PRIOR-ENLIGHTENED POSITIVE-UNLABELED LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Positive-Unlabeled (PU) learning trains models from limited labeled positive samples and unlabeled data with mixed positive and negative instances. A core challenge resides in extracting credible negative candidates from unlabeled data, a procedure that critically determines the ranking supervision quality of pairwise PU learning. Current sampling paradigms rely solely on intrinsic data statistics and real-time model outputs, while substantially neglecting informative external prior cues. In practical scenarios, such prior knowledge provides complementary evidence for identifying credible negative samples. This constitutes a critical research gap: *whether and how external priors can be properly integrated to alleviate the intrinsic limitations of purely model-based sampling in PU learning?* To tackle this issue, this work presents a Prior-Enlightened Positive-Unlabeled Learning (PEPU) framework for pairwise PU learning. PEPU integrates the external prior credibility of unlabeled negative candidates with model-derived pairwise utility to achieve adaptive sampling, and formulates a principled optimization objective that balances pairwise utility and prior knowledge. This work theoretically analyzes the trade-off between prior guidance and model optimization, and further investigates its effects on hidden positive contamination and downstream ranking performance. Extensive experiments on controlled PU studies and two real-data application domains validate that PEPU consistently outperforms state-of-the-art sampling strategies.

1 INTRODUCTION

Positive-Unlabeled (PU) learning trains models with labeled positive samples and unlabeled data, where explicit negative annotations are completely absent (Elkan & Noto, 2008; Bekker & Davis, 2020). The unlabeled set inherently mixes true negatives and hidden positives, bringing fundamental ambiguity when serving unlabeled instances as negative supervision. Such ambiguity severely degrades pairwise learning: each unlabeled candidate paired with an observed positive is treated as a relative negative during training, which introduces incorrect supervision once the candidate is essentially a hidden positive. Therefore, effective pair sampling for pairwise PU learning requires two indispensable properties: high pairwise utility for model optimization and reliable negative credibility. Existing sampling paradigms predominantly prioritize utility alone. Random sampling generates trivial pairs with limited informative gradients, while hard-negative sampling selects model-confusing candidates to strengthen discriminative learning (Zhang et al., 2013; Rendle & Freudenthaler, 2014; Robinson et al., 2021). However, hard candidates in PU scenarios are ambiguous, as they can be either informative true negatives or misleading hidden positives (Ding et al., 2020; Chuang et al., 2020). Model-derived hardness only reflects pairwise utility but cannot guarantee negative credibility. Current methods rely exclusively on data statistics and instantaneous model outputs, while largely neglecting external cues for credibility assessment.

External prior information provides complementary evidence for measuring negative plausibility. Such prior can be derived from external models, semantic knowledge, or other auxiliary sources, and only requires reliable relative ranking of negative candidacy instead of precisely calibrated probabilities or hard pseudo-labels. It complements model-derived hardness rather than replacing it: priors determine *which candidates are safe negatives*, while model outputs identify *which candidates yield high pairwise utility*. Pure prior-based sampling produces credible yet uninformative pairs, while pure model-based sampling overemphasizes ambiguous hidden positives. This constitutes a

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.

central research question: *whether and how external priors can be properly integrated to maximize pairwise utility while respecting prior credibility?*

This work presents **Prior-Enlightened Positive-Unlabeled Learning (PEPU)**, a unified sampling framework that integrates external prior credibility and model-derived pairwise utility into a single sampling distribution. The prior constructs a credibility-aware reference distribution over unlabeled candidates, and the pairwise BPR loss quantifies candidate-specific weights for model updates. We derive the proposed sampler from a reference-regularized pairwise sampling objective, where the temperature parameter explicitly balances learning utility and reference deviation. Theoretical analyses verify that PEPU is fully compatible with standard BPR optimization, bounds the deviation of hidden-positive contamination, and establishes a rigorous connection between prior reference quality and ranking AUC under the latent true-negative distribution.

We evaluate PEPU on controlled PU benchmarks and real-world tasks. Two controlled settings with and without adaptive encoders adopt identical PU generative processes and external priors, effectively isolating pair sampling effects from representation learning. We also validate PEPU on the CIFAR-PU dataset and three implicit-feedback recommendation datasets, with external priors adapted to downstream tasks. Our analyses examine prior credibility effectiveness, model robustness to degraded priors, and the inherent utility-reference trade-off. Experimental results yield consistent pairwise ranking improvements against state-of-the-art sampling strategies in all scenarios.

2 Related Work

Positive-Unlabeled learning. Existing PU-learning studies mainly focus on optimizing PU learning objectives, including classification risk estimation and pairwise AUC optimization (du Plessis et al., 2014; 2015; Kiryo et al., 2017; Sakai et al., 2018), or modeling latent class priors and labeling mechanisms (Ramaswamy et al., 2016; Garg et al., 2021; Bekker et al., 2019; Kato et al., 2019). Recent works enhance unlabeled supervision and representation learning through pseudo-labeling, mixup and meta-learning strategies (Chen et al., 2020; Li et al., 2022a; Long et al., 2024; Zhao et al., 2026), while standardized benchmarks unify PU evaluation assumptions and protocols (Chen et al., 2026b; Wang et al., 2026). These methods focus on objective construction, latent supervision and representation learning for PU data. Differently, our work focuses on selecting credible unlabeled candidates as relative negatives for model updates.

Negative sampling. Pairwise ranking losses such as BPR require negative sampling, and uniform sampling generally produces trivial pairs with low pairwise utility (Rendle et al., 2009; Rendle & Freudenthaler, 2014). Conventional negative sampling methods prioritize informative hard negatives via model hardness scoring, negative synthesis and theoretical hardness analysis (Zhang et al., 2013; Rendle & Freudenthaler, 2014; Robinson et al., 2021; Huang et al., 2021; Shi et al., 2023). Advanced approaches further alleviate false-negative risks through adaptive hardness scheduling and debiased optimization (Ding et al., 2020; Chuang et al., 2020; Lai et al., 2024), or adopt external side information to assist candidate selection (Loni et al., 2016; Ding et al., 2019; Wang et al., 2020). However, existing sampling paradigms only optimize sample informativeness or risk mitigation. They overlook the inherent ambiguity of PU scenarios and fail to jointly balance negative credibility and pairwise utility, which distinguishes our PEPU framework from prior work.

3 Prior-Enlightened Positive-Unlabeled Learning

3.1 Prior-Enlightened Pair Sampling

A useful sampler is supposed to favor candidates that are credible as relative negatives while still prioritizing pairs that provide informative signals for training the current model. PEPU represents these complementary roles through fixed prior *credibility* and model-dependent *utility*.

Prior credibility and reference distribution. Let $S(p, u) > 0$ quantify prior evidence that u is suitable as a relative negative for p. The pair-dependent form allows the same unlabeled candidate to have different credibility for different positives, with $S(u)$ as the sample-level special case. The score $S(p, u)$ need not be supplied directly by the prior information source. It may instead be

constructed from fixed external representations or other prior information through a task-dependent comparison or aggregation rule that remains unchanged during training. PEPU only requires S to provide meaningful relative negative credibility and to remain fixed with respect to the current model parameters θ; it need not be calibrated as a true-negative probability. Let $P_0(u \mid p)$ denote the base unlabeled-sampling distribution before incorporating the prior.

Definition 3.1 (Reference distribution)**.** For any p, suppose $0 < Z_S(p) := \mathbb{E}_{u\sim P_0(\cdot|p)} S(p,u) < \infty$. The reference distribution is defined as

$$P_S(u \mid p) = \frac{S(p,u)}{Z_S(p)} P_0(u \mid p). \tag{1}$$

Throughout training, P_0, S, and hence P_S are fixed with respect to the current model parameters θ.

The reference P_S converts prior credibility into sampling probability, the candidates that are more strongly supported as negatives receive larger reference mass. Since $S(p,u) > 0$, P_S preserves the support of P_0; if S does not distinguish among candidates, then $P_S = P_0$.

Current-model utility. Credibility indicates which candidates are more suitable as relative negatives, yet not which ones remain informative to the current model. We use the standard BPR logistic loss as a pair-level utility signal.

Definition 3.2 (Pair-level utility)**.** Given a scoring function $s_\theta(\cdot)$, define the pairwise margin and BPR loss by

$$d_\theta(p,u) = s_\theta(p) - s_\theta(u), \qquad \ell_\theta(p,u) = \log\Big(1 + e^{-d_\theta(p,u)}\Big). \tag{2}$$

We define $\ell_\theta(p,u)$ as the current utility of pair (p,u). A larger loss indicates that the pair is currently harder to rank correctly, hence regarded as carrying stronger corrective pressure under BPR. Unlike the fixed prior credibility, ℓ_θ changes with the model parameters θ during training.

Prior-Enlightened sampling. PEPU combines the two signals by exponentially tilting the prior reference toward high-utility pairs:

$$Q_\tau^{\mathrm{PE}}(u \mid p;\theta) \propto \underbrace{P_S(u \mid p)}_{\text{prior reference}} \underbrace{\exp\{\ell_\theta(p,u)/\tau\}}_{\text{current utility}}, \qquad \tau > 0. \tag{3}$$

We refer to Q_τ^{PE} as the *Prior-Enlightened (PE) sampler*. The next section provides an optimization characterization of this sampling rule.

3.2 Optimization Structure of PE Sampler

Fix the model parameters θ, and let $Q(u \mid p)$ denote any negative sampling distribution. For $p \sim P_+$, the training pair distribution induced by Q and the prior reference pair distribution are, respectively,

$$J_Q(p,u) = P_+(p)Q(u \mid p), \qquad J_S(p,u) = P_+(p)P_S(u \mid p).$$

For a sampler Q, we seek to favor pairs that are useful for the current model while limiting the deviation of J_Q from the prior reference J_S. This leads to the following formulation.

Definition 3.3 (Expected utility and reference deviation)**.** For fixed θ, define

$$\mathcal{H}_\theta(Q) = \mathbb{E}_{J_Q}[\ell_\theta], \qquad \mathcal{K}(Q) = D_{\mathrm{KL}}(J_Q \| J_S) = \mathbb{E}_{P_+} D_{\mathrm{KL}}(Q(\cdot \mid p) \,\|\, P_S(\cdot \mid p)). \tag{4}$$

$\mathcal{H}_\theta(Q)$ is the expected utility of the pairs Q selects, and $\mathcal{K}(Q)$ is its *reference deviation*. Sampler design therefore amounts to maximizing $\mathcal{H}_\theta(Q)$ while minimizing $\mathcal{K}(Q)$. The former favors pairs that remain informative for the current model, whereas the latter limits how far the sampler departs from the prior reference. We refer to this as the *utility–reference-deviation trade-off*.

Solving the trade-off for fixed τ. The quantities in Eq. (4) define a bi-objective sampling problem. We seek high expected utility while limiting the deviation of the induced pair distribution from the

prior reference. For fixed model parameters θ, a particular trade-off between these two objectives is obtained by solving

$$\max_{Q} \ \mathcal{H}_\theta(Q) - \tau \mathcal{K}(Q), \qquad \tau > 0. \tag{5}$$

The coefficient τ controls the strength of the reference-deviation penalty. Larger τ favors samplers closer to P_S, while smaller τ allows stronger reweighting toward pairs with high current utility.

Because J_Q and J_S share the same positive marginal P_+, Eq. (5) decomposes over positive samples p. Solving the resulting conditional optimization problems gives the following closed-form solution.

Proposition 3.4 (Optimal sampler for a fixed trade-off)**.** For fixed θ and $\tau > 0$, the optimizer of Eq. (5) is

$$Q^\star_\tau(u \mid p) = \frac{P_S(u \mid p) \exp\{\ell_\theta(p,u)/\tau\}}{Z_\tau(p)}, \qquad Z_\tau(p) = \mathbb{E}_{P_S(\cdot|p)} \exp\left(\frac{\ell_\theta(p,u)}{\tau}\right). \tag{6}$$

Eq. (6) is exactly the normalized form of the PE sampler introduced in Eq. (3). This establishes the optimization basis of the multiplicative sampling rule in Eq. (3). The full derivation, including the Lagrange solution, strict-concavity argument, and exact KL optimality decomposition, is provided in AppendixA.3.

The PE sampling path. Proposition 3.4 identifies the optimal sampler for each fixed $\tau > 0$. This optimality also makes Q^{PE}_τ Pareto-efficient for the original utility–reference-deviation problem: if another sampler achieved no lower utility and no larger reference deviation, with at least one strict improvement, it would attain a larger value of Eq. (5), contradicting Proposition 3.4.

Holding θ fixed and varying τ traces the *PE sampling path*, a one-parameter family of Pareto-efficient samplers. Large τ keeps the sampler close to P_S, with $Q^{\mathrm{PE}}_\tau \to P_S$ as $\tau \to \infty$; small τ increasingly favors utility, with $Q^{\mathrm{PE}}_\tau \to Q^{\max}$ as $\tau \to 0^+$. Thus, it remains to determine whether the PE sampling path together with its endpoints constitutes the entire Pareto frontier.

Theorem 3.5 (Pareto characterization of the PE sampling path)**.** Under the support conditions in Appendix A, suppose that the current utility is non-degenerate for at least one positive p, i.e., there exist u_1, u_2 with $P_S(u_i \mid p) > 0$ such that $\ell_\theta(p,u_1) \neq \ell_\theta(p,u_2)$. Then the set of Pareto-efficient samplers is

$$\{Q^{\mathrm{PE}}_\tau : \tau \in (0,\infty)\} \cup \{P_S, Q^{\max}\}.$$

Theorem 3.5 shows that the PE sampling path and its two endpoints are exactly the Pareto frontier of the utility–reference-deviation problem, which we call the *PE sampling frontier*. The proof is given in Appendix A.4.

Along this frontier, $\mathcal{K}(Q^{\mathrm{PE}}_\tau)$ increases strictly as τ decreases, and conversely larger τ moves the sampler toward P_S. Moreover, utility is a strictly concave function of reference deviation, with

$$\frac{d\mathcal{H}_\theta}{d\mathcal{K}} = \tau, \qquad \frac{d^2\mathcal{H}_\theta}{d\mathcal{K}^2} < 0. \tag{7}$$

Thus, τ is the local slope of the frontier and measures the marginal utility gained per unit additional reference deviation. The derivation is provided in Appendix A.5.

Training-time realization. The preceding analysis parameterizes the PE sampling path by the temperature τ, with each sampler along the path attaining a corresponding reference deviation $\mathcal{K}(Q^{\mathrm{PE}}_\tau)$. During training, the model parameters θ_t evolve, changing the current utilities $\ell_{\theta_t}(p,u)$ and hence the mapping between τ and reference deviation. We therefore parameterize the training-time trade-off by a target deviation δ_{tr} and adapt τ_t to realize this level as the model evolves.

Algorithm 1 Prior-Enlightened BPR training

Input: P_+, P_0, S; budget δ_{tr}, τ_0, b; $\theta^{(0)}$, α, T
1: $P_S(u \mid p) \propto S(p,u)P_0(u \mid p)$, fixed throughout
2: **for** $t = 0, \ldots, T-1$ **do**
3: Sample $p_t \sim P_+$
4: **Temperature** ($t \geq 1$): $\tau_t \leftarrow \tau_{t-1} + b(\widehat{K}_{t-1} - \delta_{\mathrm{tr}})$
5: **PE sampler:** $Q^{\mathrm{PE}}_t := Q^{\mathrm{PE}}_{\tau_t}(\cdot \mid p_t; \theta^{(t)})$
6: **Sampling:** draw $u_t \sim Q^{\mathrm{PE}}_t$
7: BPR update: $\theta^{(t+1)} \leftarrow \theta^{(t)} - \alpha \nabla_\theta \ell_\theta(p_t, u_t)$
8: **end for**
Output: $\theta^{(T)}$

Let $\widehat{K}_t$ denote the estimate of the reference deviation in t-th iteration. The computation of $\widehat{K}_t$ is given in Appendix A.6 Eq. (A.70). The one-dimensional dual derived in Appendix A.6 gives

$$\tau_{t+1} \approx \tau_t + b\,(\widehat{K}_t - \delta_{\mathrm{tr}}), \qquad b > 0. \tag{8}$$

The residual $\widehat{K}_t - \delta_{\mathrm{tr}}$ provides the feedback needed to track the target reference-deviation level as the model evolves. Algorithm 1 summarizes the complete training procedure. PEPU only modifies the candidate-sampling step through the adaptive PE sampler, while the BPR loss and the model-parameter update remain unchanged.

3.3 PE Sampler in PU Training

With the PE sampler established, we next examine its behavior in PU training. Because it depends on the current model state through ℓ_θ, we first ask whether adaptive sampling remains consistent with the standard BPR update. We then analyze hidden-positive contamination, since high-loss candidates favored by utility reweighting may be either informative hard negatives or hidden positives. Finally, all preceding results are relative to P_S, so we must also assess the quality of P_S itself by relating it to the latent true-negative distribution, and then establish connection to target AUC.

Optimization consistency. Because Q_τ^{PE} is built from the current BPR loss, it changes with θ. Proposition 3.4 identifies Q_τ^{PE} as the inner optimizer of (5) at every model state, so its adaptation is contained in the value function $\Phi_\tau(\theta) = \max_Q \{\mathcal{H}_\theta(Q) - \tau\mathcal{K}(Q)\}$. And the question becomes whether $\nabla_\theta \Phi_\tau$ contains terms beyond the BPR gradient under Q_τ^{PE}.

Theorem 3.6 (Sampler–BPR optimization consistency)**.** For any $\tau > 0$, with Q_τ^{PE} evaluated at the current θ,

$$\nabla_\theta \Phi_\tau(\theta) = \mathbb{E}_{P_+}\mathbb{E}_{Q_\tau^{\mathrm{PE}}}\left[\nabla_\theta \ell_\theta(p,u)\right]. \tag{9}$$

Thus, sampling a pair from the current Q_τ^{PE} and applying the ordinary BPR gradient gives an unbiased stochastic gradient of Φ_τ; no gradient through the adaptive sampler is required. The same gradient identity holds for the deviation-budget formulation, given in Appendix A.7.

Hidden-positive contamination. In PU training, contamination occurs when an unlabeled candidate sampled as the relative negative is in fact a hidden positive. Let $Y(p,u) \in \{0,1\}$ indicate this event. For a sampler Q, let $\eta(Q) = \mathbb{E}_{J_Q}[Y], \eta_S = \mathbb{E}_{J_S}[Y]$, in which $\eta(Q)$ is the hidden-positive contamination rate induced by Q, while η_S is the corresponding rate under the reference distribution. As utility reweighting can change this rate because high-loss candidates may include both informative hard negatives and hidden positives, the following result bounds how far the contamination rate can move from its reference level.

Theorem 3.7 (Reference-deviation bound on contamination shift)**.** For any sampler Q,

$$\mathrm{kl}(\eta(Q)\,\|\,\eta_S) \le \mathcal{K}(Q), \qquad |\eta(Q) - \eta_S| \le \min\left\{1, \sqrt{\mathcal{K}(Q)/2}\right\}. \tag{10}$$

Therefore, for any sampler Q satisfying $\mathcal{K}(Q) \le \delta_{\mathrm{tr}}$, $|\eta(Q) - \eta_S| \le \sqrt{\delta_{\mathrm{tr}}/2}$. The deviation budget thus limits how far contamination can move from the reference level η_S, but not the direction of that change or the quality of η_S itself. The direction depends on the relation between hidden-positive status and the current loss. The proof and a finer local characterization are provided in Appendix A.9.

From reference quality to true AUC. The preceding results are all relative to P_S, so they do not by themselves establish the quality of P_S as a proxy for the true negative distribution. If the latent labels were observable, let $J^*(p,u) = P_+(p)P_-(u \mid p)$ denote the true positive–negative pair distribution.

Assumption 3.8 (Reference Coverage)**.** There exists $\delta_{\mathrm{cov}} \ge 0$ such that

$$D_{\mathrm{KL}}(J^* \| J_S) \le \delta_{\mathrm{cov}}. \tag{11}$$

The radius δ_{cov} measures the quality of the reference distribution with respect to the true-negative distribution. A smaller δ_{cov} means that J_S better covers J^*, while the direction $D_{\mathrm{KL}}(J^* \| J_S)$ penalizes true-negative regions that receive insufficient reference mass.

For a true positive–negative pair, define

$$e_\theta(p,u) = \mathbf{1}\{d_\theta(p,u) < 0\} + \tfrac{1}{2}\mathbf{1}\{d_\theta(p,u) = 0\}, \qquad \mathrm{AUC}^*(\theta) = 1 - \mathbb{E}_{J^*} e_\theta.$$

Here AUC^* is the probability that a random true pair is ranked correctly (Hanley & McNeil, 1982). To state a relative performance guarantee, let θ_0 denote any fixed comparison model, such as a standard BPR baseline. It serves only as the AUC comparison point and does not enter PEPU training. Let $c = \log 2$. Since the BPR logistic loss satisfies $c\, e_\theta \le \ell_\theta$ (cf. Bartlett et al., 2006; Gao & Zhou, 2015), define the comparison margin $m_{\theta,\theta_0}(p,u) := c\, e_{\theta_0}(p,u) - \ell_\theta(p,u)$. The calibration inequality gives $c(e_{\theta_0} - e_\theta) \ge m_{\theta,\theta_0}$. Under Assumption 3.8, the unknown J^* lies in the KL neighborhood of J_S, so define the worst-case comparison margin

$$\Gamma_{\mathrm{cov}}(\theta;\theta_0) = \inf_{Q: D_{\mathrm{KL}}(J_Q \| J_S) \le \delta_{\mathrm{cov}}} \mathbb{E}_{J_Q} m_{\theta,\theta_0}(p,u).$$

Corollary 3.9 (True-negative AUC certificate)**.** Under Assumption 3.8,

$$\mathrm{AUC}^*(\theta) - \mathrm{AUC}^*(\theta_0) \ge \frac{\Gamma_{\mathrm{cov}}(\theta;\theta_0)}{\log 2}. \tag{12}$$

Hence, $\Gamma_{\mathrm{cov}}(\theta;\theta_0) > 0$ certifies a positive true-negative AUC improvement over θ_0. A smaller coverage radius restricts the admissible true-negative distributions and therefore gives a no-weaker worst-case certificate. Appendix A.10 provides the full surrogate argument and reduces Γ_{cov} to a one-dimensional KL dual in λ (Ben-Tal et al., 2013; Duchi & Namkoong, 2021).

4 Experiments

Our experiments address three complementary questions. Section 4.1 establishes whether PEPU improves pairwise learning across controlled and real settings. Section 4.2 examines whether the external prior indeed carries discriminative credibility information and how PEPU behaves as this information deteriorates. Finally, Section 4.3 directly tests whether the sampler exhibits the utility–reference-deviation geometry and contamination behavior predicted by the theory. Complete data-generation procedures, model architectures, training protocols, and hyperparameters are provided in Appendix B.

4.1 Performance Across Controlled and Real Settings

Experimental settings. We evaluate PEPU through two controlled PU studies and two real-data application domains. The controlled settings, **PU w/o Encoder** and **PU w/ Encoder**, are generated from the same latent process, the same PU observation mechanism, and the same fixed external view containing useful but imperfect credibility information. They differ solely in whether downstream representations adapt during training via a trainable residual encoder, enabling us to disentangle the effect of pair selection from representation adaptation. On **CIFAR-PU**, brightness-dependent selected-at-random (SAR) labeling (Bekker et al., 2019) leaves 37.5% hidden positives in the unlabeled pool. A frozen DINOv2 model (Oquab et al., 2024) provides the fixed prior, while a separately trained ResNet-18 (He et al., 2016) supplies the evolving BPR utility. We further evaluate **LightGCN recommendation** (He et al., 2020) on Amazon-book, Yelp, and Steam, where the fixed LLM-derived item semantics released with RLMRec (Ren et al., 2024) define user-conditioned credibility.

Baselines and evaluation. We compare PEPU against uniform random negative sampling (RNS; Rendle et al., 2009) and representative hard-negative samplers, including DNS (Zhang et al., 2013), AOBPR (Rendle & Freudenthaler, 2014), AHNS (Lai et al., 2024), and SRNS (Ding et al., 2020). Besides, we use **Prior-only** and **Hard-only** as the two ablations of PEPU. For the sampling comparison in Table 1, all methods within each setting share the same training setup and differ only in how the unlabeled candidate is selected for the BPR update. We report ROC-AUC and AP for PU tasks and user-averaged AUC for recommendation. For Top-K recommendation, we additionally compare against a broader set of recommendation baselines: the LightGCN backbone with standard training (Normal), the instance-level denoising methods WBPR (Gantner et al., 2012), T-CE (Wang et al., 2021), and BOD (Wang et al., 2023b), the graph contrastive methods SGL (Wu et al., 2021), SimGCL (Yu et al., 2022), and XSimGCL (Yu et al., 2024), and the LLM-enhanced methods RLMRec (Ren et al., 2024) and LLaRD (Wang et al., 2025a), and report Recall@K and NDCG@K (Järvelin & Kekäläinen, 2002) for $K \in \{10, 20\}$.

Table 1: Ranking performance on the three PU settings and LightGCN recommendation. Hard-only and Prior-only are PEPU ablations. Bold and underline denote the best and second-best results.

