[3] `https://huggingface.co/yimingzhang/deberta-v3-large-generation-similarity`

[4] `https://huggingface.co/Skywork/Skywork-Reward-Gemma-2-27B-v0.2`

where $\rho = 0.8$ is the official user-patience factor. Utility therefore rewards high-quality semantic classes more when they first appear earlier in the response sequence. To isolate response quality from coverage, let $\mathcal{C}_1, \ldots, \mathcal{C}_{\text{Distinct}_k}$ denote the observed classes and $r_j = \min\{i : c_i = \mathcal{C}_j\}$ the first response assigned to class $\mathcal{C}_j$. We define

$$\text{Quality}_k = \frac{1}{\text{Distinct}_k} \sum_{j=1}^{\text{Distinct}_k} u_{r_j}.$$

Classifier validation. The semantic-equivalence classifier is validated against human judgments rather than model-generated labels. NoveltyBench collected 1,100 human-annotated response pairs from NB-Curated and NB-WildChat, using 1,000 pairs for training and a disjoint 100-pair test set (Zhang et al., 2025b). On this holdout, the released DeBERTa classifier achieves 79% accuracy and an F1 score of 0.811. A subsequent comparison by Li et al. (2025a) evaluates the same 100 human-labeled pairs and reports 70% accuracy for GPT-4o without chain-of-thought, and 54% and 62% for o1-mini without and with chain-of-thought, respectively. We additionally evaluate GPT-5.5 with reasoning effort set to `none` on the same holdout and obtain 72% accuracy (Figure 7). Thus, even the strongest generative judge evaluated here remains 7 percentage points below the specialized encoder-based classifier, supporting our use of the released classifier for semantic clustering.

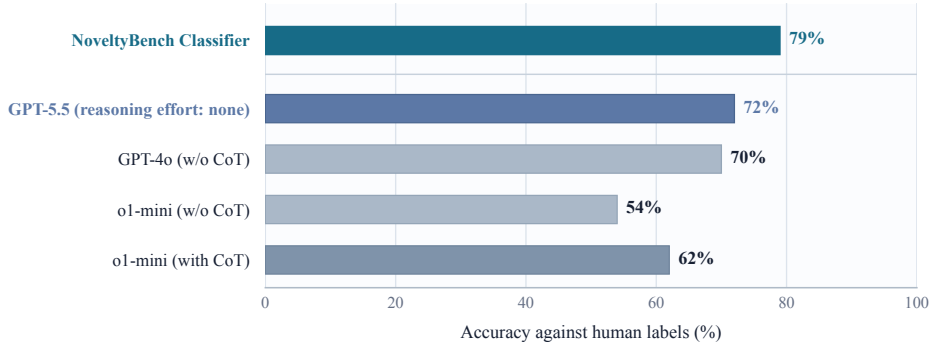


Figure 7: Accuracy against human semantic-equivalence judgments on the 100-pair NoveltyBench holdout. Published comparison values are adapted from Figure 7 of Li et al. (2025a). GPT-5.5 is evaluated on the same holdout with reasoning effort set to `none`. The specialized encoder-based classifier remains 7 percentage points more accurate than the strongest generative judge.

Artificial Hivemind. We evaluate the 100 human-verified prompts in Infinity-Chat100 (Jiang et al., 2026), sampling 50 responses per prompt. Responses are embedded using `text-embedding-3-small`[5]. Homogeneity is computed as the mean pairwise cosine similarity within each response set and averaged uniformly across prompts. We report Heterogeneity as $1 - \text{Homogeneity}$, scaled by 100 in the main table, with larger values indicating greater diversity.

Shared diversity metrics. We compute the additional diversity metrics on each prompt's full response set. Div-BLEU uses Self-BLEU-4 (Zhu et al., 2018). EAD averages expectation-adjusted Distinct-n scores over $n = 1, \ldots, 5$, with vocabulary size 50,257 (Liu et al., 2022). Sent-BERT uses embeddings from `sentence-transformers/bert-large-nli-stsb-mean-tokens` (Reimers & Gurevych, 2019). Scores are averaged uniformly over prompts within each benchmark, then equally across NoveltyBench and Artificial Hivemind, and multiplied by 100 in Table 1.

C.3 General-Capability Benchmarks

We evaluate all local models on the complete test split of each benchmark using `lm-evaluation-harness` v0.4.12 (Gao et al., 2021) with bfloat16 inference and no example limits. We fix the harness seeds to 0 for Python and 1,234 for NumPy, PyTorch, and few-shot

[5] `https://developers.openai.com/api/docs/models/text-embedding-3-small`

sampling. For generation-based tasks, we produce a single deterministic completion per example with sampling disabled, whereas MMLU-Pro is evaluated by multiple-choice likelihood.

HumanEval. HumanEval (Chen et al., 2021) evaluates functional correctness on 164 hand-written Python programming problems. We use the zero-shot completion formulation without a chat template and generate one greedy completion of at most 1,024 tokens using the task's standard code-completion stop sequences. Each completion is executed against the official unit tests, and we report pass@1 using the harness's `create_test` post-processing.

MMLU-Pro. MMLU-Pro (Wang et al., 2024b) is a reasoning-oriented multiple-choice benchmark spanning 14 academic domains, with 10 answer options per question. We evaluate all 12,032 test questions in the 5-shot setting, drawing demonstrations from the validation split. We apply each model's chat template and format the demonstrations as a multi-turn conversation. Following the `leaderboard_mmlu_pro` task, we rank the answer labels by conditional likelihood and report exact multiple-choice accuracy.

