ROLLING FAIR DICE: GROUP CREDIT ASSIGNMENT FOR LLMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Large language models are increasingly asked to sample from specified distributions, yet statistical tests reveal discrepancies between their outputs and the targets. We study this task through reinforcement learning with a reward computed on a group's pooled samples. Broadcasting this reward makes GRPO's centered advantages exactly zero, removing the distributional training signal. We introduce Jackknife Credit Assignment (JCA), which credits each sample by the decrease in group score when it is removed, and W_1 diff, a differentiable surrogate based on empirical CDFs reweighted by likelihood. For additive rewards, centered JCA credit recovers the GRPO advantage numerator. On 15 Bad Dice families, JCA improves Qwen3.5 4B from 0.585 to 0.858 TV score, and W_1 diff reaches 0.926. JCA on 9B reaches 0.929, compared with 0.945 for the strongest frontier reference under matched sampling requests. The gains extend to distribution families outside the training set within list generation, and independent requests together with natural format probes set the empirical scope of transfer, identifying deployment format as a first-class design dimension for distributional training.

1 INTRODUCTION

Large language models roll dice every day. A drafting model picks an answer position, a synthesis pipeline assigns demographic attributes, and an agent samples its next move. These choices often come with a target distribution, and downstream systems depend on the model respecting it. Yet a large audit (Zhao et al., 2026) documents systematic deviations from standard distributions, with related distortions in answer positions and demographic constraints. An external random number generator can enforce a numerical distribution, but many applications ask the model to make the choice while generating a response. This gap motivates a basic question: *Can we train LLMs to sample more faithfully from a target distribution?*

Interventions at inference have protocol dependent effects. In our Qwen3.5 4B experiments, temperature tuning produces little improvement, while thinking raises average fidelity but yields zero family test passes. An adapted String Seed of Thought reproduction (Misaki & Akiba, 2026) also retains substantial discrepancies on three discrete families. Several frontier models achieve much higher fidelity with reasoning enabled, but their behavior depends on the model and request protocol. Recent RL studies develop extended reasoning through exploration and reward design (Hou et al., 2025; Yang et al., 2025). We ask whether training can improve a small model's sampling without deliberation at inference. Known target distributions offer a controlled setting for this question: they let us evaluate an entire collection of outputs with a cheap, verifiable score.

That training opportunity hides a structural trap. Distributional fidelity belongs to the pooled samples, so a natural reward scores an entire sampling group. The reward does not supply the separate rollout scores that GRPO normally centers against one another. If every rollout instead receives the same broadcast score, all centered advantages are identically zero (Proposition 1). The distributional reward then contributes no policy gradient, regardless of how accurate the score is. Our broadcast baseline records this zero contrast throughout $1{,}800$ training steps; its additional count penalty remains a separate source of updates.

JCA (Jackknife Credit Assignment) restores contrast with a simple question: *does the group get better or worse when you are removed?* It evaluates the score with and without each sample, centers the resulting credits, and averages them within each rollout. This requires only scoring access to

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.

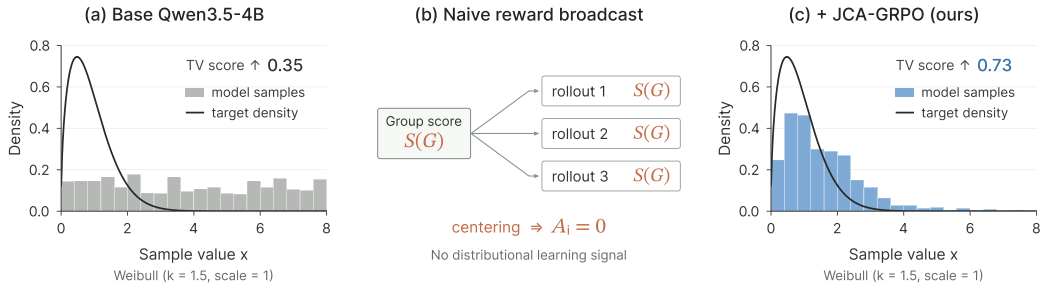


Figure 1: **Faithful sampling is learnable.** (a) Base Qwen3.5 4B draws from Weibull(k=1.5) with clear distortion (TV score 0.35). (b) A group reward broadcast to every rollout makes its centered advantages exactly zero, eliminating the distributional training signal. (c) After JCA training (seed 0), the same model better matches the target (TV score 0.73).

the original pool and its subsets formed by removing one sample. When the target CDF is available, **W_1 diff** provides a complementary route: it reweights rollout CDFs through their log probabilities and optimizes a differentiable surrogate for distribution matching. For additive rewards, centered JCA credit reduces to the GRPO numerator, linking the construction to familiar reward contrasts without equating the complete optimization algorithms.

Our contributions are threefold. **(i)** We identify the cancellation of centered broadcast credit and develop JCA and W_1 diff for training with pooled distribution objectives. **(ii)** On 15 families, JCA raises the 4B TV score from 0.585 to 0.858; W_1 diff reaches 0.926. On 9B, JCA reaches 0.929 without reasoning at inference, compared with 0.945 for the strongest frontier reference under matched requests. **(iii)** We show improved fidelity on unseen families within list generation and chart the empirical scope of transfer across request formats, from lists to independent requests, MCQ, and persona probes. We will open source our training and evaluation code upon acceptance.

2 Related Work

Sampling fidelity. Bad Dice (Zhao et al., 2026) audits 11 models across 15 distributions and downstream constraints; Coronado-Blázquez (2025) also studies LLMs as random number generators. Repeated paraphrasing can converge to cycles despite increased decoding randomness (Wang et al., 2025). Coverage objectives (Verine et al., 2025) and Fourier output heads for continuous structure (Gillman et al., 2025) offer training and architectural routes to distribution learning. SSoT (Misaki & Akiba, 2026) extracts randomness from generated strings; Appendix D describes our adapted reproduction.

Credit assignment. GRPO (Shao et al., 2024) and RLOO (Ahmadian et al., 2024) use contrasts between rollout rewards; Dr. GRPO, DAPO, and GSPO refine related optimization choices (Liu et al., 2025; Yu et al., 2025; Zheng et al., 2025). JCA instead constructs credits from a pooled reward before optimization. Its counterfactual connects to difference rewards (Wolpert & Tumer, 1999) and baselines for multiple agents (Foerster et al., 2018). SPO estimates advantages for response segments (Guo et al., 2025), while Shapley Coop redistributes outcomes among cooperating agents (Hua et al., 2025). ShapE GRPO allocates set utility among candidates (Ai et al., 2026), while OSPO assigns credit to response segments using Owen Shapley attributions (Nath et al., 2026). Our rule fixes attribution to the observed pool, using m extra score queries; Shapley allocation averages over coalitions.