	PU learning						LightGCN recommendation (AUC)		
	PU w/o Encoder		PU w/ Encoder		CIFAR-PU		Amazon-book	Yelp	Steam
Method	AUC	AP	AUC	AP	AUC	AP			
RNS	0.8384	0.7488	0.8827	0.7757	0.9091	0.8613	0.9169	0.9463	0.8909
DNS	0.7932	0.7120	0.8091	0.7130	0.8673	0.8057	0.8739	0.9344	0.7873
AOBPR	0.8602	0.7553	0.8636	0.7446	0.8686	0.8218	0.9160	0.9457	0.8716
AHNS	0.8440	0.7137	0.8652	0.7450	0.9351	0.9096	0.8817	0.9359	0.7834
SRNS	0.8066	0.7214	0.8805	0.7608	0.8631	0.8063	0.8884	0.9461	0.8415
Hard-only	0.8459	0.7598	0.8915	0.7738	0.9491	0.9093	<u>0.9248</u>	<u>0.9479</u>	<u>0.8921</u>
Prior-only	<u>0.8765</u>	<u>0.7834</u>	<u>0.9021</u>	<u>0.7948</u>	<u>0.9648</u>	<u>0.9488</u>	0.9224	0.9412	0.8507
PEPU (ours)	**0.8840**	**0.7903**	**0.9070**	**0.8012**	**0.9705**	**0.9525**	**0.9261**	**0.9585**	**0.9007**

Main results. Table 1 shows that PEPU achieves the best result on all six PU metrics and the highest AUC on all three recommendation datasets. More importantly, the two ablations reveal a clear task-dependent reversal. Across all three PU settings, Prior-only consistently outperforms Hard-only. These settings share a relatively strong alignment between the external representation and the downstream classification structure, making the credibility reference itself highly informative. PEPU nevertheless improves further over Prior-only because the fixed prior does not reflect the evolving state of the downstream model, and the current loss further identifies which credible candidates remain difficult and informative during training. Recommendation exhibits the opposite ordering, as hard-only outperforms prior-only on all three datasets, while PEPU remains the best. In this setting, semantic similarity can indicate latent relevance, but semantically similar candidates may also provide useful hard negatives as LLM-derived semantics remain imperfect proxies for user preference. This pattern suggests that relying on semantic credibility alone can be overly conservative for the downstream task, so that prior-only may therefore suppress informative hard candidates, whereas hard-only preserves their training value but ignores false-negative risk. PEPU combines the two signals by using the prior as a credibility-aware reference while retaining current-model utility.

Table 2: Top-K recommendation performance on Amazon-book, Yelp, and Steam. R/N denote Recall/NDCG, respectively. Bold and underline denote the best and second-best results, respectively.

	Amazon-book				Yelp				Steam			
Method	R@10	N@10	R@20	N@20	R@10	N@10	R@20	N@20	R@10	N@10	R@20	N@20
Normal	0.0950	0.0746	0.1415	0.0893	0.0617	0.0529	0.1011	0.0659	0.0838	0.0701	0.1317	0.0858
WBPR	0.0956	0.0749	0.1419	0.0901	0.0618	0.0531	0.1015	0.0662	0.0840	0.0704	0.1312	0.0850
T-CE	0.0990	0.0779	0.1499	0.0939	0.0740	0.0623	0.1216	0.0779	0.0877	0.0731	0.1376	0.0893
BOD	0.1273	0.0996	0.1792	0.1162	0.0843	0.0714	0.1355	0.0882	0.0955	0.0791	0.1484	0.0966
SGL	0.1091	0.0872	0.1588	0.1030	0.0762	0.0648	0.1235	0.0803	0.0890	0.0743	0.1378	0.0903
SimGCL	0.1172	0.0940	0.1681	0.1102	0.0785	0.0669	0.1265	0.0827	0.0887	0.0738	0.1373	0.0899
XSimGCL	0.1153	0.0931	0.1637	0.1084	0.0769	0.0663	0.1277	0.0830	0.0884	0.0736	0.1385	0.0903
DNS	0.1222	0.0976	0.1756	0.1146	0.0872	0.0719	0.1381	0.0888	0.0901	0.0738	0.1395	0.0901
AHNS	0.0824	0.0728	0.1381	0.0874	0.0648	0.0541	0.1063	0.0680	0.0688	0.0579	0.1138	0.0727
AOBPR	0.1022	0.0801	0.1515	0.0961	0.0668	0.0561	0.1094	0.0705	0.0935	0.0771	0.1463	0.0946
SRNS	0.1179	0.0941	0.1717	0.1112	0.0812	0.0681	0.1307	0.0845	0.0968	0.0800	0.1492	0.0973
RLMRec	0.1034	0.0788	0.1601	0.0960	0.0794	0.0652	0.1275	0.0815	0.0926	0.0746	0.1452	0.0924
LLaRD	<u>0.1408</u>	<u>0.1126</u>	<u>0.2028</u>	<u>0.1326</u>	<u>0.0975</u>	<u>0.0809</u>	<u>0.1574</u>	<u>0.1008</u>	<u>0.1054</u>	<u>0.0868</u>	<u>0.1631</u>	<u>0.1059</u>
PEPU (Ours)	**0.1457**	**0.1186**	**0.2054**	**0.1388**	**0.0983**	**0.0832**	**0.1583**	**0.1020**	**0.1061**	**0.0885**	**0.1640**	**0.1068**

Top-K recommendation. Table 2 shows PEPU achieves the best results on all 12 Recall and NDCG metrics. Together with the AUC results in Table 1, this demonstrates that gains from improved pair selection extend from global pairwise ranking to the top of the recommendation list.

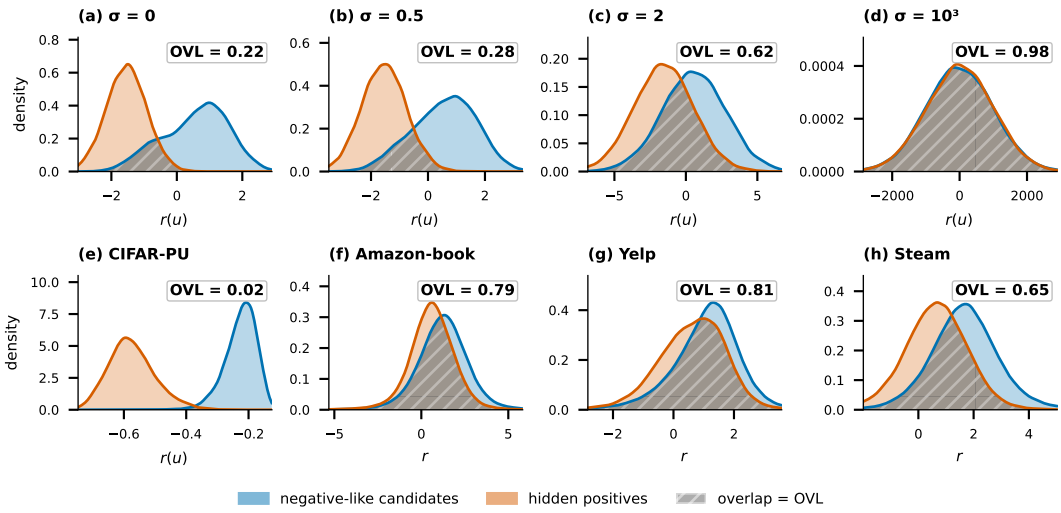


Figure 1: Prior-score distributions for auditing credibility discrimination. Larger r indicates greater negative credibility; shaded overlap corresponds to OVL in Eq. (13).

4.2 Prior Informativeness and Quality Sensitivity

PEPU requires only *relative credibility*, rather than calibrated negative probabilities. We therefore first examine whether the priors used in our experiments actually distinguish hidden-positive from negative-like candidates. Let r denote the prior score, oriented so that larger values indicate greater credibility as a negative. Collecting the scores within the two groups gives two empirical score distributions; an informative prior should shift hidden positives toward lower r and negative-like candidates toward higher r. We quantify their overlap by (Inman & Bradley, 1989)

$$\mathrm{OVL} = \int \min\{f_+(r), f_-(r)\}\,\mathrm{d}r \in [0,1], \tag{13}$$

where f_+ and f_- are the corresponding score densities. Smaller OVL indicates stronger credibility discrimination.

For PU, the two groups are identified using latent labels only for offline analysis. For recommendation, where true negatives are unavailable, we compare held-out known interactions with sampled non-interacted items; this is only an operational audit, not a ground-truth negative test.

Figure 1 shows informative but non-oracle priors across all settings. Their informativeness varies substantially: OVL is 0.28 for the default controlled prior and 0.02 for CIFAR-PU, while the recommendation priors range from 0.65 to 0.81. Thus, PEPU is evaluated under priors with widely varying credibility information.

We next test whether PEPU actually depends on this information. In PU w/o Encoder, we vary only the noise level of the auxiliary view, $a = g(c, o, s) + \sigma\epsilon$, while keeping the downstream task and training procedure fixed. This changes OVL from 0.22 to 0.98. Even at $\sigma = 0$, the prior is not oracle: removing observation noise does not remove the intrinsic ambiguity of the partial auxiliary view.

Figure 2 shows that PEPU degrades smoothly as the prior becomes less informative: its AUC decreases from 0.886 to 0.849, with AP showing the same trend. At the effectively pure-noise endpoint, Prior-only performs similarly to RNS, while PEPU approaches Hard-only empirically. Thus, the additional benefit provided by the prior diminishes as its credibility information is removed. Exact reduction to the base proposal would require the credibility score to be constant across candidates.

4.3 Empirical Geometry of the PE Sampling Frontier

We finally examine the sampler geometry predicted in Section 3.2. We freeze a trained PEPU checkpoint from PU w/o Encoder and, without further training, sweep the PE sampling path. Each temperature therefore gives one sampler and one point in the utility–reference-deviation plane, with

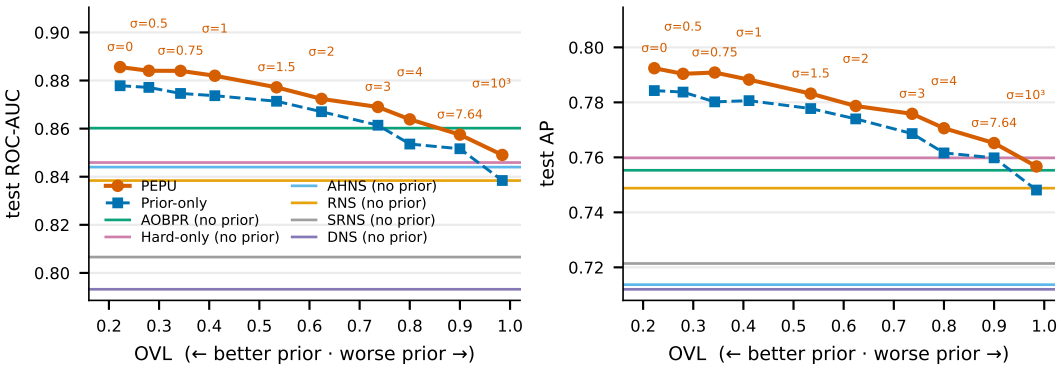


Figure 2: Effect of prior informativeness in PU w/o Encoder. Only the prior noise level varies; $\sigma = 0$ is the best prior and $\sigma = 10^3$ is effectively pure noise.

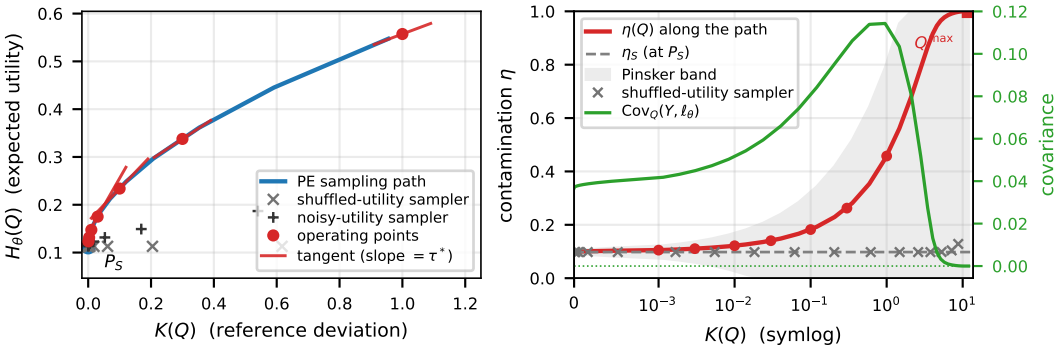


Figure 3: Empirical PE sampling path at a fixed PU w/o Encoder checkpoint. Left: utility–reference-deviation trade-off and budget-selected operating points. Right: hidden-positive contamination along the same path and its theoretical bound.

$\mathcal{K}(Q) = D_{\mathrm{KL}}(J_Q \| J_S)$ and $H_\theta(Q) = \mathbb{E}_{J_Q}[\ell_\theta]$. We also record the hidden-positive rate $\eta(Q)$ for offline analysis. Full sweep details and numerical checks are deferred to Appendix B.8.

Figure3 closely matches the theoretical predictions. In the left panel, the PE sampling path traces a concave Pareto frontier, where larger reference deviation yields higher expected utility with diminishing returns. The training budget δ_{tr} specifies the target reference-deviation level, and the tangent slope at that level gives the corresponding temperature, $\tau^* = dH_\theta/dK$ (Eq. (7)). For the budget used in training, the frozen geometry gives $\tau^* \approx 0.65$, close to the $\tau_t \approx 0.67$ maintained by the online dual tracker at the selected epoch. Shuffled or noisy utility signals fall below the frontier at comparable deviation, consistent with Theorem 3.5. The right panel illustrates the contamination result in Theorem 3.7. Here $\mathbb{E}\,\mathrm{Cov}_Q(Y, \ell_\theta) > 0$, so contamination increases as stronger utility reweighting favors harder pairs, while the entire path remains within $|\eta(Q) - \eta_S| \le \sqrt{K/2}$. Thus, reference deviation controls how far contamination can move from the reference level, but not the direction of that change.

5 Conclusion

PEPU incorporates external prior information into pairwise negative sampling: it leverages prior credibility to define a fixed reference and adopts current BPR utility for adaptive sampling around this reference, yielding a principled utility–reference-deviation formulation. Experiments on PU classification and recommendation reveal that the relative importance of prior credibility and model utility is task-dependent, and their combination consistently yields the best performance. More broadly, PEPU points to an expanded role for external priors in negative sampling, with future extensions enabling the reference itself to evolve alongside the learner.

AI USE STATEMENT

We used generative AI tools to assist with language editing and polishing of the manuscript, including improving grammar, clarity, and readability. All AI-assisted edits were reviewed and revised by the authors, who take full responsibility for the final content of the paper.

REFERENCES

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

Peter L. Bartlett, Michael I. Jordan, and Jon D. McAuliffe. Convexity, classification, and risk bounds. *Journal of the American Statistical Association*, 101(473):138–156, 2006.

Jessa Bekker and Jesse Davis. Learning from positive and unlabeled data: a survey. *Machine Learning*, 109(4):719–760, 2020.

Jessa Bekker, Pieter Robberechts, and Jesse Davis. Beyond the selected completely at random assumption for learning from positive and unlabeled data. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2019, Würzburg, Germany, September 16–20, 2019, Proceedings, Part II*, pp. 71–85, Berlin, Heidelberg, 2019. Springer-Verlag. ISBN 978-3-030-46146-1. doi: 10.1007/978-3-030-46147-8_5.

Aharon Ben-Tal, Dick den Hertog, Anja M. B. De Waegenaere, Bertrand Melenberg, and Gijs Rennen. Robust solutions of optimization problems affected by uncertain probabilities. *Manag. Sci.*, 59: 341–357, 2013.

Gilles Blanchard, Gyemin Lee, and Clayton Scott. Semi-supervised novelty detection. *Journal of Machine Learning Research*, 11(99):2973–3009, 2010.

Stéphane Boucheron, Gábor Lugosi, and Pascal Massart. *Concentration Inequalities: A Nonasymptotic Theory of Independence*. OUP Oxford, 2013.

Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

Luigi Cerulo, Charles Elkan, and Michele Ceccarelli. Learning gene regulatory networks from only positive and unlabeled data. *BMC Bioinformatics*, 11:228, 2010.

Gaode Chen, Shicheng Wang, Shikun Li, Rui Huang, Xinghua Zhang, Yunze Luo, Shipeng Li, Shiming Ge, Ruina Sun, Yinjie Jiang, and Jun Zhang. Denoising implicit feedback for cold-start recommendation. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2*, KDD '26, pp. 7059–7069, New York, NY, USA, 2026a. Association for Computing Machinery. ISBN 9798400722592. doi: 10.1145/3770855.3818367.

Qiuyi Chen, Haiyang Zhang, Leqi Zhang, Changchun Li, Jia Wang, and Wei Wang. PU-BENCH: A UNIFIED BENCHMARK FOR RIGOROUS AND REPRODUCIBLE PU LEARNING. In *The Fourteenth International Conference on Learning Representations*, 2026b.

Xuxi Chen, Wuyang Chen, Tianlong Chen, Ye Yuan, Chen Gong, Kewei Chen, and Zhangyang Wang. Self-PU: Self boosted and calibrated positive-unlabeled training. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, pp. 1510–1519, 2020.

Hyewon Choi, Jooyoung Choi, Hansol Jang, Hyun Kim, Chulmin Yun, Changwook Jun, and Stanley Jungkyu Choi. ARHN: Answer-centric relabeling of hard negatives with open-source LLMs for dense retrieval. In *Proceedings of the 49th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pp. 213–222, 2026.

Ching-Yao Chuang, Joshua Robinson, Yen-Chen Lin, Antonio Torralba, and Stefanie Jegelka. Debiased contrastive learning. In *Advances in Neural Information Processing Systems 33 (NeurIPS)*, 2020.

Stéphan Clémençon, Gábor Lugosi, and Nicolas Vayatis. Ranking and empirical minimization of u-statistics. *Annals of Statistics*, 36:844–874, 2008.

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

Imre Csiszár. I-divergence geometry of probability distributions and minimization problems. *The Annals of Probability*, 3(1):146–158, 1975.

John M. Danskin. The theory of max-min, with applications. *SIAM Journal on Applied Mathematics*, 14(4):641–664, 1966.

Jingtao Ding, Yuhan Quan, Xiangnan He, Yong Li, and Depeng Jin. Reinforced negative sampling for recommendation with exposure data. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 2230–2236, 2019.

Jingtao Ding, Yuhan Quan, Quanming Yao, Yong Li, and Depeng Jin. Simplify and robustify negative sampling for implicit collaborative filtering. In *Advances in Neural Information Processing Systems 33 (NeurIPS)*, 2020.

Monroe D. Donsker and S. R. Srinivasa Varadhan. Asymptotic evaluation of certain Markov process expectations for large time. IV. *Communications on Pure and Applied Mathematics*, 36(2):183–212, 1983.

Marthinus Christoffel du Plessis, Gang Niu, and Masashi Sugiyama. Analysis of learning from positive and unlabeled data. In *Advances in Neural Information Processing Systems 27 (NeurIPS)*, pp. 703–711, 2014.

Marthinus Christoffel du Plessis, Gang Niu, and Masashi Sugiyama. Convex formulation for learning from positive and unlabeled data. In *Proceedings of the 32nd International Conference on Machine Learning (ICML)*, pp. 1386–1394, 2015.

John C. Duchi and Hongseok Namkoong. Learning models with uniform performance via distributionally robust optimization. *The Annals of Statistics*, 49(3):1378–1406, 2021.

Charles Elkan and Keith Noto. Learning classifiers from only positive and unlabeled data. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pp. 213–220, 2008.

Zeno Gantner, Lucas Drumond, Christoph Freudenthaler, and Lars Schmidt-Thieme. Personalized ranking for non-uniformly sampled items. In *Proceedings of KDD Cup 2011 competition, San Diego, CA, USA, 2011*, volume 18 of *JMLR Proceedings*, pp. 231–247. JMLR.org, 2012.

Wei Gao and Zhi-Hua Zhou. On the consistency of AUC pairwise optimization. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 939–945, 2015.

Saurabh Garg, Yifan Wu, Alexander J. Smola, Sivaraman Balakrishnan, and Zachary C. Lipton. Mixture proportion estimation and PU learning: A modern approach. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pp. 8532–8544, 2021.

Walter Gerych, Thomas Hartvigsen, Luke Buquicchio, Emmanuel Agu, and Elke Rundensteiner. Recovering the propensity score from biased positive unlabeled data. In *Proceedings of the 36th AAAI Conference on Artificial Intelligence (AAAI)*, pp. 6694–6702, 2022.

Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the 13th International Conference on Artificial Intelligence and Statistics (AISTATS)*, pp. 249–256, 2010.

Chen Gong, Qizhou Wang, Tongliang Liu, Bo Han, Jane You, Jian Yang, and Dacheng Tao. Instance-dependent positive and unlabeled learning with labeling bias estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(8):4163–4177, 2022.

James A. Hanley and Barbara J. McNeil. The meaning and use of the area under a receiver operating characteristic (ROC) curve. *Radiology*, 143(1):29–36, 1982.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yongdong Zhang, and Meng Wang. LightGCN: Simplifying and powering graph convolution network for recommendation. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pp. 639–648, 2020.

Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (GELUs). *arXiv preprint arXiv:1606.08415*, 2016.

Yifan Hu, Yehuda Koren, and Chris Volinsky. Collaborative filtering for implicit feedback datasets. In *Eighth IEEE International Conference on Data Mining (ICDM)*, pp. 263–272, 2008.

Tinglin Huang, Yuxiao Dong, Ming Ding, Zhen Yang, Wenzheng Feng, Xinyu Wang, and Jie Tang. MixGCF: An improved training method for graph neural network-based recommender systems. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, pp. 665–674, 2021.

Henry F. Inman and Edwin L. Bradley. The overlapping coefficient as a measure of agreement between probability distributions and point estimation of the overlap of two normal densities. *Communications in Statistics – Theory and Methods*, 18(10):3851–3874, 1989.

Shantanu Jain, Martha White, and Predrag Radivojac. Estimating the class prior and posterior from noisy positives and unlabeled data. In *Advances in Neural Information Processing Systems 29 (NeurIPS)*, pp. 2685–2693, 2016.

Kalervo Järvelin and Jaana Kekäläinen. Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems*, 20(4):422–446, 2002.

Edwin T. Jaynes. Information theory and statistical mechanics. *Physical Review*, 106:620–630, 1957.

Yannis Kalantidis, Mert Bülent Sariyildiz, Noé Pion, Philippe Weinzaepfel, and Diane Larlus. Hard negative mixing for contrastive learning. In *Advances in Neural Information Processing Systems 33 (NeurIPS)*, 2020.

Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen tau Yih. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 6769–6781, 2020.

Masahiro Kato, Takeshi Teshima, and Junya Honda. Learning from positive and unlabeled data with a selection bias. In *International Conference on Learning Representations (ICLR)*, 2019.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015.

Ryuichi Kiryo, Gang Niu, Marthinus Christoffel du Plessis, and Masashi Sugiyama. Positive-unlabeled learning with non-negative risk estimator. In *Advances in Neural Information Processing Systems 30 (NeurIPS)*, 2017.

Riwei Lai, Li Chen, Yuhan Zhao, Rui Chen, and Qilong Han. Disentangled negative sampling for collaborative filtering. In *Proceedings of the 16th ACM International Conference on Web Search and Data Mining (WSDM)*, pp. 96–104, 2023.

Riwei Lai, Rui Chen, Qilong Han, Chi Zhang, and Li Chen. Adaptive hardness negative sampling for collaborative filtering. In *Proceedings of the 38th AAAI Conference on Artificial Intelligence (AAAI)*, pp. 8645–8652, 2024.

Changchun Li, Ximing Li, Lei Feng, and Jihong Ouyang. Who is your right mixup partner in positive and unlabeled learning. In *International Conference on Learning Representations (ICLR)*, 2022a.

Fuyi Li, Shuangyu Dong, André Leier, Meiya Han, Xudong Guo, Jing Xu, Xiaoyu Wang, Shirui Pan, Cangzhi Jia, Yang Zhang, Geoffrey I. Webb, Lachlan J. M. Coin, Chen Li, and Jiangning Song. Positive-unlabeled learning in bioinformatics and computational biology: a brief review. *Briefings in Bioinformatics*, 23(1):bbab461, 2022b.

Na Li, Fanghui Sun, Yan Zou, Yangfu Zhu, Xiatian Zhu, and Ying Ma. Adaptive diffusion-based augmentation for recommendation. In *Proceedings of the 40th AAAI Conference on Artificial Intelligence (AAAI)*, pp. 15117–15125, 2026.

Lin Long, Haobo Wang, Zhijie Jiang, Lei Feng, Chang Yao, Gang Chen, and Junbo Zhao. Positive-unlabeled learning by latent group-aware meta disambiguation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 23138–23147, 2024.

Babak Loni, Roberto Pagano, Martha Larson, and Alan Hanjalic. Bayesian personalized ranking with multi-channel user feedback. In *Proceedings of the 10th ACM Conference on Recommender Systems (RecSys)*, pp. 361–364, 2016.

Haokai Ma, Ruobing Xie, Lei Meng, Xin Chen, Xu Zhang, Leyu Lin, and Zhanhui Kang. Plug-in diffusion model for sequential recommendation. In *Proceedings of the 38th AAAI Conference on Artificial Intelligence (AAAI)*, pp. 8886–8894, 2024.

Haokai Ma, Ruobing Xie, Lei Meng, Fuli Feng, Xiaoyu Du, Xingwu Sun, Zhanhui Kang, and Xiangxu Meng. Negative sampling in recommendation: A survey and future directions. *ACM Transactions on Information Systems*, 44(4), 2026.

Paul Milgrom and Ilya Segal. Envelope theorems for arbitrary choice sets. *Econometrica*, 70(2): 583–601, 2002.

Hongseok Namkoong and John C. Duchi. Stochastic gradient methods for distributionally robust optimization with f-divergences. In *Advances in Neural Information Processing Systems 29 (NeurIPS)*, pp. 2208–2216, 2016.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jégou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024.

Lorenzo Perini, Vincent Vercruyssen, and Jesse Davis. Learning from positive and unlabeled multi-instance bags in anomaly detection. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, pp. 1897–1906, 2023.

Mark S. Pinsker. *Information and Information Stability of Random Variables and Processes*. Holden-Day, 1964.

Yingqi Qu, Yuchen Ding, Jing Liu, Kai Liu, Ruiyang Ren, Wayne Xin Zhao, Daxiang Dong, Hua Wu, and Haifeng Wang. RocketQA: An optimized training approach to dense passage retrieval for open-domain question answering. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pp. 5835–5847, 2021.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*, pp. 8748–8763, 2021.

Hamed Rahimian and Sanjay Mehrotra. Frameworks and results in distributionally robust optimization. *Open Journal of Mathematical Optimization*, 3:1–85, 2022.