GSM8K. GSM8K (Cobbe et al., 2021) contains linguistically diverse grade-school mathematics word problems. We evaluate all 1,319 test problems using the standard 5-shot chain-of-thought demonstrations from the training split, formatted with each model's chat template as a multi-turn conversation. We generate one response per problem with greedy decoding (temperature 0) and a maximum of 1,024 tokens. Following the official harness configuration, we extract the final numeric answer with the `flexible-extract` filter and report exact-match accuracy after standard answer normalization.

C.4 Scaling to Larger Models

To test whether the calibration gains persist beyond the 7–8B backbones used in our main experiments, we apply CADI to Qwen2.5-32B-Instruct and Llama-3.1-70B-Instruct. Both experiments use rank-16 LoRA adapters, $K = 4$, $\alpha = 0.6$, with $\beta = 0.005$ for Qwen and $\beta = 0.02$ for Llama. We compare CADI with its corresponding base model using the same NoveltyBench and Artificial Hivemind protocols described above.

Table 3: CADI scaling results on Qwen2.5-32B-Instruct and Llama-3.1-70B-Instruct. Div-BLEU, EAD, and Sent-BERT are macro-averaged across NoveltyBench and Artificial Hivemind. These metrics and Heterogeneity are multiplied by 100, while the remaining NoveltyBench metrics are reported on their native scales. All metrics are higher-is-better.

	General Diversity Metrics			**NoveltyBench**			**Artificial Hivemind**
Model / Method	**Div-BLEU**	**EAD**	**Sent-BERT**	**Distinct$_{10}$**	**Quality$_{10}$**	**Utility$_{10}$**	**Heterogeneity**
Qwen2.5-32B-Instruct	39.19	57.50	22.36	4.28	**8.18**	4.38	15.33
w. CADI	**49.39**	**66.59**	**27.39**	**5.48**	8.07	**4.97**	**19.13**
Δ	+10.20	+9.09	+5.03	+1.20	-0.11	+0.58	+3.80
Llama-3.1-70B-Instruct	40.86	60.74	26.81	4.21	**8.35**	4.30	19.29
w. CADI	**52.39**	**67.98**	**32.46**	**5.59**	8.12	**5.20**	**22.54**
Δ	+11.52	+7.24	+5.66	+1.38	-0.23	+0.90	+3.26

At the 32B scale, CADI raises Distinct$_{10}$ by 1.20, Utility$_{10}$ by 0.58, and Heterogeneity by 3.80 percentage points, while Quality$_{10}$ decreases by 0.11. The paired Heterogeneity gain has a 95% bootstrap confidence interval of $[+2.52, +5.37]$ percentage points, and 90 of the 100 prompts improve. At the 70B scale, the corresponding gains are 1.38, 0.90, and 3.26 percentage points, with a 0.23 decrease in Quality$_{10}$. Together, these results demonstrate that CADI remains effective when scaled to larger models.

C.5 Order-Calibration Protocol and Per-Prefix Results

We conduct two complementary next-token diagnostics to test whether CADI improves the relative ordering of valid continuations rather than merely flattening the output distribution.

Exact-set evaluation. We first construct controlled tasks for which the complete set of valid answers is known. The evaluation contains 240 held-out displayed-choice prompts, each specifying 6, 8, or 10 equally valid answers, and 78 prompts drawn from finite but undisplayed sets, such as US states, Greek letters, and SI base units. The displayed candidate values are excluded from the Stage 1 calibration data, and all evaluation prompts use new templates. For each task, we tokenize every valid answer and organize the resulting sequences into a prefix trie. At a given prefix, the outgoing trie edges therefore define the exact valid next-token set G_x. Figure 4(a) evaluates the initial generation position. Let $r_x(v)$ denote the vocabulary rank of token v at this position. For each $v \in G_x$, we count the invalid vocabulary tokens ranked above v, and average this quantity over valid tokens and tasks:

$$\mathrm{IAV}(x) = \frac{1}{|G_x|} \sum_{v \in G_x} \left| \{ u \notin G_x : r_x(u) < r_x(v) \} \right|.$$

Lower values indicate that valid continuations are exposed earlier in the model's ranking. This evaluation uses the full next-token distribution and requires neither sampling nor an external evaluator.

Open-ended evaluation. We further evaluate whether the ordering improvement transfers to open-ended generation using 50 fixed story prefixes. For each prefix and backbone, we construct a shared candidate pool from the union of the complete top-100 tokens and every tenth token from ranks 110 through 1,000 under the Base, Stage 1, and Stage 2 checkpoints, yielding roughly 350 candidates per prefix. Because the validity of an isolated token is generally ambiguous, we force each candidate as the next token and greedily complete the sequence for up to 96 additional tokens using the corresponding unadapted Base model. A shared `GPT-5.5` evaluator then scores the resulting continuation for grammaticality, semantic soundness and relevance, and overall quality on a 1–10 scale. A candidate is labeled valid only if all three scores are at least 9. The resulting label is shared across checkpoints, so the comparison isolates how differently they rank the same candidate continuations rather than differences in their subsequent generations.

