Escaping the Euclidean Void: Manifold-Informed Flow Matching for Sequential Recommendation

Dengzhao Fang
Jilin University
Changchun, China
fangdz25@mails.jlu.edu.cn

Jingtong Gao
City University of Hong Kong
Hong Kong, China
jt.g@my.cityu.edu.hk

Yu Li*
Jilin University
Changchun, China
liyu90@jlu.edu.cn

Xiangyu Zhao
City University of Hong Kong
Hong Kong, China
xianzhao@cityu.edu.hk

Yi Chang
Jilin University
Changchun, China
yichang@jlu.edu.cn

arXiv:2607.23762v1 [cs.IR] 26 Jul 2026

Abstract

Conventional recommenders capture users' preferences by optimizing observed user-item relations, whereas continuous generative recommendation additionally learns the trajectory of synthesizing a target item. Flow matching drives this process by gradually shaping initial noise into a definitive next-item representation through intermediate states in a continuous embedding space. However, item catalogs are discrete and sparsely supported, meaning even a straight Euclidean path can cross continuous regions that contain little evidence of valid item semantics. Formalizing this failure as the *Euclidean void*, we propose **MIRAGE**, a **M**anifold-**I**nformed **R**ectification framework for **A**ccelerated **G**eneration of **E**mbeddings in sequential recommendation, which rectifies the learned embedding geometry around an unchanged straight probability path. By leveraging an item co-occurrence graph as a proxy for the underlying semantic manifold, MIRAGE aligns interpolated path states with local anchors, reorganizing the embedding space to ground the trajectory in valid item support. MIRAGE retains the original probability path and uses the graph only during training, thereby enabling accurate and efficient one-step inference. Extensive experiments on four real-world datasets reveal that MIRAGE consistently outperforms state-of-the-art baselines, effectively boosting performance on sparsely observed targets while achieving robust overall accuracy. Our code will be made publicly available upon publication.

CCS Concepts

• **Information systems** → **Recommender systems**.

Keywords

Sequential Recommendation, Generative Models, Flow Matching, Topology Regularization

1 Introduction

Sequential recommendation (SR) predicts a user's next interaction from an ordered behavior history, whose encoding has been progressively strengthened by recurrent, convolutional, Transformer, and graph architectures [15, 17, 24, 44, 45, 49]. Most established objectives remain focused on target discrimination, as SASRec [17] scores candidate items against a sequence representation, LightGCN [14] scores items after propagating relational signals, and ranking or contrastive objectives make the observed target more compatible with the user than competing items. This design is effective because recommendation ultimately depends on the relative scores of these discrete learned points and therefore does not require the continuous region between a user and target to carry meaningful item semantics.

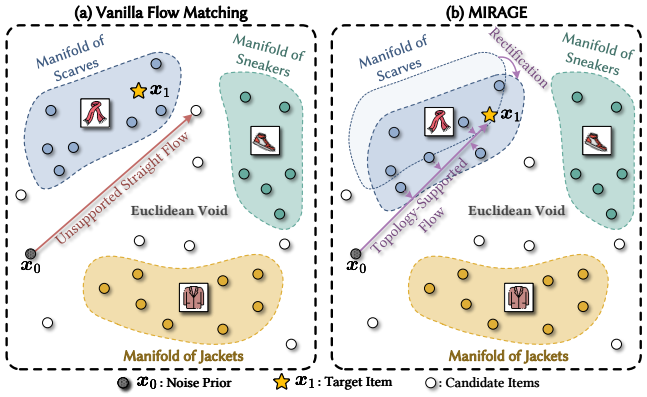

Figure 1: Euclidean void and MIRAGE rectification. (a) Vanilla straight paths traverse unsupported regions. (b) MIRAGE reorganizes item embeddings around the path via topology regularization to promote semantic support.

Continuous generative recommendation changes this requirement because the model must learn the dynamic evolution of user intent rather than merely a target score, with diffusion-based methods perturbing target embeddings and recovering them through iterative denoising [7, 16, 22, 23, 43, 52]. Flow matching instead learns a time-dependent vector field and integrates an ordinary differential equation (ODE) [25]. Since this field is trained at intermediate states and queried along the generated path, its success heavily depends on the geometry encountered between the source noise and the target item. While target discrimination remains necessary for ranking, it is no longer sufficient for learning reliable continuous transport toward the precise target semantics.

Rectified flow improves transport efficiency by learning nearly straight ODE trajectories, which incur less discretization error

*Corresponding author.

and can be integrated with fewer solver steps [30]. Following this straight-trajectory principle, FMRec [27] adapts target-recovery flow matching and ODE sampling to SR, while FAVE [40] further enables one-step transport through a history-informed prior and average-velocity modeling. However, while these methods optimize transport efficiency, they do not guarantee that the resulting straight path traverses meaningful regions of the embedding space. Consequently, a trajectory may be numerically simple yet semantically unsafe, causing the generated representations to drift into empty regions that yield inaccurate or irrelevant recommendations.

The source of this risk lies in the fundamental mismatch between a continuous Euclidean space and a discrete item catalog, where learned embeddings populate the space sparsely and unevenly. This aligns with the broader observation that high-dimensional data typically concentrates near lower-dimensional manifolds [6, 11]. However, a straight interpolation path between a source and a target is not guaranteed to remain on this manifold [2, 8]. Consequently, an intermediate state can drift far from any valid item semantics, falling into what we term the *Euclidean void*.

Unlike ODE discretization errors, the Euclidean void degrades the learning signal. Flow matching requires the neural network to infer the target direction from sampled states along the path. When a state falls into the void, it lacks surrounding geometric evidence, forcing the predictor to rely almost entirely on the user history to recover precise target semantics. Consequently, increasing the number of solver steps cannot resolve this structural failure, as finer numerical integration cannot synthesize missing catalog support. The central challenge is therefore not only to learn a short path, but also to make the space traversed by that path semantically informative. This challenge is particularly critical for long-tail items, whose representations receive limited direct supervision from user histories [19, 34, 48], relying on the local support of semantic neighbors to guide the trajectory away from the void.

Figure 1 makes this distinction concrete. A vanilla flow model projects a straight path toward the target, but crossing the Euclidean void causes the generated state to deviate from the true target. In contrast, our proposed framework retains the straight probability path while reorganizing valid item embeddings around its interior, ensuring the generated trajectory is semantically grounded and safely reaches the target. This contrast captures a shift from target-focused optimization to path-aware representation learning.

To realize this shift, we introduce **MIRAGE**, a **M**anifold-**I**nformed **R**ectification framework for **A**ccelerated **G**eneration of **E**mbeddings in SR. Rather than distorting the straight probability trajectory, MIRAGE rectifies the underlying embedding geometry around a structurally straight path. Since strict continuous manifolds are intrinsically absent in discrete item catalogs, MIRAGE treats an item co-occurrence graph as a discrete proxy for the underlying data manifold, exploiting the strong transition regularities inherent in user histories [4, 49]. Specifically, the framework applies a topology regularizer that directly aligns the interpolated path states with active graph anchors drawn from the target's neighborhood. A time modulation schedule suppresses this regularization at the source noise and the target item, concentrating the spatial restructuring exclusively on the unsupported path interior. By accumulating these geometric adjustments across sampled times and shared items, locally related embeddings are dynamically repositioned to form a cohesive semantic manifold that envelops the generative trajectories. Consequently, MIRAGE prepares a semantically safe transport route, which translates the underlying continuous trajectory into accurate next-item recommendations.

Crucially, this design separates topology supervision from model inference. The co-occurrence graph supplies neighborhood anchors exclusively during training. The generation phase relies entirely on the newly rectified space without any graph lookups or neighborhood traversals. This allows MIRAGE to retain the computational efficiency of one-step transport. Furthermore, our spatial rectification focuses entirely on whether the learned space supports the journey, making it highly complementary to existing improvements in prior distributions and numerical solvers. Extensive experiments on four real datasets confirm that ensuring semantic path support translates to consistent recommendation gains. MIRAGE significantly outperforms the best-performing baselines and demonstrates particular effectiveness for long-tail items, where geometric grounding successfully compensates for sparse direct supervision.

Our contributions are as follows:

- We identify locally supported intermediate states as a critical yet neglected requirement for reliable transport in continuous generative recommendation, formalizing the absence of this semantic grounding as the *Euclidean void*.
- We propose MIRAGE, a novel manifold-informed rectification framework centered on interpolated states, which dynamically reorganizes the embedding space around straight transport paths while preserving graph-free inference.
- Comprehensive evaluations across four datasets validate MIRAGE's superiority in state-of-the-art overall accuracy, robust long-tail gains, and one-step inference efficiency.

2 Related Work

2.1 Generative Sequential Recommendation

Sequential recommendation (SR) models encode user behavior through deep sequence models (e.g., recurrent and self-attentive networks) or contrastive and translation-based objectives [10, 12, 15, 17, 29, 38, 39, 44, 45]. Graph-based variants propagate collaborative signals over session or interaction graphs [14, 47, 49, 54], while tail-enhancement frameworks transfer information to sparsely observed targets [9, 19]. While these methods establish the predictive value of relational structure, they primarily use it to encode static representations for direct target scoring. This optimizes the latent space for endpoint similarity without requiring the continuous regions between users and items to carry meaningful semantics.

Moving beyond this static point estimation, diffusion-based SR approaches next-item prediction by modeling continuous generative trajectories. These methods perturb target representations and recover them through iterative denoising [16, 31, 43, 46, 51, 56]. Recent advancements improve this process by generating ideal target representations [52], modeling multiple interests [22], mitigating data sparsity via diffusion-based augmentation [28, 33], addressing popularity bias and corruption stability [3, 7, 23, 34, 57], or incorporating item co-occurrence graphs to strengthen diffusion guidance [50]. Although successfully introducing generative conditioning, these methods collectively inherit the discrete Markovian

dynamics of diffusion, which require complex multi-step transport processes and significant discretization errors.

2.2 Flow Matching and Geometric Support

To overcome the inefficiencies of discrete diffusion, flow matching (FM) regresses continuous vector fields along prescribed straight probability paths [1, 20, 25, 30, 37, 42]. In SR, FMRec pioneered this transition by adapting target-recovery FM with deterministic Euler sampling [27]. Subsequent methods like FlowRec [21], FAVE [40], FlowCF [26], and GMFlowRec [53] accelerate generation by optimizing priors, velocities, or mixture distributions. Although recent attempts like DAE4Rec [41] incorporate graph guidance for condition enhancement, these methods uniformly assume their continuous paths traverse semantically safe regions, largely overlooking the underlying geometric validity.