Harish G. Ramaswamy, Clayton Scott, and Ambuj Tewari. Mixture proportion estimation via kernel embeddings of distributions. In *Proceedings of the 33rd International Conference on Machine Learning (ICML)*, pp. 2052–2060, 2016.

Xubin Ren, Wei Wei, Lianghao Xia, Lixin Su, Suqi Cheng, Junfeng Wang, Dawei Yin, and Chao Huang. Representation learning with large language models for recommendation. In *Proceedings of the ACM Web Conference 2024 (WWW)*, pp. 3464–3475, 2024.

Steffen Rendle and Christoph Freudenthaler. Improving pairwise learning for item recommendation from implicit feedback. In *Proceedings of the 7th ACM International Conference on Web Search and Data Mining (WSDM)*, pp. 273–282, 2014.

Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. BPR: Bayesian personalized ranking from implicit feedback. In *Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence (UAI)*, pp. 452–461, 2009.

Joshua Robinson, Ching-Yao Chuang, Suvrit Sra, and Stefanie Jegelka. Contrastive learning with hard negative samples. In *International Conference on Learning Representations (ICLR)*, 2021.

David Rogers and Mathew Hahn. Extended-connectivity fingerprints. *Journal of Chemical Information and Modeling*, 50(5):742–754, 2010.

Lukas Ruff, Robert A. Vandermeulen, Nico Görnitz, Alexander Binder, Emmanuel Müller, Klaus-Robert Müller, and Marius Kloft. Deep semi-supervised anomaly detection. In *International Conference on Learning Representations (ICLR)*, 2020.

Tomoya Sakai, Gang Niu, and Masashi Sugiyama. Semi-supervised AUC optimization based on positive-unlabeled learning. *Machine Learning*, 107(4):767–794, 2018.

David W. Scott. *Multivariate Density Estimation: Theory, Practice, and Visualization*. Wiley, 1992.

Wentao Shi, Jiawei Chen, Fuli Feng, Jizhi Zhang, Junkang Wu, Chongming Gao, and Xiangnan He. On the theories behind hard negative sampling for recommendation. In *Proceedings of the ACM Web Conference 2023 (WWW)*, pp. 812–822, 2023.

Tianrui Song, Wen-Shuo Chao, and Hao Liu. Hard vs. noise: Resolving hard-noisy sample confusion in recommender systems via large language models. In *Proceedings of the 40th AAAI Conference on Artificial Intelligence (AAAI)*, pp. 15743–15751, 2026.

Jun Wang, Lantao Yu, Weinan Zhang, Yu Gong, Yinghui Xu, Benyou Wang, Peng Zhang, and Dell Zhang. IRGAN: A minimax game for unifying generative and discriminative information retrieval models. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pp. 515–524, 2017.

Shuyao Wang, Zhi Zheng, Yongduo Sui, and Hui Xiong. Unleashing the power of large language model for denoising recommendation. In *Proceedings of the ACM Web Conference 2025 (WWW)*, pp. 252–263, 2025a.

Wei Wang, Dong-Dong Wu, Ming Li, Jingxiong Zhang, Gang Niu, and Masashi Sugiyama. Accessible, realistic, and fair evaluation of positive-unlabeled learning algorithms. In *International Conference on Learning Representations (ICLR)*, 2026.

Wenjie Wang, Fuli Feng, Xiangnan He, Liqiang Nie, and Tat-Seng Chua. Denoising implicit feedback for recommendation. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining (WSDM)*, pp. 373–381, 2021.

Xiang Wang, Yaokun Xu, Xiangnan He, Yixin Cao, Meng Wang, and Tat-Seng Chua. Reinforced negative sampling over knowledge graph for recommendation. In *Proceedings of The Web Conference 2020 (WWW)*, pp. 99–109, 2020.

Xinrui Wang, Wenhai Wan, Chuanxing Geng, Shaoyuan Li, and Songcan Chen. Beyond myopia: Learning from positive and unlabeled data through holistic predictive trends. In *Advances in Neural Information Processing Systems 36 (NeurIPS)*, 2023a.

Yaoshu Wang, Mengyi Yan, and Wei Wang. PUER: Boosting few-shot positive-unlabeled entity resolution with reinforcement learning. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 24567–24579, 2025b.

Ye Wang, Huazheng Pan, Tao Zhang, Wen Wu, and Wenxin Hu. A positive-unlabeled metric learning framework for document-level relation extraction with incomplete labeling. In *Proceedings of the 38th AAAI Conference on Artificial Intelligence (AAAI)*, pp. 19197–19205, 2024.

Zongwei Wang, Min Gao, Wentao Li, Junliang Yu, Linxin Guo, and Hongzhi Yin. Efficient bi-level optimization for recommendation denoising. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, pp. 2502–2511, 2023b.

Wei Wei, Xubin Ren, Jiabin Tang, Qinyong Wang, Lixin Su, Suqi Cheng, Junfeng Wang, Dawei Yin, and Chao Huang. LLMRec: Large language models with graph augmentation for recommendation. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining (WSDM)*, pp. 806–815, 2024.

Jiancan Wu, Xiang Wang, Fuli Feng, Xiangnan He, Liang Chen, Jianxun Lian, and Xing Xie. Self-supervised graph learning for recommendation. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pp. 726–735, 2021.

Zhen Yang, Ming Ding, Chang Zhou, Hongxia Yang, Jingren Zhou, and Jie Tang. Understanding negative sampling in graph representation learning. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pp. 1666–1676, 2020.

Zhen Yang, Ming Ding, Xu Zou, Jie Tang, Bin Xu, Chang Zhou, and Hongxia Yang. Region or global? A principle for negative sampling in graph-based recommendation. *IEEE Transactions on Knowledge and Data Engineering*, 35(6):6264–6277, 2023.

Junliang Yu, Hongzhi Yin, Xin Xia, Tong Chen, Lizhen Cui, and Quoc Viet Hung Nguyen. Are graph augmentations necessary? Simple graph contrastive learning for recommendation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pp. 1294–1303, 2022.

Junliang Yu, Xin Xia, Tong Chen, Lizhen Cui, Quoc Viet Hung Nguyen, and Hongzhi Yin. XSimGCL: Towards extremely simple graph contrastive learning for recommendation. *IEEE Transactions on Knowledge and Data Engineering*, 36(2):913–926, 2024.

Weinan Zhang, Tianqi Chen, Jun Wang, and Yong Yu. Optimizing top-n collaborative filtering via dynamic negative item sampling. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pp. 785–788, 2013.

Hengwei Zhao, Zhengzhong Tu, Zhuo Zheng, Wei Wang, Junjue Wang, Rusty Feagin, and Wenzhe Jiao. Noisy-pair robust representation alignment for positive-unlabeled learning. In *International Conference on Learning Representations (ICLR)*, 2026.

Yunrui Zhao, Qianqian Xu, Yangbangyan Jiang, Peisong Wen, and Qingming Huang. Dist-PU: Positive-unlabeled learning from a label distribution perspective. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14441–14450, 2022.

Zhangchi Zhu, Lu Wang, Pu Zhao, Chao Du, Wei Zhang, Hang Dong, Bo Qiao, Qingwei Lin, Saravan Rajmohan, and Dongmei Zhang. Robust positive-unlabeled learning via noise negative sample self-correction. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, pp. 3663–3673, 2023.

APPENDIX CONTENTS

A Theoretical Appendix: Definitions, Conditions, and Step-by-Step Proofs **17**

A.1 Pair-Level Utility Interpretation of the BPR Loss 17

A.2 Normalization and Degeneracy of the Reference Distribution 17

A.3 Variational Decomposition of the BPR Pair-Sampling Objective and the Gibbs Optimum 18

A.4 Complete Pareto Frontier of the Gibbs Family 20

A.5 Path Geometry, KL Budgets, and Scale Invariance of the Gibbs Family 21

A.6 One-Dimensional Dual for the Reference-Deviation Budget and Temperature Tracking 23

A.7 Gradient Consistency of Adaptive Gibbs Sampling 25

A.8 Measure-Theoretic Extension to General Measurable Spaces 26

A.9 Local Direction and Global Bounds for Hidden-Positive Contamination 27

A.10 Reference Quality, True-Negative Matching, and an AUC Certificate 28

A.11 Roles of the Symbols and Scope of the Theory 31

B Experimental Setup and Reproducibility Details **31**

B.1 Controlled PU Data-Generation Protocol 32

B.2 Matched Controlled Scorer Settings: PU w/o Encoder and PU w/ Encoder 36

B.3 CIFAR-PU: Visual Prior Construction 38

B.4 Recommendation: Semantic Prior Construction and Candidate-Pool Training . . . 41

B.5 Sampler Implementation, Training Protocol, and PU Baselines 43

B.6 Sensitivity to the Deviation Budget 44

B.7 Prior Scores, OVL Computation, and Controlled Prior Degradation 45

B.8 Pareto-Geometry Scan and Numerical Checks 46

C Additional Related Work **47**

C.1 Positive–Unlabeled Learning 48

C.2 Pairwise Learning and Negative Sampling 49

C.3 Mathematical Foundations of the Analysis 51

D Applications and Future Work **52**

D.1 Application Scenarios 52

D.2 Future Directions 53

A Theoretical Appendix: Definitions, Conditions, and Step-by-Step Proofs

This appendix provides complete derivations for the core theoretical results used in the main text. The main text writes Q_τ^{PE}, Z_τ, and $Q^{\max}$ for the PE sampler, its normalizer, and the maximum-utility endpoint; this appendix makes the dependence on the current parameters explicit and writes $Q_{\theta,\tau}$, $Z_{\theta,\tau}$, and $Q_\theta^{\max}$, and, because of its exponential-tilt form, refers to the PE sampling path $\{Q_{\theta,\tau} : \tau > 0\}$ as the Gibbs family. Unless stated otherwise, we fix the model parameter θ and impose the following standing assumptions. The positive-mass support of P_+ is finite; for every p in this support, the positive-mass support of $P_0(\cdot \mid p)$ is finite and nonempty; P_S is strictly positive on this support; and $\ell_\theta(p,u)$ is finite. All admissible conditional samplers $Q(\cdot \mid p)$ are defined on this common reachable support, equivalently $Q(\cdot \mid p) \ll P_S(\cdot \mid p)$. When differentiating with respect to θ, we further assume that $\ell_\theta(p,u)$ is continuously differentiable in θ. Finite support allows the main proofs to be written as transparent finite sums. Appendix A.8 gives the corresponding Gibbs/KL identities on a general measurable unlabeled space and discusses numerical integration of the resulting expectations for large or continuous supports.

A.1 Pair-Level Utility Interpretation of the BPR Loss

The main text uses ℓ_θ as a pair-level utility proxy. The justification comes from the local sensitivity of the BPR loss to the ranking margin: for a fixed pair, $|\partial\ell/\partial d|$ increases monotonically with the loss, indicating stronger corrective pressure on the current ranking margin. This property supports the training-time interpretation of utility, while credibility continues to be supplied independently by the prior.

From Eq. (2), for a fixed pair write $d = d_\theta(p,u)$ and $\ell = \log(1+e^{-d})$. Direct differentiation gives

$$\frac{\partial \ell}{\partial d} = \frac{1}{1+e^{-d}} \cdot (-e^{-d}) = -\frac{e^{-d}}{1+e^{-d}} = -\frac{1}{1+e^{d}}. \tag{A.1}$$

On the other hand, $e^{\ell} = 1 + e^{-d}$ implies

$$e^{-\ell} = \frac{1}{1+e^{-d}}, \tag{A.2}$$

$$1 - e^{-\ell} = \frac{e^{-d}}{1+e^{-d}} = \frac{1}{1+e^{d}}. \tag{A.3}$$

Hence

$$\left|\frac{\partial \ell}{\partial d}\right| = 1 - e^{-\ell}. \tag{A.4}$$

The derivative of the right-hand side with respect to ℓ is $e^{-\ell} > 0$. Therefore, a larger BPR loss corresponds to stronger local corrective pressure on the ranking margin. Importantly,

$$\nabla_\theta \ell_\theta(p,u) = \frac{\partial \ell}{\partial d} \nabla_\theta d_\theta(p,u), \tag{A.5}$$

so the norm of the parameter gradient also depends on $\nabla_\theta d_\theta$. We therefore use ℓ_θ only as a *pair-level utility/hardness signal*; its numerical value is not identified with the parameter-gradient norm. This is the reading of ℓ_θ adopted in Definition 3.2.

A.2 Normalization and Degeneracy of the Reference Distribution

Definition 3.1 constructs the reference distribution by reweighting P_0 with S. The calculations below verify normalization and establish the degeneracy property under a non-informative prior. In addition, $S > 0$ ensures that P_S and P_0 have the same null sets, so the reachable support of the base proposal is preserved.

Because $S(p,u) > 0$ and $0 < Z_S(p) < \infty$, Eq. (1) defines a strictly positive density relative to P_0. Under the finite discrete support used in the main text,

$$\sum_u P_S(u \mid p) = \frac{1}{Z_S(p)} \sum_u S(p,u) P_0(u \mid p) \tag{A.6}$$

$$= \frac{Z_S(p)}{Z_S(p)} = 1. \tag{A.7}$$

Thus $P_S(\cdot \mid p)$ is a valid conditional probability distribution. Since $S(p,u) > 0$, it has the same null sets as $P_0(\cdot \mid p)$. If, for a fixed p, $S(p,u) = c(p) > 0$ P_0-almost surely, then

$$Z_S(p) = c(p), \tag{A.8}$$

$$P_S(u \mid p) = \frac{c(p)}{c(p)} P_0(u \mid p) = P_0(u \mid p). \tag{A.9}$$

Hence, when the prior has no relative discriminative power, the method reduces exactly to the base proposal and introduces no additional change in the sampling distribution. The two calculations of this section are the well-posedness of Definition 3.1 and the degeneracy $P_S = P_0$ recorded after it.

A.3 Variational Decomposition of the BPR Pair-Sampling Objective and the Gibbs Optimum

The main text starts from the BPR pair-sampling problem and obtains Eq. (5), where the model parameter θ is fixed and the optimization variable is the conditional sampler Q used to form training pairs. We first solve this constrained optimization problem directly. The shared positive marginal decomposes the objective row by row in p, and the Lagrange first-order conditions on each probability simplex yield the exponential reweighting form. We then use an exact KL variational decomposition to verify global uniqueness and obtain the optimal value. Thus the Gibbs form is an optimization consequence rather than an assumed parametric family.

Using the common positive marginal of $J_Q = P_+ Q$ and $J_S = P_+ P_S$, the joint KL divergence expands as

$$\mathcal{K}(Q) = \sum_p \sum_u P_+(p) Q(u \mid p) \log \frac{P_+(p) Q(u \mid p)}{P_+(p) P_S(u \mid p)} \tag{A.10}$$

$$= \sum_p P_+(p) \sum_u Q(u \mid p) \log \frac{Q(u \mid p)}{P_S(u \mid p)} \tag{A.11}$$

$$= \mathbb{E}_{P_+} D_{\mathrm{KL}}(Q(\cdot \mid p) \| P_S(\cdot \mid p)) \,. \tag{A.12}$$

Therefore Eq. (5) can be solved separately for each p with positive P_+ mass. Fix such a p and write

$$r_u = P_S(u \mid p), \qquad l_u = \ell_\theta(p,u), \qquad q_u = Q(u \mid p).$$

The corresponding conditional objective is

$$f_p(q) = \sum_u q_u l_u - \tau \sum_u q_u \log \frac{q_u}{r_u}, \qquad q_u \geq 0, \quad \sum_u q_u = 1.$$

Under the finite positive-support assumptions of the main text, $r_u > 0$ and l_u is finite. The function $-D_{\mathrm{KL}}(q \| r)$ is strictly concave on the probability simplex. Moreover, because the derivative of the entropy contribution diverges toward $+\infty$ as any $q_u \downarrow 0$, the maximizer has strictly positive mass on every reachable coordinate. We may therefore introduce only the normalization multiplier λ_p:

$$\mathcal{L}_p(q, \lambda_p) = \sum_u q_u l_u - \tau \sum_u q_u \log \frac{q_u}{r_u} + \lambda_p \left(\sum_u q_u - 1 \right).$$

The first-order condition for each q_u is

$$0 = \frac{\partial \mathcal{L}_p}{\partial q_u} \tag{A.13}$$

$$= l_u - \tau \left(\log \frac{q_u}{r_u} + 1 \right) + \lambda_p, \tag{A.14}$$

so

$$\log \frac{q_u}{r_u} = \frac{l_u}{\tau} + \frac{\lambda_p}{\tau} - 1.$$

Let $C_p = \exp(\lambda_p/\tau - 1)$. Then

$$q_u = C_p \, r_u e^{l_u/\tau}.$$

Imposing $\sum_u q_u = 1$ gives

$$1 = C_p \sum_u r_u e^{l_u/\tau}, \qquad C_p = \frac{1}{Z_\tau(p)},$$

where

$$Z_\tau(p) = \sum_u r_u e^{l_u/\tau}, \qquad q_u^\tau = \frac{r_u e^{l_u/\tau}}{Z_\tau(p)}. \tag{A.15}$$

Thus the Lagrange first-order conditions directly imply

$$\log \frac{q_u^\tau}{r_u} = \frac{l_u}{\tau} - \log Z_\tau(p), \tag{A.16}$$

i.e., q_u^τ is the normalized exponential reweighting of the reference r_u by the utility l_u. Since f_p is strictly concave, this stationary point is the unique global maximizer of the conditional problem for the fixed p.

The Lagrange calculation above explains where the exponential form comes from. We next give an exact KL variational decomposition that expresses the objective gap between an arbitrary sampler Q and the optimal sampler as a nonnegative KL remainder, while simultaneously yielding the optimal-value expression used in the main text.

Lemma A.1 (KL variational decomposition of the BPR pair-sampling objective)**.** For any conditional distribution Q and any $\tau > 0$,

$$\mathcal{H}_\theta(Q) - \tau \mathcal{K}(Q) = \tau \mathbb{E}_{P_+} \log Z_\tau(p) - \tau \mathbb{E}_{P_+} D_{\mathrm{KL}}(Q(\cdot \mid p) \| Q_{\theta,\tau}(\cdot \mid p)) \, . \tag{A.17}$$

Proof. Fix p. By Eq. (A.16),

$$D_{\mathrm{KL}}(q \| q^\tau) = \sum_u q_u \log \frac{q_u}{q_u^\tau} \tag{A.18}$$

$$= \sum_u q_u \left(\log \frac{q_u}{r_u} - \frac{l_u}{\tau} + \log Z_\tau(p) \right) \tag{A.19}$$

$$= D_{\mathrm{KL}}(q \| r) - \frac{1}{\tau} \sum_u q_u l_u + \log Z_\tau(p), \tag{A.20}$$

where the last equality uses $\sum_u q_u = 1$. Multiplying by $-\tau$ and rearranging gives

$$\sum_u q_u l_u - \tau D_{\mathrm{KL}}(q \| r) = \tau \log Z_\tau(p) - \tau D_{\mathrm{KL}}(q \| q^\tau). \tag{A.21}$$

Taking expectation over $p \sim P_+$ and using Eq. (A.12) proves Eq. (A.17). □

By nonnegativity of KL divergence, the right-hand side of Eq. (A.17) is maximized only when $Q(\cdot \mid p) = Q_{\theta,\tau}(\cdot \mid p)$ for P_+-almost every p. Hence

$$Q^*_{\theta,\tau}(u \mid p) = \frac{P_S(u \mid p) e^{\ell_\theta(p,u)/\tau}}{Z_{\theta,\tau}(p)}$$

is the unique maximizer of Eq. (5), up to P_+-null anchors, and the optimal value is

$$\max_Q \{ \mathcal{H}_\theta(Q) - \tau \mathcal{K}(Q) \} = \tau \mathbb{E}_{P_+} \log Z_{\theta,\tau}(p).$$

This agrees with the strict-concavity argument above and gives the exact KL gap between any candidate sampler and the optimum. The displayed maximizer is Eq. (6), so Proposition 3.4 follows.

A.4 Complete Pareto Frontier of the Gibbs Family

The main text has established that, for every fixed $\tau > 0$, the unique optimizer of Eq. (5) has Gibbs form. Varying τ traces the Gibbs family. This section further proves that this family, together with its two limiting endpoints, gives exactly all Pareto-efficient samplers for the utility–reference-deviation bi-objective problem, rather than only a subset of them.

Introduce the inverse temperature $\beta = 1/\tau$ and write the exponential tilt as

$$q_{\beta,p}(u) = \frac{r_p(u)e^{\beta l_p(u)}}{Z_p(\beta)}, \qquad Z_p(\beta) = \sum_u r_p(u)e^{\beta l_p(u)}. \tag{A.22}$$

Let

$$M_\theta(p) = \arg\max_{u \in \operatorname{supp} r_p} l_p(u), \qquad m_p = \max_u l_p(u),$$

and define

$$q_{\max,p}(u) = \frac{r_p(u)\mathbf{1}\{u \in M_\theta(p)\}}{r_p(M_\theta(p))}. \tag{A.23}$$

The completeness argument contains both sufficiency and necessity. We first characterize the two limiting endpoints and verify Pareto efficiency of every finite-temperature point. We then use the KKT conditions for the KL-budget problem to show that every interior Pareto point lies on the Gibbs path, and finally handle the saturated region to exclude any additional Pareto-efficient sampler.

Theorem A.2 (Exact characterization of the complete Pareto frontier)**.** Suppose there exists at least one p with positive P_+ mass and two candidates u_1, u_2 with positive r_p mass such that $l_p(u_1) \neq l_p(u_2)$. Then

$$\{Q_{\theta,\tau} : \tau \in (0,\infty)\} \cup \{P_S, Q_\theta^{\max}\}$$

is exactly the set of Pareto-efficient samplers for the bi-objective problem of maximizing $\mathcal{H}_\theta(Q)$ and minimizing $\mathcal{K}(Q)$. Its image in the $(\mathcal{K}, \mathcal{H}_\theta)$ plane is therefore the complete Pareto frontier.

Proof. The argument has five steps. Steps 1 and 2 identify the two endpoints of the family, Step 3 shows that every finite temperature is efficient, Step 4 solves the KL-budget problem, and Step 5 uses that solution to show that nothing outside the family is efficient.

Step 1: Endpoint limits. As $\beta \downarrow 0$, $e^{\beta l_p(u)} \to 1$, and finite summation permits termwise passage to the limit. Hence

$$q_{\beta,p}(u) \to r_p(u) = P_S(u \mid p). \tag{A.24}$$

As $\beta \to \infty$, divide both numerator and denominator of Eq. (A.22) by $e^{\beta m_p}$:

$$q_{\beta,p}(u) = \frac{r_p(u)e^{\beta(l_p(u)-m_p)}}{\sum_v r_p(v)e^{\beta(l_p(v)-m_p)}}. \tag{A.25}$$

If $u \notin M_\theta(p)$, then $l_p(u) - m_p < 0$ and the corresponding exponential term converges to 0; if $u \in M_\theta(p)$, the exponential term is identically 1. Therefore

$$q_{\beta,p}(u) \to \frac{r_p(u)\mathbf{1}\{u \in M_\theta(p)\}}{r_p(M_\theta(p))} = q_{\max,p}(u). \tag{A.26}$$

Step 2: Minimum-KL representation among maximum-utility samplers. Clearly,

$$\mathcal{H}_\theta(Q) \leq \mathbb{E}_{P_+} m_p =: H_{\max,\theta}, \tag{A.27}$$

with equality if and only if, for P_+-almost every p, the support of $Q(\cdot \mid p)$ is contained in $M_\theta(p)$. For such a row q, let $r_M = r_p(M_\theta(p))$ and $r_p^M(u) = r_p(u)/r_M$ for $u \in M_\theta(p)$. Then

$$D_{\mathrm{KL}}(q\|r_p) = \sum_{u \in M} q_u \log \frac{q_u}{r_p(u)} \tag{A.28}$$

$$= \sum_{u \in M} q_u \log \frac{q_u}{r_p^M(u)} + \sum_{u \in M} q_u \log \frac{1}{r_M} \tag{A.29}$$

$$= D_{\mathrm{KL}}(q\|r_p^M) - \log r_M \tag{A.30}$$

$$\geq -\log r_M, \tag{A.31}$$

with equality if and only if $q = r_p^M = q_{\max,p}$. Averaging over P_+ shows that, among all samplers attaining $H_{\max,\theta}$, $Q_\theta^{\max}$ uniquely minimizes KL divergence. Define

$$\delta_{\text{sat}}(\theta) = \mathcal{K}(Q_\theta^{\max}) = \mathbb{E}_{P_+}\big[-\log P_S(M_\theta(p) \mid p)\big]. \tag{A.32}$$

Step 3: Pareto efficiency of finite-temperature points. Fix $\tau > 0$. Suppose there were a Q' such that

$$\mathcal{K}(Q') \le \mathcal{K}(Q_{\theta,\tau}), \qquad \mathcal{H}_\theta(Q') \ge \mathcal{H}_\theta(Q_{\theta,\tau}),$$

with at least one inequality strict. Since $\tau > 0$, this would imply

$$\mathcal{H}_\theta(Q') - \tau\mathcal{K}(Q') > \mathcal{H}_\theta(Q_{\theta,\tau}) - \tau\mathcal{K}(Q_{\theta,\tau}), \tag{A.33}$$

contradicting the unique optimality of $Q_{\theta,\tau}$ for the scalarized objective established in Appendix A.3. Hence every finite-temperature point is Pareto efficient. The endpoint P_S is Pareto efficient because $\mathcal{K}(Q) \ge 0$, with equality only when $Q = P_S$ up to P_+-null anchors. The endpoint $Q_\theta^{\max}$ is Pareto efficient by Step 2. Every member of the displayed set is therefore efficient, and what remains is the reverse inclusion. The next two steps obtain it through the KL-budget problem.