For each prefix, let $\mathcal{V}_x$ and $\mathcal{I}_x$ denote the evaluator-labeled valid and invalid candidates, respectively. For an evaluated checkpoint m, we report valid-candidate recovery at rank 1,000,

$$\mathrm{Recall@1000}_m(x) = \frac{\sum_{v \in \mathcal{V}_x} \mathbb{I}[r_{m,x}(v) \le 1000]}{|\mathcal{V}_x|},$$

and the sampled valid-versus-invalid ordering AUC,

$$\mathrm{AUC}_m(x) = \frac{1}{|\mathcal{V}_x||\mathcal{I}_x|} \sum_{v \in \mathcal{V}_x} \sum_{u \in \mathcal{I}_x} \mathbb{I}[r_{m,x}(v) < r_{m,x}(u)].$$

The latter is the probability that a labeled valid candidate is ranked above a labeled invalid candidate within the shared candidate pool. We additionally report precision within each checkpoint's fully evaluated top-100 candidates in the per-prefix figures. Aggregate scores are macro-averaged over all 50 prefixes, and paired 95% confidence intervals are estimated from 10,000 prefix-level bootstrap samples. For readability, the per-prefix figures below visualize eight prefixes sampled uniformly at random from this 50-prefix evaluation set.

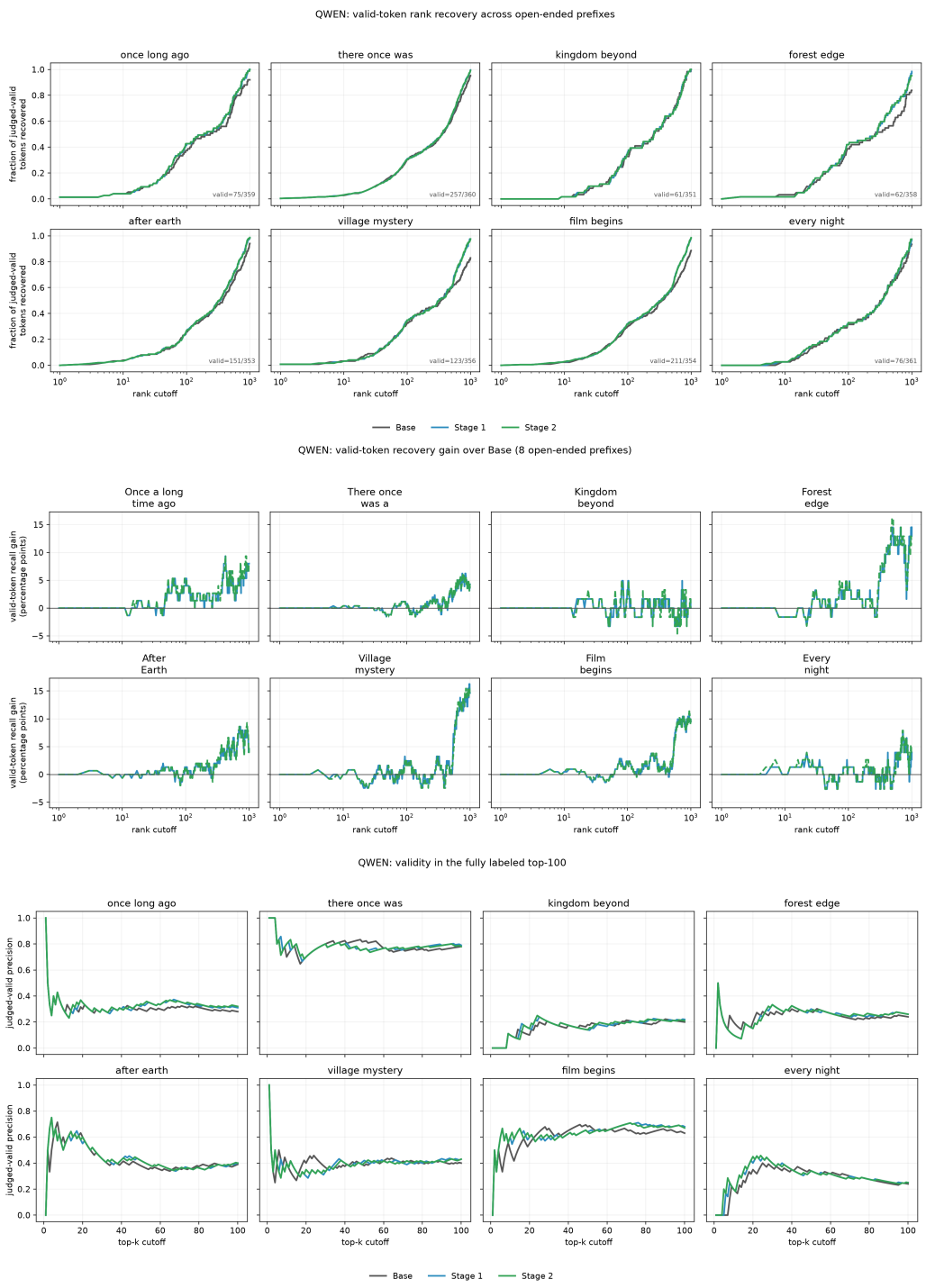


Figure 8: Per-prefix order-calibration diagnostics for eight randomly selected prefixes from the 50-prefix Qwen evaluation. From top to bottom: cumulative valid-token rank recovery, recovery gain over Base in percentage points, and judged-valid precision within the fully labeled top-100 candidates.