The vulnerability of these unsupported continuous paths is rooted in representation geometry. The manifold hypothesis posits that high-dimensional data concentrates near structured, lower-dimensional subsets [6, 11]. Standard Euclidean interpolants often deviate from these curved manifolds, leading to regions devoid of meaningful semantics [2, 8]. While classical manifold regularization promotes smoothness among observed samples via graph structures [5], and Riemannian FM constructs explicit flows on known continuous manifolds [8], generative recommendation lacks an explicit continuous manifold to guide the transport. Building upon this geometric perspective, we propose **MIRAGE** to resolve the Euclidean void. Rather than redesigning the transport velocity or requiring a known continuous manifold, MIRAGE leverages training-only discrete topology to rectify the shared embedding geometry around straight probability paths. This decouples spatial semantic safety from trajectory acceleration, ensuring that generative paths remain reliably grounded in valid representation geometry.

3 Preliminaries

3.1 Sequential Recommendation

Let $\mathcal{U}$ and $\mathcal{I}$ denote the user set and the item catalog, respectively. For a given user $u \in \mathcal{U}$, the ordered history is $S_u = (i_1, \ldots, i_m)$ with $i_\ell \in \mathcal{I}$. Sequential recommendation ranks candidate items as the next interaction i_{m+1}, denoted by i^+. Each item i is associated with a trainable embedding $\boldsymbol{e}_i \in \mathbb{R}^d$.

Generative sequential recommendation models depart from direct discrimination over the discrete item catalog, modeling a transport process in a continuous latent space to generate the target representation. Given a tractable source distribution p_0 (e.g., standard Gaussian) and a target distribution p_1 over next-item embeddings, the model learns to transport a source sample $\boldsymbol{x}_0 \sim p_0$ toward a target sample $\boldsymbol{x}_1 = \boldsymbol{e}_{i^+} \sim p_1$. Diffusion-based recommenders instantiate this paradigm through iterative denoising [23, 52], whereas flow-based recommenders learn deterministic probability paths [27].

3.2 Flow Matching

Flow matching learns a time-dependent vector field that transports samples from p_0 to p_1 through an ordinary differential equation (ODE) [25, 30]. Under the common linear probability path, a training pair $(\boldsymbol{x}_0, \boldsymbol{x}_1)$ defines an interpolated state

$$\boldsymbol{x}_t = (1-t)\boldsymbol{x}_0 + t\boldsymbol{x}_1, \quad t \in [0,1]. \tag{1}$$

The corresponding conditional velocity is

$$\boldsymbol{u}_t\left(\boldsymbol{x}_t \mid \boldsymbol{x}_0, \boldsymbol{x}_1\right) = \frac{d\boldsymbol{x}_t}{dt} = \boldsymbol{x}_1 - \boldsymbol{x}_0. \tag{2}$$

In the sequential recommendation setting, conditional flow matching trains a neural vector field $\boldsymbol{v}_\theta$ conditioned on the user history S_u to approximate this direction by minimizing the objective

$$\mathcal{L}_{\mathrm{CFM}} = \mathbb{E}_{t,\boldsymbol{x}_0,\boldsymbol{x}_1,S_u} \left\|\boldsymbol{v}_\theta(\boldsymbol{x}_t, t, S_u) - (\boldsymbol{x}_1 - \boldsymbol{x}_0)\right\|_2^2. \tag{3}$$

Target-recovery flow matching instead directly generates the clean target embedding $\boldsymbol{x}_1$ and derives the velocity from that generation. MIRAGE adopts this parameterization in Section 4.

3.3 Co-occurrence Neighborhoods

Flow matching operates in a continuous latent space, but the item catalog is discrete. We construct an item graph $\mathcal{G} = (\mathcal{I}, \mathcal{E})$ from training-set co-occurrences only. Edges count ordered item pairs co-occurring within a context window of size c, and $\mathcal{N}_K(i)$ contains the K items with the largest nonzero counts for item i. The graph is precomputed and fixed during optimization.

The graph provides a criterion for measuring whether an interpolated state is close to the target-local co-occurrence neighborhood. For an interpolated state $\boldsymbol{x}_t$ associated with target item i^+, we define its graph-neighborhood distance as

$$d_{\mathcal{G}}(\boldsymbol{x}_t, i^+) = \min_{j \in \mathcal{N}_K(i^+)} \left\|\boldsymbol{x}_t - \boldsymbol{e}_j\right\|_2^2. \tag{4}$$

A large value indicates $\boldsymbol{x}_t$ is far from the target's co-occurrence neighborhood despite being a valid Euclidean vector, thus quantifying the Euclidean void introduced in Section 1.

4 Methodology

4.1 Overview

As illustrated in Figure 2, MIRAGE introduces a manifold-informed rectification mechanism over target-local co-occurrence neighborhoods. This dynamically reshapes the shared item geometry during training to encourage straight target-recovery trajectories to remain semantically supported while leaving inference graph-free. For a user history S_u, source noise $\boldsymbol{x}_0 \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$, and target $\boldsymbol{x}_1 = \boldsymbol{e}_{i^+}$, a flow matching network f_θ predicts the clean target $\hat{\boldsymbol{x}}_1$ directly from the interpolated state at time $t \in [\epsilon, 1]$.

4.2 The Euclidean Void and Manifold Mismatch

Building upon the conceptual intuition provided in Section 1, we now formalize the *Euclidean void* as a measurable geometric mismatch inherent to continuous recommendation spaces. Standard flow matching optimizes the vector field to approximate the target direction by minimizing $\mathcal{L}_{\mathrm{CFM}}$ in Eq. (3). However, this objective strictly supervises the velocity field, leaving the spatial and semantic validity of the intermediate states $\boldsymbol{x}_t$ unconstrained.

Since the ideal linear interpolation path in Eq. (1) is constructed in a global Euclidean space, it remains blind to the discrete underlying item structure. The exposure to the Euclidean void during training can be quantified by the expected geometric deviation along

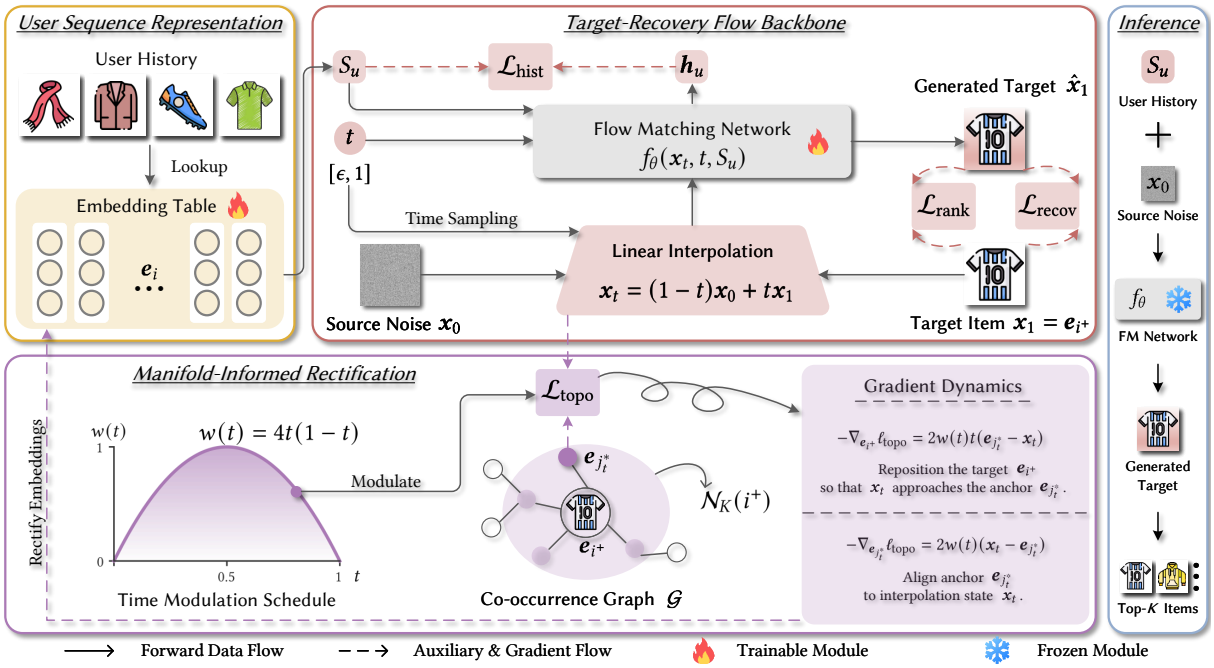

Figure 2: Overview of MIRAGE. The backbone f_θ predicts targets from interpolated user-conditioned states. A manifold-informed rectification mechanism dynamically reshapes the embedding space by aligning interpolated path states with local graph anchors, ensuring the generative path remains grounded in a valid semantic manifold.

this prescribed probability path. Using the graph-neighborhood distance from Eq. (4), the expected path deviation is defined as

$$\mathcal{D}_{\text{void}}(i^+) = \mathbb{E}_{t\sim U(0,1)}\left[d_{\mathcal{G}}(\boldsymbol{x}_t, i^+)\right]. \tag{5}$$

For standard continuous recommenders, this expected deviation $\mathcal{D}_{\text{void}}(i^+)$ is usually large. The mismatch is most consequential in the path interior, where the model must infer target-directed transport without the semantic certainty available at the endpoints, easily drifting into unsupported continuous regions.

Eq. (5) translates the Euclidean void into a quantifiable training proxy. In standard models, the generator is forced to decode precise target semantics from intermediate vectors lacking local semantic support. To ensure that the actual generative trajectories remain reliable, the underlying space must actively suppress $\mathcal{D}_{\text{void}}(i^+)$. Rather than computing an explicit continuous manifold, MIRAGE treats this prescribed deviation as a direct topology regularizer, letting the geometric penalties reshape the trainable item geometry to safely envelop generative trajectories.

4.3 Target-Recovery Flow Backbone

We adopt a target-recovery backbone as the generative foundation, optimized for target recovery, item discrimination, and historical reconstruction. We detail these base objectives before integrating our core manifold-informed rectification.