Step 4: Gibbs representation of the KL-budget problem. For any $0 < \delta < \delta_{\text{sat}}(\theta)$, consider

$$W_\delta(\theta) = \sup_{Q:\mathcal{K}(Q)\le\delta} \mathcal{H}_\theta(Q). \tag{A.34}$$

The feasible set is the intersection of a finite-dimensional product of probability simplices with a KL sublevel set, and is therefore compact and convex. Since P_S has strictly positive mass on every reachable coordinate and satisfies $\mathcal{K}(P_S) = 0 < \delta$, Slater's condition holds relative to the affine normalization constraints. Consequently, the KKT conditions are necessary and sufficient for this convex program, equivalently for minimizing the linear function $-\mathcal{H}_\theta(Q)$. Let $\lambda \ge 0$ be an optimal multiplier for the KL constraint. If $\lambda = 0$, an optimizer would have to maximize $\mathcal{H}_\theta$ without the KL constraint and would therefore attain $H_{\max,\theta}$. Step 2 shows that attaining this value requires KL divergence at least $\delta_{\text{sat}}(\theta) > \delta$, a contradiction. Hence $\lambda > 0$, and complementary slackness gives $\mathcal{K}(Q^*) = \delta$. For this fixed λ, Q^* must maximize

$$\mathcal{H}_\theta(Q) - \lambda\mathcal{K}(Q).$$

By the uniqueness in Lemma A.1, $Q^* = Q_{\theta,\lambda}$. Every interior budget is thus met by a point of the Gibbs path, which is the fact the final step uses.

Step 5: Completeness of the Pareto set. Take any Pareto-efficient Q and let $\delta = \mathcal{K}(Q)$. If $\delta = 0$, then $Q = P_S$ up to P_+-null anchors. If $0 < \delta < \delta_{\text{sat}}$, let Q^δ denote the unique optimizer of Eq. (A.34). If $Q \ne Q^\delta$, uniqueness implies

$$\mathcal{K}(Q^\delta) \le \delta = \mathcal{K}(Q), \qquad \mathcal{H}_\theta(Q^\delta) > \mathcal{H}_\theta(Q),$$

so Q would be strictly dominated, contradicting Pareto efficiency. Therefore $Q = Q^\delta$, and Step 4 places it on the Gibbs path. Finally, if $\delta \ge \delta_{\text{sat}}$, then $Q_\theta^{\max}$ satisfies

$$\mathcal{K}(Q_\theta^{\max}) = \delta_{\text{sat}} \le \delta, \qquad \mathcal{H}_\theta(Q_\theta^{\max}) = H_{\max,\theta} \ge \mathcal{H}_\theta(Q).$$

Pareto efficiency forces both inequalities to be equalities. Step 2 then implies, by uniqueness of the minimum-KL maximum-utility sampler, that $Q = Q_\theta^{\max}$. This proves completeness. □

The set characterized in Theorem A.2 is the one stated in Theorem 3.5 of the main text, which is therefore proved.

A.5 Path Geometry, KL Budgets, and Scale Invariance of the Gibbs Family

The main text shows that the Gibbs family forms the complete utility–reference-deviation Pareto frontier in the $(\mathcal{K}, \mathcal{H}_\theta)$ plane. We now characterize the geometry of this path. Reference deviation is strictly monotone in the inverse temperature β, the temperature τ equals the local slope of the Pareto curve, and the interior frontier is strictly concave. These properties further imply a one-to-one correspondence between every interior KL budget and a Gibbs temperature, together with the associated scale invariance.

Continue to use Eq. (A.22). For fixed p,

$$\log q_{\beta,p}(u) = \log r_p(u) + \beta l_p(u) - \log Z_p(\beta), \tag{A.35}$$

$$\frac{d}{d\beta} \log Z_p(\beta) = \frac{\sum_u r_p(u) e^{\beta l_p(u)} l_p(u)}{Z_p(\beta)} = \mathbb{E}_{q_{\beta,p}} l_p. \tag{A.36}$$

Therefore

$$\frac{d}{d\beta} \log q_{\beta,p}(u) = l_p(u) - \mathbb{E}_{q_{\beta,p}} l_p. \tag{A.37}$$

Since $q' = q(\log q)'$,

$$\frac{d}{d\beta} \mathbb{E}_{q_{\beta,p}} l_p = \sum_u l_p(u) q_{\beta,p}(u) \left(l_p(u) - \mathbb{E}_q l_p\right) \tag{A.38}$$

$$= \mathbb{E}_q[l_p^2] - (\mathbb{E}_q l_p)^2 \tag{A.39}$$

$$= \mathrm{Var}_q(l_p). \tag{A.40}$$

Averaging over p, define

$$H_\theta(\beta) = \mathbb{E}_{P_+} \mathbb{E}_{q_{\beta,p}} l_p, \qquad V_\theta(\beta) = \mathbb{E}_{P_+} \mathrm{Var}_{q_{\beta,p}}(l_p). \tag{A.41}$$

Then

$$H'_\theta(\beta) = V_\theta(\beta). \tag{A.42}$$

Next expand the KL divergence. Since $\log(q_{\beta,p}/r_p) = \beta l_p - \log Z_p(\beta)$,

$$K_p(\beta) := D_{\mathrm{KL}}(q_{\beta,p} \| r_p) \tag{A.43}$$

$$= \sum_u q_{\beta,p}(u) \big(\beta l_p(u) - \log Z_p(\beta)\big) \tag{A.44}$$

$$= \beta \mathbb{E}_q l_p - \log Z_p(\beta). \tag{A.45}$$

Differentiating and using Eqs. (A.36)–(A.40),

$$K'_p(\beta) = \mathbb{E}_q l_p + \beta \,\mathrm{Var}_q(l_p) - \mathbb{E}_q l_p \tag{A.46}$$

$$= \beta \,\mathrm{Var}_q(l_p). \tag{A.47}$$

Hence

$$K'_\theta(\beta) = \beta V_\theta(\beta) \geq 0. \tag{A.48}$$

Under the nondegeneracy condition of Theorem A.2, at least one positive-mass anchor contains two positive-r_p candidates with different losses. For every finite β, the corresponding $q_{\beta,p}$ has full reachable support, so that row has strictly positive loss variance. Therefore $V_\theta(\beta) > 0$ and $K'_\theta(\beta) > 0$ for every $\beta > 0$: the reference deviation is strictly increasing along the interior path.

Whenever $\beta > 0$ and $V_\theta(\beta) > 0$, the chain rule yields

$$\frac{dH_\theta}{dK_\theta} = \frac{H'_\theta(\beta)}{K'_\theta(\beta)} = \frac{V_\theta(\beta)}{\beta V_\theta(\beta)} = \frac{1}{\beta} = \tau, \tag{A.49}$$

$$\frac{d^2 H_\theta}{dK_\theta^2} = \frac{d(1/\beta)/d\beta}{K'_\theta(\beta)} = \frac{-1/\beta^2}{\beta V_\theta(\beta)} = -\frac{1}{\beta^3 V_\theta(\beta)} < 0. \tag{A.50}$$

Thus τ is also the tangent slope of the Pareto curve at the corresponding point, and the interior frontier is strictly concave; Eqs. (A.49) and (A.50) are the two statements collected in Eq. (7). This geometric interpretation is a supplementary property of the Gibbs family and is not required for the definition of the method.

By the endpoint limits in Appendix A.4 and continuity of finite sums,

$$K_\theta(0) = 0, \qquad \lim_{\beta \to \infty} K_\theta(\beta) = \delta_{\mathrm{sat}}(\theta). \tag{A.51}$$

Together with strict monotonicity, this implies that for every $0 < \delta < \delta_{\mathrm{sat}}$ there exists a unique β^* satisfying $K_\theta(\beta^*) = \delta$. By Step 4 of Theorem A.2, this Gibbs point is exactly the unique optimizer of the KL-budget problem.

If a KL budget is used as the training-time coordinate for sampling strength, the monotonicity above guarantees that every interior budget corresponds to a unique Gibbs temperature. The same temperature is also the Lagrange multiplier of the constrained optimization problem and the tangent slope of the Pareto frontier.

Theorem A.3 (Equivalence of budget, tangent slope, and Lagrange multiplier)**.** For $0 < \delta < \delta_{\mathrm{sat}}(\theta)$, the unique optimizer of the budget problem is $Q_{\theta,1/\beta^*}$, where $K_\theta(\beta^*) = \delta$. The corresponding $\tau^* = 1/\beta^*$ satisfies

$$\tau^* = \left.\frac{dH_\theta}{dK_\theta}\right|_{K=\delta},$$

and equals the optimal Lagrange multiplier of the KL constraint.

Proof. The first two claims follow from Eq. (A.49) and the budget optimality established above. For the multiplier claim, the Lagrangian of the budget problem is

$$L(Q,\lambda) = \mathcal{H}_\theta(Q) - \lambda\big(\mathcal{K}(Q) - \delta\big) = \big[\mathcal{H}_\theta(Q) - \lambda\mathcal{K}(Q)\big] + \lambda\delta. \tag{A.52}$$

For fixed $\lambda > 0$, Lemma A.1 shows that the bracketed term is uniquely maximized by $Q_{\theta,\lambda}$. At an interior-budget optimum the constraint is active, so the KKT conditions require $\mathcal{K}(Q_{\theta,\lambda}) = \delta$. Since $K_\theta(1/\lambda)$ is strictly monotone, this λ is unique and equals $\tau^* = 1/\beta^*$. □

Scale invariance. Let $l'_p(u) = a l_p(u) + b(p)$ with $a > 0$, and set $\tau' = a\tau$. Then

$$\frac{r_p(u)e^{l'_p(u)/\tau'}}{\sum_v r_p(v)e^{l'_p(v)/\tau'}} = \frac{r_p(u)e^{l_p(u)/\tau}e^{b(p)/(a\tau)}}{\sum_v r_p(v)e^{l_p(v)/\tau}e^{b(p)/(a\tau)}} \tag{A.53}$$

$$= \frac{r_p(u)e^{l_p(u)/\tau}}{\sum_v r_p(v)e^{l_p(v)/\tau}}. \tag{A.54}$$

Thus the same distributional point Q has the same KL coordinate. Moreover, the additive term $b(p)$ contributes the constant $\mathbb{E}_{P_+} b(p)$ to the budget objective, while the positive factor a only rescales utility; therefore the same KL budget selects the same optimizer Q^*. Only the corresponding temperature rescales, as $\tau^{*\prime} = a\tau^*$. Hence a reference-deviation budget controls the actual distributional displacement rather than the numerical scale of the loss.

A.6 One-Dimensional Dual for the Reference-Deviation Budget and Temperature Tracking

When training uses δ_{tr} as a reference-deviation budget, the distribution-level operating point must be translated into the Gibbs temperature τ. The two are connected by a one-dimensional dual problem. Its derivative is exactly the residual between the target deviation and the current deviation, so the corresponding temperature can be tracked by a single scalar update.

For any $\delta \geq 0$, define

$$\Psi_\delta(\theta,\tau) = \tau\delta + \tau\mathbb{E}_{P_+}\log Z_{\theta,\tau}(p), \qquad \tau > 0, \tag{A.55}$$

where

$$Z_{\theta,\tau}(p) = \mathbb{E}_{P_S(\cdot|p)}\exp(\ell_\theta(p,u)/\tau)\,.$$

By Lemma A.1, for any Q and $\tau > 0$,

$$\mathcal{H}_\theta(Q) - \tau\mathcal{K}(Q) \leq \tau\mathbb{E}_{P_+}\log Z_{\theta,\tau}(p).$$

If $\mathcal{K}(Q) \leq \delta$, then

$$\mathcal{H}_\theta(Q) = \big[\mathcal{H}_\theta(Q) - \tau\mathcal{K}(Q)\big] + \tau\mathcal{K}(Q) \tag{A.56}$$

$$\leq \tau\mathbb{E}_{P_+}\log Z_{\theta,\tau}(p) + \tau\delta \tag{A.57}$$

$$= \Psi_\delta(\theta,\tau). \tag{A.58}$$

Therefore

$$W_\delta(\theta) \leq \inf_{\tau>0}\Psi_\delta(\theta,\tau). \tag{A.59}$$

For any interior budget $0 < \delta < \delta_{\mathrm{sat}}(\theta)$, Appendix A.5 establishes a unique $\tau^* > 0$ such that

$$\mathcal{K}(Q_{\theta,\tau^*}) = \delta.$$

Taking $Q = Q_{\theta,\tau^*}$ in the Gibbs identity makes the KL remainder vanish, and therefore

$$W_\delta(\theta) = \mathcal{H}_\theta(Q_{\theta,\tau^*}) \tag{A.60}$$
$$= \tau^* \mathbb{E}_{P_+} \log Z_{\theta,\tau^*}(p) + \tau^* \mathcal{K}(Q_{\theta,\tau^*}) \tag{A.61}$$
$$= \Psi_\delta(\theta, \tau^*). \tag{A.62}$$

Thus strong duality holds for every interior budget. At the endpoints, $\delta = 0$ and $\delta \geq \delta_{\mathrm{sat}}(\theta)$ are obtained respectively through the limits $\tau \to \infty$ and $\tau \to 0^+$; the infimum need not be attained at a finite positive temperature. Consequently,

$$W_\delta(\theta) = \inf_{\tau > 0} \Psi_\delta(\theta, \tau). \tag{A.63}$$

We next compute the one-dimensional dual derivative. For fixed p, write

$$r_u = P_S(u \mid p), \qquad l_u = \ell_\theta(p, u), \qquad q_u = Q_{\theta,\tau}(u \mid p),$$

and

$$Z = \sum_u r_u e^{l_u/\tau}, \qquad q_u = \frac{r_u e^{l_u/\tau}}{Z}.$$

Then

$$\frac{\partial}{\partial \tau} \log Z = -\frac{\mathbb{E}_q l}{\tau^2}, \tag{A.64}$$
$$\frac{\partial}{\partial \tau}\big(\tau \log Z\big) = \log Z - \frac{\mathbb{E}_q l}{\tau}. \tag{A.65}$$

On the other hand, Eq. (A.45) with $\beta = 1/\tau$ gives

$$D_{\mathrm{KL}}(q \| r) = \frac{\mathbb{E}_q l}{\tau} - \log Z. \tag{A.66}$$

Hence

$$\frac{\partial \Psi_\delta}{\partial \tau} = \delta + \mathbb{E}_{P_+}\left(\log Z - \frac{\mathbb{E}_q l}{\tau}\right) \tag{A.67}$$
$$= \delta - \mathbb{E}_{P_+} D_{\mathrm{KL}}(q \| r) \tag{A.68}$$
$$= \delta - K_\theta(1/\tau). \tag{A.69}$$

Here $K_\theta(1/\tau) = \mathcal{K}(Q_{\theta,\tau})$ is precisely the reference deviation of the current Gibbs sampler from P_S. Using $K'_\theta(\beta) = \beta V_\theta(\beta)$ from Appendix A.5 further yields

$$\frac{\partial^2 \Psi_\delta}{\partial \tau^2} = \frac{V_\theta(1/\tau)}{\tau^3} \geq 0.$$

Thus Ψ_δ is convex in τ and, under the nondegeneracy condition above, strictly convex for finite τ. Every interior budget therefore corresponds to a unique τ^*, whose first-order condition is exactly

$$K_\theta(1/\tau^*) = \delta.$$

For a fixed current model state, a gradient-descent step on this one-dimensional dual is therefore

$$\tau_{t+1} = \tau_t + b\big(K_t - \delta_{\mathrm{tr}}\big), \qquad K_t = \mathcal{K}(Q_{\theta_t,\tau_t}),$$

where $b > 0$ is a constant dual stepsize.

In practice, K_t is estimated on the mini-batch B_t of observed-positive anchors used at iteration t. For each $p \in B_t$, the conditional KL is available in closed form from Eq. (A.45), so

$$\widehat{K}_t = \frac{1}{|B_t|} \sum_{p \in B_t} D_{\mathrm{KL}}(Q_{\theta_t,\tau_t}(\cdot \mid p) \,\|\, P_S(\cdot \mid p))$$
$$= \frac{1}{|B_t|} \sum_{p \in B_t} \left[\frac{1}{\tau_t} \sum_u Q_{\theta_t,\tau_t}(u \mid p)\, \ell_{\theta_t}(p, u) - \log Z_{\theta_t,\tau_t}(p)\right], \tag{A.70}$$

where

$$Z_{\theta_t,\tau_t}(p) = \sum_u P_S(u \mid p) \exp\left(\frac{\ell_{\theta_t}(p,u)}{\tau_t}\right).$$

where the inner expectation and $Z_{\theta_t,\tau_t}(p)$ are computed over the candidate set of p (Appendix A.8 discusses large or continuous supports). Since the anchors are drawn from P_+, $\widehat{K}_t$ is an unbiased estimate of K_t. Replacing K_t by $\widehat{K}_t$ gives Eq. (8) in the main text:

$$\tau_{t+1} \approx \tau_t + b\big(\widehat{K}_t - \delta_{\mathrm{tr}}\big).$$

This is the update quoted as Eq. (8) in Section 3.2. If the current deviation exceeds the target, it increases τ and weakens the reweighting toward high-utility pairs; if the deviation is below the target, it decreases τ and strengthens that reweighting. The scalar update therefore tracks the reference-deviation operating point specified by δ_{tr}.

A.7 Gradient Consistency of Adaptive Gibbs Sampling

Section 3.3 defines $\Phi_\tau(\theta)$ as the optimal value of the inner pair-sampling problem at a fixed θ. We first differentiate its log-partition form directly and recover the standard BPR gradient under the Gibbs sampler. We then give the corresponding Danskin/envelope characterization for the KL-budget formulation.

For fixed τ, direct differentiation of the log-partition function produces normalized exponential weights that are exactly $Q_{\theta,\tau}$. Consequently, sampling from this distribution and computing the standard BPR gradient gives a Monte Carlo estimator of the exact value-function gradient.

Theorem A.4 (Exact gradient at fixed temperature)**.** Let

$$\Phi_\tau(\theta) = \tau \mathbb{E}_{P_+} \log Z_{\theta,\tau}(p),$$

and assume that P_S does not depend on the current θ. Then

$$\nabla_\theta \Phi_\tau(\theta) = \mathbb{E}_{P_+} \mathbb{E}_{Q_{\theta,\tau}} \nabla_\theta \ell_\theta(p,u). \tag{A.71}$$

Proof. For fixed p, finite summation permits termwise differentiation:

$$\nabla_\theta Z_{\theta,\tau}(p) = \sum_u P_S(u \mid p) e^{\ell_\theta(p,u)/\tau} \frac{1}{\tau} \nabla_\theta \ell_\theta(p,u). \tag{A.72}$$

Therefore

$$\nabla_\theta \big[\tau \log Z_{\theta,\tau}(p)\big] = \frac{\tau}{Z_{\theta,\tau}(p)} \nabla_\theta Z_{\theta,\tau}(p) \tag{A.73}$$

$$= \sum_u \frac{P_S(u \mid p) e^{\ell_\theta(p,u)/\tau}}{Z_{\theta,\tau}(p)} \nabla_\theta \ell_\theta(p,u) \tag{A.74}$$

$$= \mathbb{E}_{Q_{\theta,\tau}(\cdot|p)} \nabla_\theta \ell_\theta(p,u). \tag{A.75}$$

Averaging over $p \sim P_+$ proves Eq. (A.71). □

For the budget value function

$$W_{\delta_{\mathrm{tr}}}(\theta) = \max_{Q \in \mathcal{C}} \mathcal{H}_\theta(Q), \qquad \mathcal{C} = \{Q : \mathcal{K}(Q) \le \delta_{\mathrm{tr}}\},$$

note that $\mathcal{C}$ does not vary with θ because P_S and δ_{tr} are fixed. If the current solution is an interior-budget unique optimizer $Q^*(\theta)$, then the finite-dimensional differentiable form of Danskin's theorem (equivalently, the envelope theorem) gives

$$\nabla_\theta W_{\delta_{\mathrm{tr}}}(\theta) = \nabla_\theta \mathcal{H}_\theta(Q)\big|_{Q=Q^*(\theta)} = \mathbb{E}_{P_+} \mathbb{E}_{Q^*} \nabla_\theta \ell_\theta(p,u). \tag{A.76}$$

There is no $\nabla_\theta Q^*$ term: the envelope theorem states that, when differentiating the optimized value at an optimal inner distribution, only the explicit dependence of the outer integrand on θ contributes to the first-order derivative.

If $p \sim P_+$ and $u \sim Q^*(\cdot \mid p)$ are sampled and

$$g = \nabla_\theta \ell_\theta(p, u),$$

then, by conditional expectation,

$$\mathbb{E}[g \mid Q^*] = \sum_p P_+(p) \sum_u Q^*(u \mid p) \nabla_\theta \ell_\theta(p, u) \tag{A.77}$$

$$= \mathbb{E}_{P_+} \mathbb{E}_{Q^*} \nabla_\theta \ell_\theta(p, u) \tag{A.78}$$

$$= \nabla_\theta W_{\delta_{\mathrm{tr}}}(\theta). \tag{A.79}$$

Thus the standard BPR gradient computed on a sampled pair is an unbiased Monte Carlo estimator of the target gradient. No differentiation through the discrete sampled index, REINFORCE correction, or extra multiplication by the sampling probability is required. Equation (A.71) is Eq. (9), so Theorem 3.6 holds at fixed temperature, and the envelope argument just given extends it to the budget form. If the implementation instead uses an online approximation $\widetilde{Q}_t \neq Q^*$, the first two lines still show that g_t is unbiased for the risk gradient under the frozen current distribution $\widetilde{Q}_t$; in that case, however, it should not be claimed to equal $\nabla W_{\delta_{\mathrm{tr}}}$ exactly.

A.8 Measure-Theoretic Extension to General Measurable Spaces

The main text uses discrete notation $P(u \mid p)$ to match the finite-candidate sampling implementation. On a general measurable space, the same construction can be written in terms of probability measures $P(\mathrm{d}u \mid p)$ and Radon–Nikodym derivatives, preserving the Gibbs/KL structure without assuming an ordinary density or a finite state space.

The main text and the preceding proofs use finite discrete support only so that every identity can be checked term by term using finite sums. The method itself does not require the unlabeled space to be discretized into a finite set. We now express the same construction on a general measurable unlabeled space.

Fix p, let $(\mathcal{X}, \mathcal{F})$ be the unlabeled sample space, and let $P_0(\mathrm{d}u \mid p)$ be a base probability measure on it. Assume that $S(p, u) > 0$ is measurable and

$$0 < Z_S(p) := \int S(p, u) P_0(\mathrm{d}u \mid p) < \infty. \tag{A.80}$$

The reference distribution is then defined through the Radon–Nikodym derivative

$$\frac{\mathrm{d}P_S(\cdot \mid p)}{\mathrm{d}P_0(\cdot \mid p)}(u) = \frac{S(p, u)}{Z_S(p)}. \tag{A.81}$$

For a temperature $\tau > 0$ under consideration, further assume that $\ell_\theta(p, \cdot)$ is finite $P_S(\cdot \mid p)$-almost surely and that

$$Z_{\theta,\tau}(p) := \int \exp\left(\frac{\ell_\theta(p, u)}{\tau}\right) P_S(\mathrm{d}u \mid p) < \infty, \tag{A.82}$$

with the additional first-moment condition

$$\int |\ell_\theta(p, u)| \exp\left(\frac{\ell_\theta(p, u)}{\tau}\right) P_S(\mathrm{d}u \mid p) < \infty \tag{A.83}$$

whenever a finite KL value or differentiation of the log-partition function is invoked. Define the Gibbs tilt by

$$\frac{\mathrm{d}Q_{\theta,\tau}(\cdot \mid p)}{\mathrm{d}P_S(\cdot \mid p)}(u) = \frac{\exp(\ell_\theta(p, u)/\tau)}{Z_{\theta,\tau}(p)}. \tag{A.84}$$

For any $Q(\cdot \mid p) \ll P_S(\cdot \mid p)$ such that $D_{\mathrm{KL}}(Q \| P_S) < \infty$ and $\mathbb{E}_Q |\ell_\theta| < \infty$, the strict positivity of the Gibbs density implies $Q \ll Q_{\theta,\tau}$. Equation (A.84) gives

$$\log \frac{\mathrm{d}Q_{\theta,\tau}}{\mathrm{d}P_S}(u) = \frac{\ell_\theta(p, u)}{\tau} - \log Z_{\theta,\tau}(p), \tag{A.85}$$

$$\log \frac{\mathrm{d}Q}{\mathrm{d}Q_{\theta,\tau}}(u) = \log \frac{\mathrm{d}Q}{\mathrm{d}P_S}(u) - \frac{\ell_\theta(p, u)}{\tau} + \log Z_{\theta,\tau}(p). \tag{A.86}$$

Integrating with respect to Q yields

$$D_{\mathrm{KL}}(Q\|Q_{\theta,\tau}) = D_{\mathrm{KL}}(Q\|P_S) - \frac{1}{\tau}\mathbb{E}_Q\ell_\theta + \log Z_{\theta,\tau}(p), \tag{A.87}$$

and hence

$$\mathbb{E}_Q\ell_\theta - \tau D_{\mathrm{KL}}(Q\|P_S) = \tau \log Z_{\theta,\tau}(p) - \tau D_{\mathrm{KL}}(Q\|Q_{\theta,\tau}). \tag{A.88}$$

This is exactly the Gibbs variational identity obtained in the finite discrete case. Therefore the Gibbs optimality and KL-dual structure do not fundamentally rely on finite discrete support; what is required is that the relevant exponential and first moments be well defined. The fixed-temperature gradient identity additionally requires the usual differentiability and domination conditions that justify interchanging ∇_θ and integration. Finite support is used in the main theorems primarily to avoid technical questions about attainment and the essential supremum as $\tau \to 0^+$. On a general space, the corresponding endpoint statements should be formulated using essential suprema together with suitable tightness or attainment conditions.