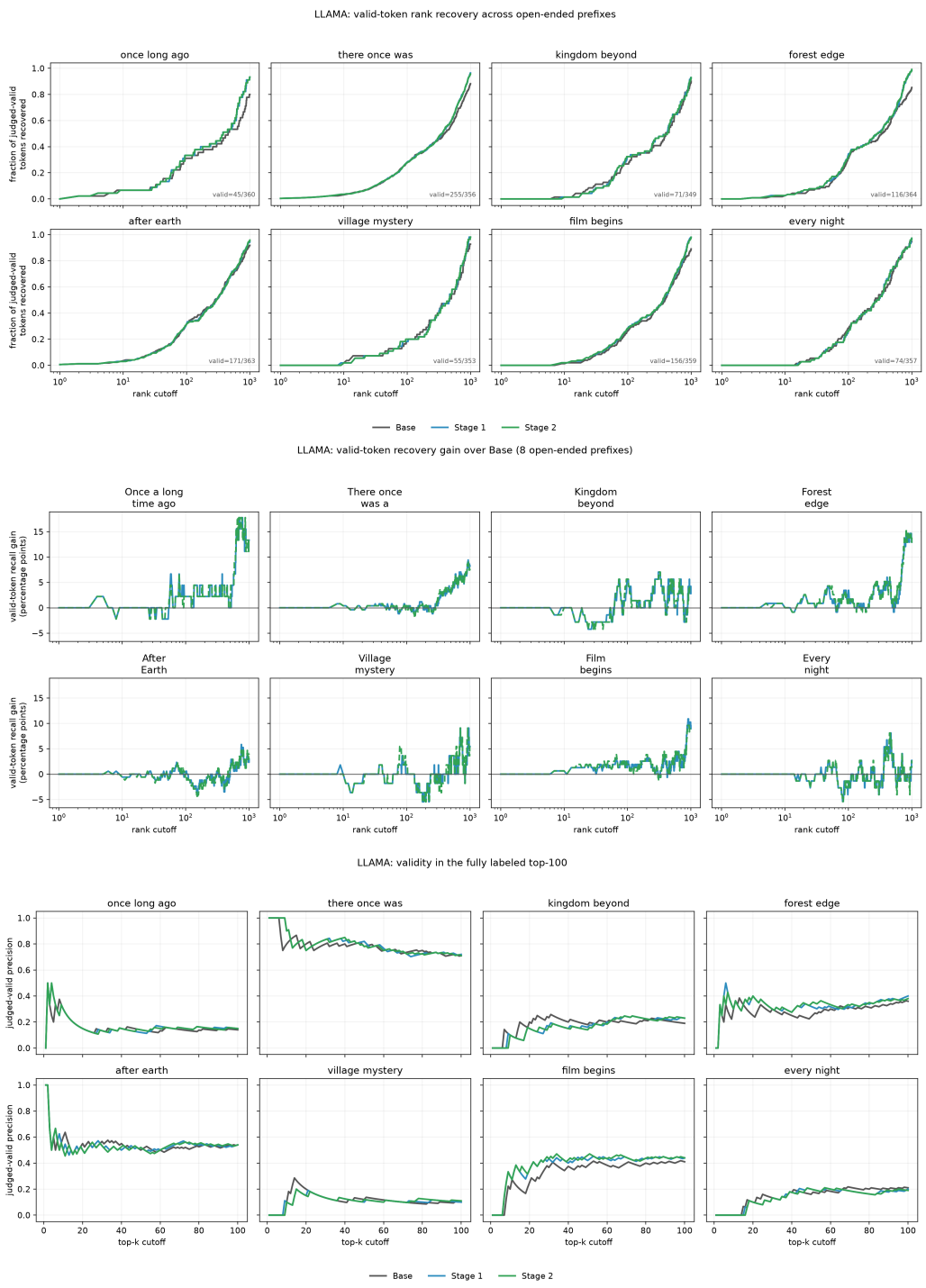


Figure 9: Per-prefix order-calibration diagnostics for eight randomly selected prefixes from the 50-prefix Llama evaluation. From top to bottom: cumulative valid-token rank recovery, recovery gain over Base in percentage points, and judged-valid precision within the fully labeled top-100 candidates.

Figures 8 and 9 visualize eight prefixes sampled uniformly at random from the 50-prefix evaluation set underlying the aggregate order-calibration results in Figure 4. For each displayed prefix, we sweep the rank cutoff and compare the Base, Stage 1, and Stage 2 checkpoints.

The top row reports the fraction of judged-valid candidate tokens recovered by each rank cutoff, so an upward shift indicates that valid continuations become accessible earlier in the model's ranking. The middle row subtracts the Base recovery curve from the Stage 1 and Stage 2 curves, so values above zero indicate improved recovery over Base. The bottom row reports judged-valid precision as the cutoff is swept within the fully annotated top-100 pool. It serves as a control for indiscriminate flattening: improved recovery accompanied by preserved precision indicates that CADI preferentially promotes valid alternatives rather than merely moving arbitrary tokens upward.

Qwen Results. For Qwen, valid-token recall at rank 1,000 increases from 91.28% for Base to 98.71% for CADI, while valid-versus-invalid ordering AUC rises from 53.17% to 56.94%. These gains do not come at the expense of top-ranked validity, with precision at rank 100 increasing from 39.50% to 41.75%. Together, these results indicate that CADI promotes valid alternatives without indiscriminately flattening the distribution.

Llama Results. Llama exhibits the same pattern. Recall at rank 1,000 increases from 88.99% for Base to 96.17% for CADI, while ordering AUC improves from 49.33% to 51.48%. Precision at rank 100 is also maintained, increasing from 33.25% to 34.75%. Although the magnitude varies across prefixes and cutoffs, the dominant gains occur at moderate-to-large cutoffs, where valid but previously suppressed alternatives are recovered.

