Towards Bayes-Consistent Dense Retrieval: LLM-Guided Flip-and-Train

Anonymous ACL submission

Abstract

Dense retrievers are commonly trained with dynamically mined hard negatives, but the hardest mined passages often contain hidden positives. Treating these latent positives as negatives corrupts contrastive supervision and caps ranking performance. We introduce **LLM-guided Flip-and-Train (FAT)**, a plug-in loop for dense retrieval that uses an LLM judge to recover high-confidence hidden positives from mined candidate pools and retrains on corrected lists. Unlike denoising methods that filter suspected false negatives, FAT converts them into additional supervision. Theoretically, we analyze FAT under judge over-flip and under-flip errors and under non-stationary mined-pool drift, proving geometric contraction of hidden-positive prevalence and pairwise misranking risk toward an explicit floor. We instantiate FAT within a standard dual-encoder retriever and cross-encoder ranker co-training framework, and empirically demonstrate on Natural Questions, TriviaQA, and MS MARCO Passage that it achieves absolute R@K gains of up to **2%** over strong dense retrieval baselines under standard evaluation, widening to **3%** under diagnostic correction of test-set false negatives.

1 Introduction

Dense retrieval (DR) is the standard first stage in many open-domain QA and search pipelines, a dual encoder maps queries and documents into a shared vector space, and approximate nearest-neighbor (ANN) search retrieves a small top-K candidate set from a large corpus (Karpukhin et al., 2020; Xiong et al., 2020). Much of the progress in DR comes from **dynamic hard-negative mining**. Rather than training only on random or BM25 negatives, modern retrievers periodically re-embed the corpus, refresh the ANN index, and mine new model-hard candidates under the current retriever. ANCE made this refresh-and-mine loop central to dense retrieval training by using approximate nearest neighbor search to obtain negatives that track the evolving model (Xiong et al., 2020); subsequent systems further strengthen this signal through retriever-ranker co-training, and filtered-hard-negative sampling (Qu et al., 2021; Ren et al., 2021; Zhang et al., 2022a; Zhou et al., 2022). Hard negative mining also creates a failure mode that is easy to overlook. The closer a mined candidate is to the query, the more likely it is to be a relevant passage that was simply never annotated. These false negatives are then used as counter-examples in the contrastive loss. Prior dense retrieval methods mainly respond by suppressing suspicious examples through filtering, down-weighting, or sampling around ambiguous candidates (Qu et al., 2021; Zhou et al., 2022; Wang et al., 2024; Yang et al., 2024). This denoising view is conservative, it prevents some damage, but it discards supervision from the most informative unlabeled positives.

Recent work shows that this limitation is not merely hypothetical, LLMs can identify and relabel false negatives in retrieval training data, and such relabeling improves both retrievers and rerankers (Thakur et al., 2025). However, this line of work is primarily a *static, post-hoc data-cleaning* procedure, where hard negatives are mined from an already-trained model, relabeled once, and the retriever is then trained/warmed on the resulting fixed set. This misses a central property of dual-encoder training with refreshed ANN indices. A converged retriever has, by construction, already suppressed many unlabeled positives, having been trained against them as hard negatives, they no longer rank near the top, and a one-shot audit of the final pool simply cannot see them. A hidden pos-

itive is only auditable in rounds where it actually surfaces among the mined candidates, and for many positives that window is early, before the model learns to bury them. Re-auditing at every index refresh catches these positives while they are still visible; relabeling once, after convergence, audits the pool at precisely the point where suppression has already done its damage. Compounding this, LLM relabeling is not noise-free, under-flips leave true positives in the negative set, while over-flips contaminate the positive set with true negatives. A dynamic relabeling loop must therefore both follow the drifting mined pool and weight flipped examples by estimated judge precision. FAT is a dynamic continuation of this LLM-relabeling view. It uses an offline LLM judge to inspect a small, high-risk subset of mined hard negatives, flip high-confidence latent positives to positives, and retrain on corrected candidate lists. The LLM is not deployed as a retriever. It performs local semantic audits over a tiny mined subset during training, while the deployed model remains a millisecond-latency dual encoder with no LLM calls at inference. This distinction is crucial, the retriever supplies global recall over millions of passages; the LLM supplies high-precision local verification.

We instantiate FAT in the AR2 retriever-ranker framework (Zhang et al., 2022a), where a cross-encoder ranker teaches a dual-encoder retriever via distillation while the retriever continues to mine hard examples. This setting raises a further issue. FAT corrects the labels seen by the cross-encoder teacher, but the student retriever is not trained by distillation alone, since AR2-style co-training also keeps an adversarial term that pushes the retriever to surface difficult candidates for the ranker. This adversarial pressure is useful early, when the teacher still needs informative hard examples. Once the teacher has converged, however, a fixed adversarial weight keeps pulling the dual encoder away from the corrected teacher distribution, even though the desired endpoint is now faithful distillation of that corrected signal. We introduce Adversarial Loss Decay (ALD) to resolves this, it leaves the adversarial signal active during the early exploration phase, then decays it once the ranker stabilizes, so the final retriever converges toward the corrected distillation target. ALD introduces no new labels, it is what allows benefits of FAT's corrections to reach the deployed dual encoder. Our contributions are:

- **Dynamic hidden-positive recovery.** We introduce FAT, a training loop that repeatedly audits the high-risk region induced by retriever updates and ANN-index refreshes, recovering latent positives from dynamically mined hard-negative pools rather than merely filtering them. Unlike static LLM relabeling, FAT follows the moving mined pool and uses precision-weighted flips to limit judge contamination.
- **Drift-aware theory.** We prove contraction of hidden-positive prevalence and pairwise misranking risk under judge over-flip, under-flip, and mined-pool drift, yielding Bayes consistency only under explicit idealized limits.
- **Corrected dual-encoder distillation.** We instantiate FAT in AR2-style retriever-ranker co-training, where corrected candidate lists train the cross-encoder teacher, which in turn supervises the retriever. ALD then decays late-stage adversarial pressure, letting the dual encoder inherit the corrected teacher distribution.
- **Empirical validation and false-negative diagnostics.** On NQ, TriviaQA, and MS MARCO Passage, FAT improves strong dense retrieval baselines, including a static LLM-relabeling ablation. An independent ensemble-judged diagnostic shows that standard benchmark labels undercount recovered positives, showing that standard evaluation understates FAT's gains.

2 Related Work

Dense retrieval and hard negatives. DPR established dual encoders as effective first-stage retrievers for open-domain QA (Karpukhin et al., 2020). ANCE introduced model-hard negatives from a refreshed ANN index (Xiong et al., 2020). AR2 and RocketQAv2 couple a dual-encoder retriever with a cross-encoder ranker for stronger supervision (Zhang et al., 2022a; Ren et al., 2021). Retrieval-oriented pretraining methods such as Condenser, coCondenser, RetroMAE, and Contriever improve the encoder via pre-training before task-specific mining (Gao and Callan, 2021, 2022; Xiao et al., 2022; Izacard et al., 2022).

False negatives in hard-negative mining. RocketQA, SimANS, TriSampler, and CCR reduce damage from noisy negatives by filtering, reweighting, or avoiding ambiguous hard candidates (Qu et al., 2021; Zhou et al., 2022; Yang et al., 2024;

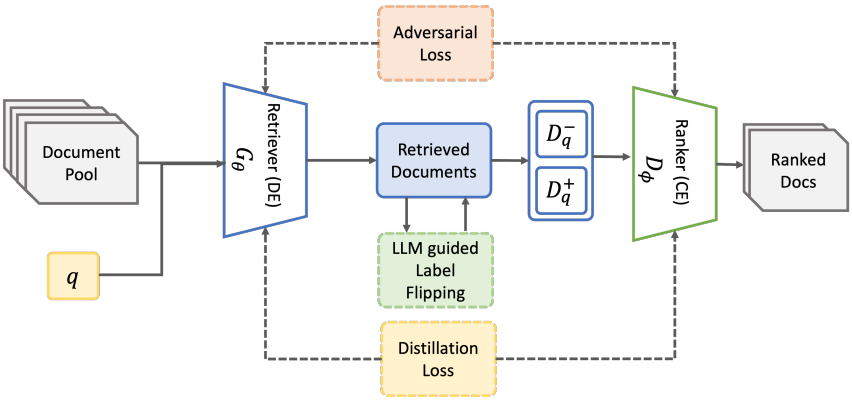

Figure 1: **FAT in AR^2 co-training.** At each index refresh, the retriever mines hard passages and the LLM judge recovers high-confidence hidden positives. The corrected list trains the cross-encoder teacher, which is distilled back into the dual encoder; once the teacher stabilizes, ALD decays adversarial pressure.