Numerical computation on large or continuous supports. When Z_S, $Z_{\theta,\tau}$, or the associated expectations cannot be computed analytically, they can be treated as ordinary integrals with respect to P_0 or P_S and approximated using Monte Carlo, importance sampling, MCMC, or a task-specific numerical sampler. This is a numerical approximation of probability measures and expectations, not an additional finite-discretization assumption in the definition of the method. For the simplest Monte Carlo construction, if $U_i \overset{\text{iid}}{\sim} P_0(\cdot \mid p)$, then for an integrable function f,

$$\widehat{\mathbb{E}}_{P_S} f = \frac{\sum_{i=1}^m S(p,U_i) f(U_i)}{\sum_{i=1}^m S(p,U_i)}, \tag{A.89}$$

$$\widehat{\mathbb{E}}_{Q_{\theta,\tau}} f = \frac{\sum_{i=1}^m S(p,U_i) e^{\ell_\theta(p,U_i)/\tau} f(U_i)}{\sum_{i=1}^m S(p,U_i) e^{\ell_\theta(p,U_i)/\tau}}. \tag{A.90}$$

These are self-normalized ratio estimators and are not claimed to be exactly unbiased at finite m. If the corresponding weights and weighted functions have finite second moments and the denominator expectations are strictly positive, the numerator and denominator each fluctuate at the $O_p(m^{-1/2})$ scale by the central limit theorem. Applying the delta method to the ratio then gives the same $O_p(m^{-1/2})$ convergence rate for the self-normalized estimators in Eq. (A.90). The samples $U_1, \ldots, U_m$ are used only to approximate the integrals; the algorithm need not preconstruct or maintain a discrete state space.

Candidate-pool implementation. In large discrete spaces such as recommendation, the construction above is exactly what candidate-pool sampling implements. For each anchor p, a finite pool $\mathcal{C}_p = \{U_1, \ldots, U_m\}$ is drawn from the base proposal $P_0(\cdot \mid p)$, and the PE weights of Eq. (3) are formed and normalized within that pool alone. Conditional on $\mathcal{C}_p$, the resulting sampler is the Gibbs optimizer of Proposition 3.4 on the restricted support, with the reference distribution renormalized to the same support; every identity of Appendices A.3–A.5 therefore holds verbatim on $\mathcal{C}_p$. Viewed against the full-support sampler, drawing a fresh pool at each step is the self-normalized Monte Carlo approximation of Eq. (A.90), whose error decreases at the $O_p(m^{-1/2})$ rate under the moment conditions above. When this implementation is used, the reference-deviation estimate $\widehat{K}_t$ driving the temperature update is formed over the same pool, so the budget δ_{tr} refers to the deviation of the pool-restricted sampler from the pool-restricted reference.

A.9 Local Direction and Global Bounds for Hidden-Positive Contamination

The main text first identifies the source of this risk: utility is supplied by the current loss, while a high-loss pair in PU data may be either a useful true hard negative or a hidden positive. Theorem 3.7 in the main text therefore gives only a global statement that does not depend on the unknown mixture of these two cases: reference deviation limits how far the hidden-positive rate can move away from its value under P_S. This section proves that bound in full and adds the local directional analysis omitted from the main text. The exponential-family derivative expresses the change in contamination rate under stronger utility reweighting as a covariance between Y and the loss, so its sign is determined by the actual data composition.

Fix θ and let $Y(p,u) \in \{0,1\}$ be independent of β. Along the Gibbs path, define

$$\eta_\theta(\beta) = \mathbb{E}_{P_+}\mathbb{E}_{q_{\beta,p}} Y(p,u). \tag{A.91}$$

For a fixed p, Eq. (A.37) implies that any function $f_p(u)$ with no explicit dependence on β satisfies the exponential-family identity

$$\frac{d}{d\beta}\mathbb{E}_{q_{\beta,p}} f_p = \sum_u f_p(u) q_{\beta,p}(u)\,(l_p(u) - \mathbb{E}_q l_p) \tag{A.92}$$

$$= \mathbb{E}_q[f_p l_p] - \mathbb{E}_q f_p \mathbb{E}_q l_p \tag{A.93}$$

$$= \mathrm{Cov}_q(f_p, l_p). \tag{A.94}$$

Taking $f_p = Y(p,\cdot)$ and averaging over P_+ gives

$$\eta'_\theta(\beta) = \mathbb{E}_{P_+}\,\mathrm{Cov}_{q_{\beta,p}}(Y, l_p). \tag{A.95}$$

Combining this with Eq. (A.42), whenever $V_\theta(\beta) > 0$,

$$\frac{d\eta_\theta}{dH_\theta} = \frac{\eta'_\theta(\beta)}{H'_\theta(\beta)} = \frac{\mathbb{E}_{P_+}\mathrm{Cov}_q(Y,l)}{\mathbb{E}_{P_+}\mathrm{Var}_q(l)}. \tag{A.96}$$

This gives the local directional characterization mentioned only verbally in the main text and makes explicit that a reference-deviation constraint alone cannot determine whether contamination increases or decreases.

We now prove the global contamination bound. Let $T(p,u) = Y(p,u) \in \{0,1\}$ and push the joint distributions J_Q and J_S forward through the measurable map T. The resulting Bernoulli success probabilities are

$$\eta(Q) = \mathbb{E}_{J_Q} Y, \qquad \eta_S = \mathbb{E}_{J_S} Y.$$

The data-processing inequality for KL divergence gives

$$\mathrm{kl}\big(\eta(Q)\|\eta_S\big) \le D_{\mathrm{KL}}(J_Q\|J_S), \tag{A.97}$$

where the left-hand side is Bernoulli KL divergence. In addition, the probability difference of the event $\{Y=1\}$ is bounded by total variation:

$$|\eta(Q) - \eta_S| \le \mathrm{TV}(J_Q, J_S). \tag{A.98}$$

Under the natural-log convention, Pinsker's inequality (Pinsker, 1964) gives

$$\mathrm{TV}(J_Q, J_S) \le \sqrt{\frac{1}{2} D_{\mathrm{KL}}(J_Q\|J_S)}. \tag{A.99}$$

Therefore

$$|\eta(Q) - \eta_S| \le \min\left\{1, \sqrt{\frac{1}{2} D_{\mathrm{KL}}(J_Q\|J_S)}\right\}. \tag{A.100}$$

If a sampler additionally satisfies $\mathcal{K}(Q) \le \delta_{\mathrm{tr}}$, Eq. (A.100) immediately gives

$$|\eta(Q) - \eta_S| \le \sqrt{\delta_{\mathrm{tr}}/2}.$$

Equations (A.97) and (A.100) are the two inequalities of Eq. (10), so Theorem 3.7 is proved, and the display above is its bound specialized to the training constraint.

A.10 Reference Quality, True-Negative Matching, and an AUC Certificate

The AUC result in the main text has three steps. We first represent the true-negative pair distribution by J^* and quantify the match between J_S and J^* through Reference Coverage. We then fix a comparison model θ_0 and use the BPR surrogate to obtain a relative AUC statement. Finally, we take the worst case over all true-negative distributions compatible with the coverage condition and reduce the resulting comparison margin to a one-dimensional KL dual.

True-negative distribution and reference quality. If all latent labels were observable, let $P_-(u \mid p)$ denote the conditional distribution of true negatives given the positive example p, and define

$$J^*(p,u) = P_+(p)P_-(u \mid p).$$

The distribution J^* is the ideal target distribution for pairwise ranking, whereas

$$J_S(p,u) = P_+(p)P_S(u \mid p)$$

is the reference pair distribution induced by the prior evidence. We do not require $P_S = P_-$, nor do we require the prior score S to be a calibrated true-negative probability. Instead, reference quality is measured by the discrepancy between J_S and J^*. The Reference Coverage condition is

$$D_{\mathrm{KL}}(J^* \| J_S) \le \delta_{\mathrm{cov}}. \tag{A.101}$$

Because the KL direction is $J^* \| J_S$, regions that occur frequently under the true-negative distribution but receive extremely small mass under J_S incur a large penalty in Eq. (A.101). Thus a smaller δ_{cov} means that J_S better matches the true-negative pair distribution J^*, while $\delta_{\mathrm{cov}} = 0$ implies equality almost everywhere. Since J^* depends on unobserved latent labels, δ_{cov} is a theoretical quantity describing reference quality rather than a quantity controlled by the sampler during training.

Pairwise ranking error and BPR calibration. For a true positive–negative pair, define the ranking error with half penalty for ties by

$$e_\theta(p,u) = \mathbf{1}\{d_\theta(p,u) < 0\} + \frac{1}{2}\mathbf{1}\{d_\theta(p,u) = 0\}, \qquad \mathrm{AUC}^*(\theta) = 1 - \mathbb{E}_{J^*} e_\theta. \tag{A.102}$$

Hence $\mathbb{E}_{J^*} e_\theta$ is exactly the pairwise ranking-error rate under the target distribution, and reducing this error is equivalent to increasing AUC^*. Let $c = \log 2$. The BPR logistic loss satisfies the pointwise calibration inequality

$$c\, e_\theta(p,u) \le \ell_\theta(p,u). \tag{A.103}$$

This follows by considering three cases. If $d_\theta > 0$, the left-hand side is 0. If $d_\theta = 0$, the left-hand side is $c/2$, whereas $\ell_\theta = c$. If $d_\theta < 0$, then $e_\theta = 1$ and

$$\ell_\theta = \log(1 + e^{-d_\theta}) > \log 2 = c.$$

Therefore Eq. (A.103) holds for every pair.

From unknown J^* to a reference-centered worst case. Fix an arbitrary comparison model θ_0. It may be a standard BPR baseline trained with random negative sampling (RNS), or any other fixed baseline. It serves only as the AUC comparison point and does not enter the sampling optimization. Define

$$m_{\theta,\theta_0}(p,u) = c\, e_{\theta_0}(p,u) - \ell_\theta(p,u). \tag{A.104}$$

By Eq. (A.103),

$$c\big(e_{\theta_0}(p,u) - e_\theta(p,u)\big) \ge m_{\theta,\theta_0}(p,u). \tag{A.105}$$

We then define the worst-case comparison margin over all candidate true-negative distributions compatible with Reference Coverage,

$$\Gamma_{\mathrm{cov}}(\theta;\theta_0) = \inf_{Q : D_{\mathrm{KL}}(J_Q \| J_S) \le \delta_{\mathrm{cov}}} \mathbb{E}_{J_Q} m_{\theta,\theta_0}(p,u). \tag{A.106}$$

Since the actual J^* belongs to this feasible set by Eq. (A.101),

$$\mathbb{E}_{J^*} m_{\theta,\theta_0} \ge \Gamma_{\mathrm{cov}}(\theta;\theta_0).$$

Theorem A.5 (AUC-improvement certificate from the comparison margin)**.** If $D_{\mathrm{KL}}(J^* \| J_S) \le \delta_{\mathrm{cov}}$, then

$$\mathrm{AUC}^*(\theta) - \mathrm{AUC}^*(\theta_0) \ge \frac{\Gamma_{\mathrm{cov}}(\theta;\theta_0)}{c}. \tag{A.107}$$

Proof. By Eq. (A.102),

$$\begin{aligned} c\big(\mathrm{AUC}^*(\theta) - \mathrm{AUC}^*(\theta_0)\big) &= c\mathbb{E}_{J^*}\big[e_{\theta_0} - e_\theta\big] && \text{(A.108)}\\ &\ge \mathbb{E}_{J^*} m_{\theta,\theta_0} && \text{(A.109)}\\ &\ge \Gamma_{\mathrm{cov}}(\theta;\theta_0), && \text{(A.110)} \end{aligned}$$

where the first inequality follows from Eq. (A.105) and the second because J^* is feasible for Eq. (A.106). Dividing by $c = \log 2 > 0$ proves the claim. □

One-dimensional dual of the worst-case margin. Equation (A.106) is a KL-robust lower expectation. For any $\lambda > 0$, define

$$G_{\mathrm{cov}}(\theta, \lambda; \theta_0) = -\lambda\delta_{\mathrm{cov}} - \lambda \mathbb{E}_{P_+} \log \mathbb{E}_{P_S(\cdot|p)} \exp\left(-\frac{m_{\theta,\theta_0}(p,u)}{\lambda}\right). \tag{A.111}$$

We now verify the strong dual representation

$$\Gamma_{\mathrm{cov}}(\theta; \theta_0) = \sup_{\lambda>0} G_{\mathrm{cov}}(\theta, \lambda; \theta_0). \tag{A.112}$$

Fix $\lambda > 0$ and define

$$Q^m_\lambda(u \mid p) = \frac{P_S(u \mid p) \exp(-m_{\theta,\theta_0}(p,u)/\lambda)}{\mathbb{E}_{P_S(\cdot|p)} \exp(-m_{\theta,\theta_0}(p,u)/\lambda)}.$$

Applying the same Gibbs variational identity to $-m_{\theta,\theta_0}$ gives, for any Q,

$$\mathbb{E}_Q m_{\theta,\theta_0} + \lambda D_{\mathrm{KL}}(Q\|P_S) = -\lambda \log \mathbb{E}_{P_S} e^{-m_{\theta,\theta_0}/\lambda} + \lambda D_{\mathrm{KL}}(Q\|Q^m_\lambda). \tag{A.113}$$

If $D_{\mathrm{KL}}(J_Q\|J_S) \le \delta_{\mathrm{cov}}$, averaging over $p \sim P_+$ and using nonnegativity of the KL remainder yields

$$\mathbb{E}_{J_Q} m_{\theta,\theta_0} \ge -\lambda\delta_{\mathrm{cov}} - \lambda \mathbb{E}_{P_+} \log \mathbb{E}_{P_S} e^{-m_{\theta,\theta_0}/\lambda} \tag{A.114}$$

$$= G_{\mathrm{cov}}(\theta, \lambda; \theta_0). \tag{A.115}$$

Thus $\Gamma_{\mathrm{cov}} \ge G_{\mathrm{cov}}(\theta, \lambda; \theta_0)$ for every $\lambda > 0$.

For completeness, define the deviation along this comparison-induced path by

$$K_m(\lambda) = D_{\mathrm{KL}}(J_{Q^m_\lambda}\|J_S).$$

Exactly as in Appendix A.6, differentiation gives

$$\frac{\partial}{\partial\lambda} G_{\mathrm{cov}}(\theta, \lambda; \theta_0) = K_m(\lambda) - \delta_{\mathrm{cov}}, \tag{A.116}$$

and

$$\frac{\partial^2}{\partial\lambda^2} G_{\mathrm{cov}}(\theta, \lambda; \theta_0) = -\frac{1}{\lambda^3} \mathbb{E}_{P_+} \mathrm{Var}_{Q^m_\lambda(\cdot|p)}\big(m_{\theta,\theta_0}(p,u)\big) \le 0. \tag{A.117}$$

Whenever the coverage radius lies strictly between the zero-deviation endpoint and the saturation deviation associated with the row-wise minimizers of m_{θ,θ_0}, the unique first-order solution satisfies $K_m(\lambda^*) = \delta_{\mathrm{cov}}$ and attains the lower bound above with zero KL remainder. At the two endpoint regimes, the same equality is obtained through $\lambda \to \infty$ or $\lambda \to 0^+$. This proves Eq. (A.112) for all $\delta_{\mathrm{cov}} \ge 0$ under the standing finite-support assumptions.

Combining Eq. (A.112) with Theorem A.5 gives

$$\mathrm{AUC}^*(\theta) - \mathrm{AUC}^*(\theta_0) \ge \frac{\Gamma_{\mathrm{cov}}(\theta; \theta_0)}{c} = \frac{\sup_{\lambda>0} G_{\mathrm{cov}}(\theta, \lambda; \theta_0)}{c}. \tag{A.118}$$

Corollary A.6 (Threshold for a strict AUC improvement)**.** If there exist $\lambda > 0$ and $\gamma > 0$ such that

$$G_{\mathrm{cov}}(\theta, \lambda; \theta_0) \ge c\gamma,$$

then

$$\mathrm{AUC}^*(\theta) - \mathrm{AUC}^*(\theta_0) \ge \gamma.$$

Proof. Since $\Gamma_{\mathrm{cov}} \ge G_{\mathrm{cov}}(\theta, \lambda; \theta_0)$, Eq. (A.107) gives

$$\mathrm{AUC}^*(\theta) - \mathrm{AUC}^*(\theta_0) \ge \frac{\Gamma_{\mathrm{cov}}}{c} \ge \frac{G_{\mathrm{cov}}(\theta, \lambda; \theta_0)}{c} \ge \gamma.$$

□

The derivation above makes explicit the full chain used in the main text. Reference Coverage links P_S to the unknown true-negative distribution J^*. After fixing a comparison model θ_0, BPR calibration turns the AUC difference into the comparison margin m_{θ,θ_0}. Equation (A.106) then takes the worst case over all true-negative distributions allowed by the coverage condition, and Eq. (A.112) reduces this quantity to the one-dimensional dual G_{cov}. The dual variable λ is used only to evaluate the certificate and is unrelated to the training-time sampling temperature τ. Theorem A.5 is Corollary 3.9, and Eq. (A.107) is its Eq. (12).

A.11 Roles of the Symbols and Scope of the Theory

The framework revolves around two distinct sources of information. The prior supplies credibility through S and thereby induces the reference distribution P_S, whereas the current BPR loss ℓ_θ supplies utility. The actual sampler $Q_{\theta,\tau}$ is obtained by utility-aware reweighting on top of this reference distribution. The remaining symbols describe either the training implementation or the boundary of the theory: δ_{tr} is an optional hyperparameter controlling reference deviation, Y is used only for contamination analysis, and δ_{cov} quantifies how well the reference pair distribution matches the true-negative distribution. The table below summarizes the role of each object.

Structurally, the method first constructs the reference distribution P_S from the prior and then uses the current utility to form $Q_{\theta,\tau}$ on top of that reference. The variational and Pareto results explain why the sampler takes this form and which efficient tradeoffs it spans. The later results introduce no additional sampling principle; they separately show how this sampler is compatible with BPR updates, how its PU contamination risk is controlled by reference deviation, and how, under an additional Reference Coverage condition, it can be connected to true-negative AUC.

Object	Role
P_0	Base unlabeled sampling distribution before incorporating the prior; it determines the base measure of the reference distribution.
$S(p,u)$	Fixed pair-level credibility evidence; it represents external support for treating u as a relative negative for p and need not be a calibrated probability.
P_S	Reference distribution obtained by reweighting P_0 with S; it is fixed with respect to the current θ during training.
ℓ_θ	Current model's pair-level utility signal; it describes current ranking difficulty but does not provide a latent-class judgment.
$Q_{\theta,\tau}$	PE sampler (Q_τ^{PE} in the main text); it exponentially reweights the current utility on top of P_S, and varying τ gives the PE sampling path of utility–reference-deviation efficient samplers.
δ_{tr}	Optional training-time hyperparameter; when reference-deviation control is used, it specifies the target KL displacement of the sampler from P_S.
τ	Utility-reweighting temperature in Eq. (3); under δ_{tr} control, its numerical value adapts with the training state and equals the dual variable of the corresponding KL constraint.
Y	Hidden-positive indicator used only for theoretical analysis or auditing; it is not observed by the base training algorithm.
δ_{cov}	Reference-quality radius measuring the KL discrepancy between the unknown true-negative pair distribution and J_S.

These distinctions also delimit the scope of the theory. A reference-deviation constraint controls only how far the sampler moves from P_S; it cannot replace an assumption on the quality of the reference distribution itself. The Reference Coverage condition required by the AUC certificate concerns the match between P_S and the true-negative distribution, and because the true labels are unobserved, this condition cannot be identified automatically from PU observations alone. On continuous or extremely large supports, if normalizing constants, KL divergences, or expectations are approximated by Monte Carlo or other numerical integration schemes, the resulting approximation error must be assessed separately from stochastic training, temperature tracking, and solver stability. These limitations do not change the base algorithm; they make explicit which information and numerical approximations each theoretical conclusion depends on.

B Experimental Setup and Reproducibility Details

This appendix provides the full construction and implementation details for the experiments in Section 4, organized around the role of each design choice. Appendices B.1–B.4 describe the four settings in turn: the controlled PU generator, the two matched scorers built on it, CIFAR-PU, and recommendation. Each of them supplies a scorer, a fixed prior, and a candidate support, and the last of

these is the only place where the settings genuinely differ, since recommendation samples on a pool while the others expose their full unlabeled set. Appendix B.5 then gives the sampler implementation and training protocol that all of them share, and Appendices B.6–B.8 report the budget sweep, the prior-quality audit, and the frontier scan. All numerical settings needed to reproduce the reported results are retained below.

B.1 Controlled PU Data-Generation Protocol

Design objective and protocol overview. The controlled study is designed to separate several mechanisms that are difficult to disentangle in real PU data. For an unlabeled sample, difficulty can come from its true class structure, from the process that hides positive labels, or from ambiguity in the external prior. When these mechanisms vary together, an observed failure of negative sampling is hard to attribute to any one of them. We therefore construct the controlled data from a small set of explicit generative components whose roles can be adjusted separately. The construction controls the prevalence and composition of the population, the class-related structure of each sample, the propensity of a positive sample to be labeled, and the information available to the external prior. This makes it possible to create hidden positives and hard negatives through specified mechanisms and then examine whether pair sampling can use prior credibility while still exploiting model-relevant hardness.

The protocol proceeds in four stages. We first define a latent population whose samples carry separate class, observability, and prior-evidence factors. We then turn each latent sample into two concrete vectors, a sample representation used by the downstream scorer and a prior representation used to construct the external prior. Next, we apply a selected-at-random (SAR) labeling mechanism to the generated population and obtain the observed-positive set P and unlabeled set U. Finally, we use only PU-observable information to convert the prior representations into a fixed reference distribution over U. Figure 4 summarizes this sequence.

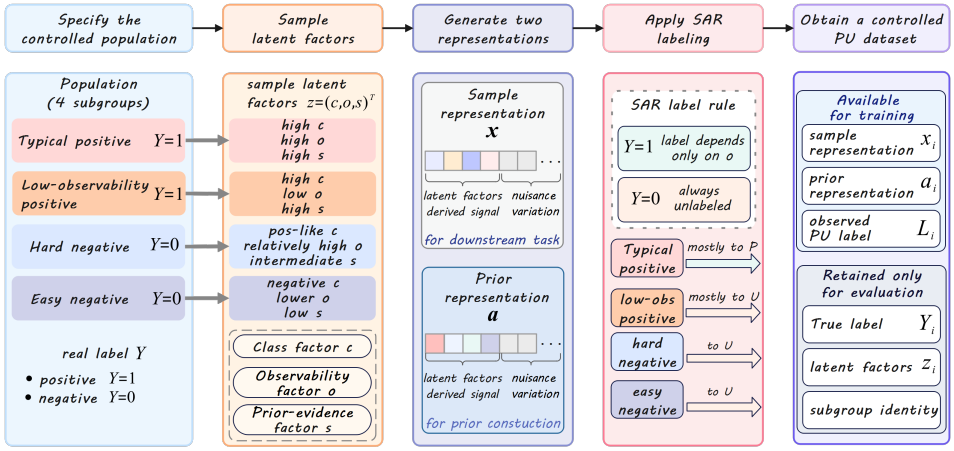


Figure 4: The controlled PU data-generation protocol. A controlled population is first specified through four subgroups and three latent factors: the class factor c, the observability factor o, and the prior-evidence factor s. Each latent sample then generates two representations, the sample representation x for the downstream task and the prior representation a for prior construction. SAR labeling uses the observability factor to split the population into observed positives and an unlabeled set, yielding a controlled PU dataset with both training-visible quantities and retained evaluation-only information.

Step 1. Define and sample the controlled latent population. The starting point is a population with true class label $Y \in \{0, 1\}$. Its overall positive prevalence is controlled by $\pi = \Pr(Y = 1)$. Within the positive and negative classes, we further control how much of the population belongs to the difficult cases that the experiment is intended to study. Let ρ_{low} denote the fraction of positives

assigned to a low-observability subgroup and let ρ_{hard} denote the fraction of negatives assigned to a hard-negative subgroup. The remaining positives are typical positives and the remaining negatives are easy negatives. Thus the four latent subgroups are

$$G \in \{P_{\text{typ}}, P_{\text{low}}, N_{\text{hard}}, N_{\text{easy}}\}.$$

Here P_{low} refers to a positive subgroup with low labeling propensity. A sample becomes a *hidden positive* only after SAR labeling when it has $Y = 1$ and its positive label is not observed. The population prevalence π and the two subgroup fractions ρ_{low} and ρ_{hard} are therefore direct controls over the amount and composition of the generated data.

Each sampled individual is then assigned three latent factors

$$z = (c, o, s)^\top.$$

The factors have distinct generative roles. The *class factor* c controls the class-related signal expressed by the sample. The true label Y is fixed by the population subgroup, while the distribution of c controls how similar the class-related structure of one subgroup is to that of another. For example, placing the c distribution of hard negatives closer to the positive distributions makes those true negatives more positive-like to a downstream scorer. The *observability factor* o controls whether a true positive is likely to appear with an observed positive label. It enters the SAR labeling probability for $Y = 1$ and can also be encoded in the sample representation, allowing the generated features to carry the selection bias created by SAR labeling. For a true negative, o never changes the PU label because negatives always remain unlabeled, although its contribution to the sample representation can still make a negative resemble the observed-positive population. The *prior-evidence factor* s controls the information made available to the prior representation. Its distribution determines how clearly the prior-side vector can separate positive-like samples from credible negatives, while s has no direct role in deciding whether a positive label is observed. Conditional on G, the three factors are sampled independently from subgroup-specific distributions.

The four subgroups are defined by choosing these factor distributions to realize the desired cases. The two positive subgroups use the same distributions for c and s and differ in o. Their underlying class-related structure and prior evidence are therefore matched, while their labeling propensities differ. This creates positive hiddenness through the observability mechanism alone. The two negative subgroups both have $Y = 0$. Hard negatives receive class-related structure closer to the positive population and can also receive more positive-like observability and prior-evidence factors, making them difficult for the scorer and less decisive for the prior. Easy negatives are placed farther from the positive population in these factors. Because the subgroup proportions and the three factor distributions are specified separately, the protocol exposes explicit controls over class difficulty, positive hiddenness, prior ambiguity, and the frequency with which each case appears.