4.3.1 Target Recovery. To align with how sequential recommendation retrieves items via inner products, MIRAGE is designed to generate the target embedding directly. Consequently, we parameterize the conditioned vector field $\boldsymbol{v}_\theta$ to approximate the ground-truth conditional velocity from Eq. (2) by predicting the target state:

$$\boldsymbol{v}_\theta(\boldsymbol{x}_t, t, S_u) = f_\theta(\boldsymbol{x}_t, t, S_u) - \boldsymbol{x}_0, \tag{6}$$

where f_θ denotes the flow matching network. Substituting this parameterization into Eq. (3) gives

$$\begin{aligned}\mathcal{L}_{\text{recov}} &= \mathbb{E}_{t,\boldsymbol{x}_0,\boldsymbol{x}_1,S_u}\left\|(f_\theta(\boldsymbol{x}_t, t, S_u) - \boldsymbol{x}_0) - (\boldsymbol{x}_1 - \boldsymbol{x}_0)\right\|_2^2 \\ &= \mathbb{E}_{t,\boldsymbol{x}_0,\boldsymbol{x}_1,S_u}\left\|\hat{\boldsymbol{x}}_1 - \boldsymbol{x}_1\right\|_2^2,\end{aligned} \tag{7}$$

where $\hat{\boldsymbol{x}}_1 = f_\theta(\boldsymbol{x}_t, t, S_u)$ is the generated target state. By eliminating the source noise $\boldsymbol{x}_0$, this formulation shows that matching the conditional velocity is equivalent to minimizing the Euclidean distance to the clean target embedding $\boldsymbol{x}_1$. Appendix A.1 gives the step-by-step equivalence.

4.3.2 Catalog Ranking. Recovery alone pulls generations toward target coordinates and can collapse trainable item embeddings [7, 52]. To enforce robust contrastive separation, we augment training with a standard catalog-ranking objective over either the full item set or a sampled candidate set C_u that includes i^+. The objective is

$$\mathcal{L}_{\text{rank}} = -\log \frac{\exp(\hat{\boldsymbol{x}}_1^\top \boldsymbol{e}_{i^+})}{\sum_{a\in C_u}\exp(\hat{\boldsymbol{x}}_1^\top \boldsymbol{e}_a)}. \tag{8}$$

This auxiliary term preserves catalog-level discrimination while $\mathcal{L}_{\text{recov}}$ learns the continuous transport map.

4.3.3 History Reconstruction. To preserve historical preference semantics under continuous noise perturbations, we incorporate a history reconstruction objective [27, 40]. Specifically, after the flow backbone encodes the user sequence into a final hidden state $\boldsymbol{h}_u$, we train a lightweight decoder g_ω to reconstruct the multi-hot

user history vector $\boldsymbol{r}_u \in \{0,1\}^{|I|}$ directly from this representation, yielding the predicted interaction probabilities $\hat{\boldsymbol{r}}_u = g_\omega(\boldsymbol{h}_u)$.

To maintain computational efficiency, the loss is evaluated over a targeted subset Ω_u, which consists of all observed items and a sampled set of unobserved items. The reconstruction objective is formally defined as the mean squared error over this subset:

$$\mathcal{L}_{\text{hist}} = \frac{1}{|\mathcal{B}|} \sum_{u \in \mathcal{B}} \frac{1}{|\Omega_u|} \sum_{a \in \Omega_u} \left(\hat{r}_{u,a} - r_{u,a}\right)^2, \tag{9}$$

where $\mathcal{B}$ denotes a mini-batch of users, while $\hat{r}_{u,a}$ and $r_{u,a}$ denote the scalar values of the predicted vector $\hat{\boldsymbol{r}}_u$ and the ground-truth vector $\boldsymbol{r}_u$ at index a, respectively.

4.4 Manifold-Informed Rectification

To implement the manifold-informed rectification, MIRAGE overcomes the Euclidean void by anchoring each interpolated state to its target-local neighborhood. The resulting gradients dynamically reshape the embedding space around the unchanged probability path. We formulate this through a time-modulated topology regularizer.

4.4.1 Target-Local Alignment. To explicitly enforce spatial validity, MIRAGE relies on *hard alignment*, which anchors an interpolated state to a single valid item in the target-local neighborhood. Recalling Eq. (4) and assuming a unique nearest neighbor, we define the active anchor j_t^* and its gradient:

$$j_t^* = \underset{j \in \mathcal{N}_K(i^+)}{\arg\min} \|\boldsymbol{x}_t - \boldsymbol{e}_j\|_2^2, \quad \nabla_{\boldsymbol{x}_t} d_{\mathcal{G}}(\boldsymbol{x}_t, i^+) = 2(\boldsymbol{x}_t - \boldsymbol{e}_{j_t^*}). \tag{10}$$

This unique minimum renders the distance function differentiable almost everywhere, yielding a gradient that pulls the interpolant toward the semantic anchor $\boldsymbol{e}_{j_t^*}$. Alternatively, we formulate a *soft alignment* variant via a temperature-controlled smooth minimum:

$$d_{\mathcal{G},\tau}^{\text{soft}} = -\tau \log \left[\frac{1}{K} \sum_{j \in \mathcal{N}_K(i^+)} \exp\left(-\frac{\|\boldsymbol{x}_t - \boldsymbol{e}_j\|_2^2}{\tau}\right) \right], \tag{11}$$

whose gradient points toward a softmax-weighted centroid of local neighbors. Unlike standard graph-based contrastive learning where continuous averages of discrete items may fall into unsupported regions, hard alignment offers a stricter geometric constraint on the discrete item manifold while soft alignment provides a smoother optimization landscape. We evaluate both variants empirically, providing complete theoretical derivations in Appendix A.2.

4.4.2 Time Modulation Schedule. To enforce the aforementioned spatial alignment without violating the boundary conditions of the generative flow, MIRAGE applies this target-local constraint dynamically. The regularizer must strictly vanish at $t = 0$ to preserve the unstructured source prior, and at $t = 1$ to leave the clean target state unconstrained. Among symmetric quadratic schedules normalized to a peak of $w(1/2) = 1$, these standard boundary conditions yield:

$$w(t) = 4t(1-t). \tag{12}$$

This schedule concentrates semantic guidance in the path interior, where the interpolant mixes the source and target strongly. Its parabolic shape matches the uncertainty profile $t(1-t)$ of a unit Brownian bridge, serving here as a geometric analogy [18]. By applying this weight to minimize the expected path deviation $\mathcal{D}_{\text{void}}(i^+)$ defined in Eq. (5), we formulate the target-local regularizer:

$$\mathcal{L}_{\text{topo}} = \mathbb{E}_{t,\boldsymbol{x}_0,\boldsymbol{x}_1,S_u}\left[w(t) \cdot \|\boldsymbol{x}_t - \boldsymbol{e}_{j_t^*}\|_2^2 \right]. \tag{13}$$

Appendix A.3 analyzes the uniqueness of this normalized quadratic schedule and details its bridge covariance properties.

4.4.3 Gradient Dynamics and Geometric Rectification. To analyze how this regularizer reshapes the space, consider its per-sample integrand $\ell_{\text{topo}} = w(t)\|\boldsymbol{x}_t - \boldsymbol{e}_{j_t^*}\|_2^2$. The active anchor is locally constant almost surely, and $\partial \boldsymbol{x}_t / \partial \boldsymbol{e}_{i^+} = t\boldsymbol{I}$. The chain rule yields:

$$\begin{aligned} \nabla_{\boldsymbol{e}_{i^+}} \ell_{\text{topo}} &= 2w(t)t\left(\boldsymbol{x}_t - \boldsymbol{e}_{j_t^*}\right), \\ \nabla_{\boldsymbol{e}_{j_t^*}} \ell_{\text{topo}} &= -2w(t)\left(\boldsymbol{x}_t - \boldsymbol{e}_{j_t^*}\right). \end{aligned} \tag{14}$$

Through these updates, gradient descent repositions the target embedding $\boldsymbol{e}_{i^+}$ so that the interpolated state $\boldsymbol{x}_t$ approaches the anchor $\boldsymbol{e}_{j_t^*}$, while simultaneously aligning the anchor $\boldsymbol{e}_{j_t^*}$ toward the current interpolation state. The time-modulation factor $w(t)$ suppresses both updates at the source and target boundaries, concentrating the spatial reorganization exclusively in the path interior. Appendix A.4 demonstrates that these joint updates contract the residual distance under a small step size.

Notably, these pairwise forces update the shared embedding table without explicit graph message passing. Although the topology term does not directly update flow parameters, it reorganizes the Euclidean space to envelop sampled path interiors with valid item evidence. This reshapes the target trajectories into a smoother continuous vector field for the flow backbone to regress. Consequently, this path-interior topology supervision restructures the space into a well-behaved vector field, minimizing discretization errors and ensuring accurate one-step target recovery.

4.5 Training Objective and Inference

The full training objective integrates the flow backbone and the geometric regularization:

$$\mathcal{L}_{\text{MIRAGE}} = \mathcal{L}_{\text{recov}} + \lambda_{\text{rank}}\mathcal{L}_{\text{rank}} + \lambda_{\text{hist}}\mathcal{L}_{\text{hist}} + \lambda_{\text{topo}}\mathcal{L}_{\text{topo}}. \tag{15}$$

At inference time, MIRAGE operates without requiring any graph computations. Given a user history S_u, the model samples a source noise $\boldsymbol{x}_0 \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$ and generates the target representation via q-step deterministic Euler sampling ($q \in \mathbb{N}^+$). Dividing the integration interval $[\epsilon, 1]$ into q uniform steps, the network times are discretized as $t_n = \epsilon + (1-\epsilon)\frac{n}{q}$, for $n = 0, \ldots, q-1$. Following the linear velocity formulation in Eq. (2), our target-recovery network explicitly predicts the clean target $\boldsymbol{x}_1$ as $f_\theta(\tilde{\boldsymbol{x}}_n, t_n, S_u)$. Thus, the intermediate states are recursively updated via Euler integration using the exact step size $\Delta t = \frac{1-\epsilon}{q}$:

$$\tilde{\boldsymbol{x}}_{n+1} = \tilde{\boldsymbol{x}}_n + \frac{1-\epsilon}{q}\left[f_\theta(\tilde{\boldsymbol{x}}_n, t_n, S_u) - \boldsymbol{x}_0\right], \tag{16}$$

with $\tilde{\boldsymbol{x}}_0 = \boldsymbol{x}_0$. For efficient one-step generation ($q = 1$), this recursion evaluates to $\tilde{\boldsymbol{x}}_1 = (1-\epsilon)f_\theta(\boldsymbol{x}_0, \epsilon, S_u) + \epsilon\boldsymbol{x}_0$. Since the minimum network time ϵ is negligible, this expression reduces to

$$\hat{\boldsymbol{x}}_1 = f_\theta(\boldsymbol{x}_0, \epsilon, S_u). \tag{17}$$

Table 1: Main results on four datasets (%; higher is better). Best and second are marked. ± denotes standard deviation over 5 runs. MIRAGE significantly outperforms the best baselines (paired t-test, $p < 0.05$).

