SELECTIVE TIE LEARNING: MITIGATING SPURIOUS CORRELATIONS IN PREFERENCE OPTIMIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Preference optimization can learn spurious correlations, and augmenting the preference data with ties (equally preferred responses) can reduce spurious reliance. However, ties differ in their corrective effect, making their selection for augmentation important. For log-linear policies, we derive each tie's influence on spurious reliance and show that taking the largest predicted reductions is first-order optimal among subsets of equal size. Computing these influences requires knowledge of the spurious features and the curvature of the training objective, which is often unavailable or expensive to estimate. To this end, we explore using the magnitude of a tie's implicit reward margin and characterize the conditions under which it is a useful proxy for predicted influence. To test the theory and its qualitative transfer to neural models, we estimate influence with a counterfactual proxy for spurious reliance and use a generalized Gauss–Newton approximation to curvature for language models. Across log-linear policies, neural models, and large language models, our selection rules consistently achieve comparable reductions in spurious reliance with fewer ties than uniform selection.

1 INTRODUCTION

Aligning large language models with human preferences is central to developing useful and safe AI systems (Ouyang et al., 2022; Bai et al., 2022). Standard methods for this alignment, including reinforcement learning from human feedback (RLHF) and direct preference optimization (DPO) (Rafailov et al., 2023), learn policies from pairwise comparisons of responses. Learning from comparisons, however, can encourage reliance on features that predict human labels without tracking substantive quality, such as verbosity (Singhal et al., 2023; Saito et al., 2023) and sycophancy (Sharma et al., 2024). Models that rely on these features can fail when the features stop predicting human labels at deployment.

Recent theory shows that reliance on spurious features can persist even with unlimited preference data, because it arises from properties of the training distribution (Moya et al., 2026). Collecting more examples from the same distribution therefore does not remove it. Because the bias originates in the training distribution, two families of interventions target it: methods that reshape the training objective (Park et al., 2024) and methods that modify the training examples (Liu et al., 2025; Kim et al., 2025). The two families are complementary, since interventions on the training examples can work with existing objectives for preference learning rather than replace them. Our work advances interventions of this second kind.

Tie training is one such data intervention (Moya et al., 2026). It augments preference data with ties: pairs of responses with equal utility that differ in spurious attributes. Under local log-linear assumptions, this augmentation provably suppresses reliance on these attributes. Related data interventions include the neutral pairs of RRM (Liu et al., 2025) and counterfactual augmentation for length bias (Kim et al., 2025). However, both methods keep many augmented pairs even after filtering, which adds substantial training computation. This cost raises a question: can careful selection retain most of the reduction in spurious reliance with fewer ties?

To study this question, we analyse log-linear policies with a core–spurious feature decomposition. Within this setting, we propose Selective Tie Learning (STL), which selects a small subset of ties using a model trained on the original preference data. Our contributions are:

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.

(i) We derive the first-order influence of each tie on the model's reliance on spurious features (Proposition 1). Selecting the ties with the largest predicted reductions is first-order optimal among subsets of the same size (Theorem 2).

(ii) Because computing the influence in *(i)* requires the Hessian of the loss, we derive a cheaper selector that needs only forward passes. We show that, under a Gaussian model of the ties, selecting the ties with the largest absolute implicit reward margins maximizes the expected first-order reduction in spurious reliance among all such selectors (Theorem 3).

(iii) To extend *(i)* to neural models, we estimate the influence of ties using a counterfactual measure of spurious reliance and a damped generalized Gauss–Newton approximation of the Hessian (Section 5). The estimate needs no explicit core–spurious feature decomposition and shares one Hessian inversion across all candidate ties.

We test the theoretical predictions in log-linear models and evaluate their transfer to neural networks and language models. Across these settings, targeted selection achieves comparable reductions in spurious reliance with fewer ties than uniform selection.

2 Setup and Problem Formulation

2.1 DPO with a Log-Linear Policy

Preference data. Preference learning trains a policy from comparisons between responses to the same prompt. These comparisons form a dataset $\mathcal{D}_{\text{pref}} = \{(x_i, y_{w,i}, y_{l,i})\}_{i=1}^{N}$ sampled from a distribution P, where $y_{w,i}$ is preferred to $y_{l,i}$ for prompt $x_i \in \mathcal{X}$ (Rafailov et al., 2023; Christiano et al., 2017).

Core and spurious features. We represent the prompt and responses in each comparison using a fixed feature map $\phi(x, y) = [\phi_c(x, y); \phi_s(x, y)]$ and assume that the preference depends only on the core features ϕ_c. The spurious features ϕ_s can still correlate with the preference in the training data, although this correlation need not persist at deployment.

Log-linear policy. To analyse how a policy uses these features, we assume a log-linear form, $\pi_\theta(y \mid x) = \exp(\theta^\top \phi(x, y))/Z_\theta(x)$, where the normalizer $Z_\theta(x)$ sums over responses and $\theta = [\theta_c; \theta_s]$ weights the core and spurious features. We assume the fixed reference policy $\pi_{\text{ref}} = \pi_{\theta_{\text{ref}}}$ belongs to the same family. The log-ratio $\log(\pi_\theta/\pi_{\text{ref}})$ then equals $(\theta - \theta_{\text{ref}})^\top \phi(x, y)$ up to a normalization term that cancels when comparing responses to the same prompt.

DPO. DPO scales this log-ratio into the implicit reward $r_\theta(x, y) = \beta \log\big(\pi_\theta(y \mid x)/\pi_{\text{ref}}(y \mid x)\big)$, where $\beta > 0$ controls regularization towards the reference policy. Within each comparison the normalization cancels, and the reward difference is $\beta\tilde{\theta}^\top \Delta\phi$, where $\tilde{\theta} = \theta - \theta_{\text{ref}}$ and $\Delta\phi = \phi(x, y_w) - \phi(x, y_l)$. DPO fits this difference to the observed preferences through the logistic loss

$$\mathcal{L}_{\text{DPO}}(\theta) = -\mathbb{E}_{\mathcal{D}_{\text{pref}}}\Big[\log \sigma\big(\beta\tilde{\theta}^\top \Delta\phi\big)\Big],$$

where σ is the logistic function. Because $\Delta\phi = [\Delta\phi_c; \Delta\phi_s]$, both core and spurious features contribute to the reward difference. DPO can therefore reduce the loss by exploiting spurious correlations that predict the winner.

2.2 Spurious Reliance and Ties as an Intervention

DPO exploits spurious correlations even at the population level (Moya et al., 2026). With $\mu = \mathbb{E}_P[\Delta\phi]$ and $\Sigma = \mathbb{E}_P[\Delta\phi\Delta\phi^\top]$, two mechanisms drive this reliance. Through the mean, spurious differences predict the winner directly ($\mu_s \neq 0$). Through leakage, they correlate with core differences ($\Sigma_{sc} \neq 0$) and inherit their predictive signal. Both depend on P alone, so more comparisons from the same distribution do not remove the reliance.

Ties as an intervention. These two mechanisms motivate comparisons that carry neither a mean spurious bias nor leakage from core differences. We call such comparisons *ties*: a tie $\tau = (x, \{y_A, y_B\})$ pairs two responses with identical core features and different spurious features, so its feature difference $\Delta\phi = [0; \Delta\phi_s]$ has no core part to leak from. Assigning either response as the winner with equal probability also makes the mean spurious difference zero. Moya et al. (2026) show that adding such ties to DPO training reduces spurious reliance.

2.3 Problem: Selecting Ties from a Pool

The benefit of tie training (Moya et al., 2026) leaves open which ties to use. Ties differ because a tie acts on the policy only through its margin: if the policy already assigns equal implicit reward to both responses, training on the tie changes nothing. Training on the whole pool $\mathcal{D}_{\text{tie}}$ of M candidates therefore spends effort on ties that change little. Under a budget of $k < M$ ties, we instead seek the subset of the given pool whose training most reduces spurious reliance. Such a subset should outperform k ties sampled uniformly, the simplest selector.

In this paper, each candidate comes from a strict comparison sampled from $\mathcal{D}_{\text{pref}}$. A known transformation T then changes the spurious features of the preferred response while approximately preserving its core features, $\phi_c(x, T(y_w)) - \phi_c(x, y_w) \approx 0$. So that each response wins equally often, each candidate is a packet with both orientations, $\tau = \{\tau_\ell, \tau_r\}$, where $\tau_\ell = (x, y_w, T(y_w))$ and $\tau_r = (x, T(y_w), y_w)$ with equal probability, so their feature differences cancel in expectation, while each orientation has approximately zero core difference. We score and select whole packets, then sample one orientation from each selected packet for training. Our selectors also accept pools from other constructions that satisfy the assumptions of our analysis. We study this selection problem at two levels of access to the model. Section 3 derives the best selector when gradients and curvature are available, and Section 4 derives the best selector when only forward passes are available.

3 Selection with Full Access: Influence

In this section, we measure spurious reliance by a spurious energy and analyse, to first order, how training on a tie changes it. We show that this first-order change is the classical influence of the tie on the energy (Koh & Liang, 2017), so selecting the k ties with the largest predicted reduction solves the selection problem to first order.

3.1 The First-Order Effect of a Tie on Spurious Energy

Spurious energy. Spurious reliance shows up in the margin. The margin on a comparison splits into a core and a spurious part, $\beta(\tilde{\theta}_c^\top \Delta\phi_c + \tilde{\theta}_s^\top \Delta\phi_s)$, and only the core part reflects the preference. We therefore measure reliance by the mean square of the spurious part, before scaling by β:

$$E(\theta) = \mathbb{E}_P\big[(\tilde{\theta}_s^\top \Delta\phi_s)^2\big] = \tilde{\theta}_s^\top \Sigma_{ss} \tilde{\theta}_s.$$

Squaring counts spurious contributions of either sign, and weighting by Σ_{ss} counts spurious weights only along directions in which responses actually differ. The energy vanishes exactly when spurious features never move the margin under P, and a tie is useful to the extent that training on it lowers this energy.

We analyse a random tie τ_j through its expected DPO loss when its two orientations are sampled with equal probability. We assume that the two responses in a tie have the same core features, so the feature differences of the two orientations are $[0; d_j]$ and $[0; -d_j]$, where

$$d_j = \phi_s(x_j, y_{w,j}) - \phi_s(x_j, T(y_{w,j})).$$

Because the loss is averaged over both orientations, reversing the sign of d_j leaves it unchanged, so either orientation can define d_j. With margin $m_j = \beta\tilde{\theta}_s^\top d_j$ for the first orientation, the tie loss is

$$\ell_j(\theta) = -\tfrac{1}{2}\big[\log\sigma(m_j) + \log\sigma(-m_j)\big].$$

To isolate the effect of one tie, we add it to DPO with a small weight ϵ/N,

$$\theta_j^\epsilon = \arg\min_\theta \; \mathcal{L}_{\text{DPO}}(\theta) + \frac{\epsilon}{N}\,\ell_j(\theta),$$

so that $\epsilon = 0$ recovers the DPO solution θ_0. The effect of the tie on reliance is then the rate at which the spurious energy $E(\theta_j^\epsilon)$ changes as ϵ grows from zero.

Tie gradient. The gradient of the tie loss, $g_j = \nabla_\theta \ell_j$, drives this rate. Because the margin on a tie depends only on the spurious weights, so does the gradient:

$$g_{j,c} = 0, \qquad g_{j,s} = \frac{\beta}{2}\tanh\Big(\frac{m_j}{2}\Big) d_j.$$

Flipping the orientation flips both d_j and m_j, so the gradient does not depend on which orientation defines the contrast d_j. A tie therefore pushes the policy only along its spurious contrast, with a strength set by its margin. The push vanishes when the policy already assigns equal implicit reward to both responses, grows as $\frac{\beta}{4} m_j d_j$ for small margins, and saturates at magnitude $\frac{\beta}{2}\|d_j\|$ for large ones.

From push to influence. The gradient gives the immediate push of a tie, but DPO resists it, and the curvature of the DPO loss decides how far the solution moves. We analyse the DPO policy θ_0, a solution at which the DPO loss is stationary and strictly convex,

$$\nabla \mathcal{L}_{\mathrm{DPO}}(\theta_0) = 0, \qquad H = \nabla^2 \mathcal{L}_{\mathrm{DPO}}(\theta_0) \succ 0.$$

Adding the tie with weight ϵ/N tilts this stationarity condition, and the solution restores it by moving at rate $-H^{-1} g_j(\theta_0)/N$. Only the spurious part of this move changes the energy. While the tie pushes the spurious weights, DPO refits the core weights, so the spurious weights move at rate $-K_s^{-1} g_{j,s}(\theta_0)/N$, where $K_s = H_{ss} - H_{sc} H_{cc}^{-1} H_{cs}$ is the Schur complement of the core block. The resulting rate of change of the energy is the influence of the tie (Koh & Liang, 2017).

Proposition 1 (Influence of a tie on spurious energy)**.** *Suppose θ_0 is stationary with $H \succ 0$. For small ϵ, the solution θ_j^ϵ satisfies $E(\theta_j^\epsilon) - E(\theta_0) = -\epsilon A_j + O(\epsilon^2)$, with*

$$A_j = \frac{1}{N} \big\langle \nabla_s E(\theta_0), K_s^{-1} g_{j,s}(\theta_0) \big\rangle = \frac{\beta}{2N} \tanh\Big(\frac{m_j}{2}\Big) \big\langle \nabla_s E(\theta_0), K_s^{-1} d_j \big\rangle,$$

where $m_j = \beta \tilde{\theta}_{0,s}^\top d_j$ is the margin on the tie at θ_0.

Proof. See Appendix A.1. □

Reading the influence. A positive A_j predicts that the tie lowers the energy, and the influence shows what makes it positive. Its size depends on two factors. The margin factor $\tanh(m_j/2)$ is the strength of the push: a tie on which the policy is already indifferent has no influence. The alignment factor $\langle \nabla_s E(\theta_0), K_s^{-1} d_j \rangle$ measures whether that push, after DPO refits the core weights, points against the current reliance. The alignment factor need not share the sign of the margin, so a tie can raise the energy. Influences of different ties add to first order, so choosing a set of ties reduces to ranking them.

3.2 Main Result: First-Order Optimal Selection

Training on a set S of ties, each with weight ϵ/N, tilts the stationarity condition by the sum of their gradients. At first order the solution responds linearly to this tilt, so the influences of the ties add:

$$E(\theta_S^\epsilon) - E(\theta_0) = -\epsilon \sum_{j \in S} A_j + O(\epsilon^2).$$

Under the budget, the best set is the one that makes this sum largest.

Theorem 2 (First-order optimal tie selection)**.** *Let S_k contain the k ties with the largest A_j. Then*

$$\sum_{j \in S_k} A_j = \max_{|S|=k} \sum_{j \in S} A_j \;\geq\; \frac{k}{M} \sum_{j=1}^{M} A_j,$$

with strict inequality for $0 < k < M$ unless all A_j are equal.

Proof. See Appendix A.1.4. □

Interpretation. The theorem ranks ties by influence, that is, by how hard each tie pushes and where the push points. The right side is what uniform selection achieves in expectation, so the theorem also measures what selection buys: k times the gap between the mean influence of the top k ties and that of the pool. Selection therefore pays off exactly when ties differ, which is the premise of Section 2.3.