Step 2. Generate a sample representation and a prior representation for each sample. Repeating Step 1 produces latent samples described by (Y_i, G_i, z_i). For each such sample, we next construct two numerical vectors from the same latent state. The *sample representation* x_i is the feature vector that represents sample i to the downstream model. The scorer receives x_i during training and uses it to produce the model-dependent utility in the pair sampler. The *prior representation* a_i is a second vector associated with the same sample. It is computed before downstream training and is used only to construct the fixed external prior.

The two representations are designed for different experimental roles. For the sample representation, we want the prediction problem to remain learnable from genuine class-related information while also exposing the bias introduced by selective positive labeling. We therefore construct x from a mixture that retains class information from c and includes substantial information from o. Once SAR preferentially labels positives with larger o, the observability component of x makes observed positives systematically different from many hidden positives and creates a usable shortcut under PU supervision. Additional signal directions retain class information so that the underlying class remains recoverable despite this shortcut.

For the prior representation, we want a fixed source of evidence that can still recognize positive-like samples even when their labels are hidden because of low observability. We therefore design a to depend strongly on the prior-evidence factor s, retain a weaker contribution from the class factor c, and contain little information from the observability factor o. Under this design, typical and

low-observability positives remain similar in the prior representation because they share the same c and s distributions, even though their o distributions and labeling propensities differ substantially. Hard negatives can occupy an intermediate region when their s values are shifted toward the positive population. This gives the prior informative yet imperfect evidence about negative credibility. The random noises used to generate x and a are independent so that the prior representation is generated separately from the downstream sample vector.

Step 3. Convert the generated population into a PU dataset through SAR labeling. After Steps 1–2, we have a population of samples with paired vectors (x_i, a_i) and generator-side information (Y_i, G_i, z_i). The next stage converts this fully specified population into the PU observation available to the learning problem. For every true positive, let $L_i \in \{0, 1\}$ indicate whether its positive label is observed and draw

$$L_i \mid (Y_i = 1, o_i) \sim \text{Bernoulli}(e(o_i)).$$

Every true negative has $L_i = 0$. The observed-positive and unlabeled sets are then

$$P = \{i : L_i = 1\}, \qquad U = \{i : L_i = 0\}.$$

The SAR probability depends on the observability factor o of a true positive. The labeling rule does not inspect x_i or a_i, and neither c_i nor s_i enters the labeling probability directly. The subgroup-specific distribution of o therefore controls which positives are likely to be hidden, while the subgroup proportions control how many such positives are present in the population before labeling.

At this point the controlled PU dataset has been formed. The experimental pipeline retains (x_i, a_i, L_i) for every sample. The downstream learner receives the sample representations together with the PU indicators, while prior construction receives the prior representations together with the same observed P/U split. The true labels Y_i, subgroup identities G_i, and latent factors z_i are retained only as generator-side metadata for evaluation and offline auditing. They are not used by the scorer, the sampler, or the prior-construction rule. This information boundary allows the generator to provide complete ground truth for controlled analysis while keeping the learning procedure restricted to PU-observable information.

Step 4. Construct the fixed prior from the generated PU data. The final stage starts from the observed sets P and U produced in Step 3 and the prior representations a generated in Step 2. The prior-construction rule compares the prior representations of P and U to assign each $u \in U$ a positive-likeness score. This score is converted into a negative-credibility weight $S(u)$ and normalized over the unlabeled set to obtain the reference distribution P_S. The construction uses the observed PU split and prior representations only. The resulting P_S is computed once before downstream training and remains fixed while the scorer evolves. The complete protocol can therefore be summarized as

$$(\pi, \rho_{\text{low}}, \rho_{\text{hard}}) \longrightarrow G \longrightarrow (Y, c, o, s) \longrightarrow (x, a) \longrightarrow L \longrightarrow (P, U) \longrightarrow P_S.$$

The first terms specify the controlled population, the middle terms generate the sample-level data and PU observation, and the final term provides the fixed prior used by PEPU.

Concrete instantiation used in our experiments. We use a true positive prevalence of $\pi = 0.30$. Among positives, $\rho_{\text{low}} = 0.35$, so 65% are typical positives and 35% are low-observability positives. Among negatives, $\rho_{\text{hard}} = 0.25$, so 25% are hard negatives and 75% are easy negatives. These proportions give the unlabeled pool substantial numbers of both hidden positives and true negatives while retaining distinct hard- and easy-negative populations. Conditional on subgroup membership, each latent factor follows a Gaussian distribution $\mathcal{N}(\mu, \varsigma^2)$ with the parameters in Table 3.

Table 3: Latent-factor distributions for the four controlled-data subgroups. Each entry reports (μ, ς) for $\mathcal{N}(\mu, \varsigma^2)$.

True label	Subgroup (fraction)	c: class structure	o: observability	s: prior evidence
$Y = 1$	Typical positive (0.65)	(0.60, 0.75)	(1.50, 0.45)	(1.20, 0.60)
$Y = 1$	Low-observability positive (0.35)	(0.60, 0.75)	(−1.50, 0.45)	(1.20, 0.60)
$Y = 0$	Hard negative (0.25)	(−0.35, 0.70)	(1.15, 0.50)	(0.45, 0.60)
$Y = 0$	Easy negative (0.75)	(−0.60, 0.80)	(−0.40, 0.75)	(−1.20, 0.70)

The parameter choices implement the roles described above directly. Both positive subgroups use $c \sim \mathcal{N}(0.60, 0.75^2)$ and $s \sim \mathcal{N}(1.20, 0.60^2)$, so their class-related structure and prior evidence are matched. Their observability means are separated at 1.50 and -1.50, which creates a large difference in labeling propensity without changing those two other factors. Hard negatives have a class mean of -0.35, closer to the positive mean than the easy-negative mean of -0.60, and their observability mean of 1.15 makes their sample representations more similar to the high-observability positive population once o is encoded in x. Their prior-evidence mean of 0.45 also lies between the positive mean of 1.20 and the easy-negative mean of -1.20, so the prior treats them as more ambiguous than easy negatives.

Both generated representations are 20-dimensional, with four signal coordinates followed by sixteen independent nuisance coordinates. For the sample representation, we use

$$x_{\text{sig}} = W_x z + D_x \epsilon_x, \qquad W_x = \begin{bmatrix} 1 & 1.4 & 0.55 \\ 0 & 1 & 0 \\ 0.6 & 0 & 0.65 \\ 0.6 & -0.35 & 0 \end{bmatrix}, \qquad D_x = \text{diag}(0.65, 0.65, 0.80, 0.85),$$

with $\epsilon_x \sim \mathcal{N}(0, I_4)$, and then form

$$x = \begin{bmatrix} x_{\text{sig}} \\ \eta_x \end{bmatrix}, \qquad \eta_x \sim \mathcal{N}(0, I_{16}).$$

This loading matrix implements the intended sample-level structure coordinate by coordinate. The first signal coordinate combines c, o, and s and gives a strong weight to observability. The second coordinate exposes o directly. The third combines c with s, and the fourth combines c with a smaller negative contribution from o. The resulting vector therefore carries the SAR-related shortcut through o while preserving several directions that remain informative about the underlying class structure. Under full supervision, a linear model reaches development AUC 0.924, compared with Bayes-optimal AUC 0.943. Uniform sampling that treats every unlabeled example as a negative reaches AUC 0.832, leaving substantial room for pair selection to affect the learned ranking.

For the prior representation, we use

$$a = \begin{bmatrix} W_a z + \sigma \epsilon_a \\ \sigma \eta_a \end{bmatrix}, \qquad W_a = \begin{bmatrix} 0 & 0 & 1 \\ 0.4 & 0 & 0 \\ 0 & 0.05 & 0 \\ -0.1 & 0 & 0.3 \end{bmatrix},$$

where $\epsilon_a \sim \mathcal{N}(0, I_4)$, $\eta_a \sim \mathcal{N}(0, I_{16})$, and the default noise scale is $\sigma = 0.5$. The first signal coordinate reads s directly, the second retains a weaker class contribution from c, the third carries only a very small observability contribution, and the fourth mixes a small amount of c with s. This implements the intended prior-side design in which s is dominant, c supplies additional class-related evidence, and o contributes little. Typical and low-observability positives therefore remain similar in a despite their very different labeling propensities, while hard negatives receive intermediate prior evidence through their shifted s distribution. The noise scale σ controls how clearly this structure can be recovered from the prior representation and is varied in the prior-quality experiment of Section 4.2.

For SAR labeling, every true positive is labeled with probability

$$e(o) = \text{clip}(\text{sigmoid}(\alpha + 3o),\ 0.02,\ 0.55)\,.$$

We choose α by bisection so that the mean labeling rate among positives is 0.15, which gives $\alpha = -6.09$. The resulting labeling rates are 0.225 for typical positives and 0.017 for low-observability positives. The two positive subgroups share the same c and s distributions, so this labeling-rate difference is generated by o. The train/development/test sizes are $50{,}000/10{,}000/20{,}000$. A single Bernoulli realization on the training set produces $2{,}263$ observed positives and $47{,}737$ unlabeled examples, with hidden positives accounting for 26.5% of U.

Finally, we construct the prior from the prior representations after this PU split. Let $\bar{a}_P$ and $\bar{a}_U$ be the mean prior representations of the observed-positive and unlabeled sets and define

$$v = \frac{\bar{a}_P - \bar{a}_U}{\|\bar{a}_P - \bar{a}_U\| + \epsilon}, \qquad A(u) = (a_u - \bar{a}_U)^\top v.$$

The direction v points from the mean unlabeled representation toward the mean observed-positive representation. A larger $A(u)$ therefore indicates that u is more positive-like under the prior representation and is less credible as a negative. We sort U by $A(u)$, partition it into ten equal-frequency bins in increasing order, denote the bin index by $b(u)$, and set

$$S(u) = \exp\left(\log(0.1)\,\frac{b(u)-1}{9}\right), \qquad P_S(u) = \frac{S(u)}{\sum_{w\in U} S(w)}. \tag{B.1}$$

The credibility weight decreases from 1 in the least positive-like bin to 0.1 in the most positive-like bin, while every unlabeled candidate retains positive support. This construction uses only P, U, and the prior representations a and therefore follows the information boundary defined by the protocol.

B.2 Matched Controlled Scorer Settings: PU w/o Encoder and PU w/ Encoder

Purpose of the two controlled settings. Appendix B.1 fixes the controlled PU problem before any downstream model is trained. For every sample, the protocol produces a sample representation x, a prior representation a, an observed PU label that places the sample in P or U, and the fixed reference distribution P_S constructed from the prior representations. The next component to specify is the scorer that turns the sample representation x into the model score used by BPR.

We use two matched scorer settings to separate two ways in which pair sampling can affect learning. When x is fed directly to the scorer, changing the sampler changes which positive–unlabeled pairs update the scoring function while every sample vector remains fixed. When a trainable encoder is placed on top of x, the selected pairs also determine how the sample vectors are transformed before scoring. The two settings therefore use exactly the same generated samples, sample representations, prior representations, PU split, and fixed reference distribution. Their only architectural difference is the presence of the encoder. We refer to them as **PU w/o Encoder** and **PU w/ Encoder**. The term "w/o Encoder" refers specifically to the absence of this additional representation module; the scoring head itself remains trainable in both settings. At a glance, the two settings share the same input representation and the same linear scoring head, and differ only in whether the representation itself is allowed to adapt:

PU w/o Encoder: $\underbrace{x}_{\text{fixed representation}} \xrightarrow{\text{trainable linear head}} h,$

PU w/ Encoder: $\underbrace{x}_{\text{fixed input}} \xrightarrow{\text{trainable residual encoder}} \underbrace{\tilde{x}}_{\text{adapted representation}} \xrightarrow{\text{same linear head}} h.$

PU w/o Encoder. This setting uses the sample representation x from Appendix B.1 directly as the input to the scorer. After the 20-dimensional vector x is standardized, a linear head maps it to a scalar logit,

$$h(x) = w^\top x + b.$$

The head contains 20 weights and one bias, so these 21 parameters are the only trainable quantities in the scorer. The vector assigned to each sample is fixed throughout training. Consequently, all sampling methods operate on the same set of sample vectors and the same fixed prior, and their only difference is which unlabeled candidates they pair with the observed positives for BPR updates. This setting isolates the effect of pair selection under a fixed representation: any change in the learned ranking is produced by changing the sequence of training pairs seen by the same linear scoring model.

PU w/ Encoder. The second setting starts from the same standardized sample representation x and the same linear scoring head, and adds a shared trainable transformation before the head. For every sample, the encoder produces

$$\tilde{x} = x + W_{\text{up}}\ \text{GELU}\big(W_{\text{down}}\,\text{LN}(x)\big), \qquad 20 \to 4 \to 20,$$

and the linear head is then applied to $\tilde{x}$. The same encoder is applied to every sample, and its shared parameters are updated from the sampled pairs across the population. LayerNorm (Ba et al., 2016) has no affine parameters, GELU is used as the nonlinearity (Hendrycks & Gimpel, 2016), and W_{up} is initialized to zero. The zero initialization gives $\tilde{x} = x$ at the start of training, which makes the two

controlled settings begin from exactly the same sample representations and the same initial scoring function.

The encoder is deliberately small because x already contains the controlled structure introduced in Appendix B.1. The residual form keeps the original sample representation as the base and allows training to learn corrections to it. The four-dimensional bottleneck restricts these corrections to a small number of shared directions. This design introduces genuine representation adaptation while keeping the adaptive setting closely matched to PU w/o Encoder. Once training begins, the sampled BPR pairs update both the linear head and the encoder, so pair selection can influence the scoring function directly and can also influence how the sample representation evolves. The prior representation a and the resulting P_S remain fixed throughout this process and receive no gradients.

Common scoring and optimization. Both settings use the same scalar scoring rule after the corresponding representation has been obtained. If h denotes the output of the linear head, the score entering both the BPR loss and the sampler is

$$s_\theta = 5 \tanh(h/5).$$

The bounded score keeps the sampling weights in Eq. (3) finite as the logits grow, preventing it from growing without bound as the logits increase. Both settings are trained solely with the BPR objective $\mathcal{L}_{\mathrm{BPR}}$. We use Adam (Kingma & Ba, 2015) with learning rate 0.01 for the linear head and weight decay 10^{-4}; PU w/ Encoder additionally uses learning rate 0.001 for the encoder. Thus, the two controlled settings differ at exactly one modeling point. PU w/o Encoder learns the scoring head on a fixed x. PU w/ Encoder learns the same type of scoring head together with a shared transformation of x from the sampled BPR pairs.

Training dynamics of the two scorer settings. To make the functional difference between the two scorer settings explicit, we track both the evolution of the representation supplied to the scoring head and the resulting pairwise-ranking dynamics throughout training. The representation diagnostics are used only to monitor whether the added encoder actually adapts the sample representation; they are not additional training objectives.

Let $r_t(x)$ denote the representation supplied to the linear head after epoch t, evaluated on a fixed probe set $\{x_i\}_{i=1}^N$. We measure the epoch-to-epoch representation change using two complementary quantities:

$$D_{\cos}^{(t)} = \frac{1}{N} \sum_{i=1}^{N} \left[1 - \frac{r_t(x_i)^\top r_{t-1}(x_i)}{\|r_t(x_i)\|_2 \, \|r_{t-1}(x_i)\|_2} \right],$$

and

$$D_{\mathrm{Euc}}^{(t)} = \frac{1}{N} \sum_{i=1}^{N} \|r_t(x_i) - r_{t-1}(x_i)\|_2 .$$

The cosine drift measures changes in representation direction, whereas the Euclidean drift measures the absolute displacement between consecutive epochs. For PU w/o Encoder, $r_t(x) = x$ for every t, so both quantities are zero by construction. For PU w/ Encoder, $r_t(x) = \tilde{x}_t$ evolves with the encoder parameters; nonzero drift therefore provides direct evidence that the encoder is actively adapting the representation during training.

To separately characterize the ranking dynamics, we also fix a probe set of positive–unlabeled pairs $\{(p_j, u_j)\}_{j=1}^M$ and define the score margin of each pair at epoch t as

$$m_t(p_j, u_j) = s_{\theta_t}(p_j) - s_{\theta_t}(u_j).$$

We report its mean

$$\bar{m}_t = \frac{1}{M} \sum_{j=1}^{M} m_t(p_j, u_j)$$

together with the positive-margin fraction

$$R_t = \frac{1}{M} \sum_{j=1}^{M} \mathbb{I}[m_t(p_j, u_j) > 0] .$$

The two quantities capture complementary aspects of the pairwise ordering: $\bar{m}_t$ measures the strength of the score separation, whereas R_t records only whether each pair follows the ordering preferred by the BPR objective. For an individual pair,

$$\ell_{\mathrm{BPR}}^{(t)}(p,u) = -\log \sigma(m_t(p,u)) \,,$$

so a larger margin corresponds directly to a smaller BPR loss for that pair.

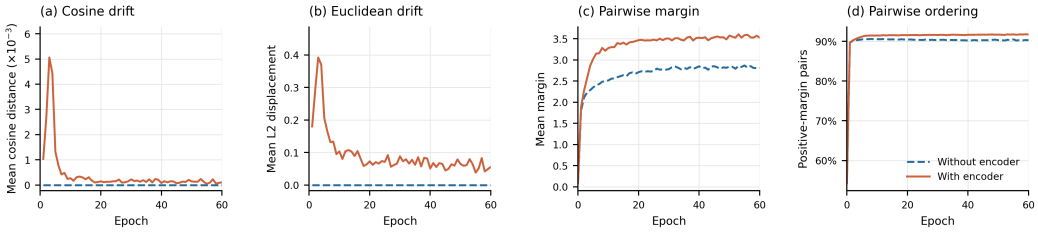


Figure 5: Training diagnostics for the matched PU w/o Encoder and PU w/ Encoder settings. Panels (a,b) measure epoch-to-epoch representation drift on a fixed probe set, while panels (c,d) report the mean P–U score margin and positive-margin fraction on a fixed pair set. Only PU w/ Encoder exhibits representation drift, whereas both settings continue to optimize the pairwise ranking.

Figure 5(a,b) first confirms the intended distinction in representation learning. PU w/o Encoder remains exactly at zero drift because the same standardized sample vector x is supplied to the linear head throughout training. In contrast, PU w/ Encoder exhibits clear nonzero cosine and Euclidean drift, showing that the encoder does not simply add architectural capacity but actually changes the representation seen by the scorer. The largest changes occur during the first few epochs, indicating that most of the representation adaptation takes place early in training, after which the epoch-to-epoch updates become substantially smaller. The cosine drift decreases more strongly than the Euclidean displacement, indicating that directional changes become small relatively early while smaller absolute changes in the representation continue later in training.

The lower panels show that representation adaptation and ranking optimization should be distinguished. Figure 5(c) shows that the mean P–U margin increases in both settings. Thus, the absence of an encoder does not imply the absence of iterative learning: in PU w/o Encoder, the representation is fixed, but the trainable linear head continues to update and progressively enlarges the score separation. PU w/ Encoder performs the same type of scoring-head optimization while simultaneously adapting the representation on which that head operates, and reaches larger mean margins in this diagnostic run.

Figure 5(d) provides a complementary view. The positive-margin fraction rises rapidly during the first few epochs and then changes only modestly in both settings. Combined with the continued increase of the mean margin in Fig. 5(c), this shows that later optimization is driven less by converting large numbers of negative-margin pairs into positive-margin pairs, and more by increasing the separation of pairs that already satisfy the BPR-preferred ordering. This is consistent with the monotonic relation between the margin and the BPR loss above: the loss can continue to decrease even after the positive-margin fraction has largely stabilized.

Taken together, these diagnostics establish the intended functional distinction between the two controlled settings. PU w/o Encoder performs iterative ranking optimization over a fixed representation space, whereas PU w/ Encoder jointly optimizes the ranking function and adapts the representation from the sampled BPR pairs.

B.3 CIFAR-PU: Visual Prior Construction

A common protocol for constructing fixed priors from external representations. The controlled setting of Appendix B.1 generates the prior representation directly, which lets us control how much credibility information it contains. In real data, such a representation is not given by the data-generating process. We therefore need a concrete procedure that starts from an external information

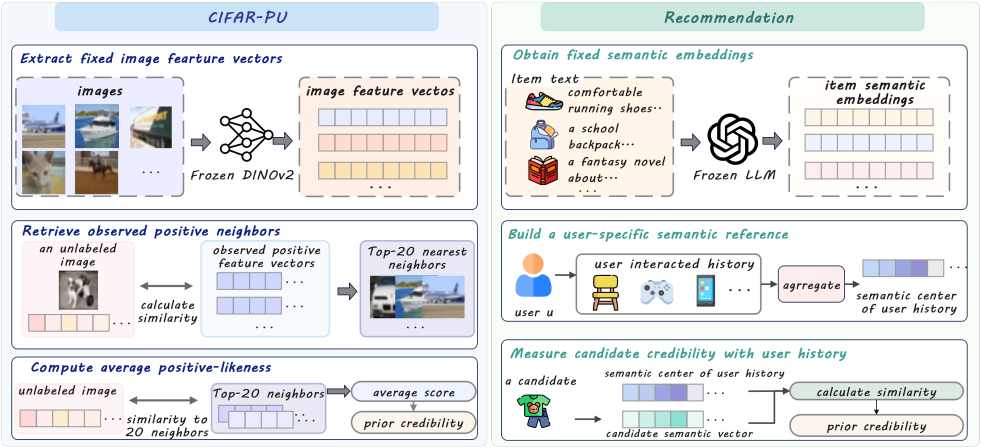


Figure 6: Prior construction in the two real-data settings. CIFAR-PU derives negative credibility from local similarity to observed positives in a frozen visual space, while recommendation derives user-conditioned credibility from similarity to a fixed semantic representation of user history. Both priors are constructed before downstream training and remain fixed with respect to the downstream model.

source and ends with the credibility score S required by Definition 3.1. The CIFAR-PU and recommendation experiments follow the same high-level construction. First, we obtain a fixed representation of each candidate from information that is separate from the downstream model. Second, we use the observed positive information available in the task to define what it means for a candidate to look positive-like under that representation. Third, we convert this positive-likeness into negative credibility, assigning smaller S to candidates that look more compatible with the positive side of the task. Finally, these credibility scores define the prior-side reference rule used to form P_S on the candidate support. All prior representations and credibility rules are fixed before downstream training; recommendation performs only the support-specific normalization when a fresh candidate pool is drawn. The external representation, the comparison rule, and the form of S are task-specific, while this sequence is shared across the two real-data domains. Figure 6 summarizes the two instantiations.

Step 1. Construct a PU task in which hiddenness is driven by a real image attribute. We use CIFAR-10 to move from the fully synthetic mechanism of Appendix B.1 to real images. The four vehicle classes are treated as positives and the six animal classes as negatives. The class-wise split contains 45,000 training images and 5,000 development images, with 4,500/500 images per class, while the official 10,000-image test set is left untouched. The training split therefore contains 18,000 true positives and 27,000 true negatives.

To create PU supervision, we need the probability that a true positive is labeled to vary across images. We use mean image brightness as the observable image attribute that controls this probability. Within each positive class, brightness is converted to a class-specific quantile and passed through a sigmoid propensity with range $[0.02, 0.40]$ and slope 5. The propensity is calibrated so that the mean labeling rate over true positives is 0.10. This gives the selected-at-random mechanism (Bekker et al., 2019) used in Appendix B.1, now implemented through a real image property. Brighter positive images have larger labeling propensity, while darker positive images are more likely to remain hidden in U. One Bernoulli realization produces 1,802 observed positives and 43,198 unlabeled images, of which 16,198, or 37.50%, are hidden positives.

This construction also determines what the external prior must accomplish. Brightness drives whether a positive is observed, so useful prior information should preserve visual evidence that a dark vehicle and a bright vehicle can belong to the same positive class while remaining independent of the downstream scorer. We obtain that evidence from a separately pretrained visual representation.

Step 2. Extract one fixed prior representation for every image. We use a frozen DINOv2 ViT-S/14 (Oquab et al., 2024) pretrained self-supervised on external data. This network is never trained on the CIFAR labels used in our pipeline and is not updated during PEPU training. Each image v is passed through DINOv2 once before downstream training to obtain a 384-dimensional feature vector, which we L_2-normalize as

$$\phi(v) \in \mathbb{R}^{384}, \qquad \|\phi(v)\|_2 = 1.$$

The vector $\phi(v)$ is the prior representation of image v in this setting. Because the vectors have unit norm, the inner product $\phi(v)^\top \phi(v')$ is the cosine similarity between the two prior representations. After this preprocessing step, every image in $P \cup U$ has a fixed prior vector and the frozen DINOv2 network is no longer needed during training.

The role of this representation parallels the prior representation a in the controlled protocol. It supplies information used only to assess candidate credibility. The downstream classifier does not receive $\phi(v)$ as input, and gradients from the BPR objective never reach DINOv2 or the stored prior vectors. This separation keeps prior credibility and current-model utility as two distinct information sources.

Step 3. Measure how strongly each unlabeled image resembles observed positives. Once the PU split and prior representations are available, the observed positives provide the only positive reference set that can be used without hidden labels. We therefore form the fixed observed-positive representation bank

$$\mathcal{B}_P = \{\phi(v) : v \in P\}.$$

For each unlabeled candidate $u \in U$, we retrieve the 20 observed-positive images whose prior representations have the largest cosine similarity to $\phi(u)$. Let this neighbor set be $\mathcal{N}_{20}(u) \subset P$. We then define the positive-likeness score

$$A_{\mathrm{nn}}(u) = \frac{1}{20} \sum_{v \in \mathcal{N}_{20}(u)} \phi(u)^\top \phi(v). \tag{B.2}$$

A large $A_{\mathrm{nn}}(u)$ means that the candidate lies close to several observed positives in the fixed visual representation and therefore has stronger evidence of being positive-like. Such a candidate is less credible as a relative negative.

The use of a local neighbor set is deliberate. The positive class contains four visually different vehicle categories, so a single global centroid can blur the multimodal structure of the observed-positive population. Nearest-neighbor retrieval instead asks whether the candidate lies near a local region occupied by observed positives. A hidden positive only needs to resemble one relevant positive region to receive a high positive-likeness score. This is particularly useful here because the SAR mechanism changes brightness rather than the underlying vehicle identity.

