SNR AS A SUPERVISORY COORDINATE: COARSE-TO-FINE LEARNING FOR DIFFUSION-BASED RECOMMENDATION

Anonymous authors
Paper under double-blind review

ABSTRACT

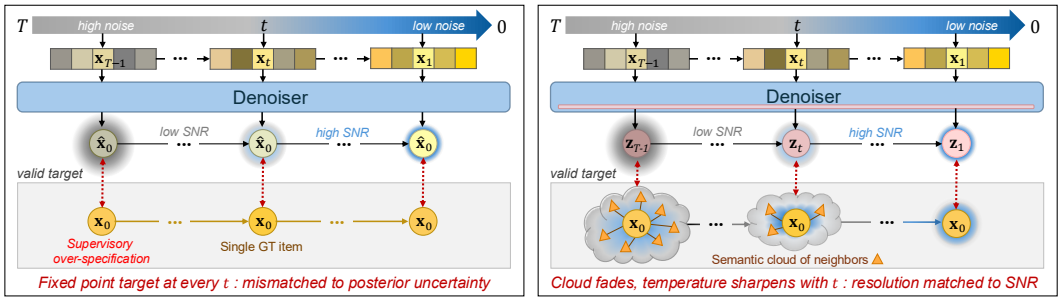
Diffusion models have emerged as a powerful paradigm for sequential recommendation (SR) due to their ability to model user preferences as complex, multi-modal distributions. However, existing frameworks typically rely on static, point-wise supervision throughout the denoising trajectory. Such constant-resolution objectives are misaligned with the time-varying information that the noisy state retains about the target. Specifically, enforcing a precise point target in high-noise regimes leads to *supervisory over-specification*, forcing the model to resolve distinctions that exceed the recoverable information. To address this, we propose **CloudDiff**, a diffusion framework for SR that adaptively modulates its supervisory granularity according to the signal-to-noise ratio (SNR). CloudDiff introduces two SNR-coupled mechanisms: (1) a *semantic cloud loss* that aligns the denoiser's intermediate representations with set-valued targets of semantic neighbors, with their influence gated by the noise level; and (2) a *log-SNR-coupled temperature schedule* that dynamically adjusts the sharpness of the contrastive distribution. We provide an information-theoretic foundation showing that both axes of CloudDiff are governed by a single coordinate, log-SNR, jointly motivating the proposed coarse-to-fine scheme. Extensive experiments on multiple benchmarks demonstrate that CloudDiff consistently outperforms state-of-the-art baselines, with detailed analyses revealing how SNR-adaptive supervision stabilizes the refinement trajectory.

1 INTRODUCTION

Diffusion models (Ho et al., 2020; Song et al.; Dhariwal & Nichol, 2021) have transformed generative modeling precisely because they embrace uncertainty: by learning to invert a noise process, they recover *distributions* rather than points, accommodating the multi-modal structure of real-world data (Rombach et al., 2022; Saharia et al., 2022; Yang et al., 2023a). In this context, sequential recommendation (SR) (Hidasi et al., 2015; Kang & McAuley, 2018) emerges as a natural application of this paradigm (Wei & Fang, 2025). User preferences are often latent and multi-faceted, while only a small portion of them is revealed through observed interactions. As a result, multiple items can plausibly satisfy a user's intent, making ambiguity modeling an important challenge in SR (Cen et al., 2020; Tan et al., 2021). However, existing diffusion-based recommenders (Li et al., 2023b; Yang et al., 2023b; Liu et al., 2024b) do not yet fully exploit this property. They supervise the denoising network with a point-wise objective anchored to a single target item, applied uniformly across every diffusion timestep, leaving the time-varying nature of the diffusion process largely unaddressed.