Overall interpretation. Across both backbones, CADI consistently raises valid-candidate coverage while preserving or modestly improving precision. This joint behavior is inconsistent with indiscriminate distribution flattening, which would be expected to promote invalid candidates as readily as valid ones. Instead, the per-prefix results support the interpretation that calibration improves the relative ordering and accessibility of plausible continuations, and that Stage 2 largely preserves this effect.

C.6 GSM8K Reasoning-Trajectory Diversity

We further test whether CADI's diversity gains transfer to out-of-domain reasoning trajectories. Using all 1,319 GSM8K test questions and the same 5-shot chat-formatted prompts as our capability evaluation, we sample 10 responses per question from Llama-3.1-8B-Instruct Base and CADI with temperature 1.0, top-$k = 10$, top-$p = 1.0$, and at most 1,024 new tokens. To measure reasoning rather than final-answer variation, we remove the final-answer segment, replace numeric literals with `<NUM>`, and embed the remaining trajectories with `sentence-transformers/bert-large-nli-stsb-mean-tokens`. Base and CADI trajectories are jointly clustered within each question by average-linkage agglomerative clustering over cosine distance. All confidence intervals are paired 95% intervals from 10,000 question-level bootstrap samples.

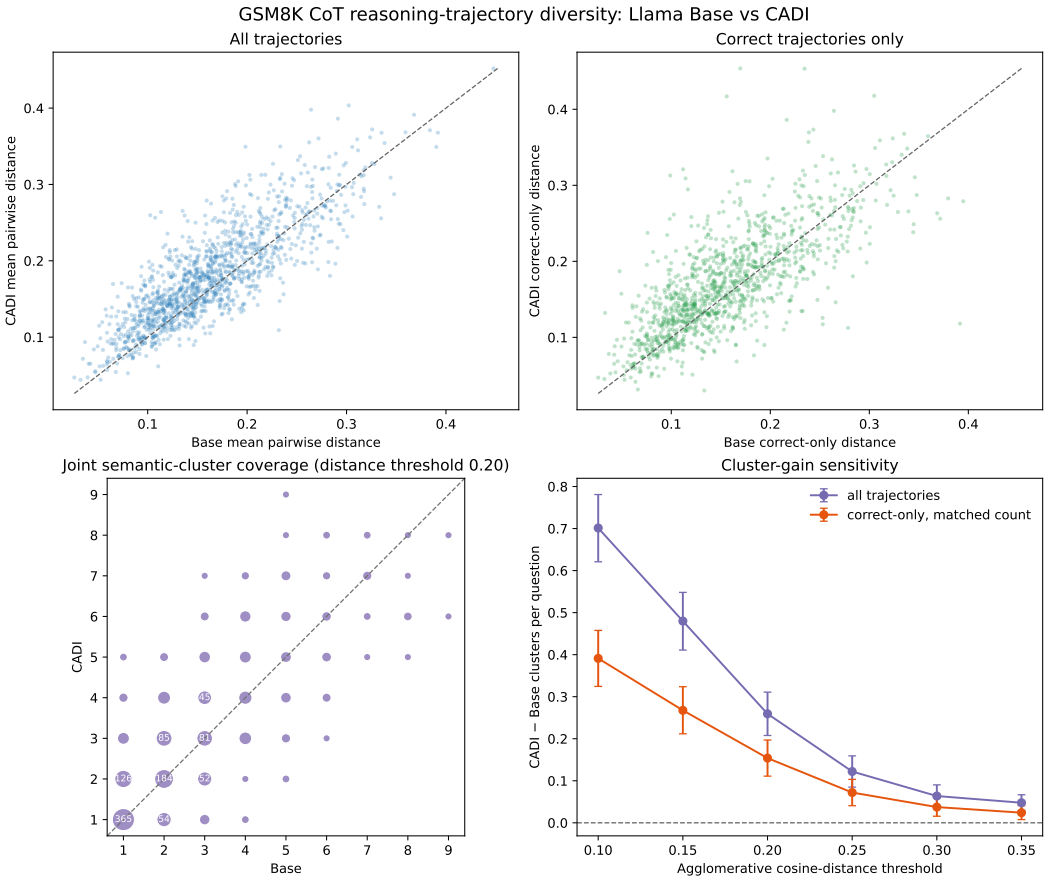


Figure 10: GSM8K reasoning-trajectory diversity for Llama Base and CADI. Each point in the top row represents one question, and points above the dashed identity line indicate greater mean pairwise semantic distance under CADI. Bottom left compares the number of jointly induced semantic clusters at cosine-distance threshold 0.20, with bubble area proportional to the number of questions at each Base–CADI count pair. Bottom right reports the mean per-question cluster-count difference, CADI minus Base, across clustering thresholds, with error bars showing paired 95% bootstrap confidence intervals. Purple uses all trajectories, while orange restricts to correct trajectories with matched sample counts.

The two diagonal plots use the same decision rule: a point above $y = x$ indicates that CADI's sampled trajectories are more semantically dispersed for that question. In the threshold-sensitivity plot, a value above zero indicates that CADI covers more semantic clusters on average. Thus, evidence above the diagonal or above zero consistently favors greater trajectory diversity under CADI.