Distribution objectives. FlowRL shapes the distribution of reasoning paths according to their rewards (Zhu et al., 2025); LAD matches distributions induced by policies and advantages (Li & Li, 2026). DivPO encourages diverse responses of high quality (Lanchantin et al., 2025), and explicit token diversity objectives promote exploration in reasoning (Yao et al., 2025). General utility RL studies concave functions of occupancy measures (Barakat et al., 2025); Flow Density Control optimizes distributional utilities in flow and diffusion models (De Santi et al., 2025). We study fidelity to a specified output distribution using credit from finite sample pools.

3 SETUP: FAITHFUL SAMPLING AS GROUP RL

Given a description in natural language of a target distribution P^*, the model is asked to emit n samples as a JSON list. The 15 Bad Dice families vary in symmetry, skew, and tail behavior (Zhao et al., 2026). At each step, N rollouts y_j produce parsed lists X_j of lengths n_j; their pooled multiset is $G = \biguplus_j X_j$, with $m = \sum_j n_j$. The requested sizes are $N = 16$ and $n = 24$ unless stated, but generated lengths can differ. The group reward is

$$S(G) = 1 - \mathrm{TV}\left(\hat{P}_G, P^*\right). \tag{1}$$

For discrete families, TV compares category frequencies; continuous families use 20 fixed bins over a truncated support (Appendix C). We report this TV score and goodness of fit tests at $\alpha = 0.01$: χ^2 for discrete families and KS against reference samples for continuous ones.

GRPO normally centers rewards for each rollout, with numerator $r_j - \bar{r}$ and optional scaling by the standard deviation. The pooled score in Eq. (1) is not an additive sum of independent sample rewards. Broadcasting it as $r_j = S(G)$ is a direct extension, but removes all reward contrast within the group.

4 THE ZERO GRADIENT DEADLOCK

Proposition 1 (Broadcast eliminates centered reward credit)**.** *If every rollout receives $r_j = S(G)$, then $A_j = r_j - \bar{r} = 0$ for every j, and $\sum_j A_j \nabla_\theta \log \pi_\theta(y_j) = 0$ for any score S and group of $N \geq 2$ rollouts.*

Proof. $\bar{r} = N^{-1} \sum_j S(G) = S(G)$. □

Scaling, clipping, or averaging zero credits leaves the distributional gradient at zero; auxiliary penalties can still update the policy. The logged standard deviation of broadcast credit is zero across three seeds and 1,800 steps (Fig. 4). The count penalty is applied afterward. A credit assignment ablation with a softmax policy isolates this cancellation without auxiliary penalties (Appendix F).

Removing centering. Broadcasting $(S(G) - \mu_t)/\max(\sigma_t, 10^{-6})$ with running statistics across groups permits nonzero joint policy gradient updates (Appendix C). Our best seed reaches 0.754 evaluation TV; valid coverage is 11, 15, and 15 families across seeds and 6 and 12 for clipped variants (Table 2). This contrasts a shared group advantage with JCA's sample contributions.

5 TWO WAYS OUT OF THE DEADLOCK

5.1 JCA: JACKKNIFE CREDIT ASSIGNMENT

For each sample x_i in the pool, compute its removal contribution and centered credit:

$$d_i = S(G) - S(G \setminus \{x_i\}), \qquad A_i = d_i - \frac{1}{m} \sum_{k=1}^{m} d_k. \tag{2}$$

The multiset difference removes one occurrence. A positive d_i means that retaining this sample improves the observed group score; a negative value means removal improves it. The rollout advantage is the average of its samples' centered credits, $a_j = n_j^{-1} \sum_{i:x_i \in X_j} A_i$, with membership tracked by sample index.

This jackknife calculation (Tukey, 1958; Efron, 1982) uses m additional calls to a scorer defined on removal pools, with no reward derivatives. Histogram distances and suitable test statistics provide such scorers. It fixes attribution to the full observed pool; Shapley allocation averages contributions over coalitions (Shapley, 1953). For histogram TV, raw credits satisfy $|d_i| \leq 1/m$; Appendix B gives the bound and discusses centering. These finite sample credits define a surrogate update. Our formal results cover credit bounds and the additive reward correspondence; experiments assess optimization behavior.

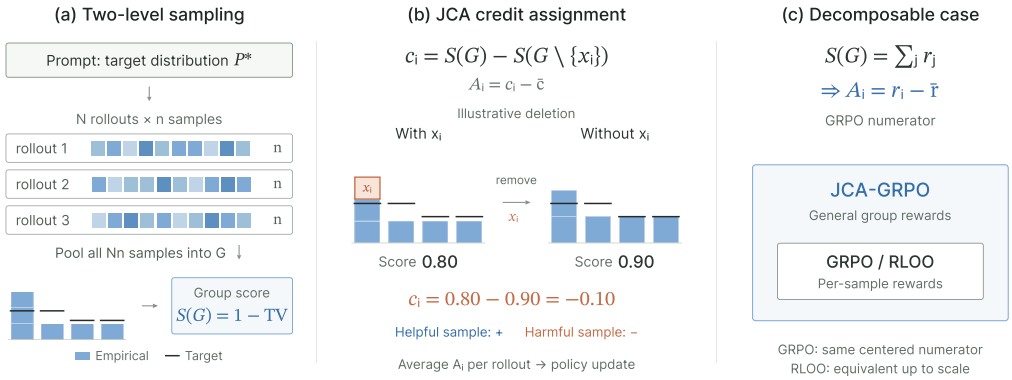


Figure 2: **JCA GRPO.** Left: one prompt yields N rollouts whose pooled samples score the group. Middle: each sample is credited by how its removal changes that score. Right: with additive rewards for each sample, centered credit recovers the GRPO numerator; RLOO is equivalent up to a factor determined by group size.

5.2 W_1 DIFF: A DIFFERENTIABLE SURROGATE FOR DISTRIBUTION MATCHING

When the target CDF is available, we can differentiate a reweighted distributional discrepancy. Let $\ell_j(\theta)$ be the summed completion log probability, ℓ_j^{old} its value at sampling time, T_j the token count, and F_j the rollout's empirical CDF. Define

$$w_j(\theta) = \exp\left(\frac{\ell_j(\theta) - \ell_j^{\text{old}}}{\tau T_j}\right), \qquad \hat{F}(y;\theta) = \frac{\sum_j w_j(\theta) F_j(y)}{\sum_j w_j(\theta)}. \tag{3}$$

On grid points y_k with quadrature weights δ_k, the surrogate is

$$\mathcal{L}(\theta) = \frac{1}{\text{span}} \sum_k \left| \hat{F}(y_k;\theta) - F^*(y_k) \right| \delta_k. \tag{4}$$

Gradients flow through the likelihood weights, with sampled values held fixed. Length and temperature normalization define a local surrogate; at the sampling policy, each rollout has unit weight. The resulting CDF equals the sample pool CDF when list lengths agree. JCA requires score queries, whereas W_1 diff additionally exposes the target CDF to the loss.

5.3 ADDITIVE REWARDS RECOVER FAMILIAR CONTRASTS

Proposition 2 (Additive reward case)**.** *If $S(G) = \sum_i r_i$ for sample rewards unaffected by other samples, then $d_i = r_i$ and centered JCA credit is $A_i = r_i - \bar{r}$.*

Proof. Removing sample i removes exactly its term from the sum. □

This is the GRPO numerator at the same sample granularity. RLOO credit is $m/(m-1)$ times this contrast; scaling by the standard deviation, clipping, and rollout aggregation are separate choices. Here leave one out acts on the reward functional, while RLOO applies it to a baseline over already available rewards.

5.4 TRAINING DETAILS

We adapt models with LoRA rank 32 for 600 steps, drawing fresh samples from the current policy at each step. We use Adam at 10^{-5} and sampling temperature 1.0. Runs that use advantages subtract $\lambda \min(|n_j - n|/n, 1)$ from rollout credit, with $\lambda = 0.3$, and use a policy gradient loss with importance sampling, without normalization by the within-group reward standard deviation. W_1 diff uses a custom loss without this count penalty. We filter short parsed lists and skip groups with too few usable rollouts; Appendix C gives the thresholds and settings.

Table 1: **Distributional fidelity on 15 families.** Trained rows use aligned requests; frontier defaults use 3×100 requests. Pass counts use $\alpha = 0.01$ and are averaged across seeds. [†]reference with target bin probabilities; [§]thinking disabled. Full TV requires valid samples on all 15 families; families with no valid samples contribute zero passes.

Method / Model	TV score ↑	Pass / 15
Interventions at inference (Qwen3.5 4B)		
base	0.585	2
temperature sweep (best of 0.7 to 1.5)	0.574	0
thinking mode (CoT)	0.718	0
Trained on Qwen3.5 4B (600 steps; RL: 3 seeds)		
naive GRPO (broadcast)	0.637 ± 0.029	1.3 ± 0.5
JCA GRPO (score queries)	0.858 ± 0.013	3.3 ± 1.2
W_1 diff (target CDF)	0.926 ± 0.004	10.7 ± 1.7
oracle bin[†]	0.929 ± 0.005	8.7 ± 0.9
Larger base model		
base Qwen3.5 9B (thinking)	0.833	1
JCA GRPO on 9B (3 seeds)	0.929 ± 0.006	8.7 ± 1.7
Frontier, default mode (thinking on where applicable)		
Qwen3.8 Max (0902)	**0.955**	**15**
Doubao Seed Evolving	0.953	15
DeepSeek V4.1 Flash	0.953	15
GLM 5.3	0.947	15
Doubao Seed 2.1 Pro	0.939	15
Doubao Seed 2.1 Turbo	0.939	15
Claude Fable 5.1	0.929	13
Kimi K3 (256K)	0.928	14
GLM 5.3 Flash	0.926	12
Claude Fable 5	0.923	13
GPT-5.6 Sol	0.906	12
Kimi K2.8 Preview	0.889	12
GPT-6 Astra	0.884	11
Doubao Seed 2.0 Lite	0.848	7
Claude Opus 5	0.805	4
Doubao Seed 2.0 Mini	0.792	6
Frontier, thinking disabled[§]		
DeepSeek V4 Pro	0.852	7
MiniMax M3	0.835	6
Matched requests (42×24, about 1,008 requested samples)		
Doubao Seed Evolving (think)	0.945	12
DeepSeek V4 Pro (thinking off)	0.857	3

6 Experiments

6.1 Protocols and baselines

We train Qwen3.5 4B and 9B (Qwen Team, 2026) with three seeds for the main RL methods. The aligned evaluation requests 42 lists of 24 samples per family (about 1,008 requested samples); the evaluation with longer lists requests 5 lists of 200. Both pool parsed outputs for scoring. Temperature, thinking, and an adapted SSoT reproduction provide controls applied at inference. Frontier references use 3 lists of 100 by default; two endpoints are also evaluated with aligned requests. Parsed output counts can differ from these request budgets. Tables report mean ± population SD across seeds where available, including pass counts; replacement credit has two seeds. Appendices C to E detail request protocols and coverage.

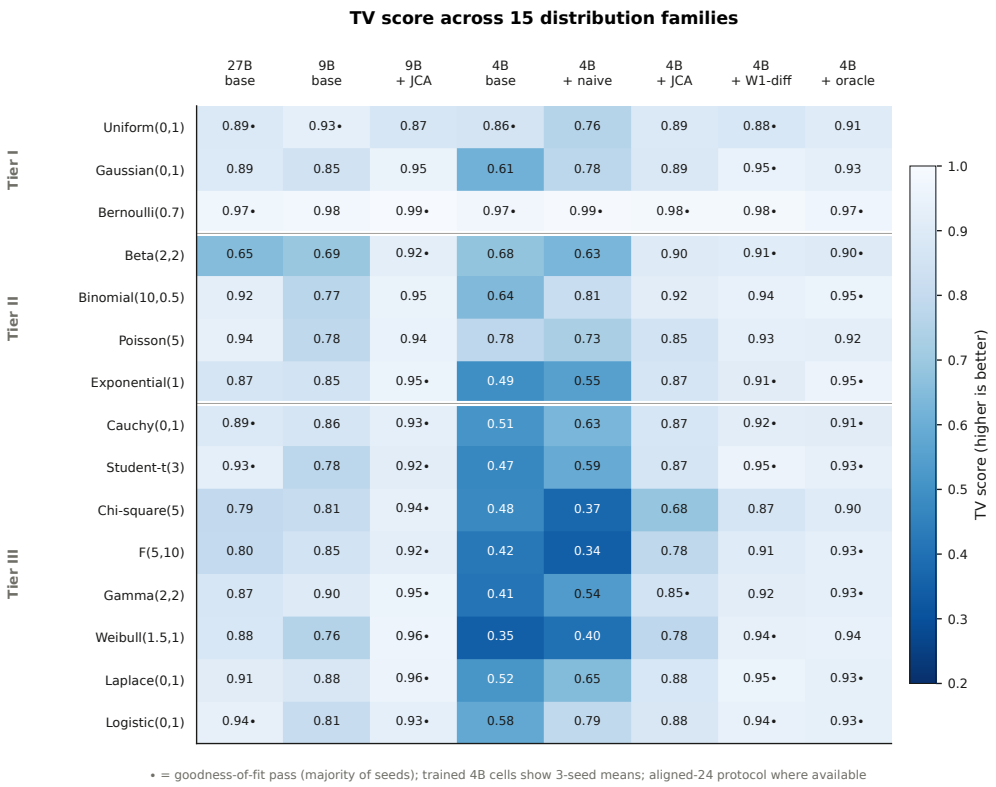


Figure 3: **TV scores for individual families.** Trained 4B columns average three seeds; their bullets denote a majority of seeds passing. The trained 9B column uses seed 0, while Table 1 averages three seeds. Base 9B and 27B use thinking mode, and 27B is Qwen3.6; model family and inference mode vary alongside size.

6.2 Main results

Table 1 shows a substantial improvement from training: JCA gains 0.273 TV over the 4B base, and W_1 diff scores approximately 0.004 below the reference with target bin probabilities. Under the goodness of fit tests, JCA and W_1 diff average 3.3 and 10.7 passing families out of 15, respectively.

The 9B JCA model reaches 0.929 ± 0.006, versus 0.945 for Doubao Seed Evolving and 0.857 for DeepSeek V4 Pro with thinking disabled under matched requests. This comparison concerns fidelity under matched requests with thinking disabled in the trained model. Latency and monetary cost require separate measurement. The default frontier panel is contextual because sample budgets differ.

Figure 3 shows that the gains vary substantially across families. For 4B, the mean JCA score reaches 0.78 on Weibull and 0.85 on Gamma, compared with base scores of 0.35 and 0.41. Bernoulli begins at 0.97 and reaches 0.98, leaving little scope for a comparable gain. Per-family gains vary: on $\chi^2(5)$, JCA improves from 0.48 to 0.68, while W_1 diff reaches 0.87. Thus, the aggregate improvement includes large corrections on initially distorted targets, alongside smaller changes where the base already matches well.

Using means over three training seeds, 4B JCA exceeds the base score on all 15 families and the broadcast score on 14. Bernoulli is the exception to the latter comparison, with both trained variants scoring above 0.98. The aggregate improvement is therefore accompanied by broad coverage across targets. These familywise comparisons summarize descriptive means across three training seeds.

The TV score and the pass count capture different aspects of fidelity. TV quantifies discrepancies between empirical and target frequencies, while test passes summarize decisions at a chosen

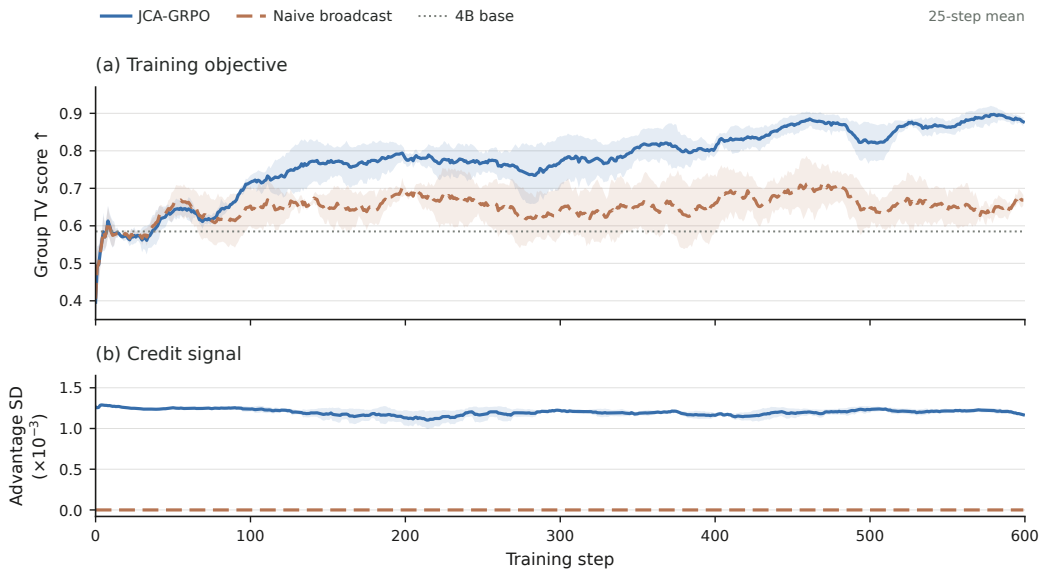


Figure 4: **Training dynamics (3 seeds, ±1 SD).** Top: training-pool TV; the dotted base reference uses aligned evaluation. Bottom: sample credit SD before the count penalty; broadcast distributional credit is zero.

Table 2: **Credit ablation on Qwen3.5 4B.** Aligned TV and pass counts are mean ± SD over three seeds, except replacement (two). Credit SD is measured before count penalties. *TV for the global baseline is conditional on valid outputs: its seeds cover 11, 15, and 15 families.

Scheme	eval TV	pass / 15	credit SD
broadcast	$0.637 \pm .029$	1.3 ± 0.5	0
global baseline*	$0.694 \pm .046$	3.7 ± 1.7	0 (within group)
replacement credit	$0.868 \pm .008$	4.5 ± 0.5	$\sim 10^{-3}$
JCA (removal)	$0.858 \pm .013$	3.3 ± 1.2	1.2×10^{-3}
W_1 diff	$0.926 \pm .004$	10.7 ± 1.7	n/a
oracle bin	$0.929 \pm .005$	8.7 ± 0.9	2.5×10^{-2}

significance level. For continuous families, TV uses 20 fixed bins, whereas KS compares empirical CDFs and can detect differences obscured by binning. A model can therefore reduce its average histogram discrepancy while retaining deviations detectable by the tests. JCA's gains concern fidelity under the reported protocol. A test pass records a decision at the chosen threshold; exact agreement and sample independence require separate analysis.

6.3 Ablating the credit signal

Both removal and replacement credits improve on broadcast training (Table 2). The replacement control substitutes a draw from the remaining pool for each sample, requiring no target information beyond score queries. Both constructions attain similar descriptive mean TV, supporting the usefulness of counterfactual contrast; comparisons between their means require a separate statistical analysis. The global baseline supplies a nonzero advantage shared by all samples and yields valid outputs on 11, 15, and 15 families across seeds; clipped variants cover 6 and 12 families in the two tested seeds. W_1 diff and oracle bin attain higher TV while accessing target structure directly.

6.4 Sensitivity and Output Counts

Table 3 characterizes variation across the tested list lengths and penalty settings; a broader grid would assess sensitivity to group size more fully. Without the count penalty, TV is high but the two runs

Table 3: **Sensitivity measured on training pools (steps 250 to 299).** The $(N, n) = (16, 24)$ entry averages three seeds; other grid settings use one. Penalty rows give mean $\pm$ SD and seed count. Pool sizes differ, so these are training diagnostics rather than matched evaluation scores.

$N\backslash n$	12	24	48	penalty	TV	seeds
8	n/a	0.808	n/a	$\lambda = 0$	$0.927 \pm .007$	2
16	0.882	0.756	0.636	$\lambda = 0.3$	$0.756 \pm .066$	3
32	n/a	0.799	n/a	$\lambda = 1.0$	$0.824 \pm .020$	2

Table 4: **Generalization to three unseen families.** Every row is evaluated on the same three families with aligned requests. Mean $\pm$ SD is computed across training seeds.

	Bernoulli(0.7)	Exponential(1)	Cauchy(0,1)	mean
base (no training)	0.973	0.491	0.513	0.659
JCA, all 15 (3 seeds)	0.982	0.865	0.869	$0.906 \pm .010$
JCA, 12 only (2 seeds)	0.980	0.889	0.930	$\mathbf{0.933 \pm .004}$

emit about 20.2 and 30.4 items per requested rollout in this window, versus about 24 at the default penalty. Thus fidelity and count compliance must be assessed separately. For W_1 diff, runs with one seed at $\tau = 0.5$ and 2.0 yield aligned TV of 0.913 and 0.927; requests for long lists still produce substantial count excesses. Appendix G provides the diagnostic plots and distinguishes alternative scales of the training rewards.

6.5 Generalization to unseen families

We exclude Bernoulli, Exponential, and Cauchy from training, leaving 12 families. On these three unseen targets, JCA reaches 0.933 ± 0.004, compared with 0.659 for the base model and 0.906 ± 0.010 after training on all families (Table 4). This supports transfer across distribution families within the same list format. The 0.933 score averages only the excluded subset; the corresponding mean across all 15 families is 0.917 ± 0.003.

6.6 Transfer across Output Formats

Table 5: **Probes in natural formats.** MCQ uses a uniform reference; persona prompts specify target probabilities. Counts include successfully parsed answers.

Probe	base	JCA trained
MCQ position (vs. uniform)	[84, 115, 140, 32]	[74, 122, 133, 39]
TV score	0.81	0.81
occupation 70/30	97%/3%	97%/3%
city (uniform) TV	0.63	0.68

MCQ position statistics and persona proportions sit near base levels after list training (Table 5). In the sweep over list lengths, TV decreases from 0.872 at $n = 24$ to 0.810 at $n = 200$, remaining above the corresponding base model (Table 6). Independent requests improve from 0.409 to 0.584; enabling thinking after training lowers the paired score on long lists from 0.824 to 0.708. Transfer is strongest within list generation; the scores under independent requests and natural formats define the measured scope of the trained policy.

7 Discussion

Group scores need attributable credit. Cancellation of broadcast credit applies to any score centered over the same group to which it is assigned. JCA provides sample credit when the scorer remains meaningful after deleting a sample. Extending this construction to diversity, fairness, or simulator rewards requires validating that scoring condition, its query cost, and performance in each application.

Table 6: **Protocol analysis (TV; one trained seed).** The sweep over list lengths uses a separate evaluation draw from Table 1. The thinking comparison uses a paired $n = 200$ evaluation.

list length n	24	50	100	200
base	0.588	0.597	0.559	0.570
JCA trained	0.872	0.845	0.821	0.810
Independent requests: base 0.409, trained 0.584.				
Trained, $n = 200$: thinking on 0.708, off 0.824 (paired draw).				

The replacement baseline further suggests that useful group contrasts need not depend on a unique counterfactual construction.

Choosing the reward interface. JCA obtains sample credit through a scalar scoring interface, with one additional call per parsed sample and no need for target CDF access or scorer derivatives. W_1 diff uses the target CDF to construct a differentiable surrogate through rollout likelihood weights. Its higher TV in these experiments accompanies differences in both target information and objective. The practical choice depends on scorer cost and available target information.

Sampling is conditioned on format. List training improves pooled fidelity, including on unseen families, and independent requests improve over their base reference (0.409 to 0.584) while scoring below list format fidelity. One possible explanation is that list training favors coverage across positions without enforcing the target marginal at every position. The existing samples exhibit position preferences (Appendix C); we state this as a hypothesis rather than a settled mechanism, since independent draws from a shared conditional can remain diverse. Both protocols pool all valid parsed outputs, so their measured fidelity also reflects count compliance.

8 Conclusion

Training can make small language models better dice players within numerical lists. We identify the zero centered credit produced by broadcasting a pooled reward, and study two remedies: jackknife removal credit and a differentiable surrogate based on CDF reweighting. Both improve the measured fidelity over broadcast training. The gains extend to unseen distribution families within list generation; fidelity under independent requests and structured tasks tracks the training format, which motivates training in the intended deployment format. These results make group credit assignment a concrete training opportunity with an empirically defined scope of transfer.

9 Limitations

Our experiments cover 15 distribution families and LoRA adaptation of 4B and 9B models for up to 600 steps; the largest gains concern list generation, and the transfer patterns of Section 6.6 define the scope for study at larger scales. Binned TV evaluates marginal histogram fidelity; joint and sequential independence of samples is a complementary measurement axis.

Comparisons differ in information access (score queries, target bins, CDFs) and request budgets: matched frontier requests cover two endpoints, the SSoT reproduction is adapted for smaller models, and main comparisons use three seeds with ablations at one or two, as recorded per setting. JCA costs one scoring call per parsed sample, so expensive simulators or human judgment change its economics, and stochastic feedback requires scorer noise small relative to removal credits, which are bounded by $1/m$ for histogram TV over a pool of m samples (Appendix B.1).

AI USE STATEMENT

We used generative AI tools to assist with manuscript drafting, language polishing, and figure preparation. The authors take responsibility for the final content, including the scientific claims and figures.

ETHICS STATEMENT

This study uses synthetic sampling tasks and outputs generated by language models. It does not involve human participants or the collection of personal data. To the best of our knowledge, the experiments reported here raise no specific ethical concerns.

REPRODUCIBILITY STATEMENT

Training settings, evaluation protocols, parsing rules, and metric definitions are described in Section 5.4 and Appendix C. Proofs and estimator analyses are provided in Appendices A and B. We will open source our training and evaluation code upon acceptance.

REFERENCES

Arash Ahmadian, Chris Cremer, Matthias Gallé, Marzieh Fadaee, Julia Kreutzer, Olivier Pietquin, Ahmet Üstün, and Sara Hooker. Back to basics: Revisiting REINFORCE style optimization for learning from human feedback in LLMs. *arXiv preprint arXiv:2402.14740*, 2024.

Rui Ai, Yu Pan, David Simchi-Levi, and Chonghuan Wang. ShapE-GRPO: Shapley-enhanced reward allocation for multi-candidate LLM training. *arXiv preprint arXiv:2603.29871*, 2026.

Anas Barakat, Souradip Chakraborty, Peihong Yu, Pratap Tokekar, and Amrit Singh Bedi. On the global optimality of policy gradient methods in general utility reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 38, pp. 79601–79633. Curran Associates, Inc., 2025. doi: 10.52202/085713-2661.

Javier Coronado-Blázquez. Deterministic or probabilistic? the psychology of LLMs as random number generators. *arXiv preprint arXiv:2502.19965*, 2025.

Riccardo De Santi, Marin Vlastelica, Ya-Ping Hsieh, Zebang Shen, Niao He, and Andreas Krause. Flow density control: Generative optimization beyond entropy-regularized fine-tuning. In *Advances in Neural Information Processing Systems*, volume 38, pp. 11056–11088. Curran Associates, Inc., 2025. doi: 10.52202/085713-0371.

Bradley Efron. *The Jackknife, the Bootstrap and Other Resampling Plans*. SIAM, 1982.

Jakob Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson. Counterfactual multi-agent policy gradients. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018.

Nate Gillman, Daksh Aggarwal, Michael Freeman, and Chen Sun. Fourier head: Helping large language models learn complex probability distributions. In *International Conference on Learning Representations*, pp. 85108–85142, 2025.

Yiran Guo, Lijie Xu, Jie Liu, Dan Ye, and Shuang Qiu. Segment policy optimization: Effective segment-level credit assignment in RL for large language models. In *Advances in Neural Information Processing Systems*, volume 38, pp. 114399–114431. Curran Associates, Inc., 2025. doi: 10.52202/085713-3815.

Zhenyu Hou, Xin Lv, Rui Lu, Jiajie Zhang, Yujiang Li, Zijun Yao, Juanzi Li, Jie Tang, and Yuxiao Dong. T1: Advancing language model reasoning through reinforcement learning and inference scaling. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 23976–24003. PMLR, 2025. URL https://proceedings.mlr.press/v267/hou25e.html.

Yun Hua, Haosheng Chen, Shiqin Wang, Wenhao Li, Xiangfeng Wang, and Jun Luo. Shapley-Coop: Credit assignment for emergent cooperation in self-interested LLM agents. In *Advances in Neural Information Processing Systems*, volume 38, pp. 88675–88702. Curran Associates, Inc., 2025. doi: 10.52202/085713-2965.

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

Wendi Li and Sharon Li. LAD: Learning advantage distribution for reasoning. *arXiv preprint arXiv:2602.20132*, 2026.

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. *arXiv preprint arXiv:2503.20783*, 2025.

Kou Misaki and Takuya Akiba. String seed of thought: Prompting LLMs for distribution-faithful and diverse generation. In *International Conference on Learning Representations (ICLR)*, 2026. arXiv:2510.21150.

Abhijnan Nath, Alireza Bagheri Garakani, Tianchen Zhou, Fan Yang, Yan Gao, and Nikhil Krishnaswamy. Owen-shapley policy optimization: A principled RL algorithm for generative search LLMs. *arXiv preprint arXiv:2601.08403*, 2026.

Qwen Team. Qwen3.5 model cards. Hugging Face, 2026. URL `https://huggingface.co/Qwen/Qwen3.5-4B`. 4B and 9B model cards, accessed September 2026.

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. *arXiv preprint arXiv:2402.03300*, 2024.

Lloyd S Shapley. A value for n-person games. In Harold W Kuhn and Albert W Tucker (eds.), *Contributions to the Theory of Games*, volume 2, pp. 307–317. Princeton University Press, 1953.

John W Tukey. Bias and confidence in not-quite large samples. *Annals of Mathematical Statistics*, 29 (2):614, 1958. Abstract.

Alexandre Verine, Florian Le Bronnec, Kunhao Zheng, Alexandre Allauzen, Yann Chevaleyre, and Benjamin Negrevergne. Improving diversity in language models: When temperature fails, change the loss. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 61266–61300. PMLR, 2025. URL `https://proceedings.mlr.press/v267/verine25a.html`.

Zhilin Wang, Yafu Li, Jianhao Yan, Yu Cheng, and Yue Zhang. Unveiling attractor cycles in large language models: A dynamical systems view of successive paraphrasing. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12740–12755. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.acl-long.624. URL `https://aclanthology.org/2025.acl-long.624/`.

David H Wolpert and Kagan Tumer. An introduction to collective intelligence. *arXiv preprint arXiv:cs/9908014*, 1999.

Shiming Yang, Yuxuan Tong, Xinyao Niu, Graham Neubig, and Xiang Yue. Demystifying long chain-of-thought reasoning. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 71177–71209. PMLR, 2025. URL `https://proceedings.mlr.press/v267/yang25ae.html`.

Jian Yao, Ran Cheng, Xingyu Wu, Jibin Wu, and KC Tan. Diversity-aware policy optimization for large language model reasoning. In *Advances in Neural Information Processing Systems*, volume 38, pp. 94801–94826. Curran Associates, Inc., 2025. doi: 10.52202/085713-3169.

Qiying Yu, Zheng Zhang, Ruofei Zhu, et al. DAPO: An open-source LLM reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*, 2025.

Minda Zhao, Yilun Du, and Mengyu Wang. Large language models are bad dice players: LLMs struggle to generate random numbers from statistical distributions. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 22942–22959. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.acl-long.1051.

Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, Jingren Zhou, and Junyang Lin. Group sequence policy optimization. *arXiv preprint arXiv:2507.18071*, 2025.

Xuekai Zhu, Daixuan Cheng, Dinghuai Zhang, Hengli Li, Kaiyan Zhang, Che Jiang, Youbang Sun, Ermo Hua, Yuxin Zuo, Xingtai Lv, Qizheng Zhang, Lin Chen, Fanghao Shao, Bo Xue, Yunchong Song, Zhenjie Yang, Ganqu Cui, Ning Ding, Jianfeng Gao, Xiaodong Liu, Bowen Zhou, Hongyuan Mei, and Zhouhan Lin. FlowRL: Matching reward distributions for LLM reasoning. *arXiv preprint arXiv:2509.15207*, 2025.

A Proofs

Proposition 1. With $r_i = S(G)$ for all i, $\bar{r} = \frac{1}{N}\sum_{i=1}^{N} S(G) = S(G)$, hence $A_i = r_i - \bar{r} = 0$ identically; the variant scaled by the standard deviation divides by $\mathrm{std}(r) = 0$ and is undefined (practical implementations either skip the update or divide by a floor, in which case the numerator is still exactly zero). □

Proposition 2. If $S(G) = \sum_{j\in G} r_j$, then $S(G \setminus \{i\}) = \sum_{j\neq i} r_j = S(G) - r_i$, so $S(G) - S(G \setminus \{i\}) = r_i$. Centering within the group gives $r_i - \bar{r}$ with $\bar{r} = \frac{1}{|G|}\sum_j r_j$, the GRPO numerator. □

B Estimator analysis

B.1 Statistical properties of JCA

A bound for histogram TV. Let p and p_{-i} be the normalized histograms before and after removal from a pool of $m > 1$ samples, and let c_k count samples in x_i's bin. The frequency of every occupied bin can change because the denominator changes. The reverse triangle inequality gives

$$|d_i| \le \mathrm{TV}(p, p_{-i}) = \frac{1}{2}\left[\frac{m - c_k}{m(m-1)} + \sum_{b\neq k}\frac{c_b}{m(m-1)}\right] = \frac{m - c_k}{m(m-1)} \le \frac{1}{m}. \tag{5}$$

Consequently $|A_i| \le 2/m$ after centering, a conservative bound. The raw bound can be attained: with target $(1/2, 1/2)$ and samples $[0, 0, 0, 1]$, removing the final sample gives $d_i = 1/4$. The bounds above apply to histogram TV with fixed bins; other test statistics require bounds for their specific scoring rules.

Interpretation and centering. Given the pool, credits are deterministic removal contributions (Tukey, 1958; Efron, 1982). Their raw signs depend on the full renormalized histogram: even removing a sample in an overrepresented bin can overshoot the target. For target $(0.6, 0.4)$ and samples $[0, 0, 1]$, the raw credit of either zero is positive $(1/30)$. Centering defines credit relative to the observed pool average, sets its sample mean to zero, and can change individual signs. The baseline depends on the sampled group; our analysis of the resulting surrogate covers the bounds above and the additive reward case.

Relation to Shapley allocation and cost. The Shapley value averages marginal contributions over coalitions (Shapley, 1953); JCA fixes the attribution context to removal from the full observed pool. This defines a local allocation whose values refer to that pool, including for nonsmooth scores such as TV. Computing all credits requires m additional score evaluations after $S(G)$. The current implementation recomputes histograms, so its total runtime depends on both the number of scoring calls and the size of each pool.

B.2 THE W_1 DIFF SURROGATE AND ITS DISCRETIZATION

With $q = \pi_{\theta_{\text{old}}}$ and $p = \pi_\theta$, the implemented weight for a sequence y is $(p(y)/q(y))^{1/(\tau T(y))}$. For a constant exponent a, the normalized weighted distribution is proportional to $q^{1-a}p^a$; $a = 1$ recovers reweighting to p. The implemented exponent $a(y) = 1/(\tau T(y))$ additionally depends on completion length, specifying a length dependent tilted distribution. At the sampling policy, weights are one and the loss supplies a local derivative through the log probabilities. Unequal list lengths distinguish the equally weighted rollout CDF used by the surrogate from the pooled CDF weighted by samples used for evaluation.

CDF discretization. For continuous families, the approximation uses 101 equally spaced points and weights $\delta_0 = 0$ and $\delta_k = y_k - y_{k-1}$. For integer support $y_0 < \cdots < y_K$, the CDF is constant on each interval $[y_k, y_{k+1})$, so exact integration uses

$$\mathcal{L}_{\text{discrete}}(\theta) = \frac{1}{y_K - y_0} \sum_{k=0}^{K-1} \left| \hat{F}(y_k; \theta) - F^*(y_k) \right| (y_{k+1} - y_k). \tag{6}$$

This integral corresponds to $\delta_k = y_{k+1} - y_k$ for $k < K$ and $\delta_K = 0$ in Eq. (4).

C IMPLEMENTATION DETAILS

Training. LoRA rank 32; Adam at 10^{-5}; one group of 16 rollouts per step; requested list length 24; sampling temperature 1.0 and a completion limit of 800 tokens. Main RL runs use 600 steps and seeds 0, 1, and 2; replacement credit and training with three excluded families use two seeds. The requested lists and update counts are matched where stated, but realized token and compute budgets vary with generated lengths and skipped groups. JCA averages sample credits within each rollout and sends them to the loss with importance sampling. W_1 diff differentiates Eq. (4) through the backend's custom loss interface, with $\tau = 1$ unless ablated.

Global baseline. This control broadcasts $(S(G) - \mu_t)/\max(\sigma_t, 10^{-6})$ using statistics computed before the current update. The running mean uses decay $\rho = 0.95$, $\mu_{t+1} = \rho\mu_t + (1-\rho)S(G)$, and the running absolute deviation is $\sigma_{t+1} = \rho\sigma_t + (1-\rho)|S(G) - \mu_{t+1}|$. We initialize μ with the first observed score and σ at 0.05; the first group's distributional credit is therefore zero. Here σ is a running absolute deviation, not a within-group reward standard deviation. Clipped variants cap this credit at ± 3 before applying the count penalty.

Counts and parsing. Runs that use advantages subtract $\lambda \min(|n_j - n|/n, 1)$ after aggregating sample credits; W_1 diff has no count penalty. Rollouts with fewer than $\max(4, \lfloor n/2 \rfloor)$ valid samples are dropped, and groups retaining fewer than half the requested rollouts are skipped. Logged credit statistics precede count penalties. Malformed items are filtered during parsing, so requested and scored sample counts can differ accordingly.

Evaluation. Aligned requests: 42×24 per family. Long lists: 5×200. The independent protocol requests 1,000 completions per family, each prompted to emit one sample. We pool every valid number parsed from each completion, including responses with multiple values; we do not select only the first value. For example, Uniform(0,1) yields 1,367 parsed values from the base and 2,629 from the trained model under these nominally single-sample requests. The temperature sweep and untrained thinking rows in Table 1 use 5×200 requests, with an 8,192-token output budget. The sweep tests temperatures 0.7, 1.0, 1.3, and 1.5 at top $p = 0.95$. Discrete tests use χ^2 after merging adjacent bins to expected counts of at least five; continuous tests compare outputs with 1,000 reference draws from the truncated target using KS, with $\alpha = 0.01$. Dependence within a list may affect calibration of these tests; assessing independence requires analysis of joint sample behavior. Reported SDs summarize variation across the available training seeds. Inference about differences between methods requires a separate statistical comparison.

Continuous TV. We use 20 bins of equal width over each family's fixed truncated support. Target masses are normalized CDF differences, and empirical masses are normalized histogram counts. The evaluator uses identical bins across methods; training methods differ in whether they access scores, target bin probabilities, or the CDF. **Reference with finite samples.** Even independent target samples fluctuate around the target histogram, so their expected TV score is below one. Individual scores

can lie above or below this reference expectation; statistical equivalence between methods requires a separate comparison.

Position and natural-format diagnostics. In the aligned Uniform(0,1) evaluation of JCA seed 0, 25 of 42 lists begin with 0.45. This descriptive count illustrates that improved pooled histograms do not establish correct marginals at each position or independent sampling within lists. The MCQ prompt specifies four options and an answer label but does not request uniform answer positions; its scores measure position bias relative to a uniform reference. The persona prompt explicitly requests a 70/30 occupation split and uniform selection among four cities.

D Adapted SSoT Reproduction

Following SSoT (Misaki & Akiba, 2026), we ask the model to generate a string and extract a category from it. For the smaller models, we bound strings to 8 to 16 characters and specify an operation that sums character codes and takes a modulus. These adaptations make the experiment distinct from an exact reproduction of the published method. At temperature 0.6 with a budget of 8,192 tokens, we request 300 independent draws for each of Bernoulli, Binomial, and Poisson. The mean TV score over these three families is 0.617 for 4B and 0.561 for 9B, with zero family test passes; valid parsing rates range from 81% to 94%. For context, the untrained 4B scores 0.799 on the same families under aligned list requests. This comparison varies both the prompting method and the request format. The SSoT subset scores are therefore excluded from the full table over all 15 families.

E Frontier Model Details

The reference panel uses September 2026 public API runs, with default requests of three lists of 100 items per family and a completion budget of 3,000 to 131,072 tokens depending on the endpoint. Table 1 distinguishes endpoints in their default mode from those with thinking disabled. Matched requests use 42×24: Doubao Seed Evolving scores 0.945 with $12/15$ passes, and DeepSeek V4 Pro with thinking disabled scores 0.857 with $3/15$ passes. Their corresponding scores with default request budgets are 0.953 and 0.852. Both sample variation and test power change with request size; pass counts should be interpreted within each protocol. The default DeepSeek V4 Pro, MiniMax M3, and Doubao Seed 2.1 Pro runs use 3,000 tokens; the matched DeepSeek V4 Pro run uses 4,096.

Endpoints and budgets. All endpoints are public APIs or relays (September 2026), queried in default mode with the request format each endpoint supports. Completion budgets range from 3,000 to 131,072 tokens: GLM 5.3 at default thinking routinely exceeds 32k tokens of reasoning on this task and required the full 131k budget to terminate, which smaller budgets can mistake for unmeasurability. Claude Fable 5 and 5.1 emit thinking blocks at their default, so their rows measure thinking-on behavior, whereas Claude Opus 5 is measured at its API default, which does not enable extended thinking.

F Ablation Study

We ablate the credit assignment rule in a softmax policy over four choices, without auxiliary penalties (Fig. 5). For each of two target distributions, we use five seeds and 800 iterations; policy gradient updates estimate the gradient from 16 sampled actions per iteration. The broadcast baseline stays at mean TV distances 0.421 and 0.449, while JCA reaches 0.036 and 0.039. The supervised cross-entropy reference reaches below 10^{-6} on both targets. This ablation isolates the effect of credit assignment on the policy update.

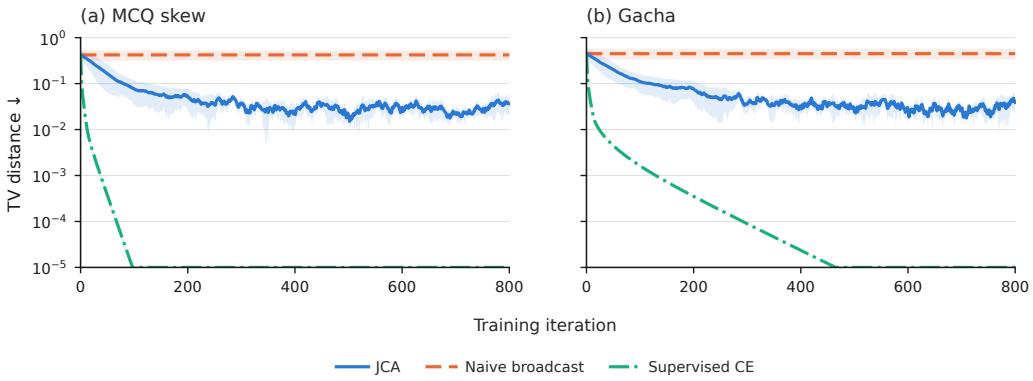


Figure 5: **Credit assignment ablation with a softmax policy (5 seeds, $\pm$1 SD).** Broadcast remains flat; JCA reaches mean TV distances 0.036 and 0.039; the supervised reference reaches below 10^{-6}. Curves use a 10^{-5} display floor.

G Additional Sensitivity Diagnostics

Figure 6 visualizes the training diagnostics and results on unseen families summarized in the main text. The sensitivity window is fixed to steps 250 to 299, including the default runs of 600 steps at that same intermediate window. This avoids comparing their final score at step 600 against the final scores of ablations of 300 steps. The grid of group sizes varies both pool size and the number of rollouts; ranking performance at a common budget requires aligned evaluation requests.

Alternative training rewards. Additional JCA runs select TV, χ^2 p-values, or a score based on Wasserstein distance. Their historical log field `tv_score` stores the selected reward, not always TV. In particular, the Wasserstein setting uses $1 - W_1/\mathrm{span}$ for continuous families and TV for discrete families; the χ^2 setting uses test p-values for discrete families and TV for continuous ones. These training rewards have different scales. Comparing their TV fidelity requires a common evaluation; Table 3 therefore compares settings that share the TV reward.

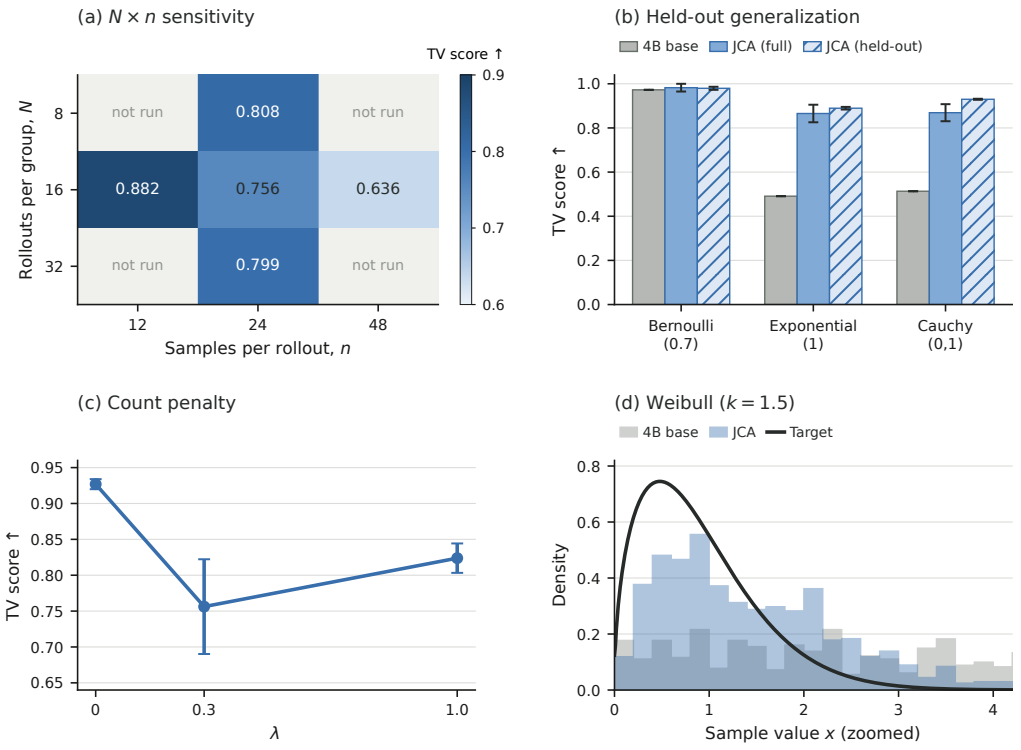


Figure 6: **Sensitivity and transfer diagnostics.** (a,c) TV of training pools over steps 250 to 299, matching Table 3; the default uses three seeds and other penalty settings use two. (b) Aligned evaluation on unseen families. (d) A zoomed Weibull illustration; the TV scores over the full support are 0.35 and 0.73.