We argue that this is not a minor implementation detail but a fundamental misalignment between supervisory resolution and the information available at each step of the diffusion process. The information retained in the noisy state about the clean target, formally measured by the mutual information $I(\mathbf{x}_0; \mathbf{x}_t)$, decreases monotonically with the signal-to-noise ratio (SNR) of the diffusion timestep t (Kingma et al., 2021; Karras et al., 2022; Lin et al., 2024). This bounds the resolution at which the target can be recovered, and a static point-wise objective is therefore poorly matched to this time-varying structure (Choi et al., 2022; Wang et al., 2023a). When SNR is low, the posterior $q(\mathbf{x}_0 \mid \mathbf{x}_t)$ becomes diffuse, as the noisy state contains limited information about the target (Kim et al., 2025). In this regime, enforcing a precise point target may require distinctions that exceed the recoverable information, a phenomenon we refer to as *supervisory over-specification* (cf. Fig. 1),

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.



(a) Existing Diffusion-based Recommenders (b) CloudDiff (Ours)

Figure 1: Comparison between existing diffusion-based recommenders and CloudDiff.

which can increase training variance and lead to unstable optimization (Hang et al., 2023; Li et al., 2024). This mismatch is particularly relevant for recommendation, where multiple items may serve similar user intents (Fan et al., 2022; 2021). In such cases, forcing a single exact target during denoising can be overly restrictive.

To address this limitation, we propose **CloudDiff**, a diffusion framework for SR that adaptively modulates supervisory granularity based on SNR. CloudDiff couples two mechanisms that control supervisory support and predictive sharpness during denoising. First, an SNR-dependent set-valued neighborhood, instantiated using frozen language model embeddings (Harte et al., 2023; Liu et al., 2024a; Li et al., 2023a), modulates supervisory support. Second, a *log-SNR-coupled temperature schedule* adjusts the sharpness of the contrastive predictive distribution. As SNR increases, supervisory support contracts while the predictive distribution sharpens, enabling broader supervision at high noise and more focused supervision as the target becomes identifiable.

A key insight of our design is that these two mechanisms are not independent. Our theoretical analysis motivates log-SNR as a shared coordinate for adapting both the spatial support of the supervisory target and the sharpness of the contrastive distribution. This unification links supervisory resolution to the information structure of the diffusion process, yielding a coarse-to-fine schedule whose granularity adapts to the recoverable information at each timestep.

Our main contributions are summarized as follows:

- **Diagnosis of fixed-resolution supervision.** We characterize the mismatch between static point-target supervision and the SNR-dependent information available at each diffusion step, and identify *supervisory over-specification* as a key failure mode in high-noise regimes.
- **CloudDiff: SNR-coupled supervisory granularity.** We replace the single point target with a *semantic cloud* and couple the contrastive temperature to log-SNR, yielding a coarse-to-fine objective whose resolution adapts to the recoverable information at each timestep.
- **Information-theoretic foundation.** We motivate log-SNR as a shared coordinate for the two axes of CloudDiff, unifying two seemingly independent design choices under a common principle for SNR-dependent adaptation.
- **Empirical validation.** Across multiple benchmarks, CloudDiff consistently outperforms strong baselines. Ablation studies validate the contribution of each SNR-coupled mechanism, while further analyses show that the intended coarse-to-fine behavior emerges along denoising.

2 Preliminaries

2.1 Sequential Recommendation Task

Problem Formulation. The goal of sequential recommendation is to capture the dynamic preferences of users from their historical interactions. Formally, let $\mathcal{U}$ be the set of users and $\mathcal{V}$ be the set of all available items. Each user $u \in \mathcal{U}$ is associated with a chronological interaction sequence $\mathcal{H}_u = [v_1, v_2, \ldots, v_n]$, where $v \in \mathcal{V}$. Given $\mathcal{H}_u$, the task is to model the conditional probability of the next item (Kang & McAuley, 2018): $\hat{v} = \arg\max_{v \in \mathcal{V}} P_\theta(v \mid \mathcal{H}_u)$.