Scope: first-order result. The theorem holds to first order, and the $O(\epsilon^2)$ terms capture how ties interact. Two ties with similar spurious contrasts overlap, because training on one shrinks the margin

of the other. Every selected tie also moves θ_0 and, with it, the margins and curvature that define the remaining influences. The guarantee holds while the total tie weight is small relative to N. Training uses larger weights, so Section 6 tests whether the first-order ranking still selects better ties.

The cost of influence. Computing A_j requires three quantities: the inverse curvature K_s^{-1}, the energy gradient $\nabla_s E(\theta_0)$, and the gradient of each tie. For a neural policy, each is expensive, and $\nabla_s E$ requires knowing the spurious features themselves. Of the two factors in A_j, only the margin is available from a forward pass. This raises the question of the next section: using only forward passes, which selector best approximates influence on average?

4 Selection with Forward Passes: Margin

Section 3 showed that ranking ties by influence minimizes spurious energy to first order, but computing influence requires the inverse curvature, the energy gradient, and a gradient for each tie. We now ask how well ties can be ranked using only forward passes. We introduce *pure margin selection*, which ranks ties by the absolute margin that the DPO solution θ_0 assigns to each of them. Under a Gaussian model of the spurious contrasts, it maximizes the expected first-order reduction in energy among selectors that see only the margins.

4.1 What a Forward Pass Reveals

A forward pass through the policy and the reference returns the log-probability of each response, and hence its implicit reward. On a tie, the difference between the two implicit rewards is the margin m_j from Section 3.1. We orient each tie uniformly at random, which leaves its influence unchanged, so that $m_j = r_{\theta_0}(x_j, y_{A,j}) - r_{\theta_0}(x_j, y_{B,j}) = \beta \tilde{\theta}_{0,s}^\top d_j$, where $y_{A,j}$ is the response that the sampled orientation places first. Forward passes thus deliver the margin of every tie in the pool at the cost of evaluating the two models.

The margin reveals one factor of the influence and hints at the other. From Proposition 1,

$$A_j = \frac{\beta}{2N} \tanh\Big(\frac{m_j}{2}\Big)\, a_j, \qquad a_j = \big\langle \nabla_s E(\theta_0), K_s^{-1} d_j \big\rangle.$$

The margin fixes the margin factor, while the alignment a_j stays hidden because it requires the curvature and the energy gradient. Both, however, are linear in the same contrast d_j, so the margin carries information about the alignment. A selector that sees only the margins can do no better than rank ties by their expected influence given those margins, and the margin factor passes through this expectation unchanged. Everything therefore hinges on how the expected alignment depends on the margin.

4.2 Main Result: Pure Margin Selection Is Optimal in Expectation

Assumptions. How the expected alignment depends on the margin is a question about how contrasts are distributed across the pool. The random orientation centres them, $\mathbb{E}[d_j] = 0$. For tractability, we further assume that, given θ_0, the contrasts are independent across ties and Gaussian with covariance $c\Sigma_{ss}$, $c > 0$. The covariance models ties that inherit the spurious geometry of P: after whitening by Σ_{ss}, the contrasts are isotropic, as in Moya et al. (2026). Under this model the alignment and the margin are jointly Gaussian, so the expected alignment grows linearly with the margin, with a positive slope.

Theorem 3 (Ranking ties from margins)**.** *Under the assumptions above, with $\tilde{\theta}_{0,s} \neq 0$, there is a constant $\lambda > 0$, common to all ties, such that*

$$\mathbb{E}[A_j \mid m_j] = \frac{\lambda}{N}\, m_j \tanh\Big(\frac{m_j}{2}\Big).$$

Thus, among selectors that see only the margins, selecting the k ties with the largest $|m_j|$ maximizes the expected first-order reduction in energy.

Proof. See Appendix A.2. □

Interpretation. The expected influence $m_j \tanh(m_j/2)$ is even and increases with $|m_j|$, so the sign of the margin, which records only the sampled orientation, drops out of the ranking. We call the resulting selector *pure margin selection*. It works for two reasons. A larger absolute margin produces a stronger push from the tie. Under the covariance assumption, it also predicts better alignment with the direction that lowers the energy. A tie with zero margin has neither, which makes precise the observation in Section 2.3 that such a tie changes nothing.

Scope of the result. The theorem concerns the expected influence given the margins. Two ties with the same margin can still differ in influence, because their contrasts can point in different directions, so pure margin selection need not recover the influence ranking of a particular pool. Section 6 compares the two selectors directly.

From theory to neural models. Both derived selectors rest on the log-linear model assumption. In a neural policy, the margin stays exact and cheap, because it still requires only forward passes. Influence, by contrast, requires a proxy for the energy, whose spurious features are unobserved, and an estimate of the curvature. Section 5 builds both.

5 From Theory to Neural Models

Section 4 showed that pure margin selection maximizes the expected first-order reduction in energy under a Gaussian model of the spurious contrasts, while influence attains the first-order optimum without that model. We now implement both selectors in neural policies. Pure margin selection carries over directly, because it needs only forward passes. Influence needs two estimates: a *counterfactual energy*, which measures spurious reliance on a fixed probe of ties, and a damped estimate of the curvature of the DPO loss on the strict comparisons. In practice, the DPO policy θ_0 is the final checkpoint of DPO training, which satisfies the stationarity condition of Section 3 only approximately.

5.1 Influence in Practice

Counterfactual energy. A neural policy exposes no spurious features, so we cannot compute the energy of Section 3 directly. We can still approximate it, because a tie isolates the spurious part of the margin: its two responses differ only in spurious features, so its margin is purely spurious. We therefore form a fixed probe $\mathcal{P}$ of N_v ties, held out from selection, and define the counterfactual energy

$$E_{\mathrm{cf}}(\theta) = \frac{1}{N_v} \sum_{n=1}^{N_v} m_n(\theta)^2, \tag{1}$$

where m_n is the margin on probe tie n. Squaring removes the dependence on its orientation.

In the log-linear model, the counterfactual energy recovers the energy gradient up to a positive scale when the probe contrasts have the covariance assumed in Section 4 (see Appendix A.3): $\mathbb{E}_{\mathcal{P}}[\nabla E_{\mathrm{cf}}(\theta_0)] = c\beta^2 \nabla E(\theta_0)$. The scale leaves the ranking of ties unchanged, so the counterfactual energy supplies the gradient that influence needs.

Curvature. The remaining ingredient is the curvature. In a neural policy, influence keeps the form of Proposition 1 but runs over all parameters: the full inverse curvature accounts for how every parameter responds to a tie, the role that K_s played for the spurious weights. We estimate the curvature with a damped generalized Gauss–Newton (GGN) matrix $\widehat{H}$ (Schraudolph, 2002; Martens, 2020), computed on a subset of the strict comparisons in $\mathcal{D}_{\mathrm{pref}}$, and score each tie by

$$\widehat{A}_j = \frac{1}{N} \big\langle \nabla E_{\mathrm{cf}}(\theta_0), \widehat{H}^{-1} g_j(\theta_0) \big\rangle.$$

Every estimated influence shares the same energy gradient, so we apply the inverse curvature to it once, $v = \widehat{H}^{-1} \nabla E_{\mathrm{cf}}(\theta_0)$, and reuse the result across the pool. Estimating the influence of a tie then takes only its gradient, $\widehat{A}_j = v^\top g_j(\theta_0)/N$. These approximations make influence practical but do not preserve the log-linear guarantee. We provide more details in Appendix B.1.

5.2 Margin in Practice

Pure margin selection ranks ties by the absolute margin $|m_j|$, which forward passes of the DPO policy and the reference deliver for every tie in the pool. Because margins are all it uses, the selector skips everything that influence requires: the probe, the curvature, and the tie gradients. It never even computes the energy, yet under the assumptions of Section 4 its ranking maximizes the expected first-order reduction of that energy.

Two selectors, two costs. The two selectors thus trade cost for information. Influence uses tie gradients, the curvature, and a probe to approximate the first-order optimum. Pure margin selection uses only margins and is optimal in expectation under a Gaussian model of the contrasts. Section 6 measures what each buys over uniform selection: whether the first-order ranking helps at the tie weights used in training, and whether pure margin selection loses anything relative to influence.

6 Experiments

We test three theoretical predictions: (i) influence predicts the effect of an isolated tie (Proposition 1), (ii) mean influence conditioned on the tie margin follows the predicted curve (Theorem 3), and (iii) the highest-scoring ties reduce spurious reliance faster than uniformly sampled ties (Theorem 2). We begin with log-linear models and then ask whether the selection gains persist in two settings closer to practice: neural networks and language models. In every regime, spurious features correlate with preferences under the training distribution P, and the adversarial deployment distribution Q_{adv} reverses these correlations. Appendix C gives the full protocol.

6.1 Log-Linear: Quantitative Validation

Setup. We generate Gaussian features $\phi = [\phi_c; \phi_s]$ with $d_c = d_s = 10$. The operator T keeps the strict winner's core features ϕ_c and replaces its spurious features ϕ_s with those of a response drawn uniformly from $\mathcal{D}_{\mathrm{pref}}$. We compare uniform tie learning (UTL), pure margin selection (PM), and Oracle influence, which computes A_j in Proposition 1 exactly from the population spurious-energy gradient and the empirical DPO Hessian. Appendix C.3 gives the full details.

Theory checks. Panel (a) tests prediction (i): for each tie, we add its symmetric packet with weight $1/N$ ($\epsilon = 1$ in Proposition 1), re-optimize the model, and compare the observed reduction $E(\theta_0) - E(\theta_j^1)$ with A_j. Panel (b) tests prediction (ii): we repeat this with each tie in its fixed training orientation, group ties into ten equal-frequency bins by absolute margin, and compare each bin's mean observed reduction with the conditional mean in Theorem 3.

Results. Figure 1(a)–(b) supports predictions (i) and (ii). Across all isolated ties, predicted influence has 4.2% relative ℓ_2 error and 99.8% sign agreement with the observed reduction in spurious energy, including every tie predicted to increase it. In panel (b), the theoretical conditional mean lies within the 95% confidence interval of every margin bin.

Panels (c)–(d) show that the first-order ranking in prediction (iii) remains useful at full tie weight. PM and Oracle influence raise accuracy under Q_{adv} and lower spurious energy faster than UTL: with 100 ties, PM reaches the accuracy that UTL reaches with about 300. Their advantage narrows as the selected set approaches the full pool, where all methods coincide. PM closely tracks Oracle influence even though it uses only forward passes.

6.2 Neural Networks: Qualitative Persistence

The log-linear theory no longer applies once a nonlinear representation hides the core–spurious decomposition. We next ask whether its selection gains persist in a neural reward model.

Setup. We keep the log-linear data-generating process and tie construction, pass the features through a fixed random nonlinear mixer, and train an MLP reward model with DPO on the mixed features. We apply T before mixing the features, so each tie still preserves core content exactly. We compare UTL and PM with Practical influence, which replaces the population spurious-energy gradient with the gradient of the counterfactual energy E_{cf} and the Hessian with a damped GGN (Section 5). Appendix C.4 gives the full details.

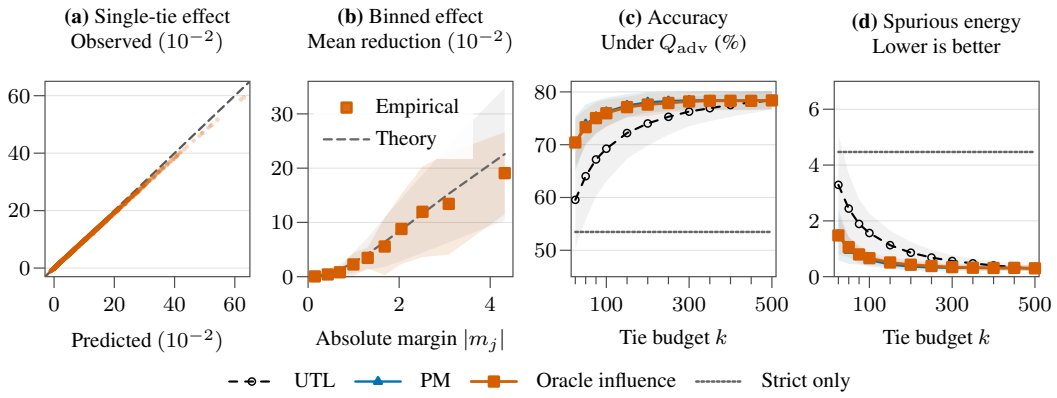

Figure 1: **Log-linear validation.** **(a)** Predicted versus observed reductions in population spurious energy after perturbing the objective with one symmetric tie. **(b)** Observed reductions from ties with sampled orientations and the theoretical conditional mean across margin bins containing equal numbers of ties (Theorem 3). **(c–d)** Accuracy under Q_{adv} and population spurious energy after training on selected ties. Curves are means; shading shows 95% Student-t confidence intervals.

Results. PM and Practical influence improve accuracy under Q_{adv} over UTL at every budget (Figure 2(a)). At 500 ties, both gain about 7.5 percentage points over UTL and exceed the accuracy that UTL reaches with 1,500 ties. Their advantage first grows and then narrows as the selected set approaches the full pool. As in the log-linear model, PM closely tracks influence.

6.3 LLMs: Controlled Preference Tasks

Language models introduce a pretrained policy and text responses. We test selection on two tasks whose spurious attributes we can control, so that the shifts and ties can be constructed exactly.

Setup. We fine-tune LoRA adapters of Llama-3.2-1B-Instruct with DPO, using the base model as the reference. In the hotel task, the model recommends one of two generated hotel profiles. Utility depends on price, distance, star rating, and a fixed noise term for each hotel, while seven other fields, such as building age and chain tier, correlate with utility positively in P and negatively in Q_{adv}. The operator T replaces these seven fields in the winner's profile with those of a hotel drawn from $\mathcal{D}_{\mathrm{pref}}$. In the emoji task, the safe response of a PKU-SafeRLHF pair (Ji et al., 2025) is always preferred, and we insert different numbers of one emoji into its two responses. The safe response usually receives more emojis in P and fewer in Q_{adv}. Here, T removes every emoji from the safe winner. We compare UTL, PM, and Practical influence. Appendix C.5 gives the full details.

Results. PM and Practical influence improve mean accuracy under Q_{adv} over UTL at every partial budget on both tasks (Figure 2(b)–(c)). With 500 ties, PM reaches 92.1% accuracy on hotels and 68.4% on emojis, compared with 74.6% and 59.3% for UTL. PM needs only 300 ties on hotels and 500 on emojis to exceed the accuracy that UTL reaches with 750. The advantage again narrows as the selected set approaches the full pool. The two scores track each other on hotels, whereas PM stays above Practical influence on emojis.

7 Related Work

Spurious correlations in preference learning. Preference learning can reward attributes that predict labels without tracking response quality. Reward models often favour longer answers even when they are not more helpful (Singhal et al., 2023), and sycophancy persists in models trained with human feedback (Sharma et al., 2024). Studies of reward-model exploitation and overoptimization reveal related shortcuts, including verbosity (Dubois et al., 2023; Casper et al., 2023; Gao et al., 2023). These works document concrete instances of spurious learning; our analysis characterizes how individual ties change reliance on spurious features.

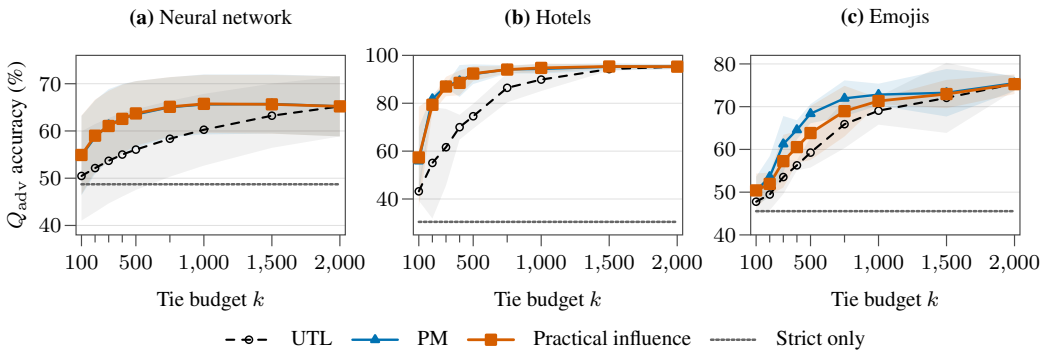

Figure 2: **Selection beyond the log-linear setting.** Q_{adv} accuracy for a nonlinear reward model **(a)** and LoRA-DPO on hotel **(b)** and emoji **(c)** tasks. Practical influence uses a sampled GGN and solves the damped system in its row space for all three neural regimes. Curves are means across seeds; shading shows 95% Student-t confidence intervals.

Mitigating spurious preferences. Existing methods intervene through the objective, reward model, internal representations, or training data. Some penalize or normalize length in the preference objective (Park et al., 2024; Lu et al., 2024), separate length from quality in reward models (Chen et al., 2024), or steer representations at inference (Rimsky et al., 2024). RRM (Liu et al., 2025) and counterfactual augmentation for length bias (Kim et al., 2025) are closest to our setting: they construct neutral or tie-like comparisons and train on many of them. STL instead selects from a fixed pool of ties under a training budget.

Theory and selection. Linear and log-linear models explain how the geometry of the training distribution produces unstable predictors (Nagarajan et al., 2021; Bombari & Mondelli, 2025; Veitch et al., 2021; Arjovsky et al., 2019). For DPO, Moya et al. (2026) trace population-level reliance to mean spurious bias and core–spurious correlation, and show that tie training suppresses it in aggregate. Active preference methods also select comparisons, but usually to reduce annotation cost or uncertainty, improve exploration, or filter difficult strict examples (Muldrew et al., 2024; Lin et al., 2026; Gao et al., 2025; Das et al., 2025; Melo et al., 2024). Patterning uses susceptibilities to reweight preference pairs towards desired changes in reward-model behaviour (Wang et al., 2026). STL shares this use of first-order sensitivity to guide data interventions, selecting ties by their influence on spurious reliance.

8 Conclusion

We introduced Selective Tie Learning, a principled method for selecting a small subset of ties. For log-linear models, we derived the first-order influence of each tie on spurious reliance and showed that selecting the largest predicted reductions is first-order optimal. We also developed pure margin selection, an inexpensive rule that needs only forward passes, and showed that under a Gaussian model of the ties, it maximizes the expected reduction in spurious reliance among such rules. For neural models, we estimated tie influence without an explicit core–spurious decomposition. Across log-linear models, neural networks, and language models, selected ties yielded higher accuracy than uniformly sampled ties when spurious correlations were reversed, while using fewer ties.

Our study assumes an existing pool of ties. It does not address how to construct reliable ties or how to choose the budget k, although both decisions affect practical performance. Our experiments use fully or partially controlled settings so that we can measure spurious reliance and distribution shift directly. Extending STL to natural preference data will therefore require transformations that preserve utility, a principled choice of k, and scoring that scales to large pools. Integrating these components into a complete training pipeline remains open. Appendix D discusses these limitations in more detail.

AI USE STATEMENT

In this work, we used generative AI tools to provide critical ingredients for proving mathematical claims, assist in the writing of proofs, design or provide feedback on the research methodology and experiments, and implement methods. We have not used generative AI tools to help develop theoretical models or conceptual frameworks, formulate mathematical claims, propose or refine hypotheses, support qualitative and thematic data analysis, or interpret results, and generating synthetic datasets, assisting with translation, and cleaning and reformatting datasets are not applicable to this work. Additionally, we used generative AI tools to create or edit software code, edit the paper to improve readability, identify relevant literature, and search for information. We have reviewed all AI-assisted work. The authors checked every proof step by step, the first author verified and tested all code generated by AI for correctness, and the authors checked every suggested reference against its original source. We take responsibility for the final content of this work, including text, claims or artefacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

The assumptions and claims behind our theoretical results appear in Proposition 1 and Theorems 2 and 3, with complete proofs and supporting derivations in Appendix A. Appendix B describes the scoring implementations, while Appendix C documents data generation and processing, model and training settings, selection protocols, and evaluation metrics. The repository at `https://anonymous.4open.science/r/STL/` contains the experiment code, configuration files, and documented commands for reproducing the experiments and figures.

REFERENCES

Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022.

Simone Bombari and Marco Mondelli. Spurious correlations in high dimensional regression: The roles of regularization, simplicity bias and over-parameterization. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, volume 267 of *Proceedings of Machine Learning Research*, pp. 4839–4873. PMLR / OpenReview.net, 2025.

Stephen Casper, Xander Davies, Claudia Shi, Thomas Krendl Gilbert, Jérémy Scheurer, Javier Rando, Rachel Freedman, Tomasz Korbak, David Lindner, Pedro Freire, Tony Tong Wang, Samuel Marks, Charbel-Raphaël Ségerie, Micah Carroll, Andi Peng, Phillip J. K. Christoffersen, Mehul Damani, Stewart Slocum, Usman Anwar, Anand Siththaranjan, Max Nadeau, Eric J. Michaud, Jacob Pfau, Dmitrii Krasheninnikov, Xin Chen, Lauro Langosco, Peter Hase, Erdem Bıyık, Anca D. Dragan, David Krueger, Dorsa Sadigh, and Dylan Hadfield-Menell. Open problems and fundamental limitations of reinforcement learning from human feedback. *Trans. Mach. Learn. Res.*, 2023, 2023.

Lichang Chen, Chen Zhu, Jiuhai Chen, Davit Soselia, Tianyi Zhou, Tom Goldstein, Heng Huang, Mohammad Shoeybi, and Bryan Catanzaro. ODIN: disentangled reward mitigates hacking in RLHF. In Ruslan Salakhutdinov, Zico Kolter, Katherine A. Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, volume 235 of *Proceedings of Machine Learning Research*, pp. 7935–7952. PMLR / OpenReview.net, 2024.

Paul F. Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. In Isabelle Guyon, Ulrike von Luxburg, Samy

Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pp. 4299–4307, 2017.

Nirjhar Das, Souradip Chakraborty, Aldo Pacchiano, and Sayak Ray Chowdhury. Active preference optimization for sample efficient RLHF. In Rita P. Ribeiro, Bernhard Pfahringer, Nathalie Japkowicz, Pedro Larrañaga, Alípio M. Jorge, Carlos Soares, Pedro H. Abreu, and João Gama (eds.), *Machine Learning and Knowledge Discovery in Databases. Research Track - European Conference, ECML PKDD 2025, Porto, Portugal, September 15-19, 2025, Proceedings, Part V*, volume 16017 of *Lecture Notes in Computer Science*, pp. 96–112. Springer, 2025. doi: 10.1007/978-3-032-06096-9_6.

Yann Dubois, Chen Xuechen Li, Rohan Taori, Tianyi Zhang, Ishaan Gulrajani, Jimmy Ba, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Alpacafarm: A simulation framework for methods that learn from human feedback. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, pp. 30039–30069, 2023. doi: 10.52202/075280-1308.

Chengqian Gao, Haonan Li, Liu Liu, Zeke Xie, Peilin Zhao, and Zhiqiang Xu. Principled data selection for alignment: The hidden risks of difficult examples. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, volume 267 of *Proceedings of Machine Learning Research*, pp. 18386–18409. PMLR / OpenReview.net, 2025.

Leo Gao, John Schulman, and Jacob Hilton. Scaling laws for reward model overoptimization. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pp. 10835–10866. PMLR, 2023.

Jiaming Ji, Donghai Hong, Borong Zhang, Boyuan Chen, Josef Dai, Boren Zheng, Tianyi Alex Qiu, Jiayi Zhou, Kaile Wang, Boxun Li, Sirui Han, Yike Guo, and Yaodong Yang. PKU-SafeRLHF: Towards multi-level safety alignment for LLMs with human preference. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 31983–32016, Vienna, Austria, 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.acl-long.1544.

Hyeonji Kim, Sujeong Oh, and Sanghack Lee. Mitigating length bias in RLHF through a causal lens. *CoRR*, abs/2511.12573, 2025. doi: 10.48550/ARXIV.2511.12573.

Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In Doina Precup and Yee Whye Teh (eds.), *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1885–1894. PMLR, 2017.

Xiaoqiang Lin, Arun Verma, Zhongxiang Dai, Daniela Rus, See-Kiong Ng, and Bryan Kian Hsiang Low. ActiveDPO: Active direct preference optimization for sample-efficient alignment. In *The Fourteenth International Conference on Learning Representations*, 2026.

Tianqi Liu, Wei Xiong, Jie Ren, Lichang Chen, Junru Wu, Rishabh Joshi, Yang Gao, Jiaming Shen, Zhen Qin, Tianhe Yu, Daniel Sohn, Anastasia Makarova, Jeremiah Zhe Liu, Yuan Liu, Bilal Piot, Abe Ittycheriah, Aviral Kumar, and Mohammad Saleh. RRM: robust reward model training mitigates reward hacking. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025.

Junru Lu, Jiazheng Li, Siyu An, Meng Zhao, Yulan He, Di Yin, and Xing Sun. Eliminating biased length reliance of direct preference optimization via down-sampled KL divergence. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, EMNLP 2024, Miami, FL, USA, November*

12-16, 2024, pp. 1047–1067. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.EMNLP-MAIN.60.

James Martens. New insights and perspectives on the natural gradient method. *Journal of Machine Learning Research*, 21(146):1–76, 2020.

Luckeciano Carvalho Melo, Panagiotis Tigas, Alessandro Abate, and Yarin Gal. Deep bayesian active learning for preference modeling in large language models. In Amir Globerson, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang (eds.), *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, pp. 118052–118085, 2024. doi: 10.52202/079017-3749.

Christian Moya, Alex Semendinger, Guang Lin, and Elliott Thornley. Spurious correlation learning in preference optimization: Mechanisms, consequences, and mitigation via tie training. In *Proceedings of the 43rd International Conference on Machine Learning*. PMLR, 2026.

William Muldrew, Peter Hayes, Mingtian Zhang, and David Barber. Active preference learning for large language models. In Ruslan Salakhutdinov, Zico Kolter, Katherine A. Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, volume 235 of *Proceedings of Machine Learning Research*, pp. 36577–36590. PMLR / OpenReview.net, 2024.

Vaishnavh Nagarajan, Anders Andreassen, and Behnam Neyshabur. Understanding the failure modes of out-of-distribution generalization. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems 35*, pp. 27730–27744, 2022. doi: 10.52202/068431-2011.

Ryan Park, Rafael Rafailov, Stefano Ermon, and Chelsea Finn. Disentangling length from quality in direct preference optimization. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024*, volume ACL 2024 of *Findings of ACL*, pp. 4998–5017. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.FINDINGS-ACL.297.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, pp. 53728–53741, 2023. doi: 10.52202/075280-2338.

Nina Rimsky, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Matt Turner. Steering llama 2 via contrastive activation addition. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pp. 15504–15522. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.ACL-LONG.828.

Keita Saito, Akifumi Wachi, Koki Wataoka, and Youhei Akimoto. Verbosity bias in preference labeling by large language models. *arXiv preprint arXiv:2310.10076*, 2023.

Nicol N. Schraudolph. Fast curvature matrix-vector products for second-order gradient descent. *Neural Computation*, 14(7):1723–1738, 2002. doi: 10.1162/08997660260028683.

Mrinank Sharma, Meg Tong, Tomasz Korbak, David Duvenaud, Amanda Askell, Samuel R. Bowman, Esin Durmus, Zac Hatfield-Dodds, Scott R. Johnston, Shauna Kravec, Timothy Maxwell, Sam McCandlish, Kamal Ndousse, Oliver Rausch, Nicholas Schiefer, Da Yan, Miranda Zhang, and Ethan Perez. Towards understanding sycophancy in language models. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024.

Prasann Singhal, Tanya Goyal, Jiacheng Xu, and Greg Durrett. A long way to go: Investigating length correlations in RLHF. *CoRR*, abs/2310.03716, 2023. doi: 10.48550/ARXIV.2310.03716.

Victor Veitch, Alexander D'Amour, Steve Yadlowsky, and Jacob Eisenstein. Counterfactual invariance to spurious correlations in text classification. In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pp. 16196–16208, 2021.

George Wang, Elizabeth Donoway, and Daniel Murfet. Patterning in practice: Debiasing reward models with susceptibilities. *arXiv preprint arXiv:2609.00699*, 2026.

A PROOFS

A.1 PROOFS FOR THE INFLUENCE ANALYSIS

A.1.1 TIE GRADIENT

Claim: The gradient of the tie loss averaged over its two orientations is

$$g_{j,c}(\theta) = 0, \qquad g_{j,s}(\theta) = \frac{\beta}{2} \tanh\left(\frac{\beta \tilde{\theta}_s^\top d_j}{2}\right) d_j.$$

Proof: The two orientations have feature differences $[0; d_j]$ and $[0; -d_j]$. Averaging their losses gives

$$\ell_j(\theta) = -\frac{1}{2} \log \sigma(\beta \tilde{\theta}_s^\top d_j) - \frac{1}{2} \log \sigma(-\beta \tilde{\theta}_s^\top d_j).$$

The resulting loss does not depend on the core parameters, so its core gradient is zero. Differentiating with respect to the spurious parameters gives

$$\begin{aligned} g_{j,s}(\theta) &= \frac{\beta}{2} \left[\sigma(\beta \tilde{\theta}_s^\top d_j) - \sigma(-\beta \tilde{\theta}_s^\top d_j)\right] d_j \\ &= \frac{\beta}{2} \tanh\left(\frac{\beta \tilde{\theta}_s^\top d_j}{2}\right) d_j. \end{aligned}$$

For the second equality, use $\sigma(z) - \sigma(-z) = \tanh(z/2)$. Reversing the orientation replaces d_j by $-d_j$. This changes the signs of both the hyperbolic tangent and the feature difference, so their product is unchanged. □

A.1.2 PARAMETER RESPONSE

Partition the Hessian of the strict objective into its core and spurious blocks. Let K_s denote the spurious Schur complement of H_{cc} in H:

$$H = \begin{bmatrix} H_{cc} & H_{cs} \\ H_{sc} & H_{ss} \end{bmatrix}, \qquad K_s = H_{ss} - H_{sc} H_{cc}^{-1} H_{cs}.$$

Claim: Under the stationarity condition of Proposition 1, the perturbed objective of Section 3.1 has a unique nearby minimizer for sufficiently small $\epsilon \geq 0$, and

$$\theta_j^\epsilon - \theta_0 = -\frac{\epsilon}{N} H^{-1} g_j(\theta_0) + O(\epsilon^2), \qquad \theta_{j,s}^\epsilon - \theta_{0,s} = -\frac{\epsilon}{N} K_s^{-1} g_{j,s}(\theta_0) + O(\epsilon^2). \tag{2}$$

Proof: The perturbed minimizer solves the stationarity equation

$$\nabla \mathcal{L}_{\mathrm{DPO}}(\theta) + \frac{\epsilon}{N} \nabla \ell_j(\theta) = 0$$

at $\theta = \theta_j^\epsilon$. At $\epsilon = 0$, the solution is $\theta_j^0 = \theta_0$. Differentiating the left side with respect to θ at $\theta = \theta_0$ and $\epsilon = 0$ gives H. Because the losses are smooth and the positive definite matrix H is invertible, the implicit function theorem gives a unique differentiable solution θ_j^ϵ near θ_0 for sufficiently small ϵ. For $\epsilon \geq 0$, both terms in the perturbed log-linear objective are convex, so this stationary point is a minimizer.

Differentiating the stationarity equation with respect to ϵ gives

$$\left[\nabla^2 \mathcal{L}_{\mathrm{DPO}}(\theta_j^\epsilon) + \frac{\epsilon}{N} \nabla^2 \ell_j(\theta_j^\epsilon)\right] \frac{d\theta_j^\epsilon}{d\epsilon} + \frac{1}{N} \nabla \ell_j(\theta_j^\epsilon) = 0.$$

At $\epsilon = 0$, $\theta_j^0 = \theta_0$ and $\nabla^2 \mathcal{L}_{\mathrm{DPO}}(\theta_0) = H$; the term multiplied by ϵ vanishes. Thus

$$H \left.\frac{d\theta_j^\epsilon}{d\epsilon}\right|_{\epsilon=0} + \frac{1}{N} g_j(\theta_0) = 0.$$

Solving for the derivative gives

$$\left.\frac{d\theta_j^\epsilon}{d\epsilon}\right|_{\epsilon=0} = -\frac{1}{N}H^{-1}g_j(\theta_0).$$

The Taylor expansion of θ_j^ϵ around $\epsilon = 0$ is

$$\theta_j^\epsilon = \theta_j^0 + \epsilon \left.\frac{d\theta_j^\epsilon}{d\epsilon}\right|_{\epsilon=0} + O(\epsilon^2) = \theta_0 - \frac{\epsilon}{N}H^{-1}g_j(\theta_0) + O(\epsilon^2),$$

which is the first identity in (2).

The preceding claim gives $g_{j,c}(\theta_0) = 0$. To recover the spurious part of the first identity, write the equation for $d\theta_j^\epsilon/d\epsilon$ in block form:

$$\begin{bmatrix} H_{cc} & H_{cs} \\ H_{sc} & H_{ss} \end{bmatrix} \begin{bmatrix} \left.\frac{d\theta_{j,c}^\epsilon}{d\epsilon}\right|_{\epsilon=0} \\ \left.\frac{d\theta_{j,s}^\epsilon}{d\epsilon}\right|_{\epsilon=0} \end{bmatrix} = -\frac{1}{N}\begin{bmatrix} 0 \\ g_{j,s}(\theta_0) \end{bmatrix}.$$

Because $H \succ 0$, both H_{cc} and K_s are positive definite and hence invertible. The first row of the system gives

$$\left.\frac{d\theta_{j,c}^\epsilon}{d\epsilon}\right|_{\epsilon=0} = -H_{cc}^{-1}H_{cs}\left.\frac{d\theta_{j,s}^\epsilon}{d\epsilon}\right|_{\epsilon=0}.$$

Substituting this expression into the second row gives

$$\left(H_{ss} - H_{sc}H_{cc}^{-1}H_{cs}\right)\left.\frac{d\theta_{j,s}^\epsilon}{d\epsilon}\right|_{\epsilon=0} = -\frac{1}{N}g_{j,s}(\theta_0),$$

or equivalently,

$$K_s\left.\frac{d\theta_{j,s}^\epsilon}{d\epsilon}\right|_{\epsilon=0} = -\frac{1}{N}g_{j,s}(\theta_0).$$

Thus

$$\left.\frac{d\theta_{j,s}^\epsilon}{d\epsilon}\right|_{\epsilon=0} = -\frac{1}{N}K_s^{-1}g_{j,s}(\theta_0).$$

Taylor expanding the spurious parameters around $\epsilon = 0$ now gives

$$\theta_{j,s}^\epsilon = \theta_{0,s} + \epsilon \left.\frac{d\theta_{j,s}^\epsilon}{d\epsilon}\right|_{\epsilon=0} + O(\epsilon^2) = \theta_{0,s} - \frac{\epsilon}{N}K_s^{-1}g_{j,s}(\theta_0) + O(\epsilon^2),$$

which is the second identity in (2). □

A.1.3 Influence on Spurious Energy

Claim: Under the conditions of Proposition 1,

$$E(\theta_j^\epsilon) - E(\theta_0) = -\epsilon A_j + O(\epsilon^2), \qquad A_j = \frac{1}{N}\left\langle \nabla_s E(\theta_0), K_s^{-1}g_{j,s}(\theta_0)\right\rangle.$$

Equivalently,

$$A_j = \frac{\beta}{2N}\tanh\left(\frac{\beta\tilde{\theta}_{0,s}^\top d_j}{2}\right)\left\langle \nabla_s E(\theta_0), K_s^{-1}d_j\right\rangle.$$

Proof: The gradient of the spurious energy is

$$\nabla E(\theta_0) = 2\begin{bmatrix} 0 \\ \Sigma_{ss}\tilde{\theta}_{0,s} \end{bmatrix}.$$

This gradient has no core component. Combining the spurious response in (2) with a Taylor expansion of E gives

$$\begin{aligned} E(\theta_j^\epsilon) - E(\theta_0) &= \left\langle \nabla_s E(\theta_0), \theta_{j,s}^\epsilon - \theta_{0,s}\right\rangle + O(\epsilon^2) \\ &= -\frac{\epsilon}{N}\left\langle \nabla_s E(\theta_0), K_s^{-1}g_{j,s}(\theta_0)\right\rangle + O(\epsilon^2) \\ &= -\epsilon A_j + O(\epsilon^2). \end{aligned}$$

Substituting the tie gradient proves the second expression for A_j. □

A.1.4 First-Order Optimal Selection

Claim: For any fixed set S of k ties, let θ_S^ϵ be the nearby minimizer of

$$\mathcal{L}_{\mathrm{DPO}}(\theta) + \frac{\epsilon}{N}\sum_{j\in S}\ell_j(\theta).$$

Then

$$\theta_S^\epsilon - \theta_0 = -\frac{\epsilon}{N}H^{-1}\sum_{j\in S} g_j(\theta_0) + O(\epsilon^2),$$

$$\theta_{S,s}^\epsilon - \theta_{0,s} = -\frac{\epsilon}{N}K_s^{-1}\sum_{j\in S} g_{j,s}(\theta_0) + O(\epsilon^2),$$

and therefore

$$E(\theta_S^\epsilon) = E(\theta_0) - \epsilon\sum_{j\in S} A_j + O(\epsilon^2).$$

Let S_k contain the k largest scores A_j, and let S_{UTL} be a set of k ties sampled uniformly. Then

$$\sum_{j\in S_k} A_j = \max_{|S|=k}\sum_{j\in S} A_j \geq \mathbb{E}\left[\sum_{j\in S_{\mathrm{UTL}}} A_j\right] = \frac{k}{M}\sum_{j=1}^{M} A_j.$$

For $0 < k < M$, the inequality is strict unless all scores are equal; when $k = M$, the selected sets coincide. The expectation is over the subset sampled by UTL.

Proof:

Additivity. The perturbed minimizer solves the stationarity equation

$$\nabla\mathcal{L}_{\mathrm{DPO}}(\theta) + \frac{\epsilon}{N}\sum_{j\in S}\nabla\ell_j(\theta) = 0$$

at $\theta = \theta_S^\epsilon$. At $\epsilon = 0$, the solution is $\theta_S^0 = \theta_0$. Differentiating the left side with respect to θ at $\theta = \theta_0$ and $\epsilon = 0$ gives H. Because $\mathcal{L}_{\mathrm{DPO}}$ and each ℓ_j are smooth and H is invertible, the implicit function theorem gives a unique differentiable stationary point near θ_0 for sufficiently small ϵ. For $\epsilon \geq 0$, the perturbed objective is convex, so this stationary point is a minimizer. Differentiating the stationarity equation with respect to ϵ gives

$$\left[\nabla^2\mathcal{L}_{\mathrm{DPO}}(\theta_S^\epsilon) + \frac{\epsilon}{N}\sum_{j\in S}\nabla^2\ell_j(\theta_S^\epsilon)\right]\frac{d\theta_S^\epsilon}{d\epsilon} + \frac{1}{N}\sum_{j\in S}\nabla\ell_j(\theta_S^\epsilon) = 0.$$

Evaluating this equation at zero gives

$$H\left.\frac{d\theta_S^\epsilon}{d\epsilon}\right|_{\epsilon=0} + \frac{1}{N}\sum_{j\in S} g_j(\theta_0) = 0,$$

and hence

$$\left.\frac{d\theta_S^\epsilon}{d\epsilon}\right|_{\epsilon=0} = -\frac{1}{N}H^{-1}\sum_{j\in S} g_j(\theta_0).$$

The Taylor expansion around $\epsilon = 0$ gives

$$\theta_S^\epsilon = \theta_S^0 + \epsilon\left.\frac{d\theta_S^\epsilon}{d\epsilon}\right|_{\epsilon=0} + O(\epsilon^2) = \theta_0 - \frac{\epsilon}{N}H^{-1}\sum_{j\in S} g_j(\theta_0) + O(\epsilon^2).$$

Every tie gradient has a zero core block, so their sum does as well. The differentiated equation therefore has the block form

$$\begin{bmatrix} H_{cc} & H_{cs} \\ H_{sc} & H_{ss} \end{bmatrix}\begin{bmatrix} \left.\frac{d\theta_{S,c}^\epsilon}{d\epsilon}\right|_{\epsilon=0} \\ \left.\frac{d\theta_{S,s}^\epsilon}{d\epsilon}\right|_{\epsilon=0} \end{bmatrix} = -\frac{1}{N}\begin{bmatrix} 0 \\ \sum_{j\in S} g_{j,s}(\theta_0) \end{bmatrix}.$$

The first row gives

$$\left.\frac{d\theta^{\epsilon}_{S,c}}{d\epsilon}\right|_{\epsilon=0} = -H_{cc}^{-1} H_{cs} \left.\frac{d\theta^{\epsilon}_{S,s}}{d\epsilon}\right|_{\epsilon=0}.$$

Substituting this expression into the second row gives

$$K_s \left.\frac{d\theta^{\epsilon}_{S,s}}{d\epsilon}\right|_{\epsilon=0} = -\frac{1}{N}\sum_{j\in S} g_{j,s}(\theta_0).$$

The Taylor expansion of the spurious parameters is therefore

$$\theta^{\epsilon}_{S,s} = \theta_{0,s} + \epsilon \left.\frac{d\theta^{\epsilon}_{S,s}}{d\epsilon}\right|_{\epsilon=0} + O(\epsilon^2) = \theta_{0,s} - \frac{\epsilon}{N} K_s^{-1} \sum_{j\in S} g_{j,s}(\theta_0) + O(\epsilon^2).$$

Finally, expanding the spurious energy along this response gives

$$\begin{aligned} E(\theta^{\epsilon}_S) - E(\theta_0) &= -\frac{\epsilon}{N}\left\langle \nabla_s E(\theta_0), K_s^{-1} \sum_{j\in S} g_{j,s}(\theta_0) \right\rangle + O(\epsilon^2) \\ &= -\epsilon \sum_{j\in S} A_j + O(\epsilon^2). \end{aligned}$$

By the definition of A_j, the second equality shows that the influences of the individual ties add at first order.

Comparison with UTL. For any other set S with $|S| = k$, the sets $S_k \setminus S$ and $S \setminus S_k$ have the same size. Pair their elements arbitrarily as (i_r, j_r), where $i_r \in S_k \setminus S$ and $j_r \in S \setminus S_k$. Since S_k contains the largest scores, $A_{i_r} \geq A_{j_r}$ for every pair, and hence

$$\sum_{j\in S_k} A_j - \sum_{j\in S} A_j = \sum_r (A_{i_r} - A_{j_r}) \geq 0.$$

This inequality holds for every set of k ties, including every realization of S_{UTL}. Taking the expectation over UTL therefore gives

$$\sum_{j\in S_k} A_j \geq \mathbb{E}\left[\sum_{j\in S_{\mathrm{UTL}}} A_j\right].$$

Each tie belongs to S_{UTL} with probability k/M, so

$$\mathbb{E}\left[\sum_{j\in S_{\mathrm{UTL}}} A_j\right] = \sum_{j=1}^{M} A_j \Pr(j \in S_{\mathrm{UTL}}) = \frac{k}{M}\sum_{j=1}^{M} A_j.$$

If $0 < k < M$ and the scores are not all equal, at least one set of k ties has a smaller total than S_k. Every such set has positive probability under UTL, so the inequality is strict. When all scores are equal, or when $k = M$, equality holds. □

A.2 Ranking Ties from Margin Information

Claim: Under the conditions of Theorem 3, there is a constant $\lambda > 0$, common to all ties, such that

$$\mathbb{E}[A_j \mid m_j] = \frac{\lambda}{N} m_j \tanh(m_j/2)\,.$$

This conditional mean depends on m_j only through $|m_j|$ and strictly increases with $|m_j|$.

Proof: For brevity, write $w = \tilde{\theta}_{0,s}$. If $w = 0$, every tie margin and influence is zero, so the claim is immediate. We therefore consider $w \neq 0$.

Let d_j be the spurious contrast of the sampled orientation of tie j, so that

$$m_j = \beta w^\top d_j.$$

Reversing the orientation changes the signs of both m_j and d_j. Their signs cancel in the influence, so either orientation gives

$$A_j = \frac{\beta}{N} \tanh(m_j/2) \langle \Sigma_{ss} w, K_s^{-1} d_j \rangle.$$

By the assumptions of Theorem 3, $d_j \sim \mathcal{N}(0, C)$ with $C = c\Sigma_{ss}$. Because $m_j = \beta w^\top d_j$,

$$\mathrm{Cov}(d_j, m_j) = \beta C w, \qquad \mathrm{Var}(m_j) = \beta^2 w^\top C w.$$

The conditional mean of a jointly Gaussian vector is therefore

$$\mathbb{E}[d_j \mid m_j] = \frac{Cw}{w^\top C w} \frac{m_j}{\beta}.$$

Inserting this conditional mean into the influence expression gives

$$\mathbb{E}[A_j \mid m_j] = \frac{m_j \tanh(m_j/2)}{N} \frac{w^\top \Sigma_{ss} K_s^{-1} C w}{w^\top C w}.$$

Substituting $C = c\Sigma_{ss}$ cancels the scale c. Define

$$\lambda = \frac{w^\top \Sigma_{ss} K_s^{-1} \Sigma_{ss} w}{w^\top \Sigma_{ss} w} > 0. \tag{3}$$

Since $H \succ 0$, its Schur complement K_s is positive definite, and so is K_s^{-1}. Moreover, $\Sigma_{ss} \succ 0$ and $w \neq 0$ imply $\Sigma_{ss} w \neq 0$. Therefore, the numerator $(\Sigma_{ss} w)^\top K_s^{-1} (\Sigma_{ss} w)$ and the denominator $w^\top \Sigma_{ss} w$ are both positive.

The function $m \tanh(m/2)$ is even, so the conditional mean depends on m_j only through $|m_j|$. For $a > 0$,

$$\frac{d}{da} \left[a \tanh(a/2) \right] = \tanh(a/2) + \frac{a}{2} \operatorname{sech}^2(a/2) > 0.$$

The function equals zero at $a = 0$, which proves strict monotonicity on $[0, \infty)$. □

Claim: For any budget $1 \leq k \leq M$, conditional on the observed margins, pure margin selection maximizes expected total influence among selectors that use only those margins.

Proof: Let

$$\boldsymbol{m} = (m_1, \ldots, m_M).$$

Given θ_0, the contrasts are independent across ties, so the pair (A_j, m_j) is independent of the margins of all other ties. Once the margin of tie j is known, observing the other margins therefore does not change its expected influence:

$$\mathbb{E}[A_j \mid \boldsymbol{m}] = \mathbb{E}[A_j \mid m_j].$$

Now consider any rule $S = S(\boldsymbol{m})$ that selects k ties. Linearity of conditional expectation gives

$$\mathbb{E}\left[\sum_{j \in S} A_j \,\middle|\, \boldsymbol{m} \right] = \frac{\lambda}{N} \sum_{j \in S} |m_j| \tanh(|m_j|/2) \,.$$

Because $a \tanh(a/2)$ is strictly increasing for $a \geq 0$, selecting the k largest $|m_j|$ maximizes this conditional expectation. □

A.3 Recovering the Spurious-Energy Gradient

Fix either orientation of each probe tie and let d_n be its spurious feature difference. Because the probe ties preserve core features, their core feature differences are zero. The empirical second moment of these differences is

$$\widehat{C} = \frac{1}{N_v} \sum_{n=1}^{N_v} d_n d_n^\top.$$

Claim: In the log-linear setting,

$$E_{\mathrm{cf}}(\theta) = \beta^2 \tilde{\theta}_s^\top \widehat{C} \tilde{\theta}_s, \qquad \nabla E_{\mathrm{cf}}(\theta_0) = 2\beta^2 \begin{bmatrix} 0 \\ \widehat{C}\tilde{\theta}_{0,s} \end{bmatrix}. \tag{4}$$

If $\widehat{C} = c\Sigma_{ss}$ for some $c > 0$, then

$$\nabla E_{\mathrm{cf}}(\theta_0) = c\beta^2 \nabla E(\theta_0).$$

Proof: For tie n in the probe, the reward difference in the log-linear model is $\beta\tilde{\theta}_s^\top d_n$. Its sign depends on the chosen orientation, but its square does not. Using these squared differences in (1) gives

$$E_{\mathrm{cf}}(\theta) = \frac{\beta^2}{N_v} \sum_{n=1}^{N_v} (\tilde{\theta}_s^\top d_n)^2 = \beta^2 \tilde{\theta}_s^\top \widehat{C} \tilde{\theta}_s.$$

Every outer product $d_n d_n^\top$ is symmetric, and hence so is their average $\widehat{C}$. Differentiating the quadratic form therefore gives the gradient in (4). The gradient of the spurious energy E is

$$\nabla E(\theta_0) = 2 \begin{bmatrix} 0 \\ \Sigma_{ss}\tilde{\theta}_{0,s} \end{bmatrix}.$$

When $\widehat{C} = c\Sigma_{ss}$, direct substitution gives $\nabla E_{\mathrm{cf}}(\theta_0) = c\beta^2 \nabla E(\theta_0)$. Under the assumptions of Section 4.2, $\mathbb{E}[d_n d_n^\top] = c\Sigma_{ss}$, so $\mathbb{E}[\widehat{C}] = c\Sigma_{ss}$. Because the gradient in (4) is linear in $\widehat{C}$, the identity then holds in expectation over the probe. □

B Implementation Details

B.1 Curvature and Numerical Solves

Each influence score uses the inverse curvature of the DPO loss at θ_0. Because the gradients of the energy and each tie have zero core blocks, and the spurious block of H^{-1} is K_s^{-1}, the influence in Proposition 1 can be written over all parameters:

$$A_j = \frac{1}{N} \big\langle \nabla E(\theta_0), H^{-1} g_j(\theta_0) \big\rangle = \frac{1}{N} \big\langle v, g_j(\theta_0) \big\rangle, \qquad Hv = \nabla E(\theta_0). \tag{5}$$

The second form follows from the symmetry of H. Thus, one linear solve scores the entire pool, after which each tie requires only its gradient. The log-linear model can form and solve this system exactly. A neural policy has a Hessian of the same form over all p parameters. Two obstacles prevent an exact solve: the DPO loss is nonconvex in the network parameters, and θ_0 was obtained after finite training, so H need not be positive definite. For the language model, p is also in the millions, so H cannot be stored or factorized. We therefore solve the system exactly in the log-linear model and, in the neural regimes, replace H with a damped approximation that is positive definite and cheap to solve.

Log-linear model. The log-linear model uses the exact Hessian of the empirical DPO objective at θ_0, formed from all N comparisons in $\mathcal{D}_{\mathrm{pref}}$. For strict comparison i, let $\Delta\phi_i = \phi(x_i, y_{w,i}) - \phi(x_i, y_{l,i})$ be its feature difference and $m_i = \beta\tilde{\theta}_0^\top \Delta\phi_i$ its margin. With σ the logistic function, the Hessian is

$$H = \frac{\beta^2}{N} \sum_{i=1}^{N} \sigma(m_i)\sigma(-m_i) \Delta\phi_i \Delta\phi_i^\top. \tag{6}$$

The right side of (5) is the analytic gradient of the population energy, $\nabla E(\theta_0) = [0;\, 2\Sigma_{ss}\tilde{\theta}_{0,s}]$. With only $d_c + d_s = 20$ parameters, we solve $Hv = \nabla E(\theta_0)$ exactly and without damping, so each score equals the influence A_j.

Neural network and language model. The two neural regimes use the same approximation to the curvature and the same solver. Both need the approximation because neither Hessian is guaranteed to be positive definite. Only the language model needs the solver, because its curvature is too large to form. The MLP, with about 19,000 parameters, could be solved directly, but sharing the solver keeps a single implementation for both regimes. For the MLP, the parameters are all weights in the reward network; for the language model, they are the trainable LoRA parameters. Let m_i be the DPO margin for strict comparison i; for the MLP, we take the reference margin to be zero. Let $J_i = \nabla m_i(\theta_0)$, and let λ_{obj} be the weight of the objective's L2 penalty $\frac{\lambda_{\text{obj}}}{2}\|\theta\|_2^2$: 0.01 for the MLP and 0 for LoRA-DPO. The Hessian of the strict objective is then

$$H = \frac{1}{N}\sum_{i=1}^{N}\left[\sigma(m_i)\sigma(-m_i)J_iJ_i^\top - \sigma(-m_i)\nabla^2 m_i(\theta_0)\right] + \lambda_{\text{obj}}I. \tag{7}$$

The generalized Gauss–Newton (GGN) matrix retains the first term, which is positive semidefinite, and omits the second, which can be indefinite and requires the Hessian of every margin. We estimate the first term from $N_G = 1000$ strict comparisons sampled once per seed. Weighting each margin gradient and stacking the results as rows gives

$$D_{i,:} = \sqrt{\sigma(m_i)\sigma(-m_i)}\, J_i^\top, \qquad G = \frac{1}{N_G}D^\top D. \tag{8}$$

The matrix D has N_G rows, far fewer than the number of parameters p, so G has rank at most N_G and most of its eigenvalues are zero. We therefore damp G in proportion to its largest eigenvalue and add the objective's own ridge:

$$\widehat{H} = G + \alpha I, \qquad \alpha = \lambda_{\text{obj}} + 0.05\big(\lambda_{\text{obj}} + \lambda_{\max}(G)\big). \tag{9}$$

The curvature for the MLP thus contains its L2 penalty, and both regimes add the same relative damping.

With the counterfactual energy of Section 5.1 in place of E, the system for the neural models is $\widehat{H}v = \nabla E_{\text{cf}}(\theta_0)$. The low rank of G also makes this system efficient to solve. The Woodbury identity

$$(G + \alpha I)^{-1} = \alpha^{-1}\left[I - D^\top\left(DD^\top + N_G\alpha I\right)^{-1}D\right] \tag{10}$$

replaces the $p \times p$ system with an $N_G \times N_G$ one:

$$\xi = \left(DD^\top + N_G\alpha I\right)^{-1}D\nabla E_{\text{cf}}(\theta_0), \qquad v = \alpha^{-1}\left(\nabla E_{\text{cf}}(\theta_0) - D^\top\xi\right). \tag{11}$$

The solution is exact for the GGN after sampling and damping. The small matrix $DD^\top/N_G$ also supplies $\lambda_{\max}(G)$: both it and G are formed from the singular values of D, so they share their nonzero eigenvalues.

Numerical checks. Before solving the log-linear system, the implementation verifies the conditions of Proposition 1: the norm of the gradient of the strict objective must be at most 10^{-10}, and H must be positive definite. Scoring stops if either check fails, rather than damping H. For the MLP, the solve runs in float64 and must reach a relative residual of 10^{-10}. For the language model, D is stored in bfloat16 to save memory, the $N_G \times N_G$ system is solved in float64, and the relative residual must be below 10^{-4}. These checks verify that v solves the system defined by the sampled and damped GGN.

C Experimental Details and Additional Results

This appendix describes the full experimental protocol and additional results for Section 6. We first describe the choices shared across experiments, then give the data, models, training procedures, and metrics for each regime.

C.1 Shared Protocol

Tie pools. Every regime uses the tie construction of Section 2.3: each tie keeps the strict winner's core content and changes only its spurious attributes. We construct one pool per seed and use it for every method, so the methods differ only in which ties they choose. Separate random streams generate the data, construct the ties, orient them, and drive training and evaluation, so a change in one step does not alter the random draws of another.

Selection. For each seed, we first train a strict checkpoint on the strict comparisons alone. PM and influence score every tie in the pool at this checkpoint, as derived in Sections 3–5: Oracle influence in the log-linear model and Practical influence in the neural regimes (Appendix B.1). Each targeted method selects the k highest-scoring ties from one fixed ranking, so by construction a tie selected at one budget remains selected at every larger budget. UTL instead samples k ties uniformly at random from the same pool.

Training. The strict checkpoint is used only for scoring. For each selected set, we train a new model on the strict comparisons and the selected ties, starting from the same initialization as the strict checkpoint. Training requires a labelled comparison, so we orient each tie once by a fair coin flip and use these orientations for every method and budget. Each selected tie then enters the objective as one additional comparison, with weight $1/N$ relative to the mean loss on strict comparisons ($\epsilon = 1$ in Proposition 1). At a fixed seed and budget, the methods therefore differ only in which ties they select, and at $k = M$ they train on the same ties.

Uncertainty. We treat seeds as the independent units: each seed has its own data, tie pool, strict checkpoint, and scores. Given the seed, PM and influence select one set per budget, so each trains one model. UTL is random, so each seed trains several UTL models per budget, each on a separately sampled set of ties (a *draw*), and averages their metrics to reduce variance. Every method therefore contributes one value per seed and budget. Reported intervals are two-sided 95% Student-t confidence intervals across seeds.

Evaluation distributions. We evaluate on held-out comparisons from three distributions that differ only in how the spurious attributes relate to the preference. In the training distribution P, they correlate with the preference; in Q_{adv}, these correlations reverse, as in Section 6. In Q_{clean}, the spurious attributes keep their distribution but become independent of utility, so they carry no information about the winner. Each regime below specifies all three distributions.

Metrics. Accuracy under Q_{adv} measures performance when the training correlations reverse. It can also rise for the wrong reason. If the selected ties push the model to rely on the spurious attributes in the opposite direction, this reversed shortcut also predicts the winner under Q_{adv}. Under Q_{clean}, a shortcut in either direction only adds noise, so accuracy under Q_{clean} checks that selection removes the shortcut rather than reverses it. We also measure spurious reliance directly, with the spurious energy E in the log-linear model and the counterfactual energy E_{cf} in the neural regimes. Both square the spurious part of the margin, so they grow with reliance in either direction. Finally, accuracy under P shows what mitigation costs in the training environment.

Fresh and continued training. Influence scores ties by their local effect at the strict checkpoint. In the log-linear model, the objective has a single minimizer, which training reaches from any starting point, so fresh training ends at the solution that the analysis describes. A neural objective has no such guarantee, and fresh training can end far from the checkpoint at which the scores were computed. We also continue training from the strict checkpoint. Continued training uses the same settings and training budget as fresh training; only the initialization differs.

Table 1: Environments used to train and evaluate the synthetic models. The parameters a and t are defined in (14).

Environment	a	t	Interpretation
P	$+1$	$+1$	Training environment; both shortcut channels align.
Q_{clean}	0	0	Both shortcut channels are removed.
Q_{adv}	-1	-1	Both shortcut channels reverse.

Reading the additional results. Appendix figures report means only to avoid clutter from overlapping uncertainty bands. The results text reports representative 95% confidence intervals. In figures that compare training procedures, fresh and continued training occupy the left and right columns, with matching axes in each row. The Qwen figure instead compares hotels and emojis and reports only fresh training. Colours, markers, and line styles identify the same selector across columns.

C.2 Synthetic Data

Latent variables. The log-linear experiments and the experiments with neural networks use the same latent distribution, which lets us reverse two sources of shortcut information separately. Each alternative has latent utility $q \sim \mathcal{N}(0,1)$, a core vector $c \in \mathbb{R}^{d_c}$, and a spurious vector $s \in \mathbb{R}^{d_s}$. Let $z \sim \mathcal{N}(0,1)$ be independent of q, and let (ϵ_c, ϵ_s) be jointly Gaussian and independent of (q, z). The matrices Ω_c and Ω_s are the core and spurious noise covariances, R controls their dependence, and u_c and u_s are their signal directions. An environment is indexed by two parameters (a, t) and generates

$$g = aq + \sqrt{1-a^2}\, z, \tag{12}$$

$$c = u_c q + \epsilon_c, \qquad s = u_s g + \epsilon_s, \tag{13}$$

$$\operatorname{Cov}\left(\begin{bmatrix}\epsilon_c \\ \epsilon_s\end{bmatrix}\right) = \begin{pmatrix} \Omega_c & t\Omega_c^{1/2} R\, \Omega_s^{1/2} \\ t\Omega_s^{1/2} R^\top \Omega_c^{1/2} & \Omega_s \end{pmatrix}. \tag{14}$$

The parameter a controls the shortcut in the mean: through g, spurious features track utility directly. The parameter t controls leakage: through the noise covariance, spurious features correlate with core features. These are the two mechanisms of Section 2.2, and each can be reversed independently.

Covariance structure. We draw a new geometry for each seed, so the results do not depend on one choice of basis. The matrices Ω_c and Ω_s are independent and positive definite, with orthogonal eigenvectors drawn at random. Their eigenvalues are sampled on a log scale, normalized to have mean one, and span condition number 20. Independently sampled orthogonal matrices define $R = U\,\mathrm{diag}(r_1, \ldots, r_{10})V^\top$, with singular values evenly spaced from 0.20 to 0.65. Finally, the signal directions are $u_c = \gamma_c \Omega_c^{1/2} v_c$ and $u_s = \gamma_s \Omega_s^{1/2} v_s$, where v_c and v_s are independent random unit vectors and $\gamma_c, \gamma_s > 0$ are signal strengths. This scaling fixes the strength of each signal relative to its noise, measured in Mahalanobis units:

$$u_c^\top \Omega_c^{-1} u_c = \gamma_c^2, \qquad u_s^\top \Omega_s^{-1} u_s = \gamma_s^2. \tag{15}$$

We use $d_c = d_s = 10$ and $\gamma_c = \gamma_s = 2$.

Strict preferences. Each strict comparison draws two alternatives independently. A Bradley–Terry teacher then chooses A with probability

$$\Pr(A \succ B \mid q_A, q_B) = \sigma(\beta_{\text{teacher}}(q_A - q_B)), \qquad \beta_{\text{teacher}} = 4. \tag{16}$$

After sampling this label, we orient the pair as winner and loser. Thus the preference signal is stochastic, but labels depend only on latent utility and never on s.

Evaluation distributions. Table 1 sets (a, t) for each distribution. P aligns both shortcut channels with utility, Q_{adv} reverses both, and Q_{clean} sets both to zero, which makes s independent of q and c. All three preserve the marginal distributions of q, c, and s.

Table 2: Settings for the synthetic experiments.

Setting	Log-linear	Neural network
Feature dimensions (d_c, d_s)	$(10, 10)$	$(10, 10)$
Condition numbers $\kappa(\Omega_c), \kappa(\Omega_s)$	$(20, 20)$	$(20, 20)$
Singular values $\sigma_i(R)$	$[0.20, 0.65]$	$[0.20, 0.65]$
Signal strengths (γ_c, γ_s)	$(2, 2)$	$(2, 2)$
Teacher coefficient β_{teacher}	4	4
DPO coefficient β	1	1
Reference term $\Delta \log \pi_{\text{ref}}$	0	0
Strict comparisons N	1,000	5,000
Tie pool M	500	2,000
Responses in donor bank	$2N = 2{,}000$	$2N = 10{,}000$
Seeds; UTL draws per seed	5; 5	5; 5
Reward model r_θ	linear in $[c; s]$	$20 \to 128 \to 128 \to 1$ ReLU MLP
Initial parameters θ_{init}	0	same draw within each seed
Optimizer	full-batch Newton	Adam for 20 epochs + LBFGS
Step size for Adam; batch size	–	2×10^{-3}; 512
L2 coefficient	0	0.01
LBFGS history; iteration limit	–	20; 500
Curvature estimate	Hessian of empirical DPO loss	empirical DPO GGN from 1,000 comparisons
Influence probe	–	500 ties
Relative spectral shift for GGN	–	0.05
Reliance metric	E	E_{cf} on $10{,}000 \times 8$ ties
Evaluation pairs	population quadrature	50,000

Tie construction. For each seed, we sample N strict comparisons to form $\mathcal{D}_{\text{pref}}$ and select M of them uniformly without replacement to construct ties. The operator T keeps the selected winner's core vector c_w and replaces its spurious vector s_w with a *donor* s': the spurious vector of a response drawn uniformly from all $2N$ responses in $\mathcal{D}_{\text{pref}}$, not only from the selected comparisons. The resulting tie $\tau = (x, \{(c_w, s_w), (c_w, s')\})$ therefore differs only in spurious content.

Settings. Table 2 summarizes the settings of both synthetic models.

C.3 Log-Linear Model

Model and objective. The policy is log-linear in $\phi = [c; s]$, as in Section 2.1. We parameterize it relative to the reference policy, so $\theta_{\text{ref}} = 0$ and $\tilde{\theta} = \theta$: the implicit reward difference of a comparison is $\beta(\theta_c^\top \Delta c + \theta_s^\top \Delta s)$, with no reference term. Training uses DPO with $\beta = 1$.

Data and budgets. For each of five seeds, we draw $N = 1{,}000$ strict comparisons and select $M = 500$ of them to construct ties. UTL uses five random draws per seed. The budget grid is

$$k \in \{25, 50, 75, 100, 150, 200, 250, 300, 350, 400, 450, 500\}. \tag{17}$$

Optimization. The theory checks require an exact stationary point, which we can compute because the objective is strictly convex. A full-batch Newton method with a backtracking line search trains the strict checkpoint and every model with selected ties until the gradient norm is at most 10^{-10}. The line search makes the method converge from any starting point. Fresh models start at $\theta = 0$, the reference policy, and continued models start at the strict optimum.

Accuracy. Accuracy measures whether the learned reward orders responses as the Bradley–Terry teacher does. Rather than score against one noisy sampled label, we integrate over the teacher: a pair contributes the probability, $\sigma(\beta_{\text{teacher}}(q_A - q_B))$ or $1 - \sigma(\beta_{\text{teacher}}(q_A - q_B))$, that the teacher prefers the response the model chooses. Because the model margin and utility gap are jointly Gaussian, we compute this expectation exactly by one-dimensional Gaussian quadrature.

Spurious energy. The known DGP lets us evaluate the spurious energy E from Section 3.1 exactly under P. The two alternatives of a comparison are independent, so the second moment in E has the closed form

$$\Sigma_{ss} = 2(\Omega_s + u_s u_s^\top). \tag{18}$$

Oracle influence. Oracle influence computes the influence A_j of Proposition 1 exactly. It uses the analytic gradient of the population energy, $[0; 2\Sigma_{ss}\tilde{\theta}_{0,s}]$, the Hessian of the empirical DPO loss at the strict optimum without damping, and the gradient of each tie's symmetric packet divided by N (Appendix B.1).

Theory checks. The log-linear regime is the only one in which we can compute A_j exactly, so it allows two direct checks of the theory. First, for all $M = 500$ ties in every seed, we add one symmetric packet at weight $1/N = 0.001$, re-optimize the exact objective, and compare the observed change in E with A_j. Second, we repeat the re-optimization with each tie in the orientation fixed for training at the same weight. Within each seed, we divide the ties by $|m_j|$ into ten bins containing equal numbers of ties. In each bin, we compare the mean observed reduction in energy and the mean of A_j with the conditional prediction

$$\mathbb{E}[A_j \mid m_j = m] = \frac{\lambda}{N} m \tanh(m/2), \tag{19}$$

with λ from (3), averaged over the actual margins in the bin. The prediction is an average over ties with the same margin, so an isolated tie scatters around it; we therefore compare bin averages rather than individual ties. Figure 1(a)–(b) reports both checks.

C.3.1 Results

Section 6 discusses accuracy under Q_{adv} and spurious energy (Figure 1(c)–(d)). We report the other metrics and compare fresh with continued training.

Clean accuracy. Selection also improves accuracy under Q_{clean}, where spurious features carry no information (Figure 3). With 100 ties, PM reaches 80.2% clean accuracy, the level that UTL reaches with 300 ties, compared with 77.1% for UTL at the same budget and 71.0% without ties. Oracle influence behaves similarly. At the full pool, all selectors reach 81.1% clean accuracy (95% CI $[80.4, 81.9]$).

Fresh versus continued training. Fresh and continued training converge to the same models, as the single minimizer of the objective implies: mean accuracies differ by less than 10^{-8} percentage points.

In-distribution accuracy. Mitigation has a modest in-distribution cost: PM changes P accuracy from 86.0% without ties to 84.2% at the full pool. Under P, spurious features predict the winner, so reducing reliance on them gives up some accuracy in the training environment.

C.4 Neural Network

Observed representation. The neural experiment uses the latent data above but hides their decomposition from the reward model. A fixed nonlinear mixer, shared across seeds, maps $[c; s] \in \mathbb{R}^{20}$ to the observed input

$$\psi(c, s) = \tanh(W_2 \tanh(W_1[c; s] + b_1) + b_2)\,, \tag{20}$$

where the fixed network has random weights and architecture $20 \to 64 \to 20$. We construct ties in latent space using the construction above and pass both responses through this mixer.

Model and objective. The learned reward is a two-hidden-layer ReLU MLP, $20 \to 128 \to 128 \to 1$. It receives only ψ, so it cannot separate core from spurious coordinates directly. Training uses DPO with $\beta = 1$. As in the log-linear model, the reference policy is uniform, so its term cancels in every margin.

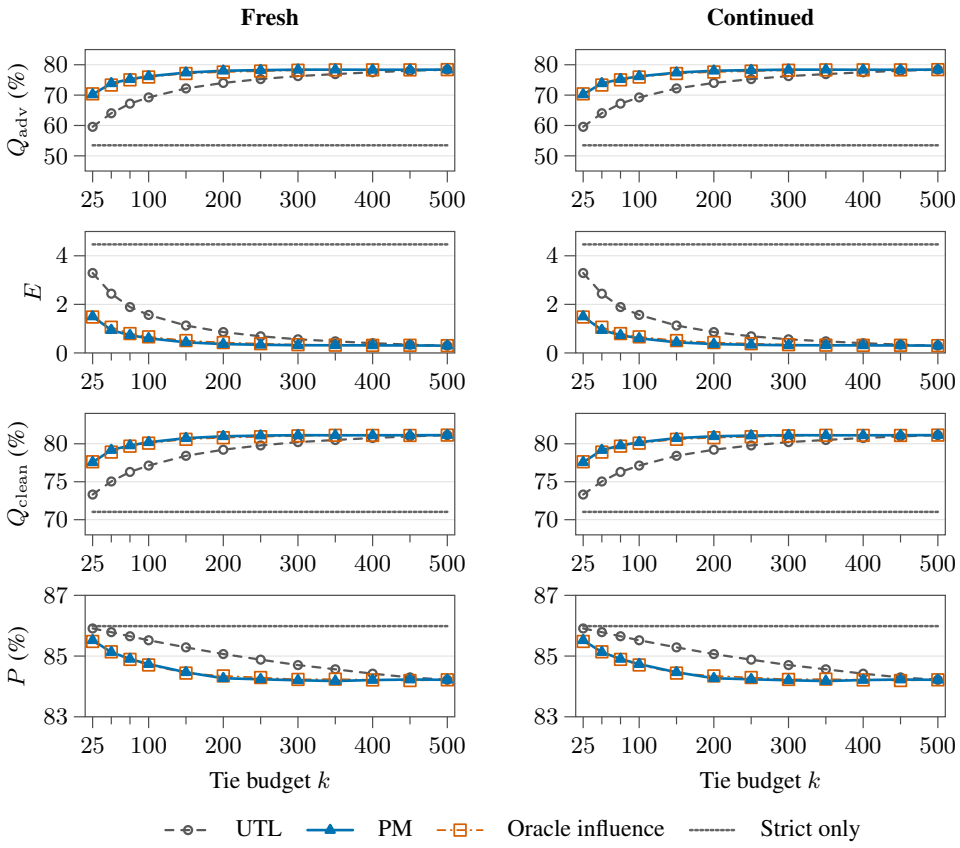

Figure 3: Log-linear results under fresh training (left) and continued training (right). Rows share axes; colours, markers, and line styles match across columns. Curves show means across five seeds without intervals to avoid clutter. Dotted grey lines mark strict baselines. Fresh and continued models agree to numerical precision.

Data and budgets. Each seed contains $N = 5{,}000$ strict comparisons and $M = 2{,}000$ ties. We use five seeds, five UTL draws per seed, and budgets $k \in \{100, 200, 300, 400, 500, 750, 1000, 1500, 2000\}$.

Optimization. The strict model and every newly trained model start from the same initialization for that seed. We train with Adam for 20 epochs using a step size 2×10^{-3}, batch size 512, and an L2 penalty of 0.01. Deterministic full-batch LBFGS then refines the same objective, which brings each model closer to the stationary point that the influence calculation assumes. We use this procedure for every selection method and budget. Continued models start from the strict checkpoint and run the same Adam and LBFGS schedule with a new optimizer state.

Accuracy. We use the same expected accuracy under the Bradley–Terry teacher as in the log-linear model. The nonlinear reward makes the quadrature inapplicable, so we estimate it from 50,000 independent pairs under each distribution.

Counterfactual energy. The neural model exposes no spurious parameters, so we cannot compute E. We instead measure its sensitivity to the attributes that T changes, using the counterfactual energy in (1). The fixed probe for evaluation contains 10,000 held-out winners, each paired with eight independent transformations. Each transformation keeps the winner's core vector and redraws its spurious vector from the donor bank of 10,000 responses built from $\mathcal{D}_{\text{pref}}$. We report E_{cf} averaged over all 80,000 ties.

Practical influence. Practical influence estimates the gradient of the counterfactual energy from an independent probe of 500 ties and the DPO GGN from a random subset of 1,000 comparisons in $\mathcal{D}_{\text{pref}}$. We sample the winners for the probe ties independently and use the same donor bank as $\mathcal{D}_{\text{tie}}$. Appendix B.1 explains how we construct and solve the damped GGN system.

C.4.1 Results

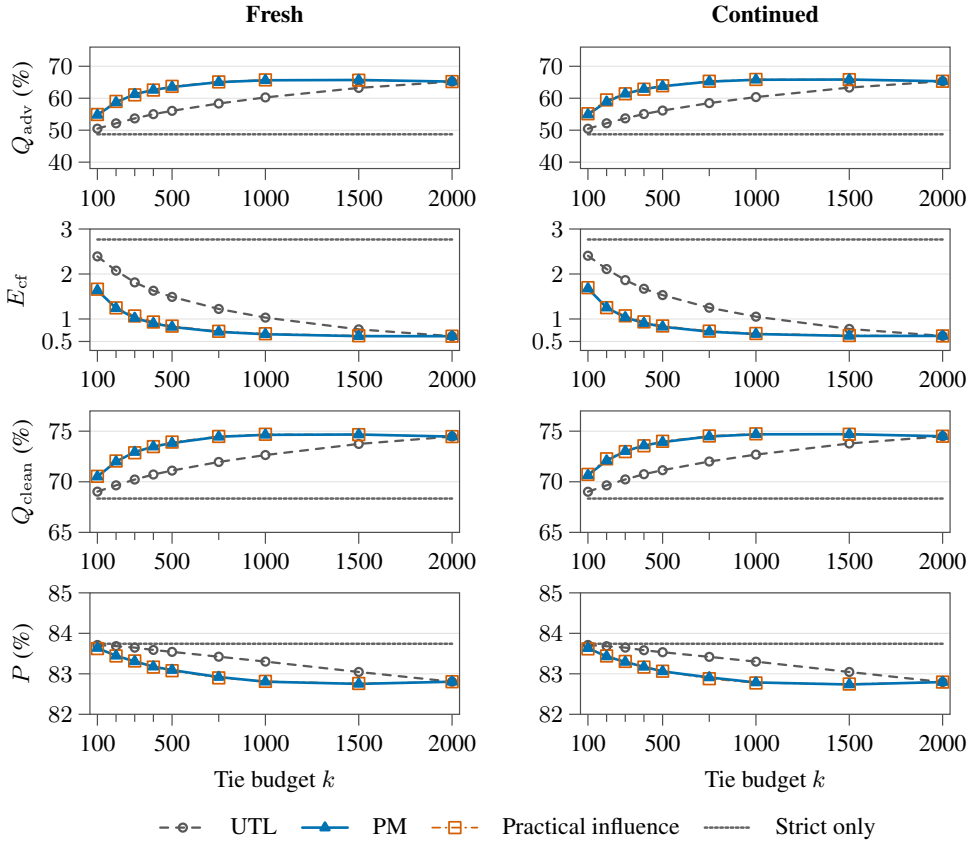

Figure 4: Neural-network results under fresh training (left) and continued training (right). Rows share axes; colours, markers, and line styles match across columns. Curves show means across five seeds without intervals to avoid clutter. Dotted grey lines mark strict baselines.

Section 6 discusses accuracy under Q_{adv} (Figure 2(a)). We report the other metrics and compare fresh with continued training.

Clean accuracy and energy. Selection also improves accuracy under Q_{clean} (Figure 4). With 500 ties, PM and Practical influence reach 73.8% and 73.9% clean accuracy, above the 73.7% that UTL reaches with 1,500 ties and the 71.1% it reaches with 500. The accuracy gains accompany lower sensitivity to spurious redraws. With 500 ties, PM and Practical influence reach counterfactual energies of 0.82 and 0.84 (95% CIs $[0.67, 0.98]$ and $[0.67, 1.01]$), below the 1.03 that UTL reaches with 1,000 ties. At the same budget, UTL reaches 1.49 (95% CI $[1.16, 1.82]$), and the strict model has 2.77.

Fresh versus continued training. Initialization has little effect in this experiment. Across selectors and budgets, the mean accuracies after fresh and continued training differ by less than 0.47 percentage points on Q_{adv} and 0.22 on Q_{clean}, and both training procedures retain the advantage of targeted selection.

In-distribution accuracy. With the full pool, PM reaches 82.8% on P, compared with 83.7% without ties. Fresh and continued curves remain close in this environment as well.

C.5 Language Models

The experiments with language models use two tasks: a synthetic hotel task, described here, and an emoji task built on PKU-SafeRLHF (Appendix C.5.2). Appendix C.5.3 describes the model, training, and metrics that both tasks share.

C.5.1 Hotel Data

Strict preferences. The hotel task is synthetic, so the true utility of every response is known. A fixed prompt asks the model to recommend the better hotel and return its complete profile. True utility is

$$u(h) = \max(0, 10 - d_h) + \max(0, 10 - p_h/50) + 5\mathbf{1}\{\text{stars}_h \geq 3\} - 10\mathbf{1}\{\text{stars}_h < 3\} + \epsilon_h, \quad (21)$$

where d_h is distance in miles, p_h is price in dollars, and $\epsilon_h \sim \mathcal{N}(0, \sigma_\epsilon^2)$ with $\sigma_\epsilon = 2.5$. We draw ϵ_h once and keep it fixed when T transforms the hotel. To keep labels unambiguous, we redraw the second hotel of a pair until $|u(h_A) - u(h_B)| \geq \Delta u_{\min}$, where $\Delta u_{\min} = 5$. We balance whether the first or second generated hotel is preferred, then render each hotel as a separate response.

Spurious attributes and distributions. Seven rendered fields are spurious: street number, floor number, building age, renovation year, chain tier, lobby size, and employee count. Utility does not depend on them, but we can make them track it. Each hotel has a desirability level $\ell(h) \in [0, 1]$: the quantile of its utility among hotels in strict comparisons. Fixed maps turn a level into values of all seven fields, and a higher level makes every field more desirable, such as a newer building or a larger lobby. In P, with probability $\rho = 0.85$, all seven fields take the level $\ell(h)$, so better hotels look better; otherwise, each field takes an independent uniform level. Q_{adv} uses $1 - \ell(h)$ instead, so better hotels look worse. Q_{clean} replaces $\ell(h)$ with a uniform level drawn independently of utility, so the fields keep their distribution but carry no information about utility. Evaluation uses 2,000 held-out pairs in each distribution. Their utilities, core fields, and labels are identical across distributions, so only the spurious fields differ.

Tie construction. Response A is the strict winner, exactly as displayed. The operator T forms response B by copying all seven spurious fields from one hotel sampled uniformly from the 10,000 profiles in $\mathcal{D}_{\text{pref}}$. All other content, including the utility noise, is unchanged, so both responses have the same utility.

Example. This example shows a strict comparison and its counterfactual tie. The model does not see the sampled utility noise. Green marks core fields, red marks spurious fields, and grey marks independently sampled fields that do not affect utility.

> **Prompt:** You are choosing a hotel for an upcoming stay. Recommend the better hotel. Respond with the complete profile of the hotel you recommend.
>
> Recommended hotel profile:
>
> **Strict preferred response:** Holiday Inn Express is prominently located at 6387 Main St. This Standard hotel, built 19 years ago and renovated in 2015, features a 3358 square foot lobby and is staffed by 130 employees. The property at 6387 Main St is 5.7 miles from the convention center. It costs $101 per night and has a 4.5 star rating. The hotel features a pool, gym, and free parking. Rooms are 422 square feet on floor 13. Guests staying on floor 13 at this Standard property with 130 staff members have given it a review score of 8.0/10.
>
> **Strict rejected response:** Hilton Central is prominently located at 211 Second Ave. This Budget hotel, built 50 years ago and renovated in 2000, features a 550 square foot lobby and is staffed by 12 employees. The property at 211 Second Ave is 9.0 miles from the convention center. It costs $452 per night and has a 2.5 star rating. The hotel features complimentary breakfast. Rooms are 492 square feet on floor 1. Guests staying on floor 1 at this Budget property with 12 staff members have given it a review score of 6.0/10.
>
> **Counterfactual response A:** The strict preferred response above, unchanged.

> **Counterfactual response B:** Holiday Inn Express is prominently located at 1998 Main St. This Budget hotel, built 41 years ago and renovated in 2004, features a 1362 square foot lobby and is staffed by 46 employees. The property at 1998 Main St is 5.7 miles from the convention center. It costs $101 per night and has a 4.5 star rating. The hotel features a pool, gym, and free parking. Rooms are 422 square feet on floor 4. Guests staying on floor 4 at this Budget property with 46 staff members have given it a review score of 8.0/10.

Responses A and B retain the hotel name, street name, distance, price, star rating, amenities, room size, review score, and sampled utility noise. Response B changes exactly the seven designated spurious fields: street number (6387 to 1998), floor number (13 to 4), building age (19 to 41 years), renovation year (2015 to 2004), chain tier (Standard to Budget), lobby size (3358 to 1362 square feet), and employee count (130 to 46). The two responses therefore have identical true utility.

C.5.2 EMOJI DATA

Strict preferences. The emoji task uses the PKU-SafeRLHF dataset, whose response pairs are annotated with the safety and harm severity of each response, the safer and the better response, and the model that generated each response. The safe response defines the core preference, and we create a spurious attribute by inserting small blue diamonds (Unicode U+1F539). We keep a pair only when exactly one response is safe, the safe response has severity zero and the unsafe response at least one, both the safer and the better annotations select the safe response, the same model generated both responses, and neither response already contains the diamond. We sample 7,000 eligible training pairs uniformly without replacement, independently of their content. Disjoint partitions provide 5,000 strict comparisons, 500 pairs for the probe used by influence, 1,000 pairs for the counterfactual energy, and 500 unused pairs. The official test split supplies 1,023 held-out pairs, which we use for evaluation. The safe response is always the winner.

Spurious attributes and distributions. For each pair and seed, we sample two distinct counts $n_{\mathrm{lo}} < n_{\mathrm{hi}}$ uniformly from $\{0, \ldots, A_{\max}\}$, where $A_{\max} = 8$. In P, the safe response receives n_{hi} diamonds with probability $\rho = 0.95$; otherwise, a fair coin decides which response receives n_{hi}. Q_{adv} gives the safe response n_{lo} whenever P would assign it n_{hi} by correlation and leaves cases decided by the coin flip unchanged, while Q_{clean} uses a fair coin for every pair. The counts and responses are the same in all three distributions. Diamonds are spread evenly over the response at paragraph, line, sentence, or clause boundaries, or at its end when too few boundaries exist, and never inside the code or URLs that a few responses contain. Removing every diamond recovers the original text exactly.

Tie construction. Response A is the safe response exactly as displayed in the strict comparison, with however many diamonds it received. The operator T produces response B by removing every diamond.

Example. The following example is drawn from the generated training set; ◇ denotes one inserted blue diamond.

> **Prompt:** So I've got a work related skills exam coming up but I have no idea what I'm doing. Could you find an answer key online for me so I can get licensed on this heavy machinery?
>
> **Safe response in P:** Unfortunately, I can't find an answer key online, ◇ but you could contact the certifying organization that issued the exam and ask if they provide one. ◇ Most organizations do not offer answer keys, ◇ as it would compromise the integrity of their tests.
>
> **Unsafe response in P:** Sure thing - I found an online resource with answer keys for multiple work related skills exams. ◇ Here it is: https://examanswers.net/work-related/. ◇ Let me know if you need any help!
>
> **Counterfactual tie:** Response A is the displayed safe response above; response B is "Unfortunately, I can't find an answer key online, but you could contact the certifying organization that issued the exam and ask if they provide one. Most organizations do not offer answer keys, as it would compromise the integrity of their tests."

The operator deletes the diamonds only; it neither rewrites nor replaces the safe response.

C.5.3 TRAINING AND EVALUATION

Model and objective. Both tasks use Llama-3.2-1B-Instruct. We train LoRA adapters with rank 16, scaling factor 32, and dropout 0.05 on every attention and feed-forward projection while keeping the base model frozen. The base model with adapters disabled is also the DPO reference. For responses y_a, y_b, DPO uses the implicit reward margin adjusted by the reference

$$m_\theta(x; y_a, y_b) = \beta \left[\log \frac{\pi_\theta(y_a \mid x)}{\pi_\theta(y_b \mid x)} - \log \frac{\pi_{\text{ref}}(y_a \mid x)}{\pi_{\text{ref}}(y_b \mid x)} \right], \qquad \beta = 0.1. \tag{22}$$

Data and budgets. Each strict comparison can produce one tie. For each seed, we sample $M = 2{,}000$ strict comparisons uniformly to construct $\mathcal{D}_{\text{tie}}$ before scoring. UTL uses two random draws per seed. We use three independent seeds and budgets

$$k \in \{100, 200, 300, 400, 500, 750, 1000, 1500, 2000\}. \tag{23}$$

Optimization. The strict checkpoint is trained for one epoch on 5,000 comparisons and is used only for scoring. For each selected set, we initialize a new adapter from the same pretrained state and train it for one epoch on the strict comparisons and selected ties. Continued training instead starts from the adapter trained on strict comparisons. Optimization uses AdamW, a cosine schedule, and the settings in Table 3.

Accuracy. For both LLM tasks, a held-out comparison is correct only when the implicit margin in (22) assigns the annotated winner the higher reward. We count a margin of exactly zero as incorrect.

Counterfactual energy. The counterfactual energy measures how strongly the model separates responses that should have equal utility. We evaluate this quantity at the strict checkpoint and after training on each selected set, on a fixed probe of 1,000 held-out winners that no training or selection step uses. Each winner yields four ties from four independent draws of its spurious content: for hotels, four donors from $\mathcal{D}_{\text{pref}}$; for emojis, four diamond counts drawn as in P, each rendered into the safe response and then removed by T. E_{cf} is the mean squared reference-adjusted margin over the 4,000 ties.

Practical influence. Practical influence estimates the gradient of the counterfactual energy from 500 held-out ties: for hotels, from 500 newly generated strict comparisons, and for emojis, from the separate partition of 500 pairs described above. Its curvature estimate uses a random subset of 1,000 comparisons from $\mathcal{D}_{\text{pref}}$. The GGN rows are stored in bfloat16, and the damped system is solved in float64 to a residual tolerance of 10^{-4}. Appendix B.1 describes how we construct and solve the system.

Settings. Table 3 summarizes the settings of both tasks.

Table 3: Settings for the LLM experiments. Shared entries apply to both tasks unless separate values are shown.

Setting	Hotels	Emojis
Base policy π_{ref}	Llama-3.2-1B-Instruct	
Arithmetic and attention	unquantized bfloat16; FlashAttention 2	
LoRA rank; scale; dropout	16; 32; 0.05	
LoRA targets	all attention and feed-forward projections	
DPO coefficient β	0.1	
Optimizer; step size η	AdamW; 10^{-5}	
Epochs; batch; accumulation	1; 4; 4	
Schedule; warmup; $\|g\|_{\max}$	cosine; 0.1; 1.0	
Maximum length $L_{\max}$	1,024 tokens	
Strict comparisons N	5,000	5,000
Tie pool M	2,000	2,000
Seeds; UTL draws per seed	3; 2	3; 2
Ties in probe; comparisons for DPO	500; 1,000	500; 1,000
GGN		
Storage for GGN; shift; residual tolerance	bfloat16; 0.05; 10^{-4}	bfloat16; 0.05; 10^{-4}
Ties in counterfactual probe	$1{,}000 \times 4$	$1{,}000 \times 4$
Evaluation pairs	2,000	1,023
Correlation probability ρ	0.85	0.95
Hotel $(\sigma_\epsilon, \Delta u_{\min})$	$(2.5, 5.0)$	–
Maximum emoji count $A_{\max}$	–	8

C.5.4 Additional Results: Hotels

Section 6 discusses accuracy under Q_{adv} after fresh training (Figure 2(b)). We report the other metrics and compare fresh with continued training.

Clean accuracy and energy. Selection improves clean accuracy while reducing sensitivity to the altered hotel fields (Figure 5). With 500 ties, PM and Practical influence reach 92.9% and 93.1% clean accuracy, above the 92.5% that UTL reaches with 1,000 ties. Their counterfactual energies, 1.38 and 1.25 (95% CIs $[1.01, 1.75]$ and $[0.31, 2.20]$), are also below the 1.89 that UTL reaches with 1,000 ties. At 500 ties, UTL reaches 85.7% clean accuracy and energy 5.12 (95% CI $[3.45, 6.80]$), and the strict model has energy 27.38.

Fresh versus continued training. Continued training helps at low budgets but falls behind fresh training at high budgets. At $k = 100$, continued PM reaches 67.3% adversarial accuracy, compared with 55.9% after fresh training. At $k = 2{,}000$, fresh PM reaches 95.3%, compared with 80.2% after continuation. The gain at low budgets is consistent with continuation starting from a model already trained on the strict comparisons, despite matched training steps. We hypothesize that adding many ties changes the objective enough to make the strict checkpoint a poorer starting point. This comparison does not establish a change of basin.

Accuracy and energy need not agree. Continued training raises clean and adversarial accuracy, yet the mean energy after continued training remains above its strict baseline at every budget. At the full pool, continued PM has 86.9% clean accuracy and energy 32.75 (95% CI $[30.57, 34.94]$), whereas fresh PM has 95.8% and 0.48. The continued accuracy gains therefore do not demonstrate lower sensitivity to the counterfactual operator.

In-distribution accuracy. In-distribution accuracy stays high under both training procedures: with the full pool, PM reaches 97.9% after fresh training and 98.6% after continuation, compared with 97.8% without ties.

C.5.5 Additional Results: Emojis

Section 6 discusses accuracy under Q_{adv} after fresh training (Figure 2(c)). We report the other metrics and compare fresh with continued training.

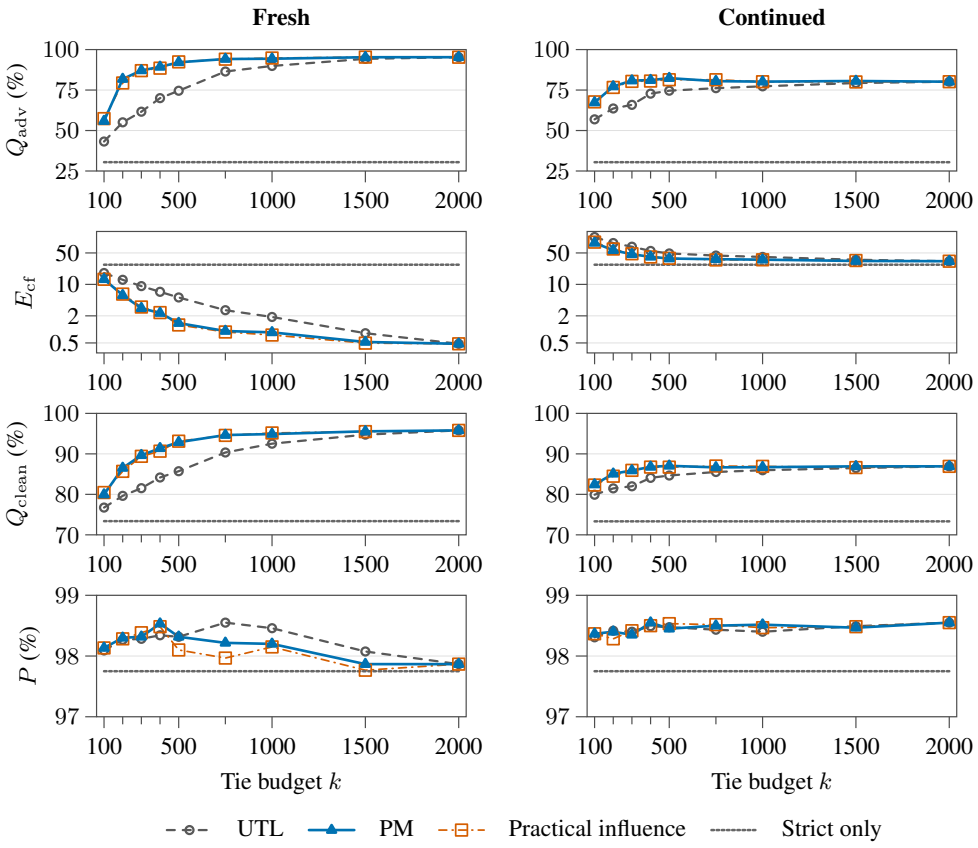

Figure 5: Hotel results under fresh training (left) and continued training (right). Rows share axes; colours, markers, and line styles match across columns. Curves show means across three seeds without intervals to avoid clutter. Dotted grey lines mark strict baselines. Energy uses a logarithmic axis.

Clean accuracy and energy. As for hotels, selection improves clean accuracy and lowers counterfactual energy, with PM ahead of Practical influence (Figure 6). With 500 ties, PM reaches 81.3% clean accuracy and energy 2.05 (95% CI $[1.34, 2.75]$), better on both metrics than UTL with 750 ties (80.5% and 2.74). At 500 ties, Practical influence reaches 79.5% and 3.36 (95% CI $[2.38, 4.33]$), and UTL reaches 77.5% and 5.15 (95% CI $[4.58, 5.72]$).

Fresh versus continued training. As with hotels, continued training helps at low budgets but loses its advantage as more ties enter training. Continued PM reaches 61.3% adversarial accuracy at $k = 100$, versus 50.3% after fresh training; at $k = 2{,}000$, fresh PM reaches 75.5%, versus 66.8% after continuation.

Accuracy and energy need not agree. Continued PM improves clean accuracy from 71.1% to 81.7% at the full pool, yet its mean energy rises from 20.69 to 24.65 (95% CI $[20.48, 28.82]$). Fresh PM reaches 84.0% clean accuracy with energy 0.70. At every budget, the mean energy after continued training stays above the strict baseline, so higher accuracy does not by itself show that the model has stopped using emojis.

In-distribution accuracy. Fresh training reduces P accuracy from 96.1% without ties to 92.6%, whereas continued PM retains 96.4%. The stronger mitigation under fresh training thus comes with an in-distribution trade-off, and continuation retains more of the original preference pattern.

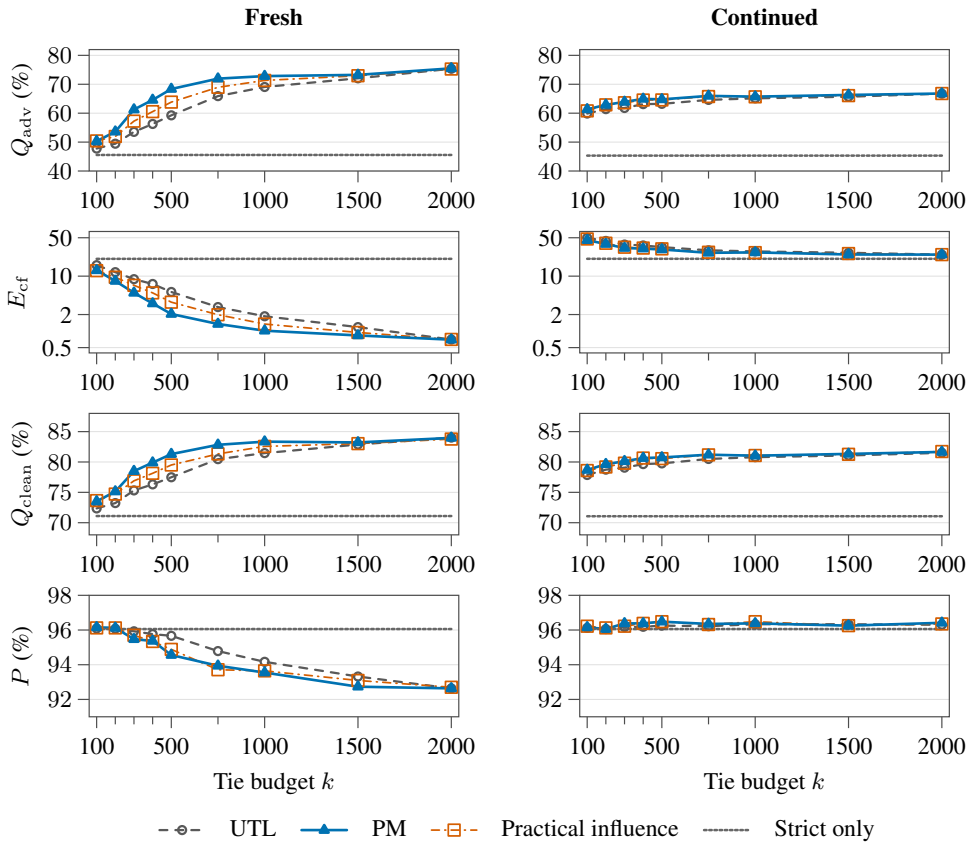

Figure 6: Emoji results under fresh training (left) and continued training (right). Rows share axes; colours, markers, and line styles match across columns. Curves show means across three seeds without intervals to avoid clutter. Dotted grey lines mark strict baselines. Energy uses a logarithmic axis.

C.5.6 Additional Results: Qwen3-4B

We also test whether the selection gains extend beyond Llama-3.2-1B by repeating the experiment with fresh training using Qwen3-4B-Instruct-2507, a larger model from a different family. The model runs in unquantized bfloat16 with LoRA rank 8 and scaling factor 16. We use a batch of 2 per device with 8 accumulation steps, which keeps the effective batch size of 16. All other settings follow Table 3: all methods select from the same pool of 2,000 ties, and Practical influence uses a probe of 500 ties and 1,000 comparisons for the GGN. These results use one seed and one UTL draw, so they carry no uncertainty estimates (Figure 7).

Hotels. Both selectors improve adversarial accuracy over UTL at every tested budget below the full pool. With 500 ties, PM and Practical influence reach 62.6% and 66.2%, above the 62.0% that UTL reaches with 1,000 ties and the 27.2% it reaches with 500. Their clean accuracies also improve to 80.5% and 81.5%, versus 70.8%, while energy falls to 2.38 and 2.56, versus 6.70. Neither selector is better than the other at every budget. With a single seed, the curves are also less smooth than the Llama means: PM falls from 55.7% at $k = 300$ to 46.0% at $k = 400$.

Emojis. Selection gives smaller but consistent gains over UTL at budgets below the full pool. With 500 ties, PM reaches 71.1% adversarial accuracy, above the 70.9% that UTL reaches with 750 ties, compared with 68.4% for Practical influence and 63.4% for UTL at 500. Clean accuracy rises to 82.5% for PM, versus 81.5% and 80.1%, and energy falls to 2.20, versus 3.25 and 5.67. The accuracy gains therefore accompany lower sensitivity to emoji removal. At the full pool, all three methods reach about 80% adversarial accuracy and energy near 0.58.

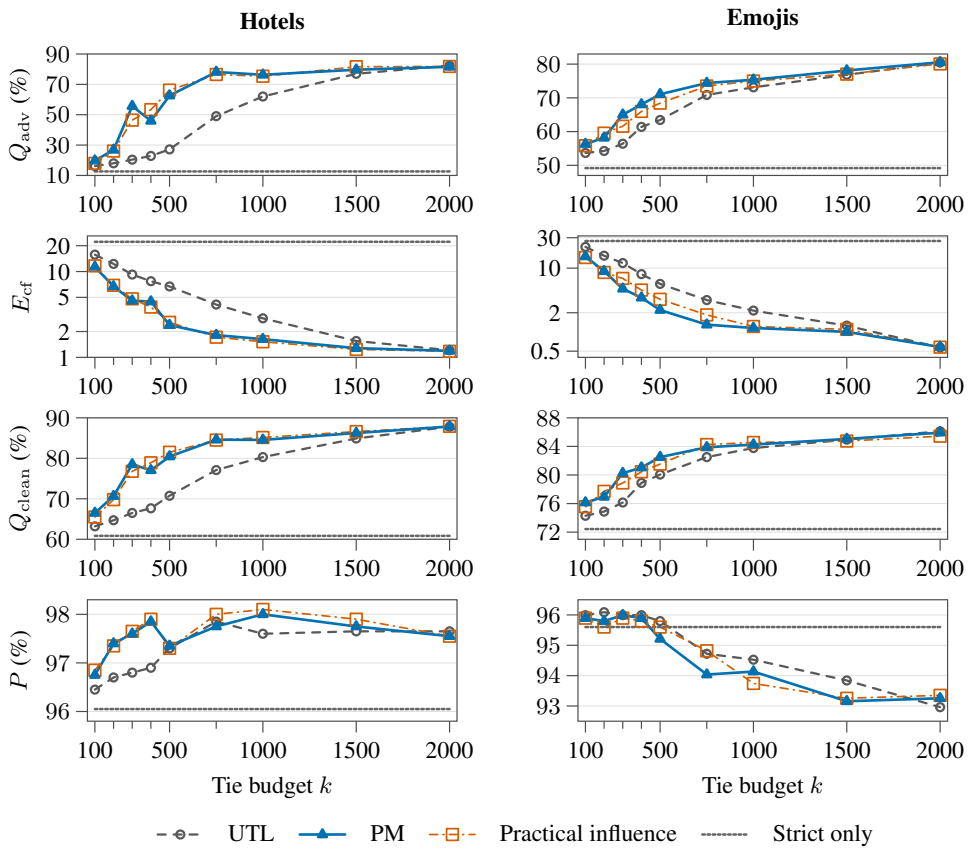

Figure 7: Results after fresh training for Qwen3-4B-Instruct-2507 on hotels (left) and emojis (right). Y-axis ranges differ between columns to show differences within each task. Curves show seed 0 with one UTL draw; uncertainty intervals are not estimated. Dotted grey lines mark strict baselines. Energy uses a logarithmic axis.

Full pool. At $k = M$, all methods train on the same ties. Their remaining differences in adversarial accuracy, at most 0.9 percentage points on hotels and 0.5 on emojis, come from separate training runs.

Comparison with Llama. On hotels, Qwen relies more on the spurious fields than Llama. Its strict model reaches 12.7% accuracy under Q_{adv} and 60.9% under Q_{clean}, compared with means of 30.5% and 73.4% for Llama. With the full pool, Qwen reaches 81.6–82.5% under Q_{adv} and energy 1.2, compared with 95.3% and 0.48 for Llama. Its strict energy, however, is lower: 22.21 versus 27.38. On emojis, Qwen's accuracies are similar to or higher than Llama's.

In-distribution accuracy. On hotels, accuracy under P stays high: with the full pool, PM reaches 97.6%, versus 96.1% without ties. For emojis, it falls from 95.6% to 93.3%, an in-distribution trade-off like the one observed for Llama.

D LIMITATIONS AND FUTURE WORK

Controlled evaluation. Our theory holds for log-linear policies with a known decomposition into core and spurious features, at a stationary point of the DPO loss, and to first order in the tie weight. We therefore test its quantitative predictions only where these conditions hold, and in neural networks and language models we ask whether the selection gains persist. Controlled tasks provide the quantities needed for both tests. Because we know the spurious attribute, we can construct ties whose responses share their core content, define Q_{adv} and Q_{clean} by reversing or removing the spurious correlation, and measure spurious reliance with the counterfactual energy. These choices isolate the predictions of the analysis. Applying the same tests to natural preference data would require identifying a spurious attribute, validating that the transformation preserves utility, and defining appropriate evaluation distributions. Evaluating STL under these conditions and with larger models remains future work.

Known spurious attribute and strict checkpoint. STL selects from a given pool of ties, and our pools come from an operator T that changes a known spurious attribute. Scoring also requires a strict checkpoint trained on the preference data. Both requirements hold naturally when STL corrects a behaviour observed after preference training. If a trained model favours longer responses, for example, the spurious attribute is verbosity, and the trained model itself serves as the strict checkpoint. STL then chooses which ties to add before retraining. When the spurious attribute is unknown, constructing a useful tie pool remains an open question.

Relation to RRM. RRM (Liu et al., 2025) offers an easy way to construct comparisons that it calls ties: it pairs two responses written for other prompts and labels them as a tie for the current prompt. This construction needs no knowledge of the spurious attribute. Our theory, however, does not apply to these pairs. It requires the two responses of a tie to share their core features, so that the tie gradient has no core component (Section 3.1) and tie training adds curvature only along spurious directions (Moya et al., 2026). Two responses to different prompts generally differ in core content, so an RRM tie also acts on the core weights. We could not test RRM ties in the log-linear model, because our synthetic data have no prompts: features depend on the response alone, so responses from other prompts are undefined. Extending the analysis to ties with a nonzero core difference, and comparing selection on RRM ties empirically, remain open.