Wang et al., 2024). FAT differs by recovering high-confidence false negatives as positives. This is especially important under sparse QA labels, where the best mined passages are often valid but unannotated.

LLM relabeling of false negatives. The most directly related recent work recovers false negatives rather than discarding them. RLHN (Thakur et al., 2025) uses cascading LLM prompts to identify and relabel false negatives mined from a *fully converged* retriever. It then warms the model for a few epochs on this static flipped dataset (positives + negatives from the final converged pool). RLHN convincingly demonstrates that LLM-based positive recovery is a reliable training signal and improves both retrievers and rerankers on BEIR and AIR-Bench. FAT builds directly on this foundation but introduces two fundamental upgrades that turn a late, one-shot post-processing step into a dynamic, drift-aware training loop. First, RLHN audits only after convergence, by that point the retriever has already learned to suppress many unlabeled positives, pushing them out of the top mined candidates. A single audit of the final pool therefore misses the very examples it was designed to recover. In contrast, FAT re-audits the mined pool $C_t(q)$ at every ANN index refresh *during* training precisely while hidden positives are still surfacing in the high-risk region. Second, FAT treats LLM judgments as noisy, as each flip is down-weighted by the estimated judge precision $(1 - \sigma_t)$ to prevent over-flips from damaging the contrastive loss. These two ideas of drift-aware re-auditing and precision-weighted flips, are formalized in our drifted recursion and micro-flip budget.

Relation to positive-unlabeled learning. Although FAT recovers hidden positives from unlabeled candidates, it is not an instance of positive-unlabeled (PU) learning. Standard PU methods (Kiryo et al., 2017; Kato et al., 2019; Bekker and Davis, 2020) assume a static unlabeled distribution and focus on estimating classification risk or class priors. Dynamic hard-negative mining, however, creates a non-stationary, model-dependent unlabeled pool whose distribution shifts with every retriever update and ANN index refresh. FAT is therefore best viewed as a retrieval-specific, drift-aware correction mechanism and not as a generic static PU classifier.

LLMs as judges. LLM-as-judge methods increasingly provide scalable semantic supervision (Zheng et al., 2023; Gu et al., 2024). Our use is deliberately local and offline, the LLM judges a tiny mined subset and the learned dual encoder performs all test-time retrieval. This gives the system access to richer semantic judgments during training without incurring LLM retrieval latency at inference.

3 Method

3.1 Dense Retrieval with Hard Negatives

Let Q be a set of queries and D a corpus. A dual encoder G_θ scores query-document pairs by $s_\theta(q, d) = \langle h_\theta^q(q), h_\theta^d(d) \rangle$. For each query q, let $D_{q,0}^+$ denote the seed positives. At round t, the current retriever G_{θ_t} re-embeds the corpus, refreshes the ANN index, and mines $C_t(q) = \mathrm{TopK}_{\mathrm{ANN}}(q; G_{\theta_t}, D)$. The unlabeled mined pool is $M_t(q) = C_t(q) \setminus D_{q,0}^+$. Standard hard-negative training treats all items in $M_t(q)$ as negatives, con-

trasting each positive anchor a against a mined list with

$$\mathcal{L}^{(1)}_{\mathrm{NCE}}(\theta; q, C, a) = \log\left(1 + \sum_{d \in C} e^{(s_\theta(q,d) - s_\theta(q,a))/\tau}\right). \tag{1}$$

Let $g(q, d) \in \{+1, -1\}$ be the latent clean relevance label. A false negative is an unlabeled mined item $d \in M_t(q)$ with $g(q, d) = +1$. We write $\pi^+_t = \Pr[g = +1 \mid d \in M_t(q)]$ for the pre-judge hidden-positive prevalence in the mined pool. After the LLM audit, let $P^-_t(\cdot \mid q)$ be the distribution over items still kept as negatives, and write $\rho_t = \mathbb{E}_q[\Pr_{d \sim P^-_t(\cdot|q)}(g(q, d) = +1)]$ for their post-judge hidden-positive prevalence. The goal of FAT is to reduce ρ_t while converting high-confidence hidden positives into additional positive supervision.

3.2 Flip-and-Train: Local LLM Audits

FAT uses the LLM as a local verifier, not as a retriever. The dual encoder supplies global recall by mining from the full corpus; the LLM only audits a small, budgeted subset of high-risk mined candidates. For each audited pair (q, d), the judge receives a compact prompt (Figure 2, left) and returns a brief rationale and a binary decision $\pi(q, d) \in \{\text{YES}, \text{NO}\}$. A candidate is flipped to positive only when the judge returns YES and the precision gate passes. Figure 2 (right) shows a representative recovered false negative.

3.3 Judge Errors & Precision-Weighted Flips

We summarize the LLM judge by two local error rates in the audited mined pool: the over-flip rate $\alpha = \Pr[\pi = \text{YES} \mid g = -1]$ and the under-flip rate $\gamma = \Pr[\pi = \text{NO} \mid g = +1]$. These errors affect different parts of the system. The under-flip rate γ controls how many hidden positives remain in the negative pool, while the over-flip rate α controls how much noise is injected into the positive set. The negative-pool contraction condition is judge advantage, $\alpha + \gamma < 1$; low α is additionally required for safe positive flips.

For refresh t, let $F_t(q) = \{d \in M_t(q) : \pi(q, d) = \text{YES}\}$ be the judge-YES stratum before precision gating. By Bayes' rule, the false-discovery rate among new judge-YES candidates is

$$\sigma_t = \Pr[g = -1 \mid \pi = \text{YES}] = \frac{\alpha(1 - \pi^+_t)}{(1 - \gamma)\pi^+_t + \alpha(1 - \pi^+_t)}. \tag{2}$$

Algorithm 1 FAT with precision-weighted LLM flips and ALD

Require: Retriever G_θ, ranker D_ϕ, corpus $\mathcal{D}$, seed positives $\mathcal{D}^+_{q,0}$, LLM judge J, iterations T, retrieval depth K_{ret}, judge budget B, flip weight w_{flip}, initial adversarial weight $\lambda_{\text{adv},0}$, ALD decay rate $\beta \in (0, 1)$

Warm-start G_θ and D_ϕ on the seed training data.
Initialize weighted positives $\mathcal{P}_q \leftarrow \{(d, 1) : d \in \mathcal{D}^+_{q,0}\}$ and $\lambda_{\text{adv}} \leftarrow \lambda_{\text{adv},0}$.
for $t = 1, 2, \ldots, T$ **do**
  Train one AR2 co-training block using weighted positives $\mathcal{P}_q$, current hard negatives, and λ_{adv}.
  if the cross-encoder development performance has plateaued **then**
    $\lambda_{\text{adv}} \leftarrow \beta\lambda_{\text{adv}}$. ▷ ALD
  end if
  Re-embed the corpus with G_θ and retrieve $C_t(q) = \mathrm{TopK}_{K_{\text{ret}}}(q; G_\theta, \mathcal{D})$.
  Form candidate false negatives $M_t(q) = C_t(q) \setminus \mathrm{supp}(\mathcal{P}_q)$.
  Select judged pairs $S_t \subseteq \{(q, d) : d \in M_t(q)\}$ by retriever hardness and cross-encoder reranking, with $|S_t| \le B$.
  for each $(q, d) \in S_t$ **do**
    Query $J(q, d)$ and cache the rationale and decision $\pi(q, d)$.
    if $\pi(q, d) = \text{YES}$ **then**
      Remove d from hard-negative training for q and add (d, w_{flip}) to $\mathcal{P}_q$. ▷ soft precision gate
    else
      Keep d eligible as a hard negative for q.
    end if
  end for
end for
return trained retriever G_θ

Thus the posterior precision of a new flip is $\Pr[g = +1 \mid \pi = \text{YES}] = 1 - \sigma_t$. This is the quantity used by the precision gate: a judge-YES item is accepted only when its estimated precision clears the threshold. Appendix B.1 derives Eq. (2).

Let $A_t(q) \subseteq F_t(q)$ be the accepted flips after precision gating, and let $A_{\le t}(q)$ be all flips accepted up to round t. Seed positives keep unit weight, while accepted flips are trained as positive anchors with reduced weight:

$$\mathcal{L}_{\text{FaT}}(\theta; q) = \sum_{d^+ \in D^+_{q,0}} \mathcal{L}^{(1)}_{\text{NCE}}(\theta; q, \widetilde{C}_t(q), d^+) + \sum_{a \in A_{\le t}(q)} w_a \, \mathcal{L}^{(1)}_{\text{NCE}}(\theta; q, \widetilde{C}_t(q), a), \tag{3}$$

with $0 \le w_a \le 1 - \widehat{\sigma}_t$. A flipped item is therefore

Compact judge prompt used by FAT

Task. Decide whether the candidate passage is a valid answer-bearing passage for the question.
Question: {question}
Gold answers: {gold answer strings, if available}
Known positive examples: {1–2 positive passages}
Candidate passage: {mined hard-negative passage}
Decision rule. Return YES only if the passage directly answers the question, entails a gold answer, or gives an unambiguous alias/paraphrase. Return NO if it is merely topically related, ambiguous, or missing the answer.
Output format.
Rationale: {one or two sentences}
Decision: YES or NO

Example false negative recovered by FAT

Question: Which U.S. president created the National Forest Service and five national parks?
Gold answers: Theodore Roosevelt; T. Roosevelt.
Candidate passage: "Roosevelt established the United States Forest Service, signed into law the creation of five National Parks ..."
Why exact matching fails: the passage says "Roosevelt" rather than the full string "Theodore Roosevelt."
Judge rationale: The passage attributes the relevant actions to Roosevelt and matches the known positive evidence for Theodore Roosevelt.
Decision: YES → flip to positive.

Figure 2: **Local LLM audit used by FAT.** The judge does not search the corpus. It verifies whether a candidate already surfaced by the retriever is a valid but unlabeled positive. The full prompt template is in Appendix A.

still positive supervision, but its positive-anchor contribution is capped by estimated judge precision. The optimal constant weight for a newly flipped stratum under squared label error is $w^{\star}_{\text{flip}} = 1 - \sigma_t$; the derivation is in Appendix B.3.

In our experiments, seed positives keep weight 1, while accepted LLM flips receive the fixed conservative weight $w_{\text{flip}} = 0.85$, calibrated in Section 3.5 from the 500-sample manual audit of flipped candidates and the posterior precision calculation.

3.4 Drift-Aware Contraction

We now connect the judge model to ranking. The analysis separates three effects. First, an advantaged judge reduces hidden-positive mass among candidates retained as negatives. Second, dynamic ANN refreshes can inject new hidden positives through mined-pool drift. Third, precision-weighted flips bound the damage from mistaken positives in the InfoNCE objective. Full proofs are in Appendix B.

Lemma 1 (Static judge contraction)**.** *For a fixed mined pool with hidden-positive rate ρ, judging the pool and retaining only items marked* NO *maps ρ to*

$$f(\rho) = \frac{\gamma\rho}{(1-\alpha)(1-\rho) + \gamma\rho}. \tag{4}$$

If $\alpha + \gamma < 1$, then $f(\rho) < \rho$ for every $0 < \rho < 1$. Moreover, with $c = (1-\alpha-\gamma)/(1-\alpha) > 0$, $f(\rho) \le \rho - c\rho(1-\rho)$. Hence, on any minority region $\rho \le \rho_{\max} < 1$, $f(\rho) \le r\rho$ for $r = 1 - c(1-\rho_{\max}) < 1$. The local slope at zero is $\kappa = f'(0) = \gamma/(1-\alpha)$.

Appendix B.4 proves the lemma. The important point is that judge advantage gives strict one-step contraction for every non-degenerate fixed pool; κ only describes the limiting late-stage slope near zero. The finite-region geometric rate used below is r.

Lemma 2 (Drifted mined-pool recursion)**.** *For each query q, let $\rho_t(q) = \Pr[g(q,d) = +1 \mid d \sim P_t^{-}(\cdot \mid q)]$, and let $\rho_t = \mathbb{E}_q[\rho_t(q)]$. Let $U_t(\cdot \mid q)$ be the reference distribution obtained by re-judging draws from $P_t^{-}(\cdot \mid q)$ with the same judge and retaining only* NO *items. Define the drift budget*

$$\delta_t = \mathbb{E}_q\big[\mathrm{TV}\big(P_{t+1}^{-}(\cdot \mid q), U_t(\cdot \mid q)\big)\big]. \tag{5}$$

If $\rho_t(q) \le \rho_{\max} < 1$ on the audited query distribution, then

$$\rho_{t+1} \le r\rho_t + \delta_t, \qquad 0 \le r < 1. \tag{6}$$

Appendix B.5 proves the query-wise recursion. The term δ_t is the non-stationarity introduced by ANN refresh, where FAT makes progress when judge contraction dominates the fresh hidden positives injected by the moving mined pool.

We connect this recurrence to ranking through a listwise-to-pairwise calibration.

Lemma 3 (Listwise-to-pairwise calibration)**.** *For any mined list $C_t(q)$, positive anchor d^+, and candidate $d^- \in C_t(q)$,*

$$\mathbf{1}[s_\theta(q, d^-) \geq s_\theta(q, d^+)] \leq \frac{\mathcal{L}^{(1)}_{\mathrm{NCE}}(\theta; q, C_t(q), d^+)}{\log 2}. \tag{7}$$

Appendix B.6 proves Lemma 3. Let $X_t = \Pr[s_{\theta_t}(q, d^-) \geq s_{\theta_t}(q, d^+)]$, where d^+ is a true positive and $d^- \sim P_t^-(\cdot \mid q)$. Let η be the residual pairwise error on true negatives after training, and let ζ_t be the additional slack from imperfect weighted flips. Appendix B.7 bounds ζ_t in terms of flip contamination, flip count, and the chosen flip weight.

Theorem 1 (Local geometric convergence)**.** *Assume that for all $t \geq T$, the audited mined pool satisfies Eq. (6) with $0 \leq r < 1$. Let $\bar{\delta} = \sup_{i \geq T} \delta_i$ and $\bar{\zeta} = \sup_{i \geq T} \zeta_i$. Then, for all $t \geq T$,*

$$\rho_t \leq r^{t-T}\rho_T + \frac{\bar{\delta}}{1-r}(1 - r^{t-T}), \tag{8}$$

$$X_t \leq \eta + (1-\eta)\left[r^{t-T}\rho_T + \frac{\bar{\delta}}{1-r}(1 - r^{t-T})\right] + \bar{\zeta}. \tag{9}$$

Consequently, the limiting risk floor is

$$X_{\mathrm{floor}} \triangleq \eta + (1-\eta)\frac{\bar{\delta}}{1-r} + \bar{\zeta}, \qquad \limsup_{t \to \infty} X_t \leq X_{\mathrm{floor}} \tag{10}$$

Appendix B.8 proves the theorem. The result is a three-gate statement. The condition $\alpha + \gamma < 1$ contracts hidden positives among retained negatives. A small α is also needed because it controls the false-discovery rate σ_t among new flips. Precision-weighted InfoNCE limits the influence of contaminated flips through ζ_t. The floor X_{floor} in Eq. (10) has three terms: residual error on true negatives η, mined-pool drift $\bar{\delta}/(1-r)$, and flip slack $\bar{\zeta}$. These correspond to distinct idealizations. The residual term vanishes only under perfect optimization of a sufficiently expressive retriever; the drift term vanishes only when the mined pool stops moving or is re-audited without lag; and the flip slack vanishes only when weighted flips become equivalent to clean positives, i.e., false-discovery and underweighting slack both vanish. Bayes-consistent ranking on the evolving mined-pool distribution is therefore recovered only in the joint limit $\eta \to 0$, $\delta_t \to 0$, and $\zeta_t \to 0$ (Appendix B.9). Theorem 1 matters because it gives geometric contraction to an explicit, measurable floor without assuming those limits hold at finite time.

Judge	$\hat{\alpha}$	$\hat{\gamma}$	$\hat{\alpha}+\hat{\gamma}$	$\hat{\kappa}$
DeepSeek-R1-Distill-Qwen-7B	0.35±0.05	0.03±0.02	0.38	0.046
DeepSeek-R1-Distill-Llama-8B	0.38±0.06	0.02±0.01	0.40	0.032
DeepSeek-R1-Distill-Qwen-32B	0.21±0.08	0.05±0.02	0.26	0.063
GPT-OSS-120B (medium)	0.09±0.04	0.05±0.01	0.14	0.055
Qwen3-235B-A22B-Thinking	0.05±0.02	0.04±0.03	0.09	0.042

Table 1: Judge error estimation on 500-sample manual audit . $\hat{\alpha}$: over-flip rate (NO labeled YES). $\hat{\gamma}$: under-flip rate (YES labeled NO); mean ± std.

3.5 Empirical Check: Judge Advantage

The theory depends on measurable judge quantities: the over-flip rate α, the under-flip rate γ, the local slope $\kappa = \gamma/(1-\alpha)$, and the posterior flip precision $1 - \sigma_t$. We therefore audit judge decisions on hard-mined NQ candidates (random 500 samples) from intermediate retriever checkpoints. Proxy-gold labels are obtained by manually audit of this small sample.

The Qwen3-235B-A22B-Thinking training judge satisfies the advantage condition by a large margin: $\hat{\alpha} + \hat{\gamma} = 0.09 < 1$, and its point-estimate local slope is $\hat{\kappa} = 0.04/(1-0.05) = 0.042$. The small $\hat{\gamma}$ supports contraction of retained negatives; the small $\hat{\alpha}$ is what makes high-precision positive flips possible.

Flip-weight calibration. We calibrate the fixed flip weight from the first high-prevalence NQ refresh. In this refresh, Qwen3 marks $\hat{y}_1 = 14{,}060/50{,}000 = 28.1\%$ of audited mined candidates as YES. Since $\hat{y}_1$ is a raw judge-YES rate rather than an independent prevalence estimate, we debias it using the audit estimates:

$$\widehat{\pi}_1^+ = \frac{\hat{y}_1 - \hat{\alpha}}{1 - \hat{\gamma} - \hat{\alpha}}, \qquad \widehat{w}_1^\star = \frac{(1-\hat{\gamma})\widehat{\pi}_1^+}{\hat{y}_1}. \tag{11}$$

With $\hat{\alpha} = 0.05$ and $\hat{\gamma} = 0.04$, this gives $\widehat{\pi}_1^+ \approx 25.4\%$ and $\widehat{w}_1^\star \approx 0.87$. Varying $\hat{\alpha}$ and $\hat{\gamma}$ within their audit standard deviations gives an approximate first-refresh precision range of 0.81–0.92. We therefore use the conservative fixed value $w_{\mathrm{flip}} = 0.85$ for accepted flips.

3.6 Adversarial Loss Decay

FAT corrects the supervision learned by the cross-encoder ranker. In AR^2-style co-training, however, the deployed model is the dual-encoder retriever.

Hyper-parameter	Value / policy
DE / CE encoder	ERNIE-2.0-base dual encoder / ERNIE-2.0-large cross encoder
Optimizer	AdamW, $\beta = (0.9, 0.999)$, gradient clipping 1.0
DE / CE learning rate	$1e{-5}$ / $1e{-6}$, warmup ratio 0.1
Iterations T / steps per iter	12 / 2,000
Steps per iter	1.5k DE updates + 0.5k CE updates
Per-GPU bsz / effective bsz	8 / 64
λ_{adv} / λ_{distill} / τ	0.5 / 0.5 / 1.0
Adversarial decay trigger / rate	CE dev plateau, patience 3 / $0.2\times$ per iter
AR2 hard negatives K	15, sampled uniformly
Refresh index	FAISS IndexFlatIP, inner product, top-100
Training judge	Qwen3-235B-A22B-Thinking-2507
Context / output / thinking	8192 / 32k / 8k tokens
Judge sampling	temperature 0.6, top-p 0.95, top-k 20
Candidate preselection	CE rerank 250k global + 250k rescue candidates
Judged subset / refresh	50k hardest
Flip weight	0.85 (dataset and Judge specific)

Table 2: FAT hyper-parameters for reproducibility.

The retriever is not trained by distillation alone; it also receives an adversarial signal that encourages it to surface difficult examples for the ranker:

$$L(\theta) = \lambda_{\text{distill}} L_{\text{distill}}(\theta; \phi) + \lambda_{\text{adv}} L_{\text{adv}}(\theta; \phi). \tag{12}$$

This adversarial term is useful early because it exposes the teacher to informative hard examples. After the ranker has stabilized on corrected lists, however, a fixed nonzero λ_{adv} keeps pulling the retriever away from the corrected teacher distribution.

Let $\phi_t \to \phi^\star$, and let $\theta^\star = \arg\min_\theta L_{\text{distill}}(\theta; \phi^\star)$. Under local strong convexity around $\theta^\star$, the stationary point $\hat{\theta}$ of the fixed-adversary objective satisfies $\hat{\theta} - \theta^\star = -\tau H_{\text{distill}}^{-1} g_{\text{adv}} + O(\tau^2)$, where $\tau = \lambda_{\text{adv}}/\lambda_{\text{distill}}$ and $g_{\text{adv}} = \nabla_\theta L_{\text{adv}}(\theta^\star; \phi^\star)$. Thus any fixed nonzero adversarial weight induces an asymptotic displacement unless $g_{\text{adv}} = 0$; Appendix C.1 gives the derivation.

ALD keeps the adversarial signal active during the exploration phase and decays it only after the ranker stabilizes. Let $L_{\phi,t}^{\text{dev}}$ be the ranker development loss and $\bar{L}_{\phi,t}^{(W)} = W^{-1} \sum_{i=0}^{W-1} L_{\phi,t-i}^{\text{dev}}$. We trigger decay when the relative improvement $\Delta_t^{\text{rel}} = (\bar{L}_{\phi,t-W}^{(W)} - \bar{L}_{\phi,t}^{(W)}) / \max(\bar{L}_{\phi,t-W}^{(W)}, \epsilon_0)$ is below a threshold for P consecutive windows. After the gate fires, we apply step-wise exponential decay to λ_{adv}. ALD is not a separate mechanism, it is the distillation stabilizer that lets FAT's corrected cross-encoder signal reach the deployed dual encoder. We keep the adversarial term during early hard-example exploration, then decay it after the ranker stabilizes so that late training follows the corrected distillation target rather than continuing to chase adversarial candidates. Appendix C.3 gives the corresponding stochastic-approximation. Without ALD, FAT convergence becomes highly unstable and the gains over AR2 are significantly reduced, despite clear improvements in cross-encoder performance, making it a critical component of FAT.

4 Experiments

We evaluate on Natural Questions (NQ) (Kwiatkowski et al., 2019), TriviaQA (TQ) (Joshi et al., 2017), and MS MARCO Passage (MS-Pas) (Bajaj et al., 2016). For NQ/TQ we report test R@5/20/100; for MS-Pas we report dev MRR@10 and R@50. We follow the AR2/CCR setup of ERNIE-2.0 base dual encoder, ERNIE-2.0 large cross encoder, batch size 64, 15 hard negatives per query, FAISS IndexFlatIP indexing, and ANN refresh every 2k retriever steps (Zhang et al., 2022a; Wang et al., 2024; Douze et al., 2024). FAT uses Qwen3-235B-A22B-Thinking as the main training judge. The Qwen3 FAT ran on $2\times$ p5en.48xlarge nodes; each node has 1 8 H200 GPUs, with 145 GB per GPU. Since this setting is expensive, we also report FAT with GPT-OSS-120B (medium), a cheaper judge that still satisfies the theory's judge-advantage condition ($\hat{\alpha} + \hat{\gamma} < 1$) and can run on a single A100. For RLHN-style static relabeling, we first train AR2 to convergence. We then audit the final AR2-mined pool once, use the flipped positives and retained negatives to form a fixed newly labeled train set, and continue training on this static set till dev performance converges. FAT runs for 12 iterations, auditing up to 50k mined candidates per iteration. To ensure fairness, the RLHN-style static baseline audits the final converged AR2 pool once and flips exactly the same total number of examples (600k hardest query-passage pairs).

4.1 Results and False-Negative Diagnostic

Tables 3–4 give a compact official-and-diagnostic view. On official labels, Qwen3 FAT is best on every metric, gaining over CCR $+1.4/+0.8/+0.3$ R@5/R@20/R@100 on NQ, $+1.8/+1.3/+0.8$ on TQ, and $+1.6/+1.0$ MRR@10/R@50 on MS-Pas. RLHN-style relabeling improves AR2 but

Method	NQ test			TQ test			MS-Pas dev	
	R@5	R@20	R@100	R@5	R@20	R@100	MRR@10	R@50
DPR (Karpukhin et al., 2020)	–	78.4	85.3	–	79.3	84.9	–	–
ANCE (Xiong et al., 2020)	71.8	81.9	87.5	–	80.3	85.3	33.0	81.1
RocketQA (Qu et al., 2021)	74.0	82.7	88.5	–	–	–	37.0	85.5
Condenser (Gao and Callan, 2021)	–	83.2	88.4	–	81.9	86.2	36.6	–
coCondenser (Gao and Callan, 2022)	75.8	84.3	89.0	76.8	83.2	87.3	38.2	–
ERNIE-Search (Lu et al., 2022)	77.0	85.3	89.7	–	–	–	40.1	87.7
MVR (Zhang et al., 2022b)	76.2	84.8	89.3	77.1	83.4	87.4	–	–
PROD (Lin et al., 2023)	75.6	84.7	89.6	–	–	–	39.3	87.0
COT-MAE (Wu et al., 2023)	75.5	84.3	89.3	–	–	–	39.4	87.0
ANCE-Tele (Sun et al., 2022)	–	84.9	89.7	–	83.4	87.3	39.1	–
AR2 (Zhang et al., 2022a)	77.9	86.0	90.1	78.2	84.4	87.9	39.5	87.8
AR2 + SimANS (Zhou et al., 2022)	78.6	86.2	90.3	78.6	84.6	88.1	40.9	88.7
AR2 + Passage Sieve (CCR) (Wang et al., 2024)	79.2	86.4	90.7	78.7	84.7	88.2	39.8	88.3
FAT (GPT-OSS-120B medium reasoning)	79.9	87.0	90.5	80.0	85.0	88.1	41.0	88.5
FAT (Qwen3-235B-A22B-Thinking)	**80.6**	**87.2**	**91.0**	**80.5**	**86.0**	**89.0**	**41.4**	**89.3**

Table 3: **Main results.** Numbers above the second midrule are quoted verbatim from the respective original papers, under each paper's own best-reported configuration (− = not reported in the source paper). AR2, AR2+CCR, AR2+SimANS,AR2+RLHN, and FAT (below the second midrule) are re-implemented by us under identical settings, for direct, controlled comparison.

Method	NQ test				TQ test			
	Std R@5	Judge R@5	Std R@20	Judge R@20	Std R@5	Judge R@5	Std R@20	Judge R@20
AR2 + RLHN-style static relabeling (Thakur et al., 2025)	78.7	83.0	86.4	90.1	78.6	82.1	84.7	88.6
AR2 + Passage Sieve (CCR)	79.2	82.5	86.4	89.9	78.7	81.8	84.7	88.1
FAT (GPT-OSS-120B med.)	79.9	84.4	87.0	91.7	80.0	84.8	85.0	89.1
FAT (Qwen3-235B-A22B-Thinking)	**80.6**	**84.9**	**87.2**	**92.0**	**80.5**	**84.9**	**86.0**	**90.3**

Table 4: **Unanimous false-negative diagnostic.** Judged columns add top-20 test-set false negatives as positives only when Gemma-4-31B-it, Qwen3-235B-A22B-Thinking, and GPT-OSS-120B (high) all return **YES**.

mostly matches CCR: it spends the matched 600k audit budget on one fixed final pool, whereas FAT spreads audits across 12 iterations and tracks the moving candidate pool. The diagnostic re-judges official top-20 false negatives with an ensemble, Gemma-4-31B-it, Qwen3-235B-A22B-Thinking, and GPT-OSS-120B (high), flipping a passage only under unanimous YES. Since the training judge sits in this ensemble, the diagnostic is not fully independent of FAT; unanimity does not remove this circularity but suppresses it, as the two non-training judges must also concur, so the training judge alone cannot inflate any score. The judged ordering matches the official one, and the gap widens for CCR and RLHN, trained without these judges, so we read it as a conservative lower bound, not a replacement for the official metric: Qwen3 FAT beats CCR by $+\mathbf{2.4}/+\mathbf{2.1}$ R@5/R@20 on NQ and $+\mathbf{3.1}/+\mathbf{2.2}$ on TQ.

5 Conclusion

We introduced FAT, a dynamic hidden-positive recovery loop for dense retrieval. Rather than filtering suspected false negatives, FAT uses an offline LLM judge to repeatedly audit the high-risk region of mined hard negatives at every ANN refresh, flipping high-confidence latent positives with precision-weighted supervision. We instantiate FAT in a standard dual-encoder retriever and cross-encoder ranker co-training framework. On Natural Questions, TriviaQA, and MS MARCO Passage, FAT delivers consistent absolute gains of 1–2% in R@K over strong baselines, including CCR and static LLM relabeling. An independent LLM-judged diagnostic reveals that standard sparse labels substantially undercount recovered positives, widening the gains to 2–3% and confirming that FAT surfaces answer-bearing passages missed by conventional evaluation. This work demonstrates that turning latent positives into training assets, rather than treating them as noise, yields stronger, more Bayes-consistent dense retrievers.

Limitations

FAT adds offline training cost, since it relies on an LLM judge and periodic ANN-index refreshes.

This cost is bounded by the micro-flip budget and by caching judged decisions, but it remains higher than that of purely local denoising methods. FaT is therefore most appropriate in settings where the observed 1–2% absolute gains in R@K carry substantial downstream value, for example, large-scale retrieval systems in which small recall improvements compound across many queries, and may not be worth the additional compute in lower-stakes or resource-constrained applications. Our evaluation focuses on open-domain QA and MS MARCO-style passage retrieval; broader benchmarks such as BEIR, along with multilingual and non-QA domains, would further test generality. The theory is also intentionally explicit about its idealizations; practical systems converge to a nonzero risk floor unless mined-pool drift, flip contamination, and optimization residue are all small, so the Bayes-consistency result should be read as a limiting statement rather than a guarantee at finite training time. Finally, LLM judges can carry biases and may be unreliable on genuinely ambiguous relevance cases. We therefore recommend independent audit ensembles or human adjudication for any final test-set diagnostics, and we treat the LLM purely as an offline training-time verifier rather than as a component of the deployed system.

Ethics Statement

FaT uses LLMs to relabel training examples, so incorrect or biased judgments can alter supervision. We mitigate this by limiting flips to high-confidence, auditable decisions and by reporting judge error estimates. In deployment, no LLM is called at query time. Any release of flipped labels should include provenance, prompts, model versions, and a description of how disagreements or ambiguous cases were handled.

References

Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, Mir Rosenberg, Xia Song, Alina Stoica, Saurabh Tiwary, and Tong Wang. MS MARCO: A human generated MAchine reading COmprehension dataset. *arXiv preprint arXiv:1611.09268*, 2016. URL https://arxiv.org/abs/1611.09268.

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

Matthijs Douze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou. The FAISS library. *arXiv preprint arXiv:2401.08281*, 2024.

Luyu Gao and Jamie Callan. Condenser: A pre-training architecture for dense retrieval. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 981–993, 2021.

Luyu Gao and Jamie Callan. Unsupervised corpus aware language model pre-training for dense passage retrieval. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, pages 2843–2853, 2022. doi: 10.18653/v1/2022.acl-long.203.

Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, Saizhuo Wang, Kun Zhang, Yuanzhuo Wang, Wen Gao, Lionel Ni, and Jian Guo. A survey on LLM-as-a-judge. *arXiv preprint arXiv:2411.15594*, 2024. URL https://arxiv.org/abs/2411.15594.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. Unsupervised dense information retrieval with contrastive learning. *Transactions on Machine Learning Research*, 2022. URL https://openreview.net/forum?id=jKN1pXi7b0.

Mandar Joshi, Eunsol Choi, Daniel S. Weld, and Luke Zettlemoyer. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pages 1601–1611, 2017.

Vladimir Karpukhin, Barlas Oguz, 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)*, pages 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*, 2019.

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

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466, 2019. doi: 10.1162/tacl_a_00276.

Zhenghao Lin, Yeyun Gong, Xiao Liu, Hang Zhang, Chen Lin, Anlei Dong, Jian Jiao, Jingwen Lu, Daxin Jiang, Rangan Majumder, and Nan Duan. PROD: Progressive distillation for dense retrieval. In *Proceedings of the ACM Web Conference 2023*, pages 3299–3308, 2023. doi: 10.1145/3543507.3583421.

Yuxiang Lu, Yiding Liu, Jiaxiang Liu, Yunsheng Shi, Zhengjie Huang, Shikun Feng, Yu Sun, Hao Tian, Hua Wu, Shuaiqiang Wang, Dawei Yin, and Haifeng Wang. ERNIE-Search: Bridging cross-encoder with dual-encoder via self on-the-fly distillation for dense passage retrieval. *arXiv preprint arXiv:2205.09153*, 2022. doi: 10.48550/arXiv.2205.09153. URL https://arxiv.org/abs/2205.09153.

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*, pages 5835–5847, 2021. doi: 10.18653/v1/2021.naacl-main.466.

Ruiyang Ren, Yingqi Qu, Jing Liu, Wayne Xin Zhao, QiaoQiao She, Hua Wu, Haifeng Wang, and Ji-Rong Wen. RocketQAv2: A joint training method for dense passage retrieval and passage re-ranking. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2825–2835, 2021. doi: 10.18653/v1/2021.emnlp-main.224.

Si Sun, Chenyan Xiong, Yue Yu, Arnold Overwijk, Zhiyuan Liu, and Jie Bao. Reduce catastrophic forgetting of dense retrieval training with teleportation negatives. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6639–6654, 2022. doi: 10.18653/v1/2022.emnlp-main.445. URL https://aclanthology.org/2022.emnlp-main.445/.

Nandan Thakur, Crystina Zhang, Xueguang Ma, and Jimmy Lin. Hard negatives, hard lessons: Revisiting training data quality for robust information retrieval with LLMs. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng, editors, *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 9064–9083, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-335-7. doi: 10.18653/v1/2025.findings-emnlp.481. URL https://aclanthology.org/2025.findings-emnlp.481/.

Shiqi Wang, Yeqin Zhang, and Cam-Tu Nguyen. Mitigating the impact of false negative in dense retrieval with contrastive confidence regularization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 19171–19179, 2024. doi: 10.1609/aaai.v38i17.29885.

Xing Wu, Guangyuan Ma, Meng Lin, Zijia Lin, Zhongyuan Wang, and Songlin Hu. ConTextual masked auto-encoder for dense passage retrieval. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 4738–4746, 2023. doi: 10.1609/aaai.v37i4.25598.

Shitao Xiao, Zheng Liu, Yingxia Shao, and Zhao Cao. RetroMAE: Pre-training retrieval-oriented language models via masked auto-encoder. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 538–548, 2022. doi: 10.18653/v1/2022.emnlp-main.35.

Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul Bennett, Junaid Ahmed, and Arnold Overwijk. Approximate nearest

neighbor negative contrastive learning for dense text retrieval. *arXiv preprint arXiv:2007.00808*, 2020.

Zhen Yang, Zhou Shao, Yuxiao Dong, and Jie Tang. TriSampler: A better negative sampling principle for dense retrieval. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 9269–9277, 2024. doi: 10.1609/aaai.v38i8.28779.

Hang Zhang, Yeyun Gong, Yelong Shen, Jiancheng Lv, Nan Duan, and Weizhu Chen. Adversarial retriever-ranker for dense text retrieval. In *International Conference on Learning Representations*, 2022a. URL https://openreview.net/forum?id=MR7XubKUFB.

Shunyu Zhang, Yaobo Liang, Ming Gong, Daxin Jiang, and Nan Duan. Multi-view document representation learning for open-domain dense retrieval. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5990–6000, 2022b. doi: 10.18653/v1/2022.acl-long.414. URL https://aclanthology.org/2022.acl-long.414/.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging LLM-as-a-judge with MT-Bench and chatbot arena. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Kun Zhou, Yeyun Gong, Xiao Liu, Wayne Xin Zhao, Yelong Shen, Anlei Dong, Jingwen Lu, Rangan Majumder, Ji-Rong Wen, and Nan Duan. Simans: Simple ambiguous negatives sampling for dense text retrieval. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 548–559, 2022.

A LLM Judge Prompt

Full judge prompt template

You are an expert relevance judge for open-domain question answering.
Your task is to decide whether the candidate passage is a valid answer-bearing passage for the question.
Return YES only if the passage directly answers the question, entails one of the gold answers, or provides an unambiguous alias or paraphrase of the answer. Return NO if the passage is merely topically related, ambiguous, missing the answer, or requires unsupported inference.
Question:
`{question}`
Gold answer strings, if available:
`{gold_answers}`
Known positive examples:
`{positive_examples}`
Candidate passage:
`{candidate_passage}`
Output exactly in the following format:
Rationale: `{step by step reasoning explaining the decision}`
Decision: `YES` or `NO`

B Proofs for Sections 3.3–3.4

This appendix gives the derivations behind the judge-error model, precision-weighted flips, and Theorem 1. We first derive posterior flip precision and the constant flip-weight rule. We then prove the static contraction map, extend it to a drifted mined-pool recursion, calibrate listwise InfoNCE to pairwise risk, and finally unroll the recursion to obtain the geometric bound.

B.1 Posterior Flip Precision

Let $\alpha = \Pr[\pi = \textsc{Yes} \mid g = -1]$ and $\gamma = \Pr[\pi = \textsc{No} \mid g = +1]$. Let $\pi_t^+ = \Pr[g = +1 \mid d \in M_t(q)]$ be the hidden-positive prevalence among audited mined candidates before judging. For a judge-YES candidate, Bayes' rule gives

$$\begin{aligned}\sigma_t &= \Pr[g = -1 \mid \pi = \textsc{Yes}] \\ &= \frac{\Pr[\pi = \textsc{Yes} \mid g = -1]\Pr[g = -1]}{\Pr[\pi = \textsc{Yes}]} \\ &= \frac{\alpha(1-\pi_t^+)}{(1-\gamma)\pi_t^+ + \alpha(1-\pi_t^+)}. \end{aligned} \tag{13}$$

Thus the posterior precision of a new flip is $\Pr[g = +1 \mid \pi = \textsc{Yes}] = 1 - \sigma_t$.

If the target minimum flip precision is $\omega_{\min}$, the precision gate requires $1 - \sigma_t \geq \omega_{\min}$, equivalently $\sigma_t \leq 1 - \omega_{\min}$. Solving the inequality in Eq. (13) for the admissible over-flip rate gives

$$\alpha \leq \frac{(1-\omega_{\min})(1-\gamma)\pi_t^+}{\omega_{\min}(1-\pi_t^+)}. \tag{14}$$

This shows why late-stage flips require stricter gates: when π_t^+ is small, the same posterior precision requires a much smaller over-flip rate α.

B.2 From Raw Judge-YES Rate to Prevalence

Let $y_t = \Pr[\pi = \textsc{Yes} \mid d \in M_t(q)]$ be the raw judge-YES rate in the audited mined pool. Under the same

confusion model,

$$\begin{aligned} y_t &= \Pr[\pi = \textsc{Yes} \mid g = +1]\Pr[g = +1] + \Pr[\pi = \textsc{Yes} \mid g = -1]\Pr[g = -1] \\ &= (1-\gamma)\pi_t^+ + \alpha(1-\pi_t^+) \\ &= \alpha + (1-\gamma-\alpha)\pi_t^+. \end{aligned} \tag{15}$$

When $\alpha + \gamma < 1$, the debiased prevalence estimate is

$$\widehat{\pi}_t^+ = \frac{\widehat{y}_t - \widehat{\alpha}}{1 - \widehat{\gamma} - \widehat{\alpha}}, \tag{16}$$

clipped to $[0, 1]$ in finite samples. Since the denominator of Eq. (13) is y_t, the plug-in posterior false-discovery estimate can also be written as

$$\widehat{\sigma}_t = \frac{\widehat{\alpha}(1 - \widehat{\pi}_t^+)}{\widehat{y}_t}. \tag{17}$$

B.3 Optimal Constant Flip Weight

Let $Y = \mathbf{1}[g = +1]$ be the clean binary label for a judge-YES candidate. If all flips in a stratum are assigned a constant weight $w \in [0, 1]$, the conditional squared label error is

$$\begin{aligned} J(w) &= \mathbb{E}[(w - Y)^2 \mid \pi = \textsc{Yes}] \\ &= (w-1)^2 \Pr(Y = 1 \mid \pi = \textsc{Yes}) + w^2 \Pr(Y = 0 \mid \pi = \textsc{Yes}). \end{aligned} \tag{18}$$

Differentiating gives $J'(w) = 2w - 2\Pr(Y = 1 \mid \pi = \textsc{Yes})$, so the unique minimizer is

$$w^\star = \Pr(Y = 1 \mid \pi = \textsc{Yes}) = \Pr[g = +1 \mid \pi = \textsc{Yes}] = 1 - \sigma_t. \tag{19}$$

Therefore posterior flip precision is the optimal constant flip weight for the stratum under squared label error. A conservative cap $w_{\text{flip}} < 1 - \sigma_t$ reduces the influence of false flips but also underuses true flips; this tradeoff is absorbed into the slack term ζ_t in Theorem 1.

B.4 Static Judge Contraction

We prove the static contraction statement. For compactness, write $A = 1 - \alpha$ and $B = \gamma$. For a fixed pool with hidden-positive rate ρ, the hidden-positive prevalence after retaining only items marked NO is

$$\begin{aligned} f(\rho) &= \Pr[g = +1 \mid \pi = \textsc{No}] \\ &= \frac{\Pr[\pi = \textsc{No} \mid g = +1]\Pr[g = +1]}{\Pr[\pi = \textsc{No}]} \\ &= \frac{B\rho}{A(1-\rho) + B\rho}. \end{aligned} \tag{20}$$

If $\alpha + \gamma < 1$, then $A > B$. For any $0 < \rho < 1$,

$$\frac{f(\rho)}{\rho} = \frac{B}{A(1-\rho) + B\rho} < 1, \tag{21}$$

because $A(1-\rho) + B\rho - B = (1-\rho)(A - B) > 0$. Hence $f(\rho) < \rho$ for every non-degenerate fixed pool.

For the stronger bound, observe that

$$\begin{aligned} \rho - f(\rho) &= \rho - \frac{B\rho}{A - (A-B)\rho} \\ &= \frac{A-B}{A-(A-B)\rho}\rho(1-\rho) \\ &\geq \frac{A-B}{A}\rho(1-\rho). \end{aligned} \tag{22}$$

Since $(A - B)/A = (1 - \alpha - \gamma)/(1 - \alpha)$, we get

$$f(\rho) \leq \rho - \frac{1-\alpha-\gamma}{1-\alpha}\rho(1-\rho). \tag{23}$$

The derivative is

$$\begin{aligned} f'(\rho) &= \frac{AB}{(A-(A-B)\rho)^2}, \\ f'(0) &= \frac{B}{A} = \frac{\gamma}{1-\alpha} \triangleq \kappa. \end{aligned} \tag{24}$$

Finally, if $\rho \leq \rho_{\max} < 1$ and $c = (1 - \alpha - \gamma)/(1 - \alpha)$, then

$$\begin{aligned} f(\rho) &\leq \rho - c\rho(1-\rho) \\ &\leq [1 - c(1 - \rho_{\max})]\,\rho. \end{aligned} \tag{25}$$

Thus $f(\rho) \leq r\rho$ with $r = 1 - c(1 - \rho_{\max}) < 1$, which is the finite-region contraction rate used in Theorem 1.

B.5 Drifted Mined-Pool Recursion

We prove the drifted recursion query-wise. For each query q, let $\rho_t(q) = \Pr[g(q, d) = +1 \mid d \sim P_t^-(\cdot \mid q)]$. Define $U_t(\cdot \mid q)$ as the distribution obtained by drawing from $P_t^-(\cdot \mid q)$, applying the same judge model, and retaining only candidates marked NO. By the static map,

$$\mathbb{E}_{d \sim U_t(\cdot|q)}\mathbf{1}[g(q, d) = +1] = f(\rho_t(q)). \tag{26}$$

For any indicator $h \in [0, 1]$, the variational characterization of total variation gives

$$\mathbb{E}_{P_{t+1}^-(\cdot|q)}h \leq \mathbb{E}_{U_t(\cdot|q)}h + \mathrm{TV}\big(P_{t+1}^-(\cdot \mid q), U_t(\cdot \mid q)\big). \tag{27}$$

Taking $h(q, d) = \mathbf{1}[g(q, d) = +1]$ yields

$$\begin{aligned} \rho_{t+1}(q) &\leq f(\rho_t(q)) + \delta_t(q), \\ \delta_t(q) &= \mathrm{TV}\big(P_{t+1}^-(\cdot \mid q), U_t(\cdot \mid q)\big). \end{aligned} \tag{28}$$

Averaging over queries gives

$$\begin{aligned} \rho_{t+1} &\leq \mathbb{E}_q[f(\rho_t(q))] + \delta_t, \\ \delta_t &= \mathbb{E}_q[\delta_t(q)]. \end{aligned} \tag{29}$$

If $\rho_t(q) \leq \rho_{\max} < 1$ on the audited query distribution, the finite-region bound in Eq. (25) gives $f(\rho_t(q)) \leq r\rho_t(q)$, so

$$\rho_{t+1} \leq r\rho_t + \delta_t, \qquad 0 \leq r < 1. \tag{30}$$

This proves the drifted mined-pool recursion used in Theorem 1.

B.6 Listwise-to-Pairwise Calibration

We prove the listwise-to-pairwise calibration lemma. Fix q, $C_t(q)$, a positive anchor d^+, and any $d^- \in C_t(q)$. Let $z = (s_\theta(q, d^-) - s_\theta(q, d^+))/\tau$. Since the InfoNCE summation contains the term for d^-,

$$\mathcal{L}_{\text{NCE}}^{(1)}(\theta; q, C_t(q), d^+) \geq \log(1 + e^z). \tag{31}$$

The normalized softplus upper-bounds the threshold indicator:

$$\mathbf{1}[z \geq 0] \leq \frac{\log(1 + e^z)}{\log 2}, \tag{32}$$

because the right-hand side is monotone, nonnegative, and equals 1 at $z = 0$. Therefore

$$\mathbf{1}[s_\theta(q, d^-) \geq s_\theta(q, d^+)] \leq \frac{1}{\log 2} \mathcal{L}^{(1)}_{\mathrm{NCE}}(\theta; q, C_t(q), d^+). \tag{33}$$

Averaging over the sampling distribution of $(q, d^+, C_t(q), d^-)$ gives the stated calibration.

B.7 Weighted-Flip Slack

Assume bounded logits $|s_\theta(q, d)| \leq B$ and list size $|C_t(q)| \leq K$. For any positive anchor a,

$$\mathcal{L}^{(1)}(\theta; q, C_t(q), a) \leq \log(1+Ke^{2B/\tau}) \triangleq \ell_{\max}(B, K, \tau). \tag{34}$$

Let $A_t(q)$ be the accepted flip set, split as $A_t(q) = A_t^+(q) \dot\cup A_t^-(q)$, where $A_t^+(q)$ are true flips and $A_t^-(q)$ are false flips. Let $m_t = \mathbb{E}_q[|A_t(q)|]$ and $\sigma_t = \Pr[g(q, a) = -1 \mid a \in A_t(q)]$.

Relative to a clean objective that gives true flips weight 1 and false flips weight 0, weighted flips create two distortions. False flips add positive-anchor loss:

$$G_+(w) \leq w_{\text{flip}} \ell_{\max} m_t \sigma_t. \tag{35}$$

True flips are also underweighted:

$$G_-(w) \leq (1 - w_{\text{flip}}) \ell_{\max} m_t (1 - \sigma_t). \tag{36}$$

Using the listwise-to-pairwise calibration, one valid pairwise-risk slack bound is

$$\zeta_t \leq \frac{\ell_{\max} m_t}{\log 2} \Big[w_{\text{flip}} \sigma_t + (1 - w_{\text{flip}})(1 - \sigma_t) \Big]. \tag{37}$$

The two one-sided contributions in Eqs. (35)–(36) are balanced when $w_{\text{flip}} \sigma_t = (1 - w_{\text{flip}})(1 - \sigma_t)$, which gives

$$w_{\text{flip}} = 1 - \sigma_t, \tag{38}$$

matching the squared-error optimum in Appendix B.3. A conservative cap $w_{\text{flip}} < 1 - \sigma_t$ reduces the false-flip term while increasing true-signal underuse; the theorem only requires that the resulting slack is bounded by $\bar\zeta$.

B.8 Local Geometric Convergence

We prove Theorem 1. Assume that for all $t \geq T$,

$$\rho_{t+1} \leq r\rho_t + \delta_t, \qquad 0 \leq r < 1. \tag{39}$$

Unrolling gives

$$\begin{aligned} \rho_t &\leq r^{t-T} \rho_T + \sum_{i=0}^{t-T-1} r^i \delta_{t-1-i} \\ &\leq r^{t-T} \rho_T + \frac{\bar\delta}{1 - r}(1 - r^{t-T}), \end{aligned} \tag{40}$$

where $\bar\delta = \sup_{i \geq T} \delta_i$.

Let $I_t = \mathbf{1}[s_{\theta_t}(q, d^-) \geq s_{\theta_t}(q, d^+)]$. By definition of η, $\mathbb{E}[I_t \mid g(d^-) = -1] \leq \eta$. If d^- is a hidden positive, the worst-case contribution to this pairwise event is at most 1. Adding the weighted-flip slack ζ_t,

$$\begin{aligned} X_t &= \mathbb{E}[I_t] \\ &\leq \rho_t \cdot 1 + (1 - \rho_t)\eta + \zeta_t \\ &= \eta + (1 - \eta)\rho_t + \zeta_t. \end{aligned} \tag{41}$$

Substituting Eq. (40) into Eq. (41) and using $\bar\zeta = \sup_{i \geq T} \zeta_i$ proves the risk bound in Theorem 1. Since $0 \leq r < 1$, taking $\limsup_{t \to \infty}$ gives the stated limiting risk floor.

B.9 Bayes-Consistency Corollary

Assume the local recursion holds after some T with $r < 1$. If $\delta_t \to 0$, then for any $\epsilon > 0$ there exists M such that $\delta_j \leq \epsilon$ for all $j \geq M$. Splitting the unrolled sum into a finite early tail and a late tail gives

$$\limsup_{t \to \infty} \rho_t \leq \frac{\epsilon}{1 - r}. \tag{42}$$

Because ϵ is arbitrary, $\rho_t \to 0$. From Eq. (41), $X_t \leq \eta + (1 - \eta)\rho_t + \zeta_t$. Therefore, if also $\zeta_t \to 0$, then $X_t \to \eta$. If the residual pairwise error vanishes as well, $\eta \to 0$, then $X_t \to 0$. This is the limited Bayes-consistency statement for FAT on the evolving mined-pool distribution.

B.10 Failure Modes Implied by the Bound

The proof identifies three failure modes. First, if $\alpha + \gamma \geq 1$, then the static judge map does not strictly contract hidden positives among retained negatives. At equality $f(\rho) = \rho$; when $\alpha + \gamma > 1$, the map can expand hidden positive mass.

Second, even when $\alpha + \gamma < 1$, a large over-flip rate can make the judge-YES stratum too contaminated. This is captured by σ_t in Eq. (13); the precision gate and $w_{\text{flip}} \leq 1 - \sigma_t$ directly control this failure mode.

Third, if δ_t is large or non-vanishing, the floor $(1 - \eta)\bar\delta/(1 - r)$ dominates the risk. This corresponds to a moving mined pool that injects new hidden positives faster than the judge removes them. Finally, if weighted flips are too contaminated or overweighted, the slack ζ_t grows through Eq. (37). These are the quantities controlled by confidence gating, micro-flip budgets, conservative flip weights, and refresh scheduling.

C Proofs for Section 3.6

C.1 Bias Under a Fixed Adversary

Let $\phi_t \to \phi^\star$ and let

$$\theta^\star = \arg\min_\theta L_{\text{distill}}(\theta; \phi^\star).$$

Define $\tau = \lambda_{\text{adv}}/\lambda_{\text{distill}}$ and

$$\begin{aligned} F(\theta, \tau) = \nabla_\theta L_{\text{distill}}(\theta; \phi^\star) \\ + \tau \nabla_\theta L_{\text{adv}}(\theta; \phi^\star). \end{aligned} \tag{43}$$

At $\tau = 0$, $F(\theta^\star, 0) = 0$. Assume $L_{\text{distill}}(\cdot; \phi^\star)$ is twice continuously differentiable and locally strongly convex at $\theta^\star$, with

$$H_{\text{distill}} = \nabla^2_\theta L_{\text{distill}}(\theta^\star; \phi^\star) \succ 0.$$

The implicit function theorem gives a differentiable curve $\theta(\tau)$ satisfying $F(\theta(\tau), \tau) = 0$ and $\theta(0) = \theta^\star$. Differentiating at $\tau = 0$ yields

$$H_{\text{distill}} \theta'(0) + g_{\text{adv}} = 0, \qquad g_{\text{adv}} = \nabla_\theta L_{\text{adv}}(\theta^\star; \phi^\star). \tag{44}$$

Thus $\theta'(0) = -H_{\text{distill}}^{-1} g_{\text{adv}}$, and the stationary point of the fixed-adversary objective satisfies

$$\hat{\theta} = \theta^\star - \tau H_{\text{distill}}^{-1} g_{\text{adv}} + O(\tau^2). \tag{45}$$

Any fixed nonzero adversarial weight therefore induces asymptotic displacement unless $g_{\text{adv}} = 0$.

C.2 Variance Inflation

Let $\widehat{g}_{\text{distill}}$ and $\widehat{g}_{\text{adv}}$ be unbiased stochastic gradients with covariance matrices Σ_{distill} and Σ_{adv}, and cross-covariance $\Sigma_\times$. The combined stochastic gradient is

$$\widehat{g} = \lambda_{\text{distill}} \widehat{g}_{\text{distill}} + \lambda_{\text{adv}} \widehat{g}_{\text{adv}}. \tag{46}$$

Therefore

$$\begin{aligned} \operatorname{Var}[\widehat{g}] = \lambda_{\text{distill}}^2 \Sigma_{\text{distill}} + \lambda_{\text{adv}}^2 \Sigma_{\text{adv}} \\ + \lambda_{\text{distill}} \lambda_{\text{adv}} (\Sigma_\times + \Sigma_\times^\top). \end{aligned} \tag{47}$$

Taking traces gives the corresponding trace-variance expression. If the cross-covariance trace is nonnegative and $\Sigma_{\text{adv}} \neq 0$, any fixed $\lambda_{\text{adv}} > 0$ strictly inflates trace variance relative to pure distillation. As ALD drives $\lambda_{\text{adv}}(t) \to 0$, this additional variance vanishes.

C.3 Convergence with ALD

Consider SGD updates

$$\begin{aligned} \theta_{t+1} = \theta_t - \eta_t \Big(\nabla L_{\text{distill}}(\theta_t; \phi_t) \\ + \lambda_{\text{adv}}(t) \nabla L_{\text{adv}}(\theta_t; \phi_t) + \xi_t \Big), \end{aligned} \tag{48}$$

where $\mathbb{E}[\xi_t \mid \mathcal{F}_t] = 0$. Assume standard stochastic approximation conditions: locally Lipschitz gradients, bounded second moments, almost-surely bounded iterates, and step sizes satisfying $\sum_t \eta_t = \infty$ and $\sum_t \eta_t^2 < \infty$.

Write

$$\nabla L_{\text{distill}}(\theta_t; \phi_t) = \nabla L_{\text{distill}}(\theta_t; \phi^\star) + \Delta_t^{(\phi)}. \tag{49}$$

If the teacher freezes after the stability gate, then $\Delta_t^{(\phi)} = 0$ after that point. More generally, assume

$$\sum_t \eta_t \|\Delta_t^{(\phi)}\| < \infty.$$

If $\lambda_{\text{adv}}(t) \to 0$ and

$$\sum_t \eta_t \lambda_{\text{adv}}(t) < \infty,$$

then the adversarial perturbation is summable whenever ∇L_{adv} is bounded along the iterates.

The recursion is therefore a Robbins–Monro scheme for the limiting ODE

$$\dot{\theta} = -\nabla L_{\text{distill}}(\theta; \phi^\star) \tag{50}$$

with square-summable noise and summable bias. Standard stochastic approximation arguments imply almost-sure convergence to the stationary set of $L_{\text{distill}}(\cdot; \phi^\star)$. Under independent sampling of the distillation and adversarial gradients, the additional variance decays as $O(\lambda_{\text{adv}}(t)^2)$; with general cross-covariance, it decays as $O(\lambda_{\text{adv}}(t))$.

C.4 False-Negative Contraction

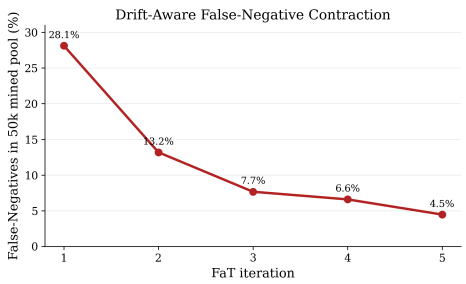

Figure 3: **Empirical contraction diagnostic on NQ.** Discoverable-hidden-positive depletion diagnostic on NQ. At each FAT refresh, we audit newly mined candidates after excluding seed positives and previously accepted flips. The plotted value is the raw judge-YES rate among newly exposed candidates, not the posterior precision of the cumulative accepted-flip pool. Its decline indicates that repeated flip-and-train depletes the finite reservoir of currently recoverable hidden positives in the high-risk mined region