Type	Model	Beauty				Sports				Toys				CDs			
		H@10	H@20	N@10	N@20	H@10	H@20	N@10	N@20	H@10	H@20	N@10	N@20	H@10	H@20	N@10	N@20
Trad.	GRU4Rec	2.53±0.10	4.26±0.22	1.21±0.05	1.64±0.10	1.92±0.12	2.96±0.18	1.01±0.07	1.27±0.07	1.79±0.09	3.19±0.17	0.86±0.07	1.21±0.06	3.77±0.25	6.29±0.37	1.86±0.10	2.50±0.12
	Caser	2.37±0.11	3.97±0.25	1.16±0.07	1.56±0.10	1.82±0.10	2.78±0.15	0.90±0.05	1.14±0.08	1.75±0.11	2.74±0.12	0.88±0.07	1.13±0.06	3.00±0.12	5.00±0.18	1.48±0.10	1.98±0.12
	HGN	1.94±0.10	3.09±0.20	0.93±0.06	1.22±0.10	1.22±0.08	2.02±0.12	0.67±0.06	0.87±0.07	1.32±0.07	2.27±0.10	0.68±0.06	0.92±0.06	0.57±0.05	0.96±0.06	0.29±0.04	0.39±0.04
	NextItNet	4.47±0.28	7.14±0.31	2.20±0.10	2.87±0.17	2.65±0.14	4.33±0.28	1.35±0.07	1.77±0.09	3.24±0.13	5.09±0.23	1.62±0.06	2.08±0.13	5.45±0.20	8.56±0.34	2.73±0.17	3.51±0.20
	LightGCN	4.38±0.17	6.90±0.35	2.12±0.14	2.76±0.16	2.79±0.12	4.73±0.31	1.43±0.08	1.92±0.12	4.30±0.26	6.33±0.27	2.14±0.14	2.65±0.11	5.18±0.20	7.98±0.27	2.62±0.13	3.32±0.12
	SASRec	4.89±0.21	7.69±0.32	2.11±0.14	2.82±0.17	2.95±0.15	4.71±0.21	1.26±0.07	1.70±0.09	5.67±0.34	8.31±0.41	2.47±0.13	3.13±0.14	4.79±0.28	7.90±0.39	2.08±0.12	2.86±0.13
	BERT4Rec	4.13±0.19	6.27±0.28	2.20±0.10	2.74±0.12	2.03±0.13	3.48±0.17	1.01±0.06	1.37±0.10	3.54±0.22	5.18±0.22	1.86±0.13	2.27±0.12	5.66±0.34	8.70±0.53	2.85±0.12	3.61±0.19
Diff.	ADRec	7.62±0.12	10.52±0.09	4.49±0.05	5.22±0.06	4.03±0.14	5.96±0.13	2.21±0.06	2.69±0.05	7.75±0.20	10.37±0.25	4.84±0.07	5.50±0.08	6.86±0.17	9.91±0.22	3.69±0.07	4.46±0.08
	DiffuRec	7.63±0.08	10.49±0.15	4.58±0.09	5.30±0.11	3.70±0.23	5.51±0.20	2.03±0.19	2.49±0.18	7.26±0.14	9.74±0.18	4.59±0.05	5.21±0.06	8.55±0.50	12.55±0.64	4.41±0.25	5.41±0.28
	DimeRec	6.25±0.10	9.76±0.18	3.07±0.10	3.96±0.11	3.70±0.11	5.89±0.13	1.93±0.08	2.48±0.08	5.86±0.27	8.86±0.31	2.91±0.12	3.66±0.12	5.52±0.16	8.85±0.26	2.76±0.09	3.60±0.11
	TA-Rec	0.42±0.04	0.85±0.06	0.21±0.02	0.35±0.03	0.38±0.03	0.72±0.06	0.18±0.03	0.25±0.03	0.55±0.05	0.92±0.06	0.25±0.03	0.38±0.03	0.48±0.05	0.81±0.06	0.22±0.02	0.31±0.02
	DreamRec	0.05±0.02	0.08±0.02	0.02±0.02	0.03±0.02	0.03±0.02	0.06±0.02	0.01±0.03	0.02±0.03	0.04±0.03	0.09±0.03	0.02±0.03	0.03±0.01	0.06±0.02	0.12±0.03	0.03±0.03	0.04±0.03
Flow	FMRec	8.20±0.10	11.47±0.16	4.79±0.04	5.61±0.04	4.36±0.23	6.45±0.23	2.43±0.19	2.95±0.19	8.31±0.08	11.21±0.14	5.17±0.04	5.90±0.05	8.77±0.23	12.97±0.32	4.49±0.11	5.55±0.13
	FAVE	7.97±0.27	11.44±0.39	4.20±0.13	5.08±0.16	2.90±1.80	4.52±2.49	1.56±0.95	1.96±1.13	5.35±4.01	7.53±5.34	2.80±2.07	3.35±2.40	8.21±0.23	12.34±0.30	4.14±0.07	5.18±0.09
	MIRAGE	**8.53**±0.13	**12.37**±0.11	**4.94**±0.07	**5.71**±0.07	**4.77**±0.15	**7.14**±0.14	**2.51**±0.07	**3.11**±0.06	**8.89**±0.37	**12.52**±0.34	**5.31**±0.35	**6.08**±0.34	**9.11**±0.24	**13.51**±0.45	**4.65**±0.10	**5.76**±0.15
	Imp.(%)	+4.02	+7.85	+3.13	+1.78	+9.40	+10.70	+3.29	+5.42	+6.98	+11.69	+2.71	+3.05	+3.88	+4.16	+3.56	+3.78

Setting the final target representation as $\hat{\boldsymbol{x}}_1 \equiv \tilde{\boldsymbol{x}}_q$, we compute the candidate scores for ranking via inner products:

$$s_a = \hat{\boldsymbol{x}}_1^\top \boldsymbol{e}_a, \quad \forall a \in \mathcal{I}. \tag{18}$$

The integration step q balances inference cost and refinement. Our training-time topological guidance enables zero-graph overhead during generation while maintaining robust recommendation quality even at $q = 1$ as demonstrated in Section 5.6.

5 Experiments

We evaluate MIRAGE through overall ranking performance, geometric rectification, component analysis, long-tail recommendation, hyperparameter sensitivity, and inference efficiency.

5.1 Experimental Setup

5.1.1 Datasets & Baselines. To evaluate across diverse topological structures and sparsity levels, we use four standard Amazon review datasets (Beauty, Sports, Toys, and CDs) [13]. [1] Following established protocols [17, 44], we apply 5-core filtering and retain each user's 20 most recent interactions (statistics in Table 2).

Table 2: Statistics of the datasets.

Dataset	#Users	#Items	#Interactions	Avg. Len.	Sparsity
Beauty	22,363	12,101	198,502	8.88	99.93%
Sports	35,598	18,357	296,337	8.32	99.95%
Toys	19,412	11,924	167,597	8.63	99.93%
CDs	75,258	64,443	1,097,592	14.58	99.98%

We evaluate MIRAGE against three groups of baselines. **Traditional models** include GRU4Rec [15] based on recurrent networks, Caser [45] and NextItNet [55] using convolutional networks, HGN [32] with hierarchical gates, graph-based LightGCN [14], and attention-based SASRec [17] and BERT4Rec [44]. **Diffusion-based models** consist of ADRec [7] for staged token-level diffusion, DiffuRec [23] for distributional target generation, DimeRec [22] for spherical geodesic diffusion, TA-Rec [35] for one-step denoising, and DreamRec [52] for oracle-like target reconstruction. **Flow-matching models** include FMRec [27], which constructs straight continuous trajectories and applies deterministic Euler solvers for direct target recovery, and FAVE [40], which jointly leverages a history-informed semantic prior and regularizes flow consistency to enable accurate one-step generation.

5.1.2 Metrics & Implementation Details. We adopt leave-one-out evaluation with full-catalog ranking and report Hit Rate (H@K) and Normalized Discounted Cumulative Gain (N@K) for $K \in \{10, 20\}$ [14]. All experiments are conducted on an NVIDIA RTX 4090 GPU. The flow network f_θ utilizes a Transformer with 4 blocks, 4 heads, and a hidden dimension $d = 128$, while the history decoder g_ω consists of 2 layers with widths 512 and 2048. We optimize using Adam with a learning rate of 10^{-3} and a batch size of 512, applying early stopping governed by validation N@20. We fix $\lambda_{\text{hist}} = 0.4$ [27], $\epsilon = 0.001$, tuning other parameters via validation and detailing sensitivity in Appendix B.2. Hard alignment proves optimal for Beauty, Sports, and Toys, whereas CDs require the soft variant. Inference strictly defaults to direct one-step generation with $q = 1$. To ensure robustness, our main performance comparisons report averages across five random seeds (2022–2026). All subsequent analyses fix seed 2026 to isolate individual experimental factors.

5.2 Performance and Geometric Rectification

5.2.1 Overall Performance. Table 1 summarizes the recommendation performance, where MIRAGE achieves the best results across all datasets and metrics. Compared to the strongest baseline FMRec, our framework delivers substantial relative improvements, highlighted by a 10.70% and 11.69% increase in H@20 on the Sports and

[1] The datasets are available at https://jmcauley.ucsd.edu/data/amazon/.

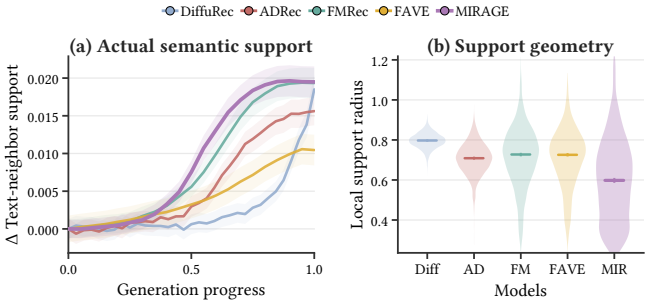

Figure 3: Geometric rectification analysis. (a) Semantic support along the actual generation trajectories, evaluated using an independent text embedding space. (b) Distributions of the normalized local support radius across the item catalog.

Toys datasets. Conversely, DreamRec and TA-Rec experience significant performance degradation. Under our rigorous full-catalog protocol, DreamRec's omission of negative sampling [7, 22] and TA-Rec's preference alignment on sparse sequences lead to reduced embedding dispersion, making target items difficult to distinguish. While continuous generative approaches generally outperform traditional models, MIRAGE maintains a distinct advantage over standard flow-based methods. This advantage validates our core structural hypothesis that optimizing for a straight trajectory leaves the intermediate states of standard methods vulnerable to unsupported continuous regions. By dynamically rectifying the embedding geometry during training to enforce continuous semantic support, MIRAGE prevents generated representations from drifting into the Euclidean void, thereby translating straight transport routes into accurate next-item predictions.

5.2.2 Geometric Rectification. To evaluate whether models escape the Euclidean void during inference, we analyze the intermediate states along the actual generated trajectories. Specifically, we retrieve the nearest neighbors of these states within the learned embedding space and measure their similarity to the true target using a frozen, independent text embedding space.[2] Figure 3a illustrates the relative gain in this text-neighbor support. While baselines traverse regions with sparse target semantics, MIRAGE exhibits an earlier and more pronounced rise in semantic alignment. This confirms our training proxy anchors actual paths, a trend that directly correlates with the overall accuracy improvements in Table 1.

Beyond securing individual trajectories, we examine the global catalog geometry in Figure 3b using the local support radius, defined as the normalized distance from each item to its semantic neighbors. The violin plots show that MIRAGE shifts this distribution downward, yielding a lower median radius than the baselines. A smaller normalized radius reflects denser semantic clustering across the catalog. This indicates that our manifold-informed regularization reorganizes the underlying Euclidean space, establishing a more compact and cohesive environment for continuous generation.

[2]Semantic similarity is computed using cosine similarity on frozen item text embeddings extracted via Sentence-T5.

Table 3: Ablation study of MIRAGE.

Variant	Beauty		Sports		Toys		CDs	
	H@20	N@20	H@20	N@20	H@20	N@20	H@20	N@20
MIRAGE	**12.27**	**5.71**	**7.22**	**3.12**	**12.78**	**6.03**	**13.84**	**5.91**
w/o Topology	11.67	5.64	6.57	3.04	10.99	5.82	13.54	5.85
w/o Time Mod.	12.25	5.63	6.55	2.82	12.74	5.90	13.63	5.78
w/ Alt. Alignment	11.98	5.45	5.67	2.44	12.66	5.98	13.75	5.77
w/ Endpoint Reg.	10.16	4.47	5.25	2.18	10.12	4.59	13.69	5.74

5.3 Ablation Study

To disentangle the contributions of MIRAGE, Table 3 compares the full MIRAGE framework against four specific variants. The ablated variants are defined as follows: 1) **w/o Topology** disables the geometric regularizer entirely by setting the weight $\lambda_{\text{topo}} = 0$ in Eq. (15); 2) **w/o Time Mod.** removes the parabolic schedule $w(t) = 4t(1-t)$ defined in Eq. (12), uniformly applying a constant topology penalty across the entire interpolation timeframe; 3) **w/ Alt. Alignment** inverts the optimal dataset-specific configuration by switching between the hard discrete anchor $\boldsymbol{e}_{j_t^*}$ and a soft weighted centroid; and 4) **w/ Endpoint Reg.** acts as standard Graph Laplacian smoothing by minimizing the L2 distance between target and neighbor embeddings, isolating our path-interior gains.

Removing the topology term yields the most severe performance degradation, highlighted by a drop in H@20 from 12.78 to 10.99 on the Toys dataset. This confirms that the generative flow backbone alone, despite its sequence modeling capabilities, struggles with the Euclidean void and relies on explicit geometric regularization to maintain valid semantic support. Furthermore, replacing the time modulation schedule with a constant penalty undermines recommendation quality, reducing H@20 from 7.22 to 6.55 on Sports. This outcome validates that applying the topological constraint indiscriminately near the boundary endpoints ($t \to 0$ or 1) corrupts the unstructured source noise and the clean target representations.

Switching alignment configurations degrades performance across all domains. Dense datasets rely on discrete hard anchors to keep intermediate states out of unsupported regions, while sparse datasets like CDs require continuous soft aggregation to stabilize local gradients. This demonstrates that alignment techniques must naturally adapt to the underlying structural sparsity. Furthermore, standard Graph Laplacian smoothing at the target endpoint performs worse than omitting the graph entirely. Forcing target items too close to their neighbors collapses their distinct representations and degrades precise ranking. This confirms that semantic support must be applied dynamically to the intermediate states along the generative trajectory, rather than statically at the final destination.

5.4 Long-Tail Performance

To evaluate performance across item popularities, we divide the catalog based on training-set frequency. Items in the bottom 80% of the catalog ranked by their training frequency form the long tail, while the remaining 20% constitute the head.

As shown in Figure 4, MIRAGE achieves its most substantial relative gains on long-tail targets. On Sports, it improves H@20 over the strongest baseline by 14.0% for the tail compared to 8.7%

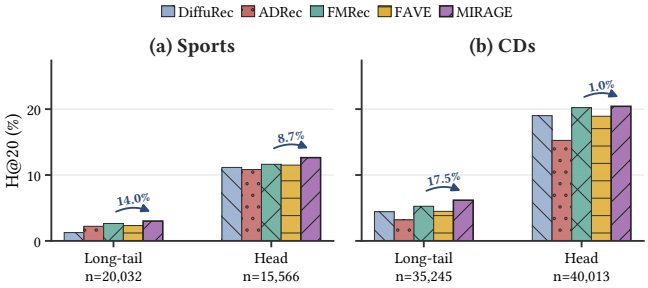

Figure 4: H@20 for Long-Tail and Head Items on Sports/CDs (n denotes the number of test targets in each group).

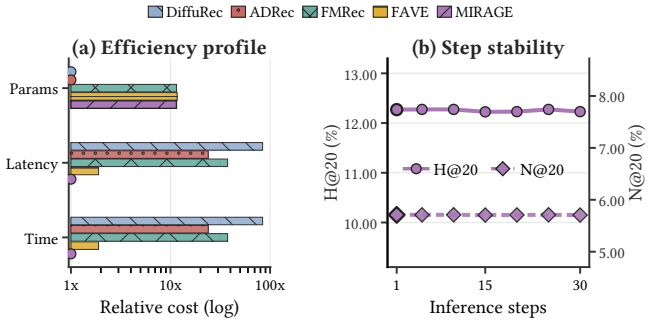

Figure 6: Efficiency profile and step stability. MIRAGE minimizes relative inference latency while maintaining robust accuracy under direct one-step generation.

for the head. On CDs, these respective gains reach 17.5% and 1.0%, with consistent trends across the remaining datasets detailed in Appendix B.6. This pronounced improvement on sparse items reflects the utility of our target-local support. Infrequent items receive fewer gradient updates during training, leaving their surrounding embedding geometry poorly defined by standard target supervision. By leveraging shared co-occurrence anchors, MIRAGE allows these sparse targets to draw geometric support from high-frequency related items, accumulating local evidence throughout the shared embedding space. Without introducing popularity-specific parameters or sacrificing head accuracy, this mechanism effectively concentrates geometric guidance exactly where the continuous generative trajectory is most vulnerable to the Euclidean void. Crucially, confining this support to the path interior prevents sparse items from collapsing into popular neighbors, avoiding popularity bias while preserving their distinct rankability.

5.5 Hyperparameter Analysis

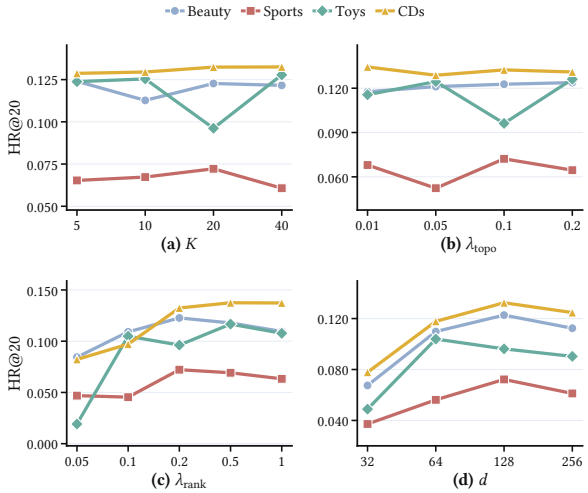

Figure 5: H@20 under one-at-a-time hyperparameter sweeps.

To systematically assess model sensitivity, Figure 5 shows that optimal neighborhood sizes and loss weights vary across datasets and thus justify our selection through validation. While MIRAGE maintains robust performance over a broad middle range of values, the irregular response on the Toys dataset to both K and λ_{topo} reveals a genuine sensitivity to local graph structure. The neighborhood size governs the balance between support and specificity; a small K provides precise but potentially unstable anchors, whereas a large K increases geometric coverage at the risk of pulling the generative trajectory toward weakly related items.

Similarly, the interior optima for the topology weight confirm that geometric rectification complements item discrimination. Sweeps over the ranking weight and embedding dimension (peaking at $d = 128$) verify that adequate spatial separation and controlled model capacity remain necessary alongside geometric reorganization. Appendix B.2 details the exact search grids and further analyzes the co-occurrence window size c, showing that dense catalogs benefit from broader semantic contexts ($c = 5$) whereas sparse datasets require strict adjacency ($c = 1$) to prevent noisy guidance.

5.6 Efficiency Analysis

The efficiency profile in Figure 6 compares the relative inference costs of MIRAGE against recent generative recommenders. MIRAGE establishes the lowest latency and total inference time, operating orders of magnitude faster than iterative models like DiffuRec while further outperforming the one-step FAVE by eliminating its prior-construction overhead. Since MIRAGE shares the exact same model capacity as FMRec and FAVE, its superior inference speed stems from its one-step generation strategy rather than a reduced model size. The topology regularizer and neighborhood retrieval operations are discarded after optimization to achieve zero graph overhead during deployment, while establishing an optimal training time trade-off detailed in Appendix B.4.

The step stability plot in Figure 6 further justifies this direct generation strategy by tracking recommendation accuracy across varying inference steps. Both metrics remain stable as the integration steps increase from 1 to 30. This stability aligns with our target recovery parameterization and straight probability path. Additional sampling steps only refine a target estimate that already receives direct supervision during training. This verifies that our training-time manifold rectification shapes a well-behaved continuous vector field, enabling safe and accurate one-step traversal.

These extra steps therefore multiply the computational cost without supplying new semantic evidence or measurable accuracy gains and confirm one-step inference as the optimal practical setting.

6 Conclusion and Future Work

In this paper, we identify the Euclidean void as a critical failure mode in continuous generative recommendation. We demonstrate that straight interpolation paths often traverse unsupported regions and cause intermediate states to lose semantic guidance. To address this limitation, we propose MIRAGE, which dynamically rectifies the underlying item geometry to ensure continuous semantic support throughout the generation process. This reorganization prevents representation drift while preserving an efficient inference mechanism. Extensive evaluations on four datasets confirm that this geometric grounding delivers robust performance improvements and yields particular benefits for sparsely observed items.

While MIRAGE operates in flat Euclidean spaces, future work will explore hyperbolic embeddings to better capture hierarchical item relationships [36]. Such non-Euclidean manifolds accommodate complex category structures and asymmetric user interactions. Integrating these geometric priors into the continuous transport process could provide even tighter bounds on intermediate representations, thereby further elevating the overall generative reliability.

References

[1] Michael S. Albergo and Eric Vanden-Eijnden. 2023. Building Normalizing Flows with Stochastic Interpolants. In *Proc. of ICLR*.

[2] Georgios Arvanitidis, Lars Kai Hansen, and Søren Hauberg. 2018. Latent Space Oddity: On the Curvature of Deep Generative Models. In *Proc. of ICLR*.

[3] Yimeng Bai, Yang Zhang, Sihao Ding, Shaohui Ruan, Han Yao, Danhui Guan, Fuli Feng, and Tat-Seng Chua. 2026. Brownian Bridge Diffusion for Sequential Recommendation. arXiv:2507.06121

[4] Oren Barkan and Noam Koenigstein. 2016. Item2vec: neural item embedding for collaborative filtering. In *2016 IEEE 26th international workshop on machine learning for signal processing (MLSP)*. 1–6.

[5] Mikhail Belkin, Partha Niyogi, and Vikas Sindhwani. 2006. Manifold Regularization: A Geometric Framework for Learning from Labeled and Unlabeled Examples. *J. Mach. Learn. Res.* 7, 85 (2006), 2399–2434.

[6] Yoshua Bengio, Aaron Courville, and Pascal Vincent. 2013. Representation Learning: A Review and New Perspectives. *IEEE Trans. Pattern Anal. Mach. Intell.* 35, 8 (2013), 1798–1828.

[7] Jialei Chen, Yuanbo Xu, and Yiheng Jiang. 2025. Unlocking the Power of Diffusion Models in Sequential Recommendation: A Simple and Effective Approach. In *Proc. of KDD*. 155–166.

[8] Ricky T. Q. Chen and Yaron Lipman. 2024. Flow Matching on General Geometries. In *Proc. of ICLR*.

[9] Mingyue Cheng, Qi Liu, Wenyu Zhang, Zhiding Liu, Hongke Zhao, and Enhong Chen. 2024. A General Tail Item Representation Enhancement Framework for Sequential Recommendation. *Front. Comput. Sci.* 18, 6 (2024), 186333.

[10] Ziwei Fan, Zhiwei Liu, Yu Wang, Alice Wang, Zahra Nazari, Lei Zheng, Hao Peng, and Philip S. Yu. 2022. Sequential Recommendation via Stochastic Self-Attention. In *Proc. of WWW*. 2036–2047.

[11] Charles Fefferman, Sanjoy Mitter, and Hariharan Narayanan. 2016. Testing the Manifold Hypothesis. *J. Amer. Math. Soc.* 29, 4 (2016), 983–1049.

[12] Ruining He, Wang-Cheng Kang, and Julian McAuley. 2017. Translation-Based Recommendation. In *Proc. of RecSys*. 161–169.

[13] Ruining He and Julian McAuley. 2016. Ups and Downs: Modeling the Visual Evolution of Fashion Trends with One-Class Collaborative Filtering. In *Proc. of WWW*. 507–517.

[14] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yong-Dong Zhang, and Meng Wang. 2020. LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation. In *Proc. of SIGIR*. 639–648.

[15] Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. 2016. Session-Based Recommendations with Recurrent Neural Networks. In *Proc. of ICLR*.

[16] Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising Diffusion Probabilistic Models. In *Proc. of NeurIPS*. 6840–6851.

[17] Wang-Cheng Kang and Julian McAuley. 2018. Self-Attentive Sequential Recommendation. In *Proc. of ICDM*. 197–206.

[18] Ioannis Karatzas and Steven E. Shreve. 1991. *Brownian Motion and Stochastic Calculus* (2 ed.). Vol. 113. Springer, New York, NY, USA.

[19] Kibum Kim, Dongmin Hyun, Sukwon Yun, and Chanyoung Park. 2023. MELT: Mutual Enhancement of Long-Tailed User and Item for Sequential Recommendation. In *Proc. of SIGIR*. 68–77.

[20] Nikita Kornilov, Petr Mokrov, Alexander V. Gasnikov, and Alexander Korotin. 2024. Optimal Flow Matching: Learning Straight Trajectories in Just One Step. In *Proc. of NeurIPS*.

[21] Li Li, Mingyue Cheng, Yuyang Ye, Zhiding Liu, and Enhong Chen. 2025. Preference Trajectory Modeling via Flow Matching for Sequential Recommendation. *arXiv preprint arXiv:2508.17618* (2025).

[22] Wuchao Li, Rui Huang, Haijun Zhao, Chi Liu, Kai Zheng, Qi Liu, Na Mou, Guorui Zhou, Defu Lian, Yang Song, Wentian Bao, Enyun Yu, and Wenwu Ou. 2025. DimeRec: A Unified Framework for Enhanced Sequential Recommendation via Generative Diffusion Models. In *Proc. of WSDM*. 726–734.

[23] Zihao Li, Aixin Sun, and Chenliang Li. 2024. DiffuRec: A Diffusion Model for Sequential Recommendation. *ACM Trans. Inf. Syst.* 42, 3 (2024), 66:1–66:28.

[24] Zihao Li, Chao Yang, Yakun Chen, Xianzhi Wang, Hongxu Chen, Guandong Xu, Lina Yao, and Michael Sheng. 2024. Graph and sequential neural networks in session-based recommendation: A survey. *Comput. Surveys* 57, 2 (2024), 1–37.

[25] Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. 2023. Flow Matching for Generative Modeling. In *Proc. of ICLR*.

[26] Chengkai Liu, Yangtian Zhang, Jianling Wang, Rex Ying, and James Caverlee. 2025. Flow Matching for Collaborative Filtering. In *Proc. of KDD*. 1765–1775.

[27] Feng Liu, Lixin Zou, Xiangyu Zhao, Min Tang, Liming Dong, Dan Luo, Xiangyang Luo, and Chenliang Li. 2025. Flow Matching Based Sequential Recommender Model. In *Proc. of IJCAI*. 3108–3116.

[28] Qidong Liu, Fan Yan, Xiangyu Zhao, Zhaocheng Du, Huifeng Guo, Ruiming Tang, and Feng Tian. 2023. Diffusion augmentation for sequential recommendation. In *Proc. of CIKM*. 1576–1586.

[29] Sijia Liu, Jiahao Liu, Hansu Gu, Dongsheng Li, Tun Lu, Peng Zhang, and Ning Gu. 2023. AutoSeqRec: Autoencoder for Efficient Sequential Recommendation. In *Proc. of CIKM*. 1493–1502.

[30] Xingchao Liu, Chengyue Gong, and Qiang Liu. 2023. Flow Straight and Fast: Learning to Generate and Transfer Data with Rectified Flow. In *Proc. of ICLR*.

[31] Jing Long, Guanhua Ye, Tong Chen, Yang Wang, Meng Wang, and Hongzhi Yin. 2024. Diffusion-based cloud-edge-device collaborative learning for next POI recommendations. In *Proc. of KDD*. 2026–2036.

[32] Chen Ma, Peng Kang, and Xue Liu. 2019. Hierarchical Gating Networks for Sequential Recommendation. In *Proc. of KDD*. 825–833.

[33] Haokai Ma, Ruobing Xie, Lei Meng, Xin Chen, Xu Zhang, Leyu Lin, and Zhanhui Kang. 2024. Plug-in diffusion model for sequential recommendation. In *Proc. of AAAI*. 8886–8894.

[34] Wenyu Mao, Shuchang Liu, Haoyang Liu, Haozhe Liu, Xiang Li, and Lantao Hu. 2025. Distinguished Quantized Guidance for Diffusion-Based Sequence Recommendation. In *Proc. of WWW*. 425–435.

[35] Wenyu Mao, Jiancan Wu, Guoqing Hu, Zhengyi Yang, Wei Ji, and Xiang Wang. 2025. On Efficiency-Effectiveness Trade-off of Diffusion-based Recommenders. In *Proc. of NeurIPS*. 92565–92591.

[36] Maximilian Nickel and Douwe Kiela. 2017. Poincaré Embeddings for Learning Hierarchical Representations. In *Proc. of NeurIPS*. 6338–6347.

[37] Aram-Alexandre Pooladian, Heli Ben-Hamu, Carles Domingo-Enrich, Brandon Amos, Yaron Lipman, and Ricky TQ Chen. 2023. Multisample Flow Matching: Straightening Flows with Minibatch Couplings. In *Proc. of ICML*. 28100–28127.

[38] Ruihong Qiu, Zi Huang, Hongzhi Yin, and Zijian Wang. 2022. Contrastive Learning for Representation Degeneration Problem in Sequential Recommendation. In *Proc. of WSDM*. 813–823.

[39] Noveen Sachdeva, Giuseppe Manco, Ettore Ritacco, and Vikram Pudi. 2019. Sequential Variational Autoencoders for Collaborative Filtering. In *Proc. of WSDM*. 600–608.

[40] Ke Shi, Yao Zhang, Feng Guo, Jinyuan Zhang, JunShuo Zhang, Shen Gao, and Shuo Shang. 2026. FAVE: Flow-Based Average Velocity Establishment for Sequential Recommendation. In *Proc. of SIGIR*. 1633–1643.

[41] Te Song, Lianyong Qi, Weiming Liu, Fan Wang, Xiaolong Xu, Xuyun Zhang, Amin Beheshti, Xiaokang Zhou, and Wanchun Dou. 2025. Enhancing diffusion model with auxiliary information mining-exploration and efficient sampling mechanism for sequential recommendation. In *Proc. of AAAI*. 12568–12576.

[42] Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. 2023. Consistency Models. In *Proc. of ICML*. 32211–32252.

[43] Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. 2021. Score-Based Generative Modeling through Stochastic Differential Equations. In *Proc. of ICLR*.

[44] Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. 2019. BERT4Rec: Sequential Recommendation with Bidirectional Encoder Representations from Transformer. In *Proc. of CIKM*. 1441–1450.

[45] Jiaxi Tang and Ke Wang. 2018. Personalized Top-N Sequential Recommendation via Convolutional Sequence Embedding. In *Proc. of WSDM*. 565–573.

[46] Wenjie Wang, Yiyan Xu, Fuli Feng, Xinyu Lin, Xiangnan He, and Tat-Seng Chua. 2023. Diffusion Recommender Model. In *Proc. of SIGIR*. 832–841.

[47] Ziyang Wang, Wei Wei, Gao Cong, Xiao-Li Li, Xian-Ling Mao, and Minghui Qiu. 2020. Global Context Enhanced Graph Neural Networks for Session-Based Recommendation. In *Proc. of SIGIR*. 169–178.

[48] Chunyu Wei, Jian Liang, Di Liu, Zehui Dai, Mang Li, and Fei Wang. 2023. Meta graph learning for long-tail recommendation. In *Proc. of KDD*. 2512–2522.

[49] Shu Wu, Yuyuan Tang, Yanqiao Zhu, Liang Wang, Xing Xie, and Tieniu Tan. 2019. Session-Based Recommendation with Graph Neural Networks. In *Proc. of AAAI*. 346–353.

[50] Shaoheng Xie, Xuan Zhang, Kunpeng Du, Jishu Wang, Xiaopeng Li, Junda Li, Weiyi Shang, and Zhi Jin. 2027. Diffusion sequential recommendation model based on item co-occurrence anchoring and user semantic collaboration. *Expert Systems with Applications* 332 (2027), 133431.

[51] Yiyuan Yang, Ming Jin, Haomin Wen, Chaoli Zhang, Yuxuan Liang, Lintao Ma, Yi Wang, Chenghao Liu, Bin Yang, Zenglin Xu, Shirui Pan, and Qingsong Wen. 2026. A Survey on Diffusion Models for Time Series and Spatio-Temporal Data. *ACM Comput. Surv.* 58, 8 (2026).

[52] Zhengyi Yang, Jiancan Wu, Zhicai Wang, Xiang Wang, Yancheng Yuan, and Xiangnan He. 2023. Generate What You Prefer: Reshaping Sequential Recommendation via Guided Diffusion. In *Proc. of NeurIPS*.

[53] Xiaoxin Ye, Chengkai Huang, Hongtao Huang, and Lina Yao. 2026. Gaussian Mixture Flow Matching with Domain Alignment for Multi-Domain Sequential Recommendation. In *Proc. of WWW*. 6159–6170.

[54] Yaowen Ye, Lianghao Xia, and Chao Huang. 2023. Graph Masked Autoencoder for Sequential Recommendation. In *Proc. of SIGIR*. 321–330.

[55] Fajie Yuan, Alexandros Karatzoglou, Ioannis Arapakis, Joemon M. Jose, and Xiangnan He. 2019. A Simple Convolutional Generative Network for Next Item Recommendation. In *Proc. of WSDM*. 582–590.

[56] Jujia Zhao, Wenjie Wang, Yiyan Xu, Teng Sun, Fuli Feng, and Tat-Seng Chua. 2024. Denoising Diffusion Recommender Model. In *Proc. of SIGIR*. 1370–1379.

[57] Yunqin Zhu, Chao Wang, Qi Zhang, and Hui Xiong. 2024. Graph Signal Diffusion Model for Collaborative Filtering. In *Proc. of SIGIR*. 1380–1390.

A Detailed Methodology

A.1 Target-Recovery Parameterization

For the linear path in Eq. (1), the conditional velocity is $\boldsymbol{x}_1 - \boldsymbol{x}_0$. MIRAGE parameterizes it by predicting the target state, as in Eq. (6). Substituting this parameterized velocity into the conditional flow-matching objective yields:

$$\begin{aligned}\mathcal{L}_{\text{CFM}} &= \mathbb{E}_{t,\boldsymbol{x}_0,\boldsymbol{x}_1,S_u} \left\| f_\theta(\boldsymbol{x}_t, t, S_u) - \boldsymbol{x}_0 - (\boldsymbol{x}_1 - \boldsymbol{x}_0) \right\|_2^2 \\ &= \mathbb{E}_{t,\boldsymbol{x}_0,\boldsymbol{x}_1,S_u} \left\| f_\theta(\boldsymbol{x}_t, t, S_u) - \boldsymbol{x}_1 \right\|_2^2 \\ &= \mathbb{E}_{t,\boldsymbol{x}_0,\boldsymbol{x}_1,S_u} \left\| \hat{\boldsymbol{x}}_1 - \boldsymbol{x}_1 \right\|_2^2 .\end{aligned} \tag{19}$$

Thus, target recovery is not an additional approximation to conditional flow matching. Under the selected parameterization, it is the same training objective expressed in target coordinates.

A.2 Alignment Derivations

A.2.1 Hard Alignment. For a target sequence and its next-item i^+, we define the squared Euclidean distance to a graph neighbor as

$$\phi_j(\boldsymbol{x}_t) = \left\| \boldsymbol{x}_t - \boldsymbol{e}_j \right\|_2^2, \quad j \in \mathcal{N}_K(i^+). \tag{20}$$

Assuming the minimizer j_t^* is unique, the continuity of the finitely many distance functions ϕ_j ensures that this specific neighbor remains the active anchor throughout a small continuous neighborhood around $\boldsymbol{x}_t$. Consequently, within this local region, the distance function and its gradient are well-defined as:

$$d_{\mathcal{G}}(\boldsymbol{x}_t, i^+) = \phi_{j_t^*}(\boldsymbol{x}_t), \qquad \nabla_{\boldsymbol{x}_t} d_{\mathcal{G}}(\boldsymbol{x}_t, i^+) = 2\left(\boldsymbol{x}_t - \boldsymbol{e}_{j_t^*}\right). \tag{21}$$

While nondifferentiable states technically arise when there are distance ties between multiple neighbors, the probability of encountering these boundaries in a continuous high-dimensional embedding space is theoretically negligible. Consequently, they do not impede stochastic gradient descent in practice. Crucially, this hard objective supplies a direct gradient toward one valid item embedding, rather than pointing toward a weighted centroid in the unsupported interstitial space.

A.2.2 Soft Alignment. To provide a differentiable alternative across the entire embedding space, we introduce a temperature parameter $\tau > 0$ and define the soft alignment using a smooth minimum over the squared Euclidean distances $\phi_j(\boldsymbol{x}_t)$. This formulation and its corresponding softmax weights are structured as:

$$\begin{aligned} d^{\text{soft}}_{\mathcal{G},\tau}(\boldsymbol{x}_t, i^+) &= -\tau \log \left[\frac{1}{K} \sum_{j \in \mathcal{N}_K(i^+)} \exp\left(-\frac{\phi_j(\boldsymbol{x}_t)}{\tau} \right) \right], \\ \alpha_{t,j} &= \frac{\exp(-\phi_j(\boldsymbol{x}_t)/\tau)}{\sum_{a \in \mathcal{N}_K(i^+)} \exp(-\phi_a(\boldsymbol{x}_t)/\tau)}. \end{aligned} \tag{22}$$

Differentiating this smooth minimum naturally produces the softmax weights $\alpha_{t,j}$ as coefficients. Since the constant scaling factor $1/K$ vanishes during differentiation, applying the chain rule directly yields the geometric gradient:

$$\begin{aligned} \nabla_{\boldsymbol{x}_t} d^{\text{soft}}_{\mathcal{G},\tau} &= \sum_{j \in \mathcal{N}_K(i^+)} \alpha_{t,j} \nabla_{\boldsymbol{x}_t} \phi_j(\boldsymbol{x}_t) \\ &= \sum_{j \in \mathcal{N}_K(i^+)} \alpha_{t,j}\, 2\left(\boldsymbol{x}_t - \boldsymbol{e}_j\right) \\ &= 2\left(\boldsymbol{x}_t - \sum_{j \in \mathcal{N}_K(i^+)} \alpha_{t,j} \boldsymbol{e}_j \right). \end{aligned} \tag{23}$$

This derivation demonstrates that the soft gradient continuously pulls the interpolant toward a convex combination of the neighbor embeddings, effectively acting as a unified topological centroid.

To establish the connection between the hard and soft alignment strategies, we examine the behavior as the temperature approaches 0. Assuming a closest neighbor j_t^*, its distance is smaller than any other neighbor j. By factoring out the dominant exponential term of this minimum distance, the relative distance gap $\Delta_j = \phi_j(\boldsymbol{x}_t) - \phi_{j_t^*}(\boldsymbol{x}_t)$ dictates the weight assigned to the active anchor:

$$\alpha_{t,j_t^*} = \left(1 + \sum_{j \neq j_t^*} \exp\left(-\frac{\Delta_j}{\tau} \right) \right)^{-1} \longrightarrow 1 \quad \text{as} \quad \tau \to 0. \tag{24}$$

Since every gap Δ_j is positive, all other weights $\alpha_{t,j \neq j_t^*}$ exponentially decay to zero. Consequently, as the temperature approaches zero, the soft formulation and its geometric gradient converge to their hard-alignment counterparts almost everywhere.

A.3 Time Modulation

A.3.1 Parabolic Time Modulation. To control the strength of the topological regularization along the flow path dynamically, we construct a general quadratic schedule $w(t) = at^2 + bt + c$. To ensure that this geometric intervention does not interfere with the prescribed source and target distributions, we impose strict boundary conditions $w(0) = w(1) = 0$. This yields $c = 0$ and $b = -a$, reducing the form to $w(t) = -at(1-t)$. Furthermore, to normalize the maximum intervention at the midpoint of the trajectory, we enforce $w(1/2) = 1$, which gives $a = -4$. Thus, the time modulation schedule is uniquely determined as:

$$w(t) = 4t(1-t). \tag{25}$$

This parabolic weight guarantees a symmetric profile that peaks at $t = 1/2$ and vanishes at both endpoints, concentrating the geometric rectification on the unconstrained path interior. This schedule mirrors the spatial uncertainty variance $\text{Var}(\boldsymbol{B}_t) = t(1-t)\boldsymbol{I}$ of a standard Brownian bridge, establishing a theoretical connection between our deterministic geometric modulation and stochastically constrained trajectories.

A.4 Gradient Dynamics

To understand how the topological loss reshapes the embedding space, we analyze the gradient dynamics during a local optimization step. Assuming a training state where the active anchor j_t^* remains locally constant and is distinct from the target item, we define the residual vector as $\boldsymbol{r} = \boldsymbol{x}_t - \boldsymbol{e}_{j_t^*}$. Applying a gradient step of size η

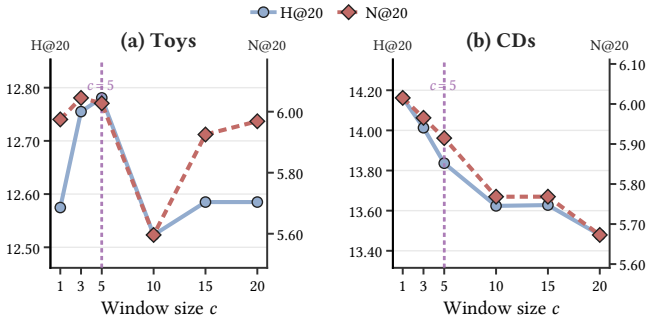

Figure 7: Effect of the co-occurrence window size c.

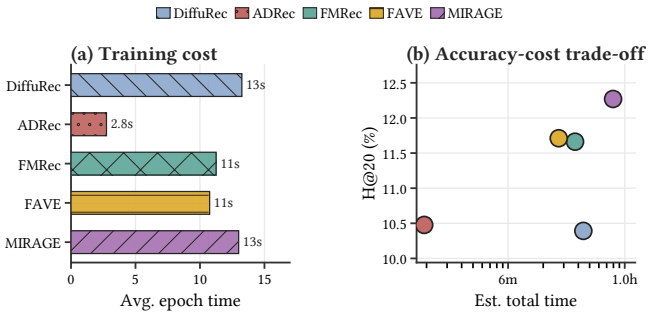

Figure 8: Training efficiency and performance trade-off. MIRAGE requires comparable per-epoch training time while establishing the optimal Pareto frontier in overall accuracy and total training cost.

updates the target and anchor embeddings according to:

$$\begin{aligned} \boldsymbol{e}'_{i^+} &= \boldsymbol{e}_{i^+} - 2\eta w(t) t \boldsymbol{r}, \\ \boldsymbol{e}'_{j_t^*} &= \boldsymbol{e}_{j_t^*} + 2\eta w(t) \boldsymbol{r}. \end{aligned} \tag{26}$$

Since the interpolation is defined as $\boldsymbol{x}_t = (1-t)\boldsymbol{x}_0 + t\boldsymbol{e}_{i^+}$, this update to the target embedding simultaneously shifts the intermediate state to $\boldsymbol{x}'_t = \boldsymbol{x}_t - 2\eta w(t) t^2 \boldsymbol{r}$. Consequently, the new residual becomes:

$$\boldsymbol{r}' = \boldsymbol{x}'_t - \boldsymbol{e}'_{j_t^*} = \left[1 - 2\eta w(t)(1+t^2)\right] \boldsymbol{r}. \tag{27}$$

A learning rate constrained within $0 < \eta w(t)(1+t^2) < 1$ ensures this local step decreases the residual norm. Even with additional gradients from the recovery, ranking, and history components during full training, the topological term enforces this local contraction toward valid semantic anchors. This dynamic accounts for the implicit geometric rectification.

A.5 Inference Consistency

To demonstrate the efficiency of single-step generation where $q = 1$, we evaluate the Euler integration over the full interval length $1 - \epsilon$. Initializing the integration with $\tilde{\boldsymbol{x}}_0 = \boldsymbol{x}_0$ at $t_0 = \epsilon$ and substituting the parameterized velocity yields:

$$\begin{aligned} \tilde{\boldsymbol{x}}_1 &= \boldsymbol{x}_0 + (1-\epsilon)\left[f_\theta(\boldsymbol{x}_0, \epsilon, S_u) - \boldsymbol{x}_0\right] \\ &= (1-\epsilon) f_\theta(\boldsymbol{x}_0, \epsilon, S_u) + \epsilon \boldsymbol{x}_0. \end{aligned} \tag{28}$$

Since the boundary margin ϵ is chosen to be negligibly small, this expression naturally reduces to the direct target prediction formulation $\hat{\boldsymbol{x}}_1$. Furthermore, because the co-occurrence graph is utilized exclusively within the topological loss during training, it remains decoupled from the learned vector field and inference equations, eliminating the need for graph lookups during deployment.

B Additional Experimental Details

B.1 Baseline Implementation Details

All evaluated baselines were strictly reproduced using their official open-source implementations and recommended protocols. Recent related models, such as FlowRec and DAE4Rec, are omitted because their official source codes are not publicly available.

B.2 Hyperparameter Sensitivity

We evaluate hyperparameter sensitivity by isolating individual variables. The optimal neighborhood size $K \in \{5, 10, 20, 40\}$ depends on data sparsity. Sparser domains like CDs and Toys benefit from broader structural support with $K = 40$, whereas the denser Beauty dataset peaks at $K = 5$. Furthermore, performance exhibits pronounced dataset-specific sensitivity to both the topology weight λ_{topo} and the ranking weight λ_{rank}. Toys, in particular, experiences degradation if these weights deviate from their respective optima of 0.2 and 0.5, underscoring the necessity of balancing geometric rectification with the contrastive ranking objective.

As illustrated in Figure 7, evaluating the co-occurrence window size $c \in \{1, 3, 5, 10, 15, 20\}$ reveals distinct geometric requirements across different data distributions. Dense catalogs like Toys achieve optimal performance at an intermediate window of $c = 5$, forming a pronounced inverted U-shaped trend. This shows that moderately larger windows capture broader semantic contexts to support hard alignment before eventually introducing overly smoothed correlations. Conversely, the sparse and sequential CDs dataset favors adjacent co-occurrences at $c = 1$. Extending the window for this sparse catalog rapidly introduces distant noisy edges, which dilute the weighted centroid of the soft alignment variant and consequently degrade the geometric guidance.

B.3 Graph Construction and Complexity

We construct the topology graph once using the training split exclusively. By concatenating the history and target for each training instance, we count ordered item pairs co-occurring within a dataset-specific context window ($c = 1$ for CDs and $c = 5$ otherwise). The stored neighborhood for every item comprises those with the highest positive co-occurrence frequencies, resolving ties deterministically to ensure reproducibility. Because this graph remains fixed during optimization, dynamic embedding updates never alter neighborhood memberships or leak future information.

Processing a mini-batch of size B with hard alignment requires retrieving up to K neighbor embeddings per target to compute their squared distances to the interpolated state. This introduces a distance computation cost of $O(BKd)$ and caches $O(|\mathcal{I}|K)$ indices. The soft alignment applies a normalized weighting over these distances, resulting in identical asymptotic complexity. Both strategies reuse the existing item table without introducing auxiliary encoders. Furthermore, we discard the cached graph and neighbor retrieval

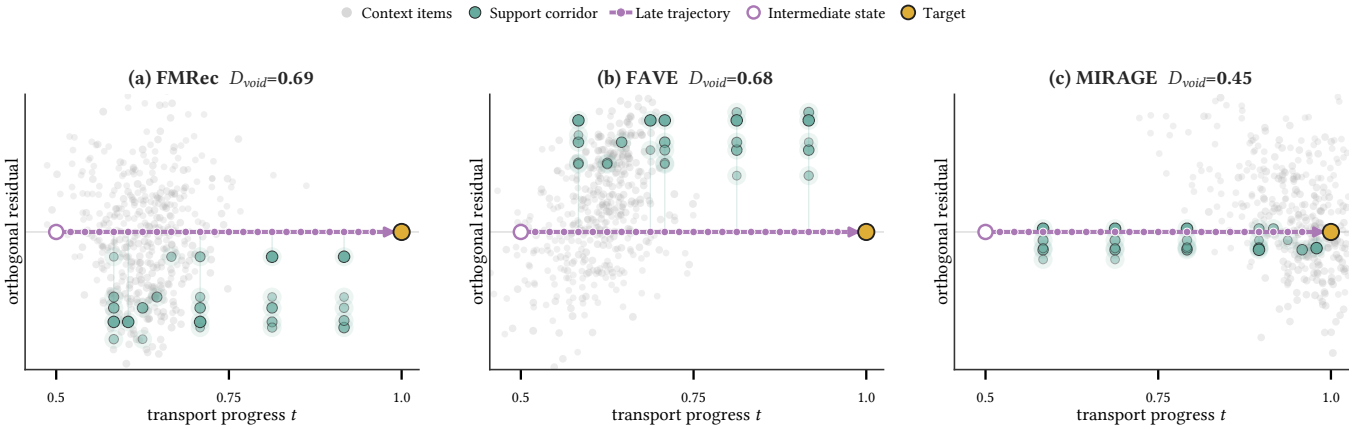

Figure 9: Target-local transport plane visualization. MIRAGE rectifies the embedding space to form a tight support corridor along the late generative trajectory, whereas baselines traverse the unsupported Euclidean void.

mechanisms after training, which preserves the baseline parameter count and ensures zero graph overhead during inference.

B.4 Training Efficiency

To address potential concerns regarding the computational overhead of topological regularization, Figure 8 compares the average epoch time and the overall accuracy-efficiency trade-off. MIRAGE requires 13 seconds per epoch, remaining comparable to standard generative baselines like FMRec and FAVE which require 11 seconds. The scatter plot confirms that MIRAGE achieves a favorable Pareto trade-off by yielding the highest performance for a similar overall training budget. This demonstrates that the geometric rectification process introduces negligible computational burden while delivering recommendation improvements.

B.5 Qualitative Trajectory Support Analysis

To qualitatively illustrate how MIRAGE reshapes the latent item geometry, Figure 9 visualizes the target-local transport plane during the late generation phase ($t \geq 0.5$). The horizontal axis represents the progress of the interpolant toward the target item, while the vertical axis measures the orthogonal residual deviation from this straight probability path. For sampled time bins along the trajectory, faint lines connect the continuous path to its nearest valid graph anchors in the learned embedding space.

Generative baselines like FMRec and FAVE project straight paths that traverse sparse continuous regions. Their nearest valid item representations exhibit large orthogonal residuals and uneven distributions, yielding long projection lines that indicate a lack of local semantic grounding. By contrast, MIRAGE reorganizes the underlying item geometry to envelop these straight generative trajectories. The nearest graph anchors tightly align with the trajectory and distribute along the generation progress. These aligned anchors form a continuous support corridor that provides dense structural support for the intermediate states toward the intended target semantics.

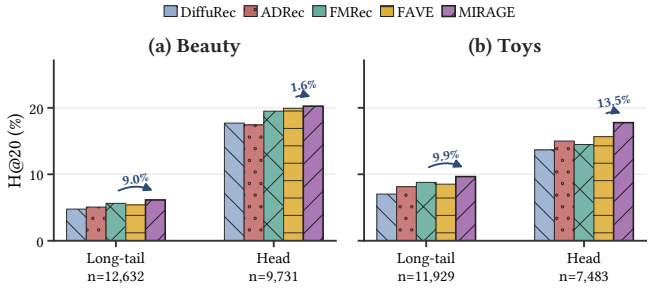

Figure 10: Additional long-tail H@20 results on Beauty and Toys. Buckets use training-set item frequency (n denotes the number of test targets in each group).

B.6 Additional Long-Tail Results

To evaluate long-tail performance, we partition the bottom 80% of the catalog based on training-set frequency as the long tail and the remaining 20% as the head. Test instances are assigned to these fixed subsets based on ground-truth targets to ensure fair, model-independent evaluations. On Beauty, MIRAGE improves H@20 over the strongest baseline by 9.0% for long-tail targets and 1.6% for head targets. Toys similarly achieves respective gains of 9.9% on the tail and 13.5% on the head. Alongside the main text results for Sports and CDs, these findings demonstrate that MIRAGE consistently enhances generative recommendations for sparsely observed items without sacrificing accuracy on popular targets.

Unlike other domains, Toys exhibits a gain of 13.5% on head targets alongside its 9.9% tail improvement. This concurrent improvement stems from the unique category characteristics of toys, where popular items share tightly coupled interaction sequences. In such densely correlated item spaces, our topological regularization effectively captures high-frequency structural paths without suffering from performance saturation, simultaneously boosting popular and sparsely observed recommendations.