Step 4. Convert positive-likeness into the fixed negative-credibility reference. The score A_{nn} ranks candidates by positive-likeness, while PEPU requires a score whose direction is negative credibility. We therefore sort the unlabeled set by $A_{\mathrm{nn}}(u)$, divide it into ten equal-frequency bins in increasing order, and apply the same rank-based exponential gate as in Appendix B.1 through Eq. (B.1). The least positive-like bin receives credibility weight 1, the most positive-like bin receives weight 0.1, and all intermediate bins are spaced exponentially between them. Every unlabeled candidate retains strictly positive support. Normalizing these weights over U gives the fixed reference distribution P_S.

The information boundary is the same as in the controlled protocol. The entire construction uses only the observed-positive set P, the unlabeled set U, and the frozen prior representations ϕ. True labels inside U are used only for offline evaluation and never enter the representation bank, neighbor retrieval, credibility mapping, or P_S. The bank, all candidate scores, and the resulting reference distribution are computed once before downstream training and remain fixed thereafter.

Two separate models for prior construction and downstream learning. The CIFAR experiment uses two models with deliberately separate roles. The frozen DINOv2 model is used only as the prior feature extractor. It processes each image once before downstream training and produces the fixed prior representation $\phi(v)$ used in Steps 2–4 to construct the credibility score and P_S. The

downstream model is a separate CIFAR-style ResNet-18 (He et al., 2016), initialized from scratch and trained on the raw images with the BPR objective. We use a 3×3 first convolution with stride 1, no max-pooling, and a scalar output. Its current scores determine the BPR loss and therefore the model-dependent utility used by the sampler.

This gives two parallel information paths with different responsibilities. DINOv2 supplies fixed prior evidence about candidate credibility, whereas ResNet-18 supplies the evolving downstream utility used for learning. The two models share no parameters, feature vectors, or gradients, and the DINOv2 representation is never used as an input to ResNet-18. PEPU combines the information from the two models only at the sampling stage through the fixed credibility reference and the current BPR utility. Consequently, prior construction requires one DINOv2 feature-extraction pass per image before training, while all subsequent model optimization is carried out by ResNet-18 alone.

B.4 Recommendation: Semantic Prior Construction and Candidate-Pool Training

Adapting the prior-construction protocol to user-conditioned relevance. Recommendation follows the same prior-construction sequence as CIFAR-PU, but the object being judged is different. For an image, positive-likeness can be defined relative to a single observed-positive image population. For recommendation, whether an item is a plausible positive depends on the user. The same candidate can therefore be a doubtful negative for one user and a credible negative for another. We implement this user dependence by constructing a fixed semantic representation for every item, summarizing each user's observed interaction history in that semantic space, and measuring each candidate relative to that user-specific history. The resulting credibility score is consequently user-conditioned, matching the pair-dependent form $S(p, u)$ in Definition 3.1.

Step 1. Obtain a fixed semantic prior representation for every item. We use the public RLMRec (Ren et al., 2024) release for Amazon-book, Yelp, and Steam. The three datasets contain 9,332, 11,010, and 5,237 items for 11,000, 11,091, and 23,310 users, respectively. The release provides the interaction data together with one 1536-dimensional semantic vector for every item.

These item vectors are produced from information outside the downstream interaction model. A large language model first receives the textual material attached to an item, including its title, description, dataset-specific attributes, and user reviews, and generates a short profile summarizing the item and the preferences it may satisfy. A text embedding model then encodes that profile into a vector. We denote the released semantic representation of item i by

$$\psi_i \in \mathbb{R}^{1536}.$$

The vectors are fixed before LightGCN training and are never updated by the recommender. They therefore serve as the prior-side representation in the recommendation setting, while the downstream model continues to learn only from the interaction graph.

Before comparing items, we remove the catalog-wide component shared by the semantic vectors and place all items on a common cosine-similarity scale. Writing $\bar{\psi}$ for the catalog mean, we define

$$e_i = \frac{\psi_i - \bar{\psi}}{\|\psi_i - \bar{\psi}\|_2}. \tag{B.3}$$

Centering removes the direction common to the catalog, which contributes little to distinguishing candidates, and the unit norm makes $e_i^\top e_j$ the cosine similarity between the centered semantic representations of two items.

Step 2. Build a user-specific semantic reference from observed interactions. For a user u, the observed training interactions provide the positive information available before negative sampling. Let P_u be the set of items that user u has interacted with in the training data. We summarize this history by the mean semantic representation

$$m_u = \frac{1}{|P_u|} \sum_{i \in P_u} e_i. \tag{B.4}$$

The vector m_u is the semantic center of the user's observed history. It is left unnormalized so that both its direction and magnitude retain information about the history. When a user's interacted items

point in similar semantic directions, $\|m_u\|$ is larger; when the history is semantically diverse, the vectors partially cancel and the center has smaller magnitude.

This user-level reference plays the role that the observed-positive neighbor bank plays in CIFAR-PU. The difference comes from the task structure. CIFAR-PU has one common positive population, whereas recommendation has a different positive context for every user. Accordingly, recommendation replaces a global observed-positive bank with one fixed semantic history representation m_u per user.

Step 3. Measure candidate compatibility with the user's observed history. For a candidate item j, we define

$$\mathrm{sim}(u, j) = e_j^\top m_u = \frac{1}{|P_u|} \sum_{i \in P_u} \cos(e_j, e_i). \tag{B.5}$$

The equality shows exactly what the score measures. It is the average semantic similarity between the candidate and the items that the user has already interacted with. A candidate with large $\mathrm{sim}(u, j)$ lies close to the user's observed semantic preferences and is therefore more plausible as an unobserved positive. Such a candidate should receive lower credibility as a negative. Conversely, a candidate far from the observed history receives weaker positive evidence and is more credible as a relative negative.

Using the history mean also prevents a single isolated match from dominating the score for a long and heterogeneous interaction history. Each observed item contributes equally to the average in Eq. (B.5), so the score reflects compatibility with the user's history as a whole. All quantities in this comparison are fixed once the training interactions and released semantic vectors are fixed.

Step 4. Convert semantic compatibility into user-conditioned negative credibility. The similarity in Eq. (B.5) already has the required ordering: larger values indicate stronger positive compatibility and therefore lower negative credibility. We convert it directly through

$$S(u, j) = \max\{1 - \mathrm{sim}(u, j),\, \epsilon\}, \qquad \epsilon = 10^{-8}. \tag{B.6}$$

The floor guarantees strictly positive support for every candidate. For a fixed user, candidates farther from the semantic history receive larger credibility weights, while candidates close to that history receive smaller weights. A user with no training positives would receive $S \equiv 1$, leaving the base distribution unchanged; the data split used here retains at least one training interaction per user, so this fallback is not activated in our experiments.

Unlike the unbounded positive-likeness scores used in the controlled settings, the semantic similarity here is already on a common bounded scale after centering and normalization. We therefore use Eq. (B.6) directly rather than introducing the rank-based binning of Eq. (B.1). The semantic vectors ψ_i, normalized vectors e_i, user centers m_u, and credibility rule are all fixed before downstream training. The same item can still receive different values of S for different users because the reference center m_u is user-specific.

Candidate-pool realization of the reference and PE sampler. The item catalogs are large enough that evaluating every unobserved item at every BPR update is unnecessary. We therefore realize both the fixed prior and the current utility on a fresh candidate pool. For each observed interaction of user u, we draw

$$\mathcal{C}_u = \{j_1, \ldots, j_m\}, \qquad m = 128,$$

uniformly and without replacement from the items that user u has not interacted with in training. The same pool is then used for both sources of information required by PEPU. The fixed semantic construction supplies $S(u, j)$ for every $j \in \mathcal{C}_u$, and the current LightGCN model evaluates the BPR utility of the same candidates. The reference distribution is normalized over $\mathcal{C}_u$, Eq. (3) forms the PE weights on that support, and one candidate is sampled for the ordinary BPR update.

This candidate-pool step changes only the support on which the distributions are realized. Conditional on the sampled pool, the PE sampler is the same reference-regularized sampler analyzed in Section 3.2, with P_S renormalized to the realized candidates. Appendix A.8 gives this restricted-support interpretation formally. Because the pool-restricted reference and attainable deviation geometry differ from the full-support PU settings, we select δ_{tr} separately for recommendation. The values are 1 on

Amazon-book, 2 on Yelp, and 1.3 on Steam, each selected once using validation NDCG@20 and then fixed.

Downstream model and information separation. Every sampling method trains the same LightGCN (He et al., 2020) from the interaction graph, with embedding dimension 64, three propagation layers, and Xavier initialization (Glorot & Bengio, 2010). Randomly initialized ID embeddings are the only model input. The LLM-derived semantic representations and their processed vectors never enter LightGCN and receive no gradients. Thus recommendation uses the same separation as CIFAR-PU: the fixed external representation supplies prior credibility, while the evolving downstream model supplies current BPR utility. PEPU changes how candidates are sampled from the pool without adding semantic features or pretrained parameters to the recommender itself.

Data split and evaluation. For the user-averaged ROC-AUC experiment, we hold out 25% of each user's interactions from the original training file with a fixed seed while retaining at least one interaction for training. For every user with test positives, ROC-AUC is computed exactly by enumerating all remaining items after removing known interactions. For Top-K evaluation, we follow the public RLMRec/LLaRD split and full-item ranking protocol (Ren et al., 2024; Wang et al., 2025a), mask known training interactions at test time, and report Recall@K and NDCG@K (Järvelin & Kekäläinen, 2002) for $K \in \{10, 20\}$.

B.5 Sampler Implementation, Training Protocol, and PU Baselines

PEPU implementation. The sampler is always built on the candidate support that the training anchor actually has: the full unlabeled set in the two controlled settings and in CIFAR-PU, and the freshly drawn pool of Appendix B.4 in recommendation. On either support the steps are the same. The current model evaluates the BPR loss of every candidate, the reference distribution is normalized over those same candidates, and Eq. (3) turns the two into the PE weights; we draw one candidate and apply an otherwise standard, unweighted BPR update to the selected pair. Sampling probabilities, KL statistics, and temperature statistics are stop-gradient quantities, so gradients reach the model only through the selected pair.

The reference deviation $\widehat{K}_t$ is computed on the support used to build the sampler and averaged over the anchor batch, Eq. (A.70); its residual against δ_{tr} drives the persistent temperature state through the dual rule of Appendix A.6, Eq. (A.69). When recommendation uses a candidate pool, both Q_τ^{PE} and its reference deviation are thus defined on the realized pool, as described in Appendix A.8. In the three PU settings the deviation budget is likewise a hyperparameter: $\delta_{\mathrm{tr}} = 0.1$ was selected once from a small candidate grid using development-set ROC-AUC, then frozen and reused across those settings and all prior-quality experiments. The latent development labels used for this controlled hyperparameter selection are not used by the sampler, the model updates, or the construction of the prior. Appendix B.6 further shows that performance is flat over more than two decades of δ_{tr} around this value.

The two ablations keep this pipeline intact and remove one of its two inputs each. Prior-only drops the loss tilt and samples directly from P_S; Hard-only drops the prior, replacing the reference with the uniform base distribution while retaining the loss tilt and the same budget update. Everything else, including the candidate support, the training schedule, and the random streams, is shared with PEPU.

Model selection. Within each PU setting, all compared methods share the same initialization and training protocol. For controlled comparison across samplers, we split the development set with a fixed seed into search and confirm halves, select the epoch by ROC-AUC on the search half, and evaluate the frozen model once on the test set. Latent development labels are used only for this model-selection step and are not used by the PEPU sampler, model updates, or prior construction. Because our PU settings follow instance-dependent SAR labeling, existing PU-only proxy-AUC selection guarantees established under conventional PU observation assumptions (Wang et al., 2026) do not directly transfer to our setting.

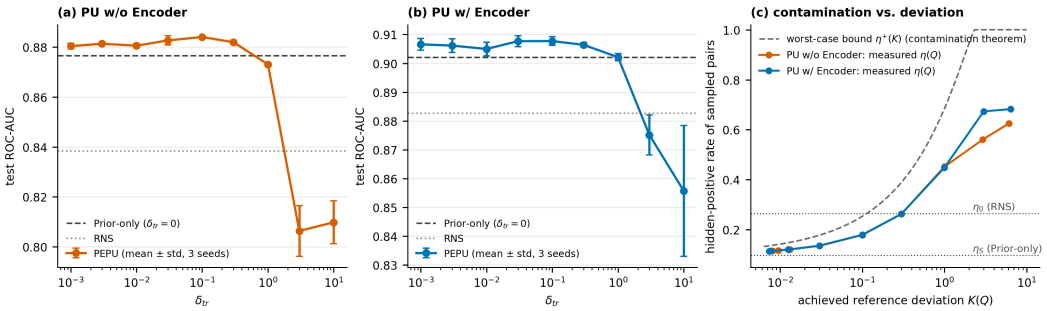


Figure 7: Sensitivity to the deviation budget. (a)–(b) Test AUC versus δ_{tr} (mean $\pm$ std over three seeds) with Prior-only and RNS as references; the vertical line is the budget used in the paper. (c) Hidden-positive rate of the sampled pairs against the achieved deviation $\widehat{K}$, together with the worst-case bound $\eta^+(K)$ of Theorem 3.7; the dotted lines mark the rates of Prior-only (η_S) and RNS (η_0).

B.6 Sensitivity to the Deviation Budget

What the theory says about δ_{tr}. The target deviation δ_{tr} is the sampler-preference hyperparameter of PEPU: it specifies the desired reference-deviation operating level, while the corresponding temperature is adapted online rather than tuned directly (Appendix A.6). Three results locate its admissible range. First, every interior budget $0 < \delta_{\mathrm{tr}} < \delta_{\mathrm{sat}}$ corresponds to a unique point on the Pareto frontier (Appendix A.5); beyond δ_{sat}, which is of order $\log|U|$ (≈ 11.9 in the controlled settings), the sampler degenerates to the argmax endpoint. Second, δ_{tr} is invariant to affine rescaling of the loss (Appendix A.5), so, unlike τ, its meaning is not tied to the numerical scale of the utility; its useful range still depends on the reference distribution and the candidate support. Third, Theorem 3.7 bounds the hidden-positive rate of any sampler with $\mathcal{K}(Q) \le \delta_{\mathrm{tr}}$ by $\eta^+(\delta_{\mathrm{tr}}) = \max\{\eta : \mathrm{kl}(\eta\|\eta_S) \le \delta_{\mathrm{tr}}\}$. The ceiling equals the hidden-positive rate of uniform sampling, $\eta_0 = \mathbb{E}_{P_0}[Y]$, at $\delta_{\mathrm{tr}} = \mathrm{kl}(\eta_0\|\eta_S)$; below this budget utility reweighting is guaranteed, for every model state, not to be more contaminated than RNS, and the guarantee weakens as the prior becomes less informative ($\eta_S \to \eta_0$). Both η_0 and η_S depend on hidden labels, so this quantity is an audit statistic: it cannot be used to set δ_{tr}, and we use it below only to interpret the sweep. The theory does not single out an optimum either: for small budgets both the utility gain and the worst-case contamination shift grow as $\sqrt{\delta_{\mathrm{tr}}}$ (Appendices A.5 and A.9), so the location of the optimum is an empirical question.

Protocol. We sweep $\delta_{\mathrm{tr}} \in \{0.001, 0.003, 0.01, 0.03, 0.1, 0.3, 1, 3, 10\}$ (the paper's value included) in PU w/o Encoder and PU w/ Encoder, where every audit quantity can be computed exactly, with Prior-only ($\delta_{\mathrm{tr}} = 0$) and RNS as references. These settings expose the full unlabeled support, so the sweep isolates the effect of the budget itself, without the candidate-pool approximation that recommendation introduces. Each configuration is trained with three seeds that change only the model initialization and the sampling random streams; the data, the PU split, the prior, and all other settings follow Appendix B.5. Besides test AUC we record, at the selected epoch, the achieved reference deviation $\widehat{K}$, the tracked temperature, and the hidden-positive rate $\eta(Q)$ of the sampled pairs; the last quantity uses hidden labels for auditing only.

Findings. Figure 7 and Table 4 summarize the sweep. Performance is flat over a wide range: in both settings the AUC stays within 0.005 of its maximum for $\delta_{\mathrm{tr}} \in [0.001, 0.3]$ and above Prior-only throughout this range, while the seed standard deviation is at most 0.002. The same budget is near-optimal in both settings: the paper's value is the best grid point in PU w/o Encoder and within 0.0002 of the best in PU w/ Encoder, consistent with the scale invariance of the budget. Beyond the plateau the sampler fails in the way the theory anticipates. At $\delta_{\mathrm{tr}} = 1$ the AUC falls below Prior-only, and for $\delta_{\mathrm{tr}} \ge 3$ below RNS, with the seed variance growing by an order of magnitude as the sampler approaches the deterministic argmax endpoint. Panel (c) shows the mechanism: the sampled hidden-positive rate rises from about 0.10 on the plateau to 0.45 at $\delta_{\mathrm{tr}} = 1$ and above 0.6 beyond, and every measured point lies below the worst-case envelope of Theorem 3.7. The measured

rate reaches the RNS level η_0 only at $\widehat{K} \approx 0.3$, well beyond the audit-only budget $\mathrm{kl}(\eta_0 \| \eta_S) \approx 0.11$ at which the worst-case guarantee of Theorem 3.7 stops excluding this level; the bound is conservative, and conservative on the safe side, and the right edge of the plateau coincides with the point where contamination overtakes the RNS level. Finally, the constant-step dual tracker closely follows the target over the intermediate-budget regime, with $\widehat{K}/\delta_{\mathrm{tr}}$ ranging from 0.98 to 1.01 for $0.03 \le \delta_{\mathrm{tr}} \le 3$. For smaller targets, the achieved deviation remains above the nominal target (e.g., $\widehat{K} \approx 0.008$ at $\delta_{\mathrm{tr}} = 0.001$ after 60 epochs), indicating that these very small target deviations are not fully reached within the fixed training horizon. At $\delta_{\mathrm{tr}} = 10$, the sampler approaches saturation and reaches only $0.61/0.64$ of the target in the two settings. Thus, the present implementation tracks the nominal budget most closely over the intermediate range, while the extreme-budget points are better interpreted through their realized deviations.

Choice of the budget. The sweep supports a simple, label-free guideline for these two settings: any δ_{tr} in $[0.03, 0.3]$ gives the same result to within seed variance, so a fixed value such as 0.1 can be used without tuning, and a coarse development-set search over this range is sufficient when one is desired. The paper's value 0.1 lies inside this range; values above 1 should be avoided because the sampler then approaches the argmax endpoint and contamination dominates. The interval is a property of the full unlabeled support used here. Recommendation samples on a pool, whose reference distribution and saturation deviation differ, and its budgets are therefore selected separately in Appendix B.4.

Table 4: Deviation-budget sweep: test AUC (mean ± std over three seeds), ratio of achieved to target deviation, and hidden-positive rate of the sampled pairs (PU w/o Encoder / PU w/ Encoder). $\delta_{\mathrm{tr}} = 0$ is Prior-only.

δ_{tr}	AUC, PU w/o Encoder	AUC, PU w/ Encoder	$\widehat{K}/\delta_{\mathrm{tr}}$	$\eta(Q)$
0 (Prior-only)	0.8765	0.9021	–	0.098
0.001	0.8804 ± 0.0011	0.9066 ± 0.0020	8.1 / 7.4	0.115
0.003	0.8813 ± 0.0007	0.9061 ± 0.0024	3.2 / 2.6	0.116
0.01	0.8805 ± 0.0009	0.9050 ± 0.0024	1.3 / 1.3	0.121
0.03	0.8826 ± 0.0019	0.9065 ± 0.0019	1.01	0.136
0.1	0.8840 ± 0.0002	0.9070 ± 0.0014	1.00	0.179
0.3	0.8819 ± 0.0008	0.9064 ± 0.0008	1.00	0.263
1	0.8730 ± 0.0007	0.9021 ± 0.0013	1.00	0.45
3	0.8064 ± 0.0101	0.8751 ± 0.0070	0.98 / 1.00	0.56 / 0.67
10	0.8098 ± 0.0086	0.8557 ± 0.0227	0.61 / 0.64	0.63 / 0.68
RNS	0.8384	0.8827	–	0.265

B.7 Prior Scores, OVL Computation, and Controlled Prior Degradation

Prior scores used by OVL. The OVL analysis is applied to the continuous score underlying the credibility reference, oriented so that larger values always indicate greater suitability as a negative. For PU w/o Encoder and PU w/ Encoder, we use $r(u) = -A(u)$. For CIFAR-PU, $r(u) = -A_{\mathrm{nn}}(u)$, the negative of the neighbourhood score in Eq. (B.2). For recommendation, we use $r(u, j) = -\operatorname{sim}(u, j)$, the negative average cosine similarity between candidate j and user u's observed history. To place the three recommendation datasets on a common plotting scale, Figure 1 shows a within-dataset standardized version of this score, with the same affine transformation applied to both audit groups. This standardization is used only for visualization and does not enter the prior construction; as a common affine rescaling, it also leaves OVL unchanged. Each score retains the origin implied by its construction: $A(u)$ is centered at the unlabeled-set mean, so $r > 0$ indicates a candidate that is more negative-like than the average unlabeled sample, whereas the CIFAR-PU and recommendation scores are negated cosine-based similarities with no analogous zero reference. Neither the score origin nor its units affect OVL, and positive affine rescaling likewise preserves the rank-based mapping to S used in the controlled and CIFAR-PU settings.

Numerical computation of OVL. For each pair of score groups, we estimate both densities with Gaussian kernel density estimation using Scott's bandwidth rule (Scott, 1992). We evaluate them on a common 2,000-point grid spanning both observed score ranges and numerically integrate their

Table 5: Eleven prior-quality levels and corresponding results for PU w/o Encoder (fixed δ_{tr}).

σ	OVL	Prior AUC	η_S	PEPU AUC	PEPU AP	Prior-only AUC	Prior-only AP	PEPU HP rate
0	0.222	0.954	0.092	0.8856	0.7924	0.8778	0.7843	0.177
0.25	0.237	0.948	0.093	0.8843	0.7897	0.8783	0.7850	0.181
0.5	0.280	0.931	0.098	0.8840	0.7903	0.8765	0.7837	0.185
0.75	0.343	0.905	0.105	0.8840	0.7908	0.8746	0.7802	0.193
1	0.411	0.873	0.115	0.8820	0.7883	0.8737	0.7806	0.212
1.5	0.534	0.810	0.137	0.8772	0.7832	0.8714	0.7777	0.235
2	0.624	0.757	0.157	0.8724	0.7787	0.8670	0.7740	0.260
3	0.737	0.686	0.185	0.8690	0.7758	0.8614	0.7686	0.291
4	0.800	0.643	0.203	0.8639	0.7706	0.8536	0.7616	0.313
7.64	0.900	0.572	0.233	0.8575	0.7652	0.8516	0.7598	0.344
1000	0.985	0.496	0.266	0.8491	0.7567	0.8384	0.7481	0.383

pointwise minimum. For recommendation, if either group contains more than 300,000 scores, we randomly subsample 300,000 values for the density estimate. Finite-sample estimation imposes an empirical ceiling on OVL: two samples drawn from the same distribution with sizes 13,000 and 35,000 yield OVL approximately 0.985 under this procedure. Accordingly, the pure-noise prior in the main text is reported as OVL 0.98 rather than 1.00.

Recommendation prior-audit protocol. Because implicit feedback does not provide observable true negatives, the recommendation audit is designed to test whether the prior separates deliberately hidden known preferences from ordinary non-interacted candidates. For each user with at least three training positives, we hide

$$\max(1, \lfloor 0.1|P_u| \rfloor)$$

known positives using a fixed seed and denote them by H_u. The user's semantic history m_u is rebuilt from the remaining positives O_u alone, so the hidden items never contribute to the score that ranks them. We then sample without replacement

$$\min(200, \max(50, 10|H_u|))$$

non-interacted items after excluding all known train and test interactions. Both groups are then scored by the same prior. The sampled non-interacted items are not claimed to be true negatives; the audit asks only whether the prior can distinguish held-out known preference from ordinary unobserved candidates. The aggregate distributional shift also appears at the user level: on Amazon-book, the median per-user AUC for recovering the hidden positives is 0.685, and 73.0% of users have AUC above 0.5.

Controlled prior-degradation levels. Table 5 reports all eleven prior-quality levels used in the controlled degradation experiment. All levels share the same base noise realization ϵ' and differ only in the noise scale σ. The default prior uses $\sigma = 0.5$, whose PEPU result matches Table 1. At $\sigma = 0$, OVL is 0.22, corresponding to the most informative prior attainable under this external-view construction; the remaining overlap arises from the latent structure itself rather than injected observation noise. As σ increases, the two score distributions progressively overlap; for $\sigma \geq 32$, OVL reaches its empirical ceiling, and $\sigma = 1000$ serves as the effectively pure-noise endpoint. Table 5 additionally reports η_S, the hidden-positive mass under the reference distribution P_S, and the "PEPU HP rate," the fraction of hidden positives actually sampled by PEPU at the selected epoch; for reference, uniform sampling has hidden-positive rate 0.265.

B.8 Pareto-Geometry Scan and Numerical Checks

This appendix documents the scan behind Figure 3 and turns the visual agreement with Section 3.2 into numbers. The scan freezes a trained model and asks two questions: does the PE sampling path $\{Q_\tau^{\mathrm{PE}}\}$ trace the geometry predicted for it (Table 6), and how the operating temperatures of the frozen frontier compare numerically with those reached by the corresponding training runs (Table 7)? No training is involved; hidden labels enter only the two audit quantities named below.

Scan construction. The model is the PU w/o Encoder PEPU checkpoint of Table 1 ($\delta_{\mathrm{tr}} = 0.1$, epoch chosen by the development criterion, trained as in Appendix B.5). We fix 1,024 observed-positive anchors and the complete unlabeled pool of 47,737 candidates, and evaluate the PE sampling path

$$Q_\beta \propto P_S \exp(\beta \ell_\theta), \qquad \beta = 1/\tau,$$

on 61 values of β spaced logarithmically in $[10^{-3}, 10^3]$ plus the two endpoints: $\beta = 0$, which is the reference P_S, and $\beta \to \infty$, which is the hardening limit $Q^{\max}$ that puts each anchor's mass on its maximum-loss candidate. At every point we record the reference deviation $K(\beta) = \mathbb{E}_p D_{\mathrm{KL}}(Q_\beta \| P_S)$, the expected utility $H(\beta) = \mathbb{E}_p \mathbb{E}_{Q_\beta}[\ell_\theta]$, and, for auditing only, the hidden-positive rate $\eta(\beta) = \mathbb{E}_p \mathbb{E}_{Q_\beta}[Y]$ and the covariance $\mathbb{E}_p \operatorname{Cov}_{Q_\beta}(Y, \ell_\theta)$. Two non-Pareto controls keep the same exponential-tilt form but tilt by a corrupted utility: an anchor-wise random permutation of ℓ_θ, and $\ell_\theta + \mathcal{N}(0, 1)$. Their utility is always measured with the true ℓ_θ.

Numerical checks (Table 6). Each row of Table 6 takes one statement proved in Appendix A and tests it on the scan. The first three rows concern the shape of the frontier: K and H grow with β, the secant slope $\Delta H / \Delta K$ between neighbouring grid points approximates the temperature at the geometric midpoint (Appendix A.5), and the slopes decrease, so $H(K)$ is concave. The next two rows concern contamination: the numerical derivative of η in β equals the recorded covariance (Appendix A.9), and the whole path satisfies the Pinsker bound of Theorem 3.7. The sixth row is the optimality statement of Theorem 3.5: at every deviation level, the two controls attain lower utility than the PE sampler with the same K; the shuffled control never rises above the reference value H_S, and the noisy control stays strictly below the frontier. The last row reports the endpoints, P_S at $K = 0$ and $Q^{\max}$ at $K = \delta_{\mathrm{sat}}$. Repeating the scan at a PU w/o Encoder Prior-only checkpoint and at a PU w/ Encoder PEPU checkpoint gives the same qualitative results.

Operating points (Table 7). The frozen frontier assigns to each budget δ_{tr} a unique point, and the tangent slope there is the temperature τ^* that realizes that budget on this model (Appendix A.5). Table 7 lists, for the budgets used in Appendix B.6, this frozen-geometry temperature next to the temperature τ_t that the online tracker had reached at the selected epoch of the training run with the same budget, together with the utility and hidden-positive rate at the point. For $0.03 \le \delta_{\mathrm{tr}} \le 1$ the two temperatures agree to within about 10% and move together with the budget; at the paper's budget they are 0.65 and 0.67. This is the empirical content of the claim that τ is not a second hyperparameter: given δ_{tr}, the tracker finds the temperature that the geometry prescribes, and the residual difference comes from θ still moving during training while the scan freezes it. At $\delta_{\mathrm{tr}} \le 0.01$ the tracker has not reached its target within the 60-epoch budget (Appendix B.6), which is why τ_t lags τ^* there.

Direction of contamination. On this dataset the covariance $\mathbb{E}_p \operatorname{Cov}_Q(Y, \ell_\theta)$ is positive along the entire interior path (about $0.05/0.08/0.11$ at $K = 0.01/0.1/1$), so η increases monotonically with the budget, as the right panel of Figure 3 shows. This is what Theorem 3.7 predicts: the budget bounds the size of the shift through the Pinsker envelope, while its sign is set by the covariance, which is a property of the data and the current model rather than of the sampler.

Table 6: Numerical checks of the predicted geometry at the fixed PU w/o Encoder PEPU checkpoint. Each row names the statement, where it is proved, how it is tested on the scan, and the outcome.

Statement	Proved in	Test on the scan	Outcome
K and H increase with β	App. A.5	finite differences on the grid	holds
$\mathrm{d}H/\mathrm{d}K = \tau$	App. A.5	secant slope vs. midpoint temperature	median rel. error 0.7%, max 0.8%
$H(K)$ strictly concave	App. A.5	second differences of secant slopes ≤ 0	holds
$\eta'(\beta) = \mathbb{E}_p \operatorname{Cov}_Q(Y, \ell_\theta)$	App. A.9	numerical derivative vs. recorded covariance	median rel. error 0.02%
$\lvert\eta(Q) - \eta_S\rvert \le \sqrt{K(Q)/2}$	Thm. 3.7	checked at every grid point	holds
PE sampling path is the Pareto frontier	Thm. 3.5	corrupted-utility controls at equal K	shuffled: $H \approx H_S$; noisy: below frontier
Endpoints P_S and $Q^{\max}$	App. A.4	$K = 0$ and $K = \delta_{\mathrm{sat}}$	$\delta_{\mathrm{sat}} = 11.9$, $H_{\max} = 1.20$, $\eta_{\max} = 1.0$

C Additional Related Work

Section 2 keeps only the two literature axes that locate the problem studied here. This appendix expands them. Appendix C.1 groups positive–unlabeled learning by what each family of methods

Table 7: Operating points of the frozen frontier for the budgets of Appendix B.6. τ^* is the tangent temperature of the frozen geometry; τ_t is the temperature reached by the online tracker at the selected epoch of the training run with that budget; H and η are the utility and hidden-positive rate at the point. The row $\delta_{\mathrm{tr}} = 0.1$ is the budget used in the paper.

δ_{tr}	τ^* (frozen geometry)	τ_t (tracker)	$H(Q^*)$	$\eta(Q^*)$
0	∞	–	0.113	0.098
0.001	5.39	2.81	0.123	0.105
0.01	1.86	2.14	0.147	0.122
0.03	1.15	1.38	0.175	0.140
0.1	0.65	0.67	0.234	0.182
0.3	0.43	0.40	0.338	0.263
1.0	0.25	0.20	0.557	0.458

corrects, Appendix C.2 groups negative sampling by how each family decides which candidate to draw, and Appendix C.3 places the theoretical analysis in the broader mathematical context of the classical results and frameworks underlying the derivations in Appendix A.

C.1 Positive–Unlabeled Learning

Unbiased and non-negative risk estimation. The first family makes the classification risk computable when the negative class is never labeled. The unbiased PU risk rewrites the negative part of the risk through the positive and unlabeled distributions, which requires a symmetric loss (du Plessis et al., 2014), and a convex formulation later removed that requirement for ordinary losses (du Plessis et al., 2015). Flexible models expose a failure of this estimator: the empirical negative risk can fall below zero and the model then fits the unlabeled data arbitrarily well, which the non-negative risk estimator (nnPU) prevents by clamping that term (Kiryo et al., 2017). The same rewriting carries over to pairwise objectives and yields a PU estimator of the AUC risk (Sakai et al., 2018). Dist-PU replaces risk correction with distribution matching, aligning the expected label distribution on the unlabeled data with the value implied by the class prior and regularizing it by entropy minimization and mixup (Zhao et al., 2022). All of these act on the objective that a pointwise classifier optimizes, while the sampler studied here leaves the downstream objective untouched and acts on the pair that enters it.

Mixture-proportion estimation. The risk estimators above are indexed by one unknown scalar, the fraction of positives inside the unlabeled sample, and a separate line of work recovers it from the data. Kernel mean embedding estimators pose this as mixture proportion estimation and read the fraction off a comparison between the positive and unlabeled distributions after embedding both in a reproducing kernel Hilbert space (Ramaswamy et al., 2016). Univariate transforms that leave the fraction unchanged allow it to be estimated together with the class posterior from noisy positives, which removes the need for density estimation in high dimensions (Jain et al., 2016). Best Bin Estimation instead locates a region where a trained model separates a subset of positives cleanly and reads the fraction off that region, and Conditional Value Ignoring Risk trains the classifier while discarding the highest-loss unlabeled examples; the two alternate in the (TED)^n algorithm (Garg et al., 2021). What this line estimates is a population-level quantity, and a single number suffices, since its role is to calibrate a risk computed over the whole unlabeled sample.

Labeling-mechanism and propensity modeling. A third family models how positives came to be labeled. The selected-completely-at-random assumption makes the labeling probability a constant multiple of the class posterior, so a classifier trained on the labeling indicator recovers the class posterior up to that constant (Elkan & Noto, 2008). SAR-EM replaces this with instance-dependent labeling and estimates the propensity together with the classifier by expectation maximization (Bekker et al., 2019). Under selection bias a scoring function that preserves the order induced by the class posterior is still learnable, which suffices for classification once a threshold is chosen (Kato et al., 2019). A graphical model over the class label and the labeling indicator makes instance-dependent labeling explicit and supports generalization bounds (Gong et al., 2022), and two assumption sets have been identified under which the propensity score is recoverable from biased positive data (Gerych

et al., 2022). Bekker & Davis (2020) survey this landscape and organize methods by the assumptions they place on the labeling mechanism. Our controlled generator and CIFAR-PU both label positives at random conditioned on an observable attribute, so this is the regime in which the sampler is evaluated, and the propensity itself is never estimated.

Label disambiguation and representation learning. A fourth family treats the latent label of each unlabeled example as a quantity to be recovered during training. Self-PU combines self-paced selection of confident positives and negatives with an instance-aware calibrated loss and a self-distillation regularizer (Chen et al., 2020). P3Mix starts from the observation that the PU decision boundary drifts toward the positive side, and mixes ambiguous unlabeled instances with positives that lie near that boundary so the augmented examples carry more accurate supervision (Li et al., 2022a). Robust-PU defines a hardness measure that separates unlabeled examples likely to be negative from those carrying label noise, and admits them under a curriculum (Zhu et al., 2023). Reading the trajectory of an example's prediction scores as a temporal point process turns disambiguation into trend detection, with a trend score whose asymptotic unbiasedness is established (Wang et al., 2023a). Two recent methods make representation quality the bottleneck: LaGAM learns grouping semantics inside PU data by hierarchical contrastive learning and then distills the labels of unlabeled examples through meta-learning (Long et al., 2024), and NcPU pairs a supervised non-contrastive loss whose gradients are dominated by clean pairs with a phantom-label scheme that revises conservative negative supervision through regret-based updates (Zhao et al., 2026). Each of these commits to a label for unlabeled examples; PEPU commits to none, and uses its evidence only to order candidates inside a sampling distribution.

Benchmarks and evaluation protocols. Because PU methods differ in the assumptions they place on the data as much as in the objectives they optimize, a fifth line fixes the protocol under which they are compared. PU-Bench supplies a configurable generator covering sampling schemes, label frequencies, and labeling mechanisms, and re-runs 18 methods over 8 datasets in one pipeline (Chen et al., 2026b). A complementary study targets model selection, which in published PU work often relies on validation data containing labeled negatives: it examines selection criteria computable from positive and unlabeled validation data alone, separates the one-sample from the two-sample setting, and calibrates for the label shift the one-sample construction introduces (Wang et al., 2026). PU learning has thus grown from risk correction under missing negative labels into a body of work that also treats the class prior, the labeling mechanism, the latent labels of the unlabeled set, and the protocols under which any of this is measured.

C.2 Pairwise Learning and Negative Sampling

Pairwise objectives and static samplers. Implicit feedback records interactions and leaves everything else unobserved (Hu et al., 2008), and BPR turns this into a learnable objective by requiring an observed item to be ranked above an unobserved one (Rendle et al., 2009). Every update therefore consumes a negative, drawn uniformly in the default formulation. Static samplers fix a different distribution before training begins. WBPR weights the BPR criterion by item popularity and draws training negatives from the same popularity distribution that generates the candidates at evaluation time (Gantner et al., 2012). MCNS analyzes negative sampling from the perspectives of the optimization objective and the variance of the estimator, and derives that the sampling distribution should be positively but sub-linearly correlated with the positive distribution (Yang et al., 2020). This family shows that the sampling distribution is part of the objective, which is also the starting point of the formulation in Section 3.2.

Model-dependent hard-negative samplers. Uniform negatives are easy to rank correctly and their gradients vanish as training proceeds, so the next family lets the current model choose. DNS draws a small candidate set and keeps the highest-scored item, which makes the sampler adaptive at negligible cost (Zhang et al., 2013). AOBPR samples by the candidate's rank under the current model with an exponentially decaying rank distribution, and refreshes the ranks by a procedure that avoids scoring the full catalog (Rendle & Freudenthaler, 2014). Contrastive learning reached the same design independently, with a sampling distribution tilted by similarity to the anchor and a user-controlled hardness level (Robinson et al., 2021). Two methods create hard negatives instead of selecting them: MoCHi mixes the hardest negatives at the feature level so that difficulty can be raised without

enlarging the batch or the memory bank (Kalantidis et al., 2020), and MixGCF synthesizes a negative embedding from a positive item and candidates drawn from its neighborhood in the user–item graph (Huang et al., 2021). IRGAN makes the difficulty itself the object of optimization, letting a generator learn which items to present to the discriminative ranker in a minimax game (Wang et al., 2017). Shi et al. (2023) give the family a target by showing that hardness-based sampling optimizes a one-way partial AUC, which explains its effect on the top of the ranking. Hardness in this family is the sole criterion, whereas the sampler studied here treats it as one of two.

False-negative-aware samplers. The score that marks a candidate as difficult also marks it as a plausible unobserved positive, and this family controls that risk. SRNS keeps a memory of candidates per user and combines current positive-label probabilities with the standard deviation of historical predictions, favoring high-variability candidates to reduce the risk of sampling persistently high-scoring false negatives (Ding et al., 2020). DENS disentangles the relevant and irrelevant factors of an item and selects negatives that differ from the positive on the relevant factors while matching it on the rest (Lai et al., 2023). AHNS reports that a sampler fixed at one hardness level suffers from both false positives and false negatives, and adapts the target hardness to the predicted score of the positive being paired (Lai et al., 2024). RecNS organizes the candidate space by distance from the positive and samples mostly from an intermediate region, since adjacent items are likely to be unobserved positives and distant ones carry little signal (Yang et al., 2023). Debiased contrastive learning corrects the objective for the probability that a negative shares the anchor's class, which removes the bias without identifying the offending samples (Chuang et al., 2020). Generative models offer a further route: PDRec excludes candidates that a diffusion model estimates the user would prefer before sampling negatives (Ma et al., 2024), and ADAR generates negatives along a controllable transition away from a positive, with the transition point chosen from the model's score (Li et al., 2026). This family and the present work share the concern; the evidence it uses to express that concern comes from the model being trained, from its history, or from a generator trained alongside it.

Knowledge-enhanced samplers. The last sampler family draws on evidence the platform records outside the interaction matrix, and spends it on making the recommender's training signal more faithful to what users actually saw and chose. MF-BPR reads the several feedback types a platform logs as channels ordered by the commitment each expresses, and draws negatives from the lower channels so that the ranking objective absorbs the graded feedback already present in the data (Loni et al., 2016). RNS takes exposure data, which records what was shown to a user without being interacted with, and trains a sampler by reinforcement learning to produce exposure-like negatives, correcting the recommender on the items its own serving system actually put in front of the user (Ding et al., 2019). KGPolicy attaches a knowledge graph to the item set and learns a policy that walks its multi-hop neighborhoods, returning negatives that are difficult and connected to the positive through explicit relations among entities (Wang et al., 2020). Ma et al. (2026) survey these designs and group them as knowledge-enhanced negative sampling, alongside the static, dynamic, adversarial, and reweighting-based families above. In each case the auxiliary source is tied into the recommender: it is consumed by a sampler trained alongside the model, and the design question it answers is which items that particular system should be corrected on.

Language-model knowledge in recommendation. Pretrained language models widened the evidence available to a recommender, and recent work spends it on the recommendation pipeline itself. RLMRec generates user and item profiles with a language model and aligns the resulting representations with the collaborative filtering representations the recommender learns, so the semantic signal reaches the model through its own embedding space (Ren et al., 2024). LLMRec edits the interaction graph before training, adding candidate edges, item attributes, and user profiles, with a denoising step that guards against low-quality additions (Wei et al., 2024). LLaRD elicits preference and relation knowledge from a language model and filters it through an information-bottleneck objective, keeping the part that serves the recommendation target (Wang et al., 2025a). LLMHNI puts the language model in the role of a judge: it combines language-model-encoded semantic relevance with generative relevance judgments about user–item pairs, projects both signals into a space aligned with user–item relevance, and applies graph contrastive learning to suppress the edges those judgments get wrong (Song et al., 2026). These methods carry a substantial language-model apparatus into training, and what they deliver is recommendation accuracy on the datasets they target. The construction in Section 3 asks a different question of such evidence, namely how a fixed external signal should be

combined with current model utility when a pair is formed, and answers it by giving the evidence a single role, as the reference measure of the sampling problem whose solution and efficient frontier are characterized there.

C.3 Mathematical Foundations of the Analysis

The theoretical development in Appendix A draws on several classical frameworks from variational optimization, information theory, convex analysis, and statistical learning. We summarize these foundations here to place the analysis in its broader mathematical context. They provide the general tools on which our derivations build, while the specific formulation and consequences for prior-referenced pair sampling are developed in Appendix A.

KL-regularized variational optimization. Optimization over probability distributions with a linear expected objective and a KL penalty,

$$\sup_{Q} \left\{ \mathbb{E}_Q[f] - \tau D_{\mathrm{KL}}(Q\|P) \right\},$$

has the classical exponentially tilted solution $Q^\star \propto P \exp(f/\tau)$, with a log-partition optimal value. This variational structure is closely related to the maximum-entropy principle (Jaynes, 1957), the geometry of information projection (Csiszár, 1975), and the Donsker–Varadhan variational representation of log-moment-generating functions (Donsker & Varadhan, 1983). Standard treatments can also be found in Cover & Thomas (2006) and Boucheron et al. (2013). These results establish the general correspondence between a reference measure, KL regularization, and exponential reweighting that underlies the distributional view adopted in our analysis.

Convex duality and value-function sensitivity. KL-regularized problems also admit constrained formulations in which the admissible distributions are restricted to a divergence neighborhood of the reference. Under standard regularity conditions, Lagrangian duality and Slater's condition connect the penalized and constrained views and give the associated multiplier its usual trade-off interpretation (Boyd & Vandenberghe, 2004). When the optimized objective depends on model parameters, Danskin-type and envelope results characterize the derivative of the resulting value function without differentiating the optimizer itself (Danskin, 1966; Milgrom & Segal, 2002). Together, these tools provide the standard convex-analytic basis for reasoning about divergence budgets, their dual variables, and parameter-dependent optimal sampling distributions.

Information-theoretic control and distributional robustness. KL divergence provides not only a regularizer but also a quantitative notion of distributional displacement. The data-processing inequality ensures that measurable coarsenings cannot increase KL divergence, while Pinsker's inequality converts KL control into control of probability differences (Cover & Thomas, 2006; Pinsker, 1964; Boucheron et al., 2013). A complementary perspective comes from distributionally robust optimization, where divergence balls around a reference distribution represent ambiguity about the target distribution. Robust optimization over ϕ-divergence sets admits tractable convex dual representations (Ben-Tal et al., 2013), with corresponding stochastic optimization and statistical theory developed for f-divergence neighborhoods (Namkoong & Duchi, 2016; Duchi & Namkoong, 2021). Rahimian & Mehrotra (2022) provide a broader review of these formulations and their duality theory. These two viewpoints make KL divergence useful both for controlling changes induced by reweighting and for expressing uncertainty around a reference distribution.

Ranking risk and surrogate losses. The downstream quantities considered in our analysis belong to the classical theory of bipartite ranking. ROC-AUC admits the probabilistic interpretation that a randomly drawn positive is scored above a randomly drawn negative (Hanley & McNeil, 1982), and empirical pairwise ranking risk has the structure of a U-statistic (Clémençon et al., 2008). Because the discrete ranking error is inconvenient for optimization, convex surrogate losses provide a standard route to tractable learning objectives and corresponding risk guarantees (Bartlett et al., 2006). For pairwise ranking, such consistency and calibration questions have been studied directly by Gao & Zhou (2015). This literature supplies the connection between pairwise surrogate losses and the ranking quantities used to interpret downstream performance.

Taken together, these classical results form the mathematical toolkit behind our analysis. Appendix A specializes and connects them in the setting of prior-referenced negative sampling, yielding the

sampling geometry, training-time characterization, contamination control, and ranking certificate developed for PEPU.

D Applications and Future Work

D.1 Application Scenarios

PEPU is a general prior-enlightened negative sampling framework tailored for pairwise positive-unlabeled learning. It integrates domain prior credibility and dynamic model utility to construct a principled sampling criterion, effectively mitigating noisy supervision caused by hidden positive samples in conventional PU sampling strategies (Lai et al., 2024; Ma et al., 2026). As a plug-in module compatible with standard pairwise ranking losses, PEPU requires minimal modifications to existing training pipelines. It is universally applicable to weak-supervised tasks with limited positive annotations, massive unlabeled data, and accessible domain prior knowledge (Bekker & Davis, 2020; Chen et al., 2026b). We elaborate on typical application scenarios as follows.

Computer Vision Computer vision hosts a wide range of weakly supervised PU tasks, including PU classification, few-shot retrieval and visual pairwise matching (Long et al., 2024; Zhao et al., 2026). Manual annotation incurs substantial labor costs, leaving only a small set of verified positive samples within a large unlabeled pool mixed with latent positives. These unmarked hidden positives easily mislead model optimization and degrade generalization performance. Fortunately, visual domains provide readily available external priors, such as category semantic distances and zero-shot similarities derived from pre-trained vision models (Radford et al., 2021; Oquab et al., 2024). PEPU can quantify sample credibility using these prior cues, select high-quality negatives for pairwise optimization, suppress sampling noise and strengthen the robustness of weak-supervised visual learning.

Natural Language Processing Natural language processing tasks such as text matching, document retrieval and entity alignment naturally fit the PU learning paradigm (Wang et al., 2024; 2025b). In practical textual scenarios, only a small fraction of positive matching pairs can be manually confirmed, while most candidate text pairs remain unlabeled and contain both true negatives and hidden positives. Conventional negative sampling strategies tend to produce low-quality or misleading training pairs, restricting model optimization (Qu et al., 2021; Choi et al., 2026). Drawing on linguistic priors including domain knowledge bases and pre-trained semantic representations, PEPU can adaptively balance prior credibility and model utility to select informative negatives, steadily improving the overall performance of textual ranking and matching tasks (Karpukhin et al., 2020).

Recommender Systems Implicit-feedback recommendation stands as one of the most practically valuable industrial PU applications (Hu et al., 2008; Rendle et al., 2009). Recommendation systems only observe explicit user positive interactions such as clicks and favorites, while unobserved user-item pairs form a large unlabeled set that mixes true negatives with hidden positive preferences. Classic pairwise ranking methods like BPR inevitably suffer from false-negative sampling, especially under data sparsity and cold-start conditions (Lai et al., 2024; Chen et al., 2026a). Leveraging rich side information, including user profiles, item taxonomies and knowledge graph structures, PEPU can build credible sampling references and dynamically optimize the sampling distribution, effectively reducing supervision noise and boosting ranking accuracy for real-world recommendation systems (Wang et al., 2020; Ren et al., 2024).

Bioinformatics Bioinformatics applications including molecular activity prediction and protein-protein interaction screening represent typical PU scenarios burdened by high experimental annotation costs (Li et al., 2022b; Cerulo et al., 2010). Wet-lab experiments can only validate a limited number of positive biological samples, while most candidate molecules and protein instances remain unlabeled and underexplored. Domain-specific priors, such as molecular structural similarity and empirical biological rules, provide reliable credibility signals for sample screening (Rogers & Hahn, 2010). PEPU can leverage these biological priors to identify high-confidence negative candidates, reduces experimental trial-and-error overhead, and enhances prediction accuracy for various biological screening tasks.

Industrial Anomaly Detection Industrial anomaly detection tasks, covering equipment fault diagnosis, network intrusion recognition and medical image screening, exhibit inherent PU learning properties (Blanchard et al., 2010; Perini et al., 2023). Anomalous samples are inherently rare in industrial scenarios, and most monitored data remains unlabeled, containing numerous undiscovered hidden anomalies that introduce persistent training bias (Ruff et al., 2020). Stable domain knowledge, including equipment physical rules and standardized medical diagnostic guidelines, can serve as effective external priors. PEPU can evaluate sample normality based on prior information and combines dynamic model utility for adaptive negative sampling, effectively improving detection precision for industrial and medical anomaly identification.

D.2 Future Directions

We outline several promising directions to further extend and enrich the proposed PEPU framework.

Dynamic Adaptive Prior Refinement PEPU adopts fixed offline priors as credibility benchmarks throughout training. Static priors cannot keep pace with evolving model representations and shifting data distributions. Future work will explore dynamically updatable prior criteria, adjusting credibility references according to real-time model states. This design can reduce static prior bias and maintains high-quality sampling across different training stages.

Multi-source Prior Fusion and Noise-adaptive Balancing Our current implementation relies on single-source, relatively clean priors. Real-world settings often contain noisy, heterogeneous and conflicting prior signals. We plan to develop a prior quality assessment module together with multi-source fusion. By adaptively weighting different priors and balancing prior credibility and model utility, the framework can better tolerate noisy inputs and gain stronger generalization in complex environments.

Generalization to Broader Weak Supervision and Contrastive Learning We validate the prior-enlightened sampling strategy only on pairwise PU ranking, though its core idea is transferable. On one hand, the strategy can be extended to point-wise PU classification to verify its generality within PU learning. On the other hand, the credible negative screening idea can be transferred to contrastive representation learning to mitigate representation degeneration caused by false negatives. We will also study the trade-off between sample credibility and negative-set diversity.

Extension to Streaming and Open-set Scenarios All experiments are conducted on static offline datasets, while real applications often involve continuously arriving unlabeled data and distribution shifts. Future work will adapt PEPU for online weak-supervised learning and open-set recognition, enabling adaptive negative sampling for incremental data. Such adaptations improve the practical deployment potential of our framework in real-time systems.