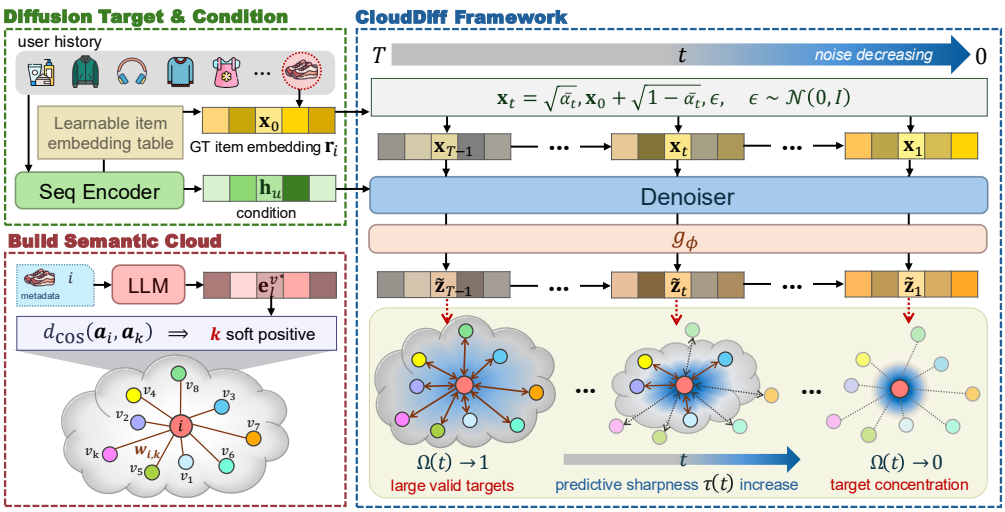


Figure 2: Overview of CloudDiff.

Sequence Encoding. To distill the user's evolving interest into a representation, we employ a sequence encoder $\Phi(\cdot)$. Each item $v \in \mathcal{H}_u$ is first transformed into its trainable embedding $\mathbf{r}_v \in \mathbb{R}^d$. These are then processed by Φ to yield the user preference vector $\mathbf{h}_u = \Phi(\mathbf{r}_{v_1}, \ldots, \mathbf{r}_{v_n}) \in \mathbb{R}^d$.

2.2 Diffusion-based Recommendation

Diffusion Model. The diffusion framework (Ho et al., 2020) operates by learning a reverse process to recover the original signal from data that has been progressively corrupted through a forward noise injection process. The *forward process* transforms a clean data point $\mathbf{x}_0$ into a sequence of increasingly noisy latents $\{\mathbf{x}_1, \ldots, \mathbf{x}_T\}$. For any timestep $t \in \{1, \ldots, T\}$, the latent state $\mathbf{x}_t$ can be sampled directly as $\mathbf{x}_t = \sqrt{\bar{\alpha}_t}\,\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\,\boldsymbol{\epsilon}$, where $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and $\bar{\alpha}_t$ is the noise schedule. The *reverse process* aims to recover $\mathbf{x}_0$ from a noisy state $\mathbf{x}_t$. In the $\mathbf{x}_0$-prediction formulation, a denoiser $f_\theta(\mathbf{x}_t, t, c)$ is trained to directly reconstruct the original signal $\mathbf{x}_0$ given a condition c. The training objective is to minimize the mean squared reconstruction error:

$$\mathcal{L}_{\text{diff}} = \mathbb{E}_{\mathbf{x}_0, t, c}\left[\|\mathbf{x}_0 - f_\theta(\mathbf{x}_t, t, c)\|^2\right]. \tag{1}$$

We apply Classifier-Free Guidance (CFG) (Ho & Salimans, 2022): $\hat{\mathbf{x}}_0 = (1+w) f_\theta(\mathbf{x}_t, t, c) - w f_\theta(\mathbf{x}_t, t, \emptyset)$, where $w \geq 1$ is the guidance scale and $\emptyset$ denotes the null condition.

Inference. For the recommendation task, we set the diffusion target as the trainable item embedding, $\mathbf{x}_0 = \mathbf{r}_i$, conditioned on $\mathbf{h}_u$. During inference, the model starts from pure noise $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and iteratively performs the reverse process to generate $\hat{\mathbf{x}}_0$. For efficient sampling, we adopt Denoising Diffusion Implicit Models (DDIM) (Song et al., 2021) to compute the transition from $\mathbf{x}_t$ to $\mathbf{x}_{t-1}$ deterministically:

$$\mathbf{x}_{t-1} = \sqrt{\bar{\alpha}_{t-1}}\,\hat{\mathbf{x}}_0 + \sqrt{1-\bar{\alpha}_{t-1}}\left(\frac{\mathbf{x}_t - \sqrt{\bar{\alpha}_t}\,\hat{\mathbf{x}}_0}{\sqrt{1-\bar{\alpha}_t}}\right). \tag{2}$$

Log-SNR Parametrization. Following Kingma et al. (2021), we adopt the log signal-to-noise ratio (log-SNR) parametrization throughout the paper:

$$\lambda(t) := \log \mathrm{SNR}(t) = \log \frac{\bar{\alpha}_t}{1-\bar{\alpha}_t}, \tag{3}$$

which serves as the central coordinate in our analysis.

3 Method

Figure 2 provides an overview of **CloudDiff**, a diffusion framework for sequential recommendation built on two SNR-coupled mechanisms: a set-valued supervisory target and an adaptive temperature schedule. We first present the framework design in §3.2, then provide its theoretical foundation in §3.3 through posterior-uncertainty and exponential-family arguments. Finally, we describe the training objective and inference procedure in §3.4.

3.1 SEMANTIC SPACE AND NOTATION

We build on the Variance Preserving (VP) diffusion formulation of §2 and extend it with a *semantic space*: for each item $v \in \mathcal{V}$, we obtain a frozen LLM embedding $\mathbf{a}_v \in \mathbb{R}^{d_\ell}$ from a pretrained language model applied to the item's textual metadata, with construction details provided in Appendix E.3. Following mean-centering (Ethayarajh, 2019) and ℓ_2-normalization, these embeddings lie on the unit sphere, where cosine distance $d_{\cos}(\mathbf{u}, \mathbf{v}) = 1 - \langle \mathbf{u}, \mathbf{v} \rangle$ directly reflects semantic similarity between items. The semantic space serves as a supervisory signal during training, and the LLM is not used at inference. We use semantic neighborhoods to provide a complementary prior beyond interaction signals, with further neighborhood analyses provided in Appendix G. For a training instance whose target item is indexed by i, we define its *semantic cloud* as

$$\mathcal{N}_i(K) = \{k \in \mathcal{V} \setminus \{i\} : \text{rank}\left(d_{\cos}(\mathbf{a}_i, \mathbf{a}_k)\right) \leq K\}. \tag{4}$$

We treat $\mathcal{N}_i(K)$ as a set of *soft positives*, with cloud members weighted by their proximity to item i:

$$w_{ik} = \frac{\exp\big(\langle \mathbf{a}_i, \mathbf{a}_k \rangle\big)}{\sum_{m \in \mathcal{N}_i(K)} \exp\big(\langle \mathbf{a}_i, \mathbf{a}_m \rangle\big)}, \qquad \sum_{k \in \mathcal{N}_i(K)} w_{ik} = 1. \tag{5}$$

3.2 THE CLOUDDIFF FRAMEWORK

3.2.1 DENOISER ARCHITECTURE

We follow the $\mathbf{x}_0$-prediction formulation of equation 1 with a simple two-layer MLP denoiser:

$$\mathbf{z}_t = \text{GELU}(W_1[\mathbf{x}_t; \mathbf{h}_u; \psi(t)]), \quad \hat{\mathbf{x}}_0 = W_2 \mathbf{z}_t \in \mathbb{R}^d \tag{6}$$

where $\psi(t)$ is a sinusoidal timestep embedding (Vaswani et al., 2017) and $[\cdot\,;\,\cdot\,;\,\cdot]$ denotes concatenation. Drawing on the insight that intermediate layers in diffusion models are well-suited for representation alignment (Yu et al., 2025), we apply the semantic cloud loss to the hidden representation $\mathbf{z}_t$. This architectural choice allows our semantic space to provide effective supervisory signals while W_2 remains focused on reconstruction. The CFG training and inference procedures follow §2.

3.2.2 LOG-SNR TEMPERATURE SCHEDULE WITH INTRINSIC CALIBRATION

Following the log-SNR parametrization of §2, we couple the contrastive temperature to $\lambda(t)$, placing supervisory sharpness on the same axis as the diffusion schedule. We parameterize:

$$\tau(t) \;=\; \tