Density-Weighted Minimax Q-Learning with ANN Retrieval: Aligning Retrieval Geometry to Policy-Mass Recall

Anonymous authors
Paper under double-blind review

Abstract

Approximate nearest-neighbor (ANN) retrieval is typically treated as a system-level detail in large-action offline reinforcement learning. This study treats it as a statistical problem: retrieval truncates the support of the target policy, biasing Bellman updates in proportion to the missing policy mass. It formalizes this as *policy-mass recall* (PMR@k), proves a value-error bound linear in missing mass, and introduces a retrieval-projection firewall (B2) that aligns ANN index geometry to the Q-ranking without allowing the alignment loss to perturb the critic. Experiments on MovieLens-25M (59,047 items, 25M ratings) show that three baselines (Naive ANN-DQN, CQL, BCQ) collapse to near zero ANN recall (0.024–0.034, majority of seeds at exactly 0.000), while DW-MQL with the B2 firewall achieves 0.960 ± 0.014 across 10 seeds. A component ablation isolates the mechanism: retaining B2 while removing density correction preserves high recall (0.951 ± 0.038), whereas removing B2 collapses recall to 0.033 and Wolpertinger behavior-cloning retrieval gives 0.025. Thus the listwise-KL projection, not density correction alone, drives the retrieval advantage. A second-dataset experiment on KuaiRand-Pure (7,583 short-video items, binary click feedback) replicates the pattern: DW-MQL achieves 0.885 ± 0.059 while all baselines collapse below 0.003. These experiments report diagnostic retrieval-support metrics rather than recommender benchmark performance.

1 Introduction

Large action catalogues are common in recommendation, control with action codebooks, and retrieval-augmented decision systems. Since scoring every action is expensive, practical systems retrieve a small candidate set and restrict the policy to it. This study treats the approximation as statistical, not only computational: retrieval truncates the target-policy support. If high-value actions lie outside the candidate set, Bellman backups target a different policy and density-ratio objectives weight transitions against a support the learner cannot reach. The bias is governed by the probability mass placed on unretrieved actions, termed *missing policy mass*.

The retrieval–objective alignment problem. A Q-function ranks actions by learned value, but an ANN index ranks them by embedding similarity. These two orderings need not agree. When they disagree, retrieval systematically excludes the critic's highest-value actions, collapsing policy-mass recall (PMR@k) even when the embedding model is otherwise well-trained. This study identifies two failure modes: misaligned embedding geometry, and index-family sensitivity when approximate search does not preserve the high-mass neighborhood.

Approach. The method introduces a *retrieval-projection firewall* (B2): separate projection heads trained on detached encoder outputs by a listwise KL distillation of the Q-ranking onto the embedding space. The ANN index and recall metric operate through these projections, keeping the alignment gradient separated from the critic by construction (stop-gradient + separate clip) so that alignment objectives cannot degrade critic stability.

Contributions.

- This study formalizes ANN-truncated target policies using PMR@k and proves a value-error bound linear in missing mass (Theorem 1).
- The B2 retrieval-projection firewall introduces listwise-KL Q-ranking distillation into ANN keys with stop-gradient isolation from the critic.
- The experiments show that DW-MQL maintains high ANN recall on MovieLens-25M (0.960 ± 0.014) and KuaiRand-Pure (0.885 ± 0.059), while Naive ANN-DQN, CQL, and BCQ collapse under ANN retrieval.
- A component ablation isolates the mechanism: retaining B2 while removing density correction preserves high recall (0.951 ± 0.038), whereas removing B2 reduces recall to 0.033 and Wolpertinger-style behavior-cloning retrieval gives 0.025.

All experimental results characterize retrieval-support behavior, not recommender-system benchmark performance.

2 Related Work

Offline RL must improve from fixed data while avoiding actions unsupported by the logging distribution. BCQ (Fujimoto et al., 2019) and TD3+BC (Fujimoto & Gu, 2021) constrain the learned policy toward behavior, CQL (Kumar et al., 2020) pessimistically penalizes unsupported actions, and IQL (Kostrikov et al., 2022) avoids explicit out-of-distribution maximization. Recent work further studies offline generalization (Mediratta et al., 2024), whether value learning is the main bottleneck (Park et al., 2024), worst-case guarantees under arbitrary data support (Miyaguchi, 2024), implicit-regularization pathologies of deep offline value learning (Gülçehre et al., 2022), and benchmark design beyond proprioceptive state spaces (Lu et al., 2023a). The concern here is complementary: even if the dataset and critic support are adequate, an ANN candidate set can remove target-policy mass before Bellman evaluation or deployment.

Density-ratio and minimax methods reason directly about distribution shift. DualDICE (Nachum et al., 2019), GenDICE (Zhang et al., 2020a), GradientDICE (Zhang et al., 2020b), regularized Lagrangian OPE (Yang et al., 2020), and minimax weight/Q-function learning (Uehara et al., 2020) estimate stationary corrections through saddle-point objectives, with recent variants extending DICE ideas to multi-agent offline RL (Bui et al., 2025). DW-MQL uses density-weighted Bellman updates from this lineage, but PMR measures a different failure mode: loss of target-policy support caused by retrieval truncation. The no-firewall ablation separates this retrieval effect from density correction.

Large discrete-action RL often avoids full action maximization by retrieving or searching over a smaller set. Wolpertinger maps states to proto-actions before nearest-neighbor lookup (Dulac-Arnold et al., 2015); stochastic Q-learning samples sublinear action subsets (Fourati et al., 2024); BraVE targets scalable offline RL in combinatorial action spaces (Landers et al., 2025); and recent ICLR work embeds Q-functions inside combinatorial solvers for restless bandits (Xu et al., 2025). These approaches treat the candidate set mainly as a computational approximation. This study instead treats retrieval as a target-policy support restriction and measures the probability mass preserved by that restriction.

ANN systems such as FAISS (Johnson et al., 2019), HNSW (Malkov & Yashunin, 2020), DiskANN (Subramanya et al., 2019), ScaNN (Guo et al., 2020), and learned search indexes (Lu et al., 2023b) optimize geometric recall, latency, and memory. In recommendation, large systems similarly separate candidate generation from ranking (Covington et al., 2016; Zhai et al., 2024), while RL recommenders study logged-policy correction and slate value decomposition (Chen et al., 2019; Ie et al., 2019). The contribution sits between these threads: the central question is whether the retrieval geometry preserves the critic-induced policy mass. The method introduces a projection interface that aligns ANN keys to the Q-ranking without sending the alignment gradient through the critic.

3 Problem Setting

The setting is batch reinforcement learning with a finite action catalogue. The offline dataset is

$$\mathcal{D} = \{(s_i, a_i, r_i, s'_i, d_i)\}_{i=1}^{n}.$$

The state is s_i. The logged action is $a_i \in \mathcal{A}$. The reward is r_i. The next state is s'_i. The terminal flag is d_i.

The catalogue is

$$\mathcal{A} = \{1, \ldots, N\}.$$

Each action has a feature vector $x_a \in \mathbb{R}^{d_a}$. The learner uses a critic $Q_\theta(s, a)$, a density-ratio model $w_\psi(s, a)$, and a dual function $f_\phi(s, a)$.

Large catalogues make full action scoring expensive. The algorithm therefore uses retrieval. For each query state s, an index returns a candidate set $C_k(s) \subseteq \mathcal{A}$. The set has size at most k. The retrieved set may be exact, stale, or approximate.

The central risk is support loss. A target policy over $\mathcal{A}$ may assign mass to actions that are not present in $C_k(s)$. The truncated learner cannot use those actions in its backup. This changes the effective target policy.

4 Methodology

4.1 ANN-Truncated Target Policies

Let $\pi(a \mid s)$ be a reference target policy over the full action catalogue. In the prototype, this policy is often instantiated as a softmax over full-catalogue or fresh-candidate Q-values:

$$\pi(a \mid s) = \frac{\exp(Q_\theta(s, a)/\tau)}{\sum_{b \in \mathcal{A}} \exp(Q_\theta(s, b)/\tau)}.$$

The retrieved policy is the renormalized restriction of π to $C_k(s)$:

$$\pi_k(a \mid s) = \begin{cases} \pi(a \mid s)/\mathrm{PMR}@k(s), & a \in C_k(s), \\ 0, & a \notin C_k(s). \end{cases}$$

The policy-mass recall is

$$\mathrm{PMR}@k(s) = \sum_{a \in C_k(s)} \pi(a \mid s).$$

The missing policy mass is

$$\epsilon_k(s) = 1 - \mathrm{PMR}@k(s).$$

The experiments also report normalized PMR:

$$\mathrm{nPMR}@k(s) = \frac{\mathrm{PMR}@k(s)}{\max(k/N, \varepsilon)}.$$

This diagnostic compares captured mass to the uniform catalogue baseline.

4.2 Retrieval-Support Audits

The audit compares a retrieved candidate set against a fresh or exact reference set. It records set recall, overlap, value gap, PMR@k, missing mass, and normalized PMR. Set recall only measures identifier overlap. PMR measures target-policy mass. This makes PMR the primary support-loss diagnostic.

Persistent-index experiments also track index age. The index may be refreshed every ρ optimization steps. The index encoder can be the online encoder, a target encoder, or an exponential moving average encoder. These choices are treated as experimental variables.

4.3 Density-Weighted Minimax Q-Learning

The critic uses a density-weighted Bellman residual:

$$\delta_\theta = Q_\theta(s,a) - \left[r + \gamma(1-d)\mathbb{E}_{a'\sim\pi_k(\cdot|s')}Q_{\bar{\theta}}(s',a')\right].$$

The raw Bellman loss is

$$\mathcal{L}_{\text{raw}} = \mathbb{E}_{\mathcal{D}}[\text{stopgrad}(w_\psi(s,a))\delta_\theta^2].$$

The density and dual updates use a stationarity moment. The density update minimizes a squared moment plus regularization. The dual update maximizes the same moment with density weights detached. The implementation uses alternating updates with gradient isolation.

4.4 Retrieval-Aware Bellman Weighting

The aware mode changes the Bellman reduction. It computes a support weight $\omega_{\text{supp}}(s')$ from PMR diagnostics:

$$\mathcal{L}_{\text{weighted}} = \mathbb{E}_{\mathcal{D}}[\text{stopgrad}(w_\psi(s,a))\omega_{\text{supp}}(s')\delta_\theta^2].$$

The current prototype supports three weighting rules:

$$\omega_{\text{supp}} = 1 + \alpha(1 - \text{PMR@}k),$$

$$\omega_{\text{supp}} = \text{clip}\left(\frac{1}{\max(\text{PMR@}k, \varepsilon)}, \omega_{\min}, \omega_{\max}\right),$$

and

$$\omega_{\text{supp}} = 1 + \alpha\max(0, 1 - \text{nPMR@}k).$$

These rules are prototype choices. They are used to test whether support loss should change the critic update. They are not claimed to be optimal.

The critic can also include an additive support penalty:

$$\mathcal{L}_{\text{critic}} = \mathcal{L}_{\text{weighted}} + \lambda_{\text{supp}}\mathbb{E}[1 - \text{PMR@}k].$$

Reports separate raw Bellman loss, support-weighted Bellman loss, support penalty, and critic total loss.

4.5 Retrieval-Projection Alignment

ANN retrieval ranks actions by an embedding similarity, while the critic ranks by Q_θ. When these orders disagree, retrieval truncates away the critic's high-value actions and policy-mass recall collapses. The method aligns the two without letting the alignment objective perturb the critic.

The method adds retrieval-projection heads g_s, g_a on top of *detached* encoder outputs. The retrieval embeddings are

$$\tilde{e}_s = g_s(\text{stopgrad}(e_s)), \qquad \tilde{e}_a = g_a(\text{stopgrad}(e_a)),$$

and the ANN index is built from $\tilde{e}_a$ and queried with $\tilde{e}_s$. Because the projections read stop-gradient inputs, no alignment gradient reaches the encoders or Q_θ; the projections are the only parameters trained by the alignment loss, and they are clipped separately from the critic so a shared gradient-norm clip cannot couple the two.

The projections are trained by a listwise distillation of the Q-ranking onto a candidate set $\mathcal{C}$ (the logged positive plus random negatives):

$$\pi_Q(a) = \text{softmax}_{a\in\mathcal{C}}\big(Q_\theta(s,a)/T_Q\big), \quad \pi_R(a) = \text{softmax}_{a\in\mathcal{C}}\big(\text{sim}(\tilde{e}_s, \tilde{e}_a)/T_R\big),$$

$$\mathcal{L}_{\text{align}} = \text{KL}\big(\pi_Q \,\|\, \pi_R\big),$$

with Q_θ detached (teacher). The similarity sim matches the index metric. For inner-product indexes it is the raw dot product; for cosine indexes the embeddings are L_2-normalized first, so the learned ranking is norm-invariant. This choice is decisive for approximate indexes: a dot-trained alignment can encode ranking partly in embedding norm, which an IVF quantizer (clustering by magnitude) scatters across unsearched cells.

When projections are present, the index-facing representation uses the projected embeddings. In the reported experiments, training uses an exact inner-product index unless otherwise stated; approximate-index behavior is audited post-hoc on the same projected keys. Training with the approximate index throughout would close the remaining train/inference retrieval gap. Projections are optional; with them disabled the method reduces to the base shared-encoder retrieval.

4.6 DW-MQL Training Algorithm

The objectives alternate on a fixed (dual, density, critic) schedule. Each critic step forms a retrieved candidate set, evaluates the density-weighted Bellman loss, and trains the B2 projections with a detached listwise-KL teacher from Q_θ. The firewall is structural: g_s, g_a read only stop-gradient encoder outputs and are gradient-clipped separately from the critic. Full pseudocode is given in Appendix A.6.

4.7 Ablation Modes

The synthetic protocol defines four modes. The `aware` mode uses support-aware Bellman weighting and optional support penalties. The `naive` mode uses the same retrieved sets but no PMR correction in the objective. The `uniform_weights` mode forces Bellman weights to one. The `no_density` mode disables the density path where possible.

These modes make the empirical claim falsifiable. If PMR support loss matters, the aware mode should differ from the naive mode under matched candidate budget. If it does not differ, the support correction is not yet useful.

5 Theory

This section proves a support-loss bound: the value error introduced by ANN truncation is controlled by the missing policy mass. All statements are for policy evaluation of a fixed target policy; assumptions are stated explicitly.

Definition 1 (Policy-mass recall)**.** *For a state s and retrieved set $C_k(s)$, define*

$$\mathrm{PMR}@k(s) = \sum_{a \in C_k(s)} \pi(a \mid s), \qquad \epsilon_k(s) = 1 - \mathrm{PMR}@k(s).$$

Proposition 1 (Policy distance from missing mass)**.** *Assume $\mathrm{PMR}@k(s) > 0$ and let $\pi_k(\cdot \mid s)$ be the normalized restriction of π to $C_k(s)$. Then*

$$\|\pi(\cdot \mid s) - \pi_k(\cdot \mid s)\|_1 \le 2\,\epsilon_k(s).$$

Proof. Write $p = \mathrm{PMR}@k(s) = 1 - \epsilon_k(s)$. For $a \in C_k(s)$, $\pi_k(a \mid s) = \pi(a \mid s)/p \ge \pi(a \mid s)$, and for $a \notin C_k(s)$, $\pi_k(a \mid s) = 0$. Hence

$$\|\pi - \pi_k\|_1 = \sum_{a \in C_k} \Big(\tfrac{\pi(a)}{p} - \pi(a)\Big) + \sum_{a \notin C_k} \pi(a) = \Big(\tfrac{1}{p} - 1\Big)p + \epsilon_k = 2\epsilon_k(s). \qquad \square$$

Proposition 2 (Bellman target bias)**.** *Assume $|Q(s,a)| \le Q_{\max}$. Let T_π and T_{π_k} be the policy-evaluation Bellman operators under π and the retrieved policy π_k,*

$$(T_\mu Q)(s,a) = r(s,a) + \gamma\, \mathbb{E}_{s'|s,a}\, \mathbb{E}_{a' \sim \mu(\cdot|s')} Q(s',a').$$

Then for all (s,a),

$$|(T_\pi Q)(s,a) - (T_{\pi_k} Q)(s,a)| \le 2\gamma Q_{\max}\, \mathbb{E}_{s'|s,a}\, \epsilon_k(s').$$

Proof. The reward and discount terms cancel, so the difference equals $\gamma\,\mathbb{E}_{s'}\big[\sum_{a'}(\pi(a' \mid s') - \pi_k(a' \mid s'))Q(s',a')\big]$. Bounding $|Q| \le Q_{\max}$ and applying Proposition 1, $|\sum_{a'}(\pi - \pi_k)Q(s',\cdot)| \le Q_{\max}\|\pi - \pi_k\|_1 \le 2Q_{\max}\epsilon_k(s')$. Taking expectation over s' and the absolute value inside gives the claim. □

Theorem 1 (Support-loss value bound)**.** *Let Q_π and Q_{π_k} be the fixed points of T_π and T_{π_k} (the true and retrieved policy-evaluation values), and let $\bar{\epsilon}_k = \sup_{s'} \epsilon_k(s')$. Assume rewards are bounded, $|r| \le R_{\max}$, so that both fixed points satisfy $\|Q\|_\infty \le Q_{\max} := R_{\max}/(1-\gamma)$. Then for $\gamma \in [0,1)$,*

$$\|Q_\pi - Q_{\pi_k}\|_\infty \;\le\; \frac{2\gamma Q_{\max}}{1-\gamma}\,\bar{\epsilon}_k.$$

Proof. T_π is a γ-contraction in $\|\cdot\|_\infty$ with fixed point Q_π. Using $Q_\pi = T_\pi Q_\pi$ and $Q_{\pi_k} = T_{\pi_k} Q_{\pi_k}$,

$$\begin{aligned}
\|Q_\pi - Q_{\pi_k}\|_\infty &= \|T_\pi Q_\pi - T_{\pi_k} Q_{\pi_k}\|_\infty \\
&\le \|T_\pi Q_\pi - T_\pi Q_{\pi_k}\|_\infty + \|T_\pi Q_{\pi_k} - T_{\pi_k} Q_{\pi_k}\|_\infty \\
&\le \gamma \|Q_\pi - Q_{\pi_k}\|_\infty + \sup_{s,a} |(T_\pi Q_{\pi_k})(s,a) - (T_{\pi_k} Q_{\pi_k})(s,a)|.
\end{aligned}$$

By Proposition 2 the last term is at most $2\gamma Q_{\max}\bar{\epsilon}_k$. Rearranging, $(1-\gamma)\|Q_\pi - Q_{\pi_k}\|_\infty \le 2\gamma Q_{\max}\bar{\epsilon}_k$. □

Remark 1 (Interpretation and design consequence)**.** *Theorem 1 bounds the retrieved-policy value error linearly in the worst-case missing mass $\bar{\epsilon}_k$, amplified by the horizon factor $1/(1-\gamma)$. Two design consequences follow. First, ANN recall should be measured in policy mass (PMR), not identifier overlap, since the bound depends on ϵ_k. Second, any mechanism that reduces ϵ_k – aligning the retrieval geometry to the Q-ranking, or choosing an index family that preserves the high-mass neighborhood – directly tightens the value bound. This motivates the retrieval-projection alignment of Section 4.5 and the support-weighted Bellman losses, and it is consistent with the empirical finding that index family and alignment space jointly determine retrieved recall.*

Scope. Theorem 1 is a policy-evaluation bound under a fixed target policy and exact operators; it isolates the ANN support-loss term. A full offline-RL bound would additionally carry the usual offline estimation error and optimization error (the decomposition $\mathrm{Err} \le \mathrm{Err}_{\mathrm{offline}} + \mathrm{Err}_{\mathrm{ANN}} + \mathrm{Err}_{\mathrm{approx}}$), with the ANN term governed by the bound above. Tightness is not claimed.

Density-ratio caveat. Density-ratio identifiability may fail when retrieval excludes the logged action: a naive objective can place the observed transition outside the target support. Future theory should study behavior-mixture support, support-restricted density ratios, and PMR bias-correction terms.

6 Experimental Protocol

All experiments are diagnostic retrieval-support studies rather than recommender-system benchmark comparisons. Results are reported as descriptive mean ± std over seeds unless noted otherwise.

Datasets and training. Phase 1 uses MovieLens-latest-small (100,836 ratings, 9,724 movies) for alignment/index diagnostics. Phases 2–3 use MovieLens-25M (25,000,095 ratings, 59,047 movies) with chronological splits, 3-dimensional user-affinity states, and 20-dimensional genre action features. Phase 4 uses KuaiRand-Pure (Gao et al., 2022) (1.44M logs, 7,583 videos, binary click reward) with 5-dimensional state/action features. All models use Adam with $\gamma = 0.99$, batch 256, grad-norm clip 5.0, target update $\tau = 0.01$, and exact inner-product training indexes unless stated otherwise.

Baselines and ablations. Naive ANN-DQN shares the architecture but removes density and retrieval alignment losses. CQL (Kumar et al., 2020) adds a sampled logsumexp conservative penalty; BCQ (Fujimoto et al., 2019) ranks the logged action above a random OOD action. DW-MQL (no-firewall) keeps density correction but zeros the B2 alignment losses. B2-only keeps the retrieval-projection firewall and listwise-KL alignment but removes density correction. The Wolpertinger-style baseline (Dulac-Arnold et al., 2015) replaces the query key with a behavior-cloned proto-action head. Appendix A.4 contains implementation details.

Table 1: MovieLens-25M diagnostic retrieval-support results (59,047 items, 10 seeds, mean ± std). "zeros" counts seeds with ANN recall@20 = 0.

Method	PMR@20	ANN recall@20	ANN PMR@20	zeros
DW-MQL	0.000988±0.000132	**0.960±0.014**	**0.000953±0.000130**	0/10
B2-only (no density)	0.001005±0.000176	0.951±0.038	0.000960±0.000165	0/10
Naive ANN-DQN	0.000796±0.000223	0.032±0.053	0.000033±0.000060	6/10
CQL	**0.002814±0.001336**	0.034±0.064	0.000092±0.000165	7/10
BCQ	0.001443±0.000312	0.024±0.047	0.000040±0.000080	7/10
DW-MQL no-firewall	0.000781±0.000208	0.033±0.054	0.000031±0.000055	4/10
Wolpertinger-style	0.000889±0.000230	0.025±0.078	0.000025±0.000077	8/10

Table 2: KuaiRand-Pure diagnostic retrieval-support results (7,583 items, 10 seeds, mean ± std). "zeros" counts seeds with ANN recall@20 = 0.

Method	PMR@20	ANN recall@20	ANN PMR@20	zeros
DW-MQL	0.002832±0.000056	**0.885±0.059**	**0.002508±0.000184**	0/10
Naive ANN-DQN	0.002798±0.000034	0.002±0.007	0.000006±0.000021	9/10
CQL	0.002822±0.000031	0.002±0.005	0.000004±0.000014	9/10
BCQ	0.002790±0.000022	0.003±0.009	0.000008±0.000024	9/10

Metrics. PMR@k is the softmax target-policy mass captured by the retrieved top-k set. ANN recall@k is overlap between retrieved top-k and full-catalogue exact-Q top-k; it is an embedding-alignment diagnostic, not a generic ANN systems metric. ANN PMR@k is PMR measured through the deployed retrieval index. Bellman loss reports support-weighted critic residual.

7 Results

All results use retrieval-support metrics (PMR@20, ANN recall@20, and Bellman loss), not recommender benchmark metrics. Phase 1 and Phase 2 audit results appear in Appendix A.1–A.2; they justify dot+HNSW as the approximate index family by showing exact-recall recovery on MovieLens-latest-small and 99.99% recovery at 59,047 items.

7.1 MovieLens-25M: Baselines and B2 Ablation

MovieLens-25M uses 5,000 training steps, 10 seeds, $d = 64$, and the full 59,047-item catalogue. Table 1 shows two facts. First, DW-MQL and its B2-only variant are the only methods with reliable ANN recall (0.960 ± 0.014 and 0.951 ± 0.038); Naive ANN-DQN, CQL, and BCQ collapse to 0.024–0.034, despite CQL achieving the highest exact-index PMR. Thus a better exact-index Q-ranking does not by itself imply ANN deployability. Second, the ablations isolate the mechanism: keeping B2 while removing density correction preserves high recall, while keeping density correction but removing B2 reduces recall to 0.033 and Wolpertinger-style behavior-cloning retrieval gives 0.025. The no-firewall and Wolpertinger ablations overlap the Naive baseline within one standard deviation, while B2-equipped methods are roughly 29×–38× higher in mean recall.

7.2 KuaiRand-Pure: Second-Dataset Check

Table 2 repeats the pattern on KuaiRand-Pure (7,583 short-video items, binary click feedback). DW-MQL obtains 0.885 ± 0.059 ANN recall, with every seed above 0.80, while all three baselines collapse to 0.002–0.003 and 9/10 zero-recall seeds. Exact-index PMR is nearly identical across methods, so the gap again lies in whether the embedding geometry lets ANN retrieval navigate to the critic's preferred actions.

8 Limitations

All experiments are recommendation-domain diagnostics; extension to robotics, dialogue, healthcare, or other offline-RL action structures remains open. Training is also unstable at 5,000 steps: every method saturates the grad-norm clip on MovieLens-25M, limiting loss-based comparisons and motivating learning-rate schedules or early stopping.

Phase 3 trains and evaluates with an exact inner-product index; approximate behavior is audited post-hoc on the same trained projected keys. Training with HNSW throughout would close this train/inference retrieval gap. Phase 2 also audits approximate indexes on one seed only, so larger multi-seed audits and additional index families are future work.

Finally, Theorem 1 is a fixed-policy evaluation bound that isolates ANN support loss; a full offline-RL bound would add estimation and optimization error. PMR@20, ANN recall@20, and Bellman loss are internal method diagnostics, not user-satisfaction or recommender benchmark metrics.

9 Conclusion

ANN retrieval is not merely a systems approximation in large-action offline RL: it truncates the target-policy support. PMR@k measures the retained policy mass, and Theorem 1 shows that missing mass induces a value error proportional to the horizon factor. The B2 retrieval-projection firewall uses listwise-KL Q-ranking distillation to align ANN keys while keeping the alignment gradient outside the critic.

Across MovieLens-25M and KuaiRand-Pure, DW-MQL maintains high ANN recall while Naive ANN-DQN, CQL, and BCQ collapse under retrieval. B2-only retains high recall after density correction is removed, whereas the no-firewall and Wolpertinger-style ablations overlap the Naive baseline within one standard deviation. This identifies the listwise-KL projection as the primary driver of the observed retrieval improvement. The results are diagnostic rather than recommender benchmark claims, but they support a practical lesson: offline RL policies intended for ANN deployment should align the retrieval geometry to the critic's ordering, not only optimize the critic itself.

References

The Viet Bui, Thanh Hong Nguyen, and Tien Anh Mai. ComaDICE: Offline cooperative multi-agent reinforcement learning with stationary distribution shift regularization. In *The Thirteenth International Conference on Learning Representations*, 2025. URL `https://openreview.net/forum?id=5o9JJJPPm6`.

Minmin Chen, Alex Beutel, Paul Covington, Sagar Jain, Francois Belletti, and Ed H. Chi. Top-K off-policy correction for a REINFORCE recommender system. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, pp. 456–464, 2019.

Paul Covington, Jay Adams, and Emre Sargin. Deep neural networks for YouTube recommendations. In *Proceedings of the 10th ACM Conference on Recommender Systems*, pp. 191–198, 2016.

Gabriel Dulac-Arnold, Richard Evans, Hado van Hasselt, Peter Sunehag, Timothy Lillicrap, Jonathan Hunt, Timothy Mann, Theophane Weber, Thomas Degris, and Ben Coppin. Deep reinforcement learning in large discrete action spaces. *arXiv preprint arXiv:1512.07679*, 2015.

Fares Fourati, Vaneet Aggarwal, and Mohamed-Slim Alouini. Stochastic Q-learning for large discrete action spaces. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 13734–13759, 2024. URL `https://proceedings.mlr.press/v235/fourati24a.html`.

Scott Fujimoto and Shixiang Shane Gu. A minimalist approach to offline reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 34, pp. 20132–20145, 2021.

Scott Fujimoto, David Meger, and Doina Precup. Off-policy deep reinforcement learning without exploration. In *Proceedings of the 36th International Conference on Machine Learning*, pp. 2052–2062, 2019. URL `https://proceedings.mlr.press/v97/fujimoto19a.html`.

Chongming Gao, Shijun Li, Wenqiang Lei, Jiawei Chen, Biao Li, Peng Jiang, Xiangnan He, Jiaxin Mao, and Tat-Seng Chua. KuaiRand: An unbiased sequential recommendation dataset with randomly exposed videos. In *Proceedings of the 31st ACM International Conference on Information and Knowledge Management*, pp. 3953–3957. ACM, 2022. doi: 10.1145/3511808.3557624.

Çağlar Gülçehre, Srivatsan Srinivasan, Jakub Sygnowski, Georg Ostrovski, Mehrdad Farajtabar, Matthew Hoffman, Razvan Pascanu, and Arnaud Doucet. An empirical study of implicit regularization in deep offline RL. *Transactions on Machine Learning Research*, 2022. URL https://openreview.net/forum?id=HFfJWx60IT.

Ruiqi Guo, Philip Sun, Erik Lindgren, Quan Geng, David Simcha, Felix Chern, and Sanjiv Kumar. Accelerating large-scale inference with anisotropic vector quantization. In *Proceedings of the 37th International Conference on Machine Learning*, pp. 3887–3896, 2020. URL https://proceedings.mlr.press/v119/guo20h.html.

Eugene Ie, Vihan Jain, Jing Wang, Sanmit Narvekar, Ritesh Agarwal, Rui Wu, Heng-Tze Cheng, Morgane Lustman, and Craig Boutilier. SlateQ: A tractable decomposition for reinforcement learning with recommendation sets. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, pp. 2592–2599, 2019.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data*, 7(3):535–547, 2019.

Ilya Kostrikov, Ashvin Nair, and Sergey Levine. Offline reinforcement learning with implicit Q-learning. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=68n2s9ZJWF8.

Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. Conservative Q-learning for offline reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 33, pp. 1179–1191, 2020.

Matthew Landers, Taylor W. Killian, Hugo Barnes, Thomas Hartvigsen, and Afsaneh Doryab. BraVE: Offline reinforcement learning for discrete combinatorial action spaces. In *Advances in Neural Information Processing Systems*, 2025. URL https://arxiv.org/abs/2410.21151.

Cong Lu, Philip J. Ball, Tim G. J. Rudner, Jack Parker-Holder, Michael A. Osborne, and Yee Whye Teh. Challenges and opportunities in offline reinforcement learning from visual observations. *Transactions on Machine Learning Research*, 2023a. URL https://openreview.net/forum?id=1QqIfGZOWu.

Zepu Lu, Jin Chen, Defu Lian, Zaixi Zhang, Yong Ge, and Enhong Chen. Knowledge distillation for high dimensional search index. In *Advances in Neural Information Processing Systems*, volume 36, 2023b. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/6a15378acabd1aef017ec79a3ed744d2-Abstract-Conference.html.

Yu. A. Malkov and D. A. Yashunin. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(4):824–836, 2020.

Ishita Mediratta, Qingfei You, Minqi Jiang, and Roberta Raileanu. The generalization gap in offline reinforcement learning. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=3w6xuXDOdY.

Kohei Miyaguchi. Worst-case offline reinforcement learning with arbitrary data support. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=63VajkIDEu.

Ofir Nachum, Yinlam Chow, Bo Dai, and Lihong Li. DualDICE: Behavior-agnostic estimation of discounted stationary distribution corrections. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Seohong Park, Kevin Frans, Sergey Levine, and Aviral Kumar. Is value learning really the main bottleneck in offline RL? In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=nyp59a31Ju.

Suhas Jayaram Subramanya, Fnu Devvrit, Harsha Vardhan Simhadri, Ravishankar Krishnaswamy, and Rohan Kadekodi. DiskANN: Fast accurate billion-point nearest neighbor search on a single node. In *Advances in Neural Information Processing Systems*, 2019.

Masatoshi Uehara, Jiawei Huang, and Nan Jiang. Minimax weight and Q-function learning for off-policy evaluation. In *Proceedings of the 37th International Conference on Machine Learning*, pp. 9659–9668, 2020. URL https://proceedings.mlr.press/v119/uehara20a.html.

Lily Xu, Bryan Wilder, Elias Boutros Khalil, and Milind Tambe. Reinforcement learning with combinatorial actions for coupled restless bandits. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=DhH3LbA6F6.

Mengjiao Yang, Ofir Nachum, Bo Dai, Lihong Li, and Dale Schuurmans. Off-policy evaluation via the regularized Lagrangian. In *Advances in Neural Information Processing Systems*, volume 33, pp. 6551–6561, 2020.

Jiaqi Zhai, Lucy Liao, Xing Liu, Yueming Wang, Rui Li, Xuan Cao, Leon Gao, Zhaojie Gong, Fangda Gu, Jiayuan He, Yinghai Lu, and Yu Shi. Actions speak louder than words: Trillion-parameter sequential transducers for generative recommendations. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 58484–58509, 2024. URL https://proceedings.mlr.press/v235/zhai24a.html.

Ruiyi Zhang, Bo Dai, Lihong Li, and Dale Schuurmans. GenDICE: Generalized offline estimation of stationary values. In *International Conference on Learning Representations*, 2020a. URL https://openreview.net/forum?id=HkxlcnVFwB.

Shangtong Zhang, Bo Liu, and Shimon Whiteson. GradientDICE: Rethinking generalized offline estimation of stationary values. In *Proceedings of the 37th International Conference on Machine Learning*, pp. 11194–11203, 2020b. URL https://proceedings.mlr.press/v119/zhang20r.html.

A Appendix

A.1 Phase 1: Alignment and Index Ablation (MovieLens-latest-small)

Setup. MovieLens-latest-small chronological-strict split (N = 9,724, train 80,670, val/test 10,083 each). Embedding dim $d = 16$, 1,000 steps, 10 seeds. Each trained model is re-evaluated under three index families without retraining. Recall@20 is the overlap between index top-20 and full-catalogue exact-Q top-20.

Table 3: Phase 1 diagnostic audit: alignment × index (MovieLens-latest-small, 10 seeds, mean ± standard error). Recall@20 is exact-Q / embedding top-k overlap. Each index row re-evaluates the same trained model as its exact row. IVF: nprobe = 8 of 98 cells; HNSW: ef = 64. "min" is the worst-seed recall.

Alignment	Index	Recall@20	min-seed	Bellman	PMR@20
Dot	Exact	0.975 ± 0.011	0.901	3.48 ± 0.39	0.00581 ± 0.00052
Dot	IVF (np = 8)	0.553 ± 0.122	0.000	—	—
Dot	**HNSW** (ef = 64)	$\mathbf{0.975 \pm 0.011}$	**0.901**	—	—
Cosine	Exact	0.891 ± 0.017	0.801	3.26 ± 0.38	0.00579 ± 0.00051
Cosine	IVF (np = 8)	0.891 ± 0.017	0.801	—	—
Cosine	HNSW (ef = 64)	0.891 ± 0.017	0.801	—	—

Index family is decisive. Table 3 shows that with dot alignment, IVF recall is 0.553 ± 0.122 and collapses to 0.000 on 2/10 seeds, while HNSW recovers exact recall on every seed (0.975 ± 0.011, worst seed 0.901). The mechanism is norm sensitivity: dot alignment encodes part of the ranking in embedding magnitude; IVF's k-means quantizer clusters by magnitude and places high-value actions in unsearched cells; HNSW navigates a similarity graph with no magnitude-based partition. Re-evaluating the dot model under cosine similarity confirms this: cosine-exact recall drops to ≈ 0.13, showing the ranking lives in the norms.

Two remedies. Cosine alignment (norm-invariant training) makes both IVF and HNSW reliable (0.891 ± 0.017 exact recall, matched by HNSW and matched by IVF to rounding), at the cost of $0.975 \rightarrow 0.891$ exact recall. HNSW keeps the higher dot recall directly. The primary configuration is *dot alignment + HNSW*; cosine + IVF is the fallback for partition-only deployments.

Retrieval choice is isolated from the critic. Bellman loss (3.26 vs. 3.48) and PMR@20 (0.00579 vs. 0.00581) overlap closely across alignment choices, consistent with the B2 firewall's gradient isolation.

A.2 Phase 2: Scale Audit at 59k Items (Gate G1)

Setup. Full MovieLens-25M dataset (25M ratings, 59,047 movies), 5,000 steps, seed 0, embedding dim $d = 64$, CUDA. ANN audit on 5,000 held-out test rows after training. This table is a single-seed systems gate for index-family viability at full catalogue scale; the 10-seed Phase 3 study supplies the multi-seed evidence for retrieval-support performance at the same catalogue size.

Table 4: Phase 2 Gate G1 ANN audit at 59,047-item scale (MovieLens-25M, single seed). Pass criteria: HNSW recall $\geq$ 95% of exact recall; cosine+IVF fallback recall $\geq$ 90% of cosine-exact recall. Recall@20 as in Table 3.

Metric	Index	Setting	Recall@20	% of exact
Dot	Exact	—	0.9099	100%
Dot	HNSW	ef = 64	**0.9098**	**99.99%** ✓
Dot	HNSW	ef = 16	0.9100	100.0%
Cosine	Exact	—	0.5687	100%
Cosine	IVF	np = 8	0.5249	92.3% ✓
Cosine	IVF	np = 2	0.2040	35.9%

Gate G1 passed. Table 4 shows dot+HNSW (ef = 64) matches dot-exact to within 0.01% at 59,047-item scale — a 6× increase in catalogue size from Phase 1. The Phase 1 finding (HNSW = exact for dot embeddings) generalises to the full large-scale setting. Cosine+IVF achieves 92.3% of cosine-exact at nprobe = 8, but collapses to 35.9% at nprobe = 2.

Dot dramatically outperforms cosine on exact recall (0.9099 vs. 0.5687), confirming that the model optimises inner products and the ranking lives in embedding norms. This makes dot+HNSW the only configuration that simultaneously achieves high policy-mass recall and faithful approximate retrieval at this scale.

A.3 Implementation Notes

The implementation contains PMR diagnostics, normalized PMR, missing-mass summaries, density-weighted minimax Q-learning with alternating update schedule, B2 retrieval-projection firewall with stop-gradient enforcement, support-aware Bellman weighting, and diagnostic retrieval-support routes for MovieLens and KuaiRand data shapes. Full hyperparameter settings are given in Appendix A.5. Extensions to additional benchmark protocols are left for future work.

A.4 Experimental Details

MovieLens-latest-small uses a chronological-strict global split with train 80,670, validation 10,083, and test 10,083. MovieLens-25M uses the same split protocol; evaluation is capped at 10,000 rows per split with a fixed seed. KuaiRand-Pure uses the same chronological split protocol, with evaluation capped at 5,000 rows per split.

The critic is a two-tower network with shared-base encoder ($d = 16$ in Phase 1, $d = 64$ in Phases 2–3, and $d = 32$ in Phase 4) and B2 retrieval-projection heads. Phase 1 runs 1,000 steps. Phases 2–4 run 5,000 steps on CUDA. Phase 1 trains dot and cosine alignment variants and evaluates each under exact, IVF (nprobe = 8), and HNSW (ef = 64) indexes. Phase 2 audits dot+HNSW and cosine+IVF at 59,047 items with the pass criteria in Table 4. Phase 3 compares DW-MQL, Naive ANN-DQN, CQL, BCQ, and the component ablations on MovieLens-25M; Phase 4 repeats the comparison on KuaiRand-Pure.

CQL uses 64 random action samples with $\lambda_{\text{CQL}} = 1.0$. BCQ uses a cross-entropy constraint with $\lambda_{\text{BCQ}} = 1.0$. DW-MQL (no-firewall) retains $\lambda_{\text{density}} = 0.01$ and $\lambda_s = 1.0$ but sets the alignment and ranking-margin weights to zero. B2-only sets $\lambda_{\text{density}} = 0$ while retaining the B2 listwise-KL and ranking losses. Wolpertinger trains a same-width proto-action MLP by MSE behavior cloning and queries the ANN index with the proto-action embedding.

A.5 Hyperparameters

Table 5 lists the settings shared by all methods in each phase; Table 6 lists the per-method loss weights. All networks are MLPs with a single hidden layer of width 32 and ReLU activations, trained with AdamW. Within each phase every method uses identical architecture, optimizer, index configuration, and seeds; only the loss weights in Table 6 differ. No per-method hyperparameter tuning was performed.

Table 5: Shared hyperparameters per experimental phase. Phase 2 (Gate G1) reuses the Phase 3 DW-MQL configuration on seed 0 with a post-training ANN audit.

	Phase 1 (ML-latest-small)	Phases 2–3 (ML-25M)	Phase 4 (KuaiRand-Pure)
Catalogue size N	9,724	59,047	7,583
Training steps T	1,000	5,000	5,000
Batch size	8	256	256
Embedding dim d	16	64	32
Hidden layer width	32	32	32
Learning rate (AdamW)	1.5×10^{-4}	10^{-4}	10^{-4}
Discount γ	0.99	0.99	0.99
Candidate budget k	20	20	20
Target soft-update τ	0.01	0.01	0.01
Target update period	1	1	1
Gradient-norm clip	5.0	5.0	5.0
Q-head L_2 weight	0.01	0.01	0.01
Q-value clip	5.0	5.0	5.0
Index metric (training)	dot	dot	dot
Index backend (training)	exact	exact	exact
Index refresh period	1 step	1 step	1 step
Alignment warmup steps	200	500	500
Eval rows per split	full	10,000	5,000
Positive-rating threshold	≥ 4.0	≥ 4.0	click = 1
Seeds	10	10	10

All methods that include the density path use stationarity weight $\lambda_s = 1.0$ and Bellman weight 1.0, with the three-step alternating schedule of Algorithm 1. Reward scale is 1.0 throughout.

Table 6: Per-method loss weights. Shared B2 settings where enabled: ranking margin 0.5 with 20 negative samples; listwise-KL teacher temperature $T_Q = 0.2$ and retrieval temperature $T_R = 0.07$; alignment margin 1.0. CQL uses 64 sampled actions for the logsumexp penalty. Wolpertinger-style adds a behavior-cloning MSE loss on the proto-action head with weight 1.0.

Method	λ_{density}	λ_{align}	λ_{margin}	Other
DW-MQL (full)	0.01	0.1	0.1	—
B2-only	0	0.1	0.1	—
DW-MQL no-firewall	0.01	0	0	—
Naive ANN-DQN	0	0	0	—
CQL	0	0	0	$\lambda_{\text{CQL}} = 1.0$
BCQ	0	0	0	$\lambda_{\text{BCQ}} = 1.0$
Wolpertinger-style	0	0	0	$\lambda_{\text{BC}} = 1.0$

A.6 DW-MQL Pseudocode

Algorithm 1 DW-MQL with B2 Retrieval-Projection Firewall

Require: Offline dataset $\mathcal{D}$, action catalogue $\mathcal{A}$, discount γ, candidate budget k, step count T

Initialize encoders e_s, e_a, critic Q_θ, density w_η, dual f_ϕ, projections g_s, g_a, target $\bar{\theta} \leftarrow \theta$
for $t = 1$ **to** T **do**
  Build index $\mathcal{I}$ from $\{g_a(\text{sg}(e_a(a))) : a \in \mathcal{A}\}$
  Sample batch $B = \{(s, a, r, s', d)\} \sim \mathcal{D}$ and retrieve $C_k(s') \leftarrow \mathcal{I}.\text{search}(g_s(\text{sg}(e_s(s'))), k)$
  if $t \bmod 3 = 0$ **then**
    Update f_ϕ by maximizing the stationarity moment
  else if $t \bmod 3 = 1$ **then**
    Update w_η by minimizing the squared moment plus regularization
  else
    Compute $y = r + \gamma(1-d)\mathbb{E}_{a' \sim \pi_k(\cdot|s')} Q_{\bar{\theta}}(s', a')$ and density-weighted Bellman loss $\mathbb{E}[\text{sg}(w_\eta)(Q_\theta - y)^2]$
    Train g_s, g_a with $\text{KL}(\text{sg}(\pi_Q) \| \pi_R)$ on candidate actions
    Clip critic and projection gradients separately; update parameters
  end if
  Soft-update target: $\bar{\theta} \leftarrow (1-\tau)\bar{\theta} + \tau\theta$
end for