Semantic dispersion. Across all trajectories (top left), mean pairwise cosine distance increases from 0.1630 for Base to 0.1798 for CADI, a paired gain of $+0.0167$ with 95% CI $[+0.0147, +0.0187]$. More importantly, the gain remains when the analysis is restricted to correct solutions (top right): distance increases from 0.1566 to 0.1707, yielding $+0.0142$ with 95% CI $[+0.0115, +0.0169]$ over the 1,163 eligible questions. This correct-only control shows both that the diversity gain is not solely explained by generating a wider variety of incorrect answers and that the correct reasoning trajectories themselves are also more varied.

Cluster coverage. At the primary cosine-distance threshold of 0.20 (bottom left), CADI covers 2.4958 clusters per question versus 2.2365 for Base, a gain of $+0.2593$ with 95% CI $[+0.2077, +0.3108]$. CADI covers more clusters on 425 questions, ties on 696, and covers fewer on 198. A bubble at $(1, 2)$, for example, denotes questions for which the ten Base trajectories collapse into one semantic cluster while CADI covers two.

Threshold robustness and interpretation. The conclusion is stable across every tested clustering threshold from 0.10 to 0.35 (bottom right): the mean cluster-count difference remains positive, and its paired confidence interval stays above zero, both for all trajectories and for the correct-only matched-count comparison. Together, these results show that CADI transfers greater semantic and procedural variation to GSM8K reasoning, including among correct solutions, rather than exploiting one favorable clustering threshold. We nevertheless interpret semantic clusters conservatively: they capture differences in decomposition or formulation and do not necessarily correspond to fundamentally different algorithms.

C.7 Sensitivity to Candidate Count K

We vary only the number of candidates in each Stage 1 calibration prompt, using $K \in \{3, 4, 6, 8\}$ for Llama-3.1-8B-Instruct while fixing $\alpha = 0.6$, $\beta = 0.02$, the training seed, 800 optimization steps, and all other settings. All variants use the same 36-field training support and are evaluated on NoveltyBench with the sampling configuration in Section 4.1.

Table 4: Stage 1 sensitivity to the number of candidates K. All results are computed on NoveltyBench, with Div-BLEU, EAD, and Sent-BERT shown on their native scales. The adopted $K = 4$ setting is marked in bold.

K	Div-BLEU	EAD	Sent-BERT	Distinct_{10}	Quality_{10}	Utility_{10}
3	0.684	0.794	0.403	7.01	7.480	5.660
4	0.695	0.792	0.400	7.09	7.438	5.655
6	0.695	0.794	0.399	7.08	7.399	5.583
8	0.699	0.790	0.407	7.24	7.321	5.586

Performance is stable over the tested range: the total spreads are only 0.23, 0.159, and 0.077 for Distinct_{10}, Quality_{10}, and Utility_{10}, respectively. Increasing K produces a mild trade-off—$K = 8$ favors diversity, whereas $K = 3$ retains slightly higher quality and utility—and $K = 4$ provides a balanced intermediate setting.

C.8 Held-Out Candidate-Probability Balance

We directly test whether Stage 1 reduces preference concentration on unseen candidate-selection prompts. The held-out suite contains 12 categories whose candidate strings and two prompt templates do not overlap the Stage 1 training data or templates. For each $K \in \{4, 8, 12\}$, we evaluate four candidate lists under four fixed candidate orderings and two templates, yielding 96 paired Base–Stage 1 conditions per backbone. For each condition, we compute the length-normalized sequence log-likelihood of every valid candidate and apply a softmax over the candidate set, following the definition in Section 3.1. This produces the model's induced preference distribution over the enumerated candidates, rather than its unrestricted next-token distribution.

Table 5: Candidate-probability balance on held-out prompts. JSD is measured from the uniform distribution in bits, with the relative reduction shown in parentheses. "Improved" counts conditions whose JSD decreases after Stage 1.

Backbone	JSD ↓ Base → Stage 1	Normalized entropy ↑ Base → Stage 1	Max. candidate prob. ↓ Base → Stage 1	Improved conditions
Qwen2.5-7B	$0.3019 \rightarrow 0.1012$ (-66.5%)	$0.5688 \rightarrow 0.8591$	$0.5744 \rightarrow 0.3414$	96/96
Llama-3.1-8B	$0.2958 \rightarrow 0.0852$ (-71.2%)	$0.5759 \rightarrow 0.8789$	$0.5680 \rightarrow 0.3217$	96/96

Stage 1 makes the held-out candidate distributions substantially more balanced for both backbones. Mean JSD decreases by 66.5% for Qwen and 71.2% for Llama, and all 96 paired conditions improve. The Stage 1-minus-Base JSD changes are -0.2008 for Qwen, with a list-clustered 95% bootstrap confidence interval of $[-0.2362, -0.1637]$, and -0.2106 for Llama, with an interval of $[-0.2554, -0.1664]$. The simultaneous increase in normalized entropy and decrease in maximum candidate probability show that probability mass is redistributed away from the dominant candidate. The improvement also persists at $K = 8$ and $K = 12$, beyond the $K = 4$ candidate count used during training.

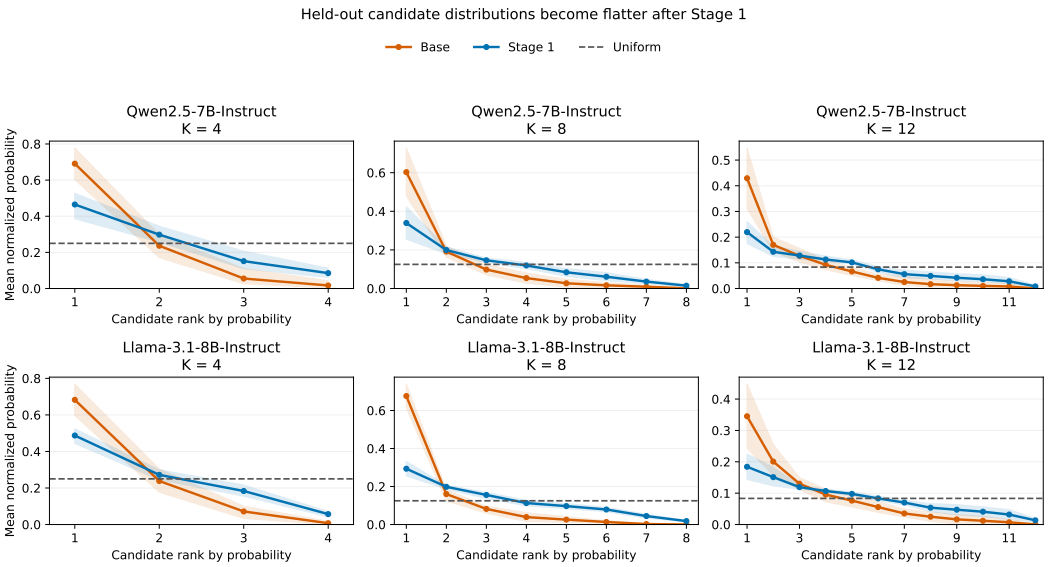


Figure 11: Sorted candidate-probability profiles on held-out prompts. Candidates are ranked by probability within each condition before averaging. Solid curves show the mean induced candidate-choice probabilities, shaded regions show list-clustered 95% bootstrap confidence intervals, and dashed lines indicate the uniform distribution. Stage 1 consistently reduces the dominant-candidate mass and redistributes it toward lower-ranked valid candidates across both backbones and all tested candidate-set sizes.

The profiles move toward, but do not collapse to, the uniform reference. This is expected because the base-aware target retains an $\alpha = 0.6$ contribution from the Base distribution. Together, Table 5 and Figure 11 provide direct evidence that Stage 1 reduces candidate-level preference concentration on prompts, candidates, and templates excluded from calibration training.

D TRAINING ALGORITHMS

Algorithms 1 and 2 summarize the end-to-end optimization procedures corresponding to Sections 3.1 and 3.2. Let $\mathcal{D}_{\text{open}}$ denote the open-ended prompt set used in Stage 2, and let $\mathcal{S}_t$ denote the accepted distillation examples at round t. A bar over model parameters denotes an immutable snapshot within a round: $\bar{\theta}_t$ is refreshed at the beginning of each round.

Algorithm 1 Stage 1: Base-Anchored Diversity Calibration

Require: Base parameters θ_0, calibration set $\mathcal{D}_{\text{cal}}$, mixture coefficient α, token-anchor weight β
Ensure: Calibrated parameters θ_{cal}

Freeze θ_0 and initialize $\theta \leftarrow \theta_0$
for each optimization minibatch $\mathcal{B} \subset \mathcal{D}_{\text{cal}}$ **do**
  for all $(x, \mathcal{A}_x) \in \mathcal{B}$ **do**
    Compute $\mathbf{q}_0 \leftarrow \mathbf{q}_{\theta_0}(x, \mathcal{A}_x)$ and $\mathbf{q}_\theta \leftarrow \mathbf{q}_\theta(x, \mathcal{A}_x)$
    Set $\mathbf{q}^\star_\alpha \leftarrow \alpha \mathbf{q}_0 + (1-\alpha)\mathbf{u}_K$
  end for
  Compute $\widehat{\mathcal{L}}_{\text{cal}}$ on $\mathcal{B}$ using $D_{\text{KL}}(\mathbf{q}^\star_\alpha \,\|\, \mathbf{q}_\theta)$
  Compute $\widehat{\mathcal{L}}_{\text{tok}}$ from the token-level KL over all candidate prefixes
  $\widehat{\mathcal{L}}_{\text{Stage1}} \leftarrow \widehat{\mathcal{L}}_{\text{cal}} + \beta \widehat{\mathcal{L}}_{\text{tok}}$
  Update θ using $\nabla_\theta \widehat{\mathcal{L}}_{\text{Stage1}}$
end for
$\theta_{\text{cal}} \leftarrow \theta$
return θ_{cal}

Algorithm 2 Stage 2: Online Guided Self-Distillation

Require: Calibrated parameters θ_{cal}, prompts $\mathcal{D}_{\text{open}}$, guidance generator G, margins $\{\tau_Z\}_{Z \in \{D,U,Q\}}$, rounds R
Ensure: Final unguided model θ_{R+1}

Set $\theta_1 \leftarrow \theta_{\text{cal}}$
for $t = 1, \ldots, R$ **do**
  Set $\bar{\theta}_t \leftarrow \theta_t$ and freeze it, then initialize $\mathcal{S}_t \leftarrow \emptyset$
  for all $x \in \mathcal{D}_{\text{open}}$ **do**
    Generate query-specific guidance $g \leftarrow G(x)$
    Sample $Y_t^{\text{plain}} \sim \pi_{\bar{\theta}_t}(\cdot \mid x)$ and $Y_t^{\text{guided}} \sim \pi_{\bar{\theta}_t}(\cdot \mid x, g)$
    for all $Z \in \{D, U, Q\}$ **do**
      $\Delta Z(x) \leftarrow Z(Y_t^{\text{guided}}) - Z(Y_t^{\text{plain}})$
    end for
    if $\Delta Z(x) \geq \tau_Z$ for every $Z \in \{D, U, Q\}$ **then**
      Partition Y_t^{guided} into m semantic clusters
      Form $\mathcal{T}(x) = \{y_1, \ldots, y_m\}$ by retaining one response per cluster
      $\mathcal{S}_t \leftarrow \mathcal{S}_t \cup \{(x, g, \mathcal{T}(x))\}$
    end if
  end for
  Initialize the unguided student $\theta \leftarrow \theta_t$
  for each student-optimization minibatch $\mathcal{B} \subseteq \mathcal{S}_t$ **do**
    for all $(x, g, \mathcal{T}(x)) \in \mathcal{B}$, $y_j \in \mathcal{T}(x)$, and $1 \leq k \leq |y_j|$ **do**
      $p^G_{t,k} \leftarrow \pi_{\bar{\theta}_t}(\cdot \mid x, g, y_{j,<k})$
      $p^S_k \leftarrow \pi_\theta(\cdot \mid x, y_{j,<k})$
    end for
    $\widehat{\mathcal{L}}_{\text{Stage2}} \leftarrow |\mathcal{B}|^{-1} \sum_{(x,\cdot) \in \mathcal{B}} \mathcal{L}_{\text{Stage2}}(x)$
    Update θ using $\nabla_\theta \widehat{\mathcal{L}}_{\text{Stage2}}$
  end for
  $\theta_{t+1} \leftarrow \theta$
end for
return θ_{R+1}

The returned model is evaluated and deployed without query-specific guidance, which is used only to construct the verified teacher trajectories during Stage 2 training.

E CASE STUDIES

Stage 1: candidate-selection calibration. A representative Stage 1 training prompt is shown below.

Calibration Prompt

```
Select exactly one item from the following city options:
[Paris, Tokyo, Berlin, Madrid]
Reply with only the selected item.
```

Stage 2: guidance-conditioned teacher trajectories. A representative accepted trajectory from the Llama-3.1-8B Stage 2 run is shown below.

Original User Prompt

```
Primeval TV series.  4-5 Season.  Write a RP fanfiction story.  Write
as Emily Merchant, through an anomaly, ends up in the house of a
fan of the series named Slava.  The guy at first thinks it's an act
of acting, but over time, she takes guardianship of his house and
demands Victorian standarts, as if she were a noble lady.  She really
was.
Also describe Emily Merchant from Primeval TV.
```

Guidance-Generation Prompt

```
Create one compact, question-specific private planning hint for a
response model.
Original user request:  [the request reproduced above]

Silently consider several plausible response directions and the most
likely way an overly specific hint would collapse them.  Then return
a short category and exactly one strongest guidance sentence.  The
guidance must be 8--28 English words, accurate, useful, and broad
enough to permit genuinely different high-quality responses.  It
should open legitimate dimensions or approaches without answering
the request, selecting one method, prescribing a fixed outline,
or listing many concrete examples.  Preserve the user's intent.
Do not mention models, evaluation, diversity, scores, or this
instruction.  Conform exactly to supplied JSON schema and include
no other commentary.
```

Generated Guidance

Guidance: Balance canon-faithful characterization, slow-burn culture clash, Victorian domestic detail, and clear consent while leaving room for an original plot.

Unguided Student — Representative Excerpt (1 Semantic Class)

"As I emerged from the anomalous vortex, my head spinning and my stomach churning, I found myself in the most unlikely of places - a small, quaint house, surrounded by the familiar sights and sounds of modern life. My eyes adjusted slowly, and I took in my surroundings. I was standing in a cozy living room, surrounded by bookshelves, a fireplace, and a large TV on the wall."

Guidance-Conditioned Teacher — One Excerpt per Semantic Class

Mode 1. "As I emerged from the anomalous vortex, my senses reeled, disoriented and confused. The last thing I remembered was navigating the labyrinthine tunnels beneath the Natural History Museum, and now - I stood in a peculiar living room, surrounded by unfamiliar furnishings and an assortment of antique trinkets."

Mode 2. "As Emily Merchant entered the small cottage, her eyes widened in dismay at the state of disarray before her. Dirty dishes scattered the countertop, dust accumulated on every surface, and a faint smell of stale air wafted from the kitchen."

Mode 3. "I stood before the imposing oak door, my gaze drifting over the intricately etched lock. The soft hum of a television drifted through the crack beneath the door, punctuated by the occasional chuckle. Intriguing. The anomaly had deposited me in a strange, unassuming neighborhood, and I spotted this house as my landing destination."

Mode 4. "Emily Merchant stood frozen, her eyes scanning the strange new surroundings with an uneasy sense of unease. One moment, she was trekking through a dense forest in 2011, searching for evidence of prehistoric creatures with the team from the Anomaly Research Centre; the next, she found herself in a modest but well-maintained Victorian-era townhouse."

Table 6: Metrics for the accepted Stage 2 trajectory.

Condition	$D(Y)$	$U(Y)$	$Q(Y)$
Unguided student	1.00	1.92	8.00
Guided teacher	4.00	5.82	9.50
Improvement	$+3.00$	$+3.89$	$+1.50$

In this accepted case, the guidance-conditioned responses cover four semantic classes while also improving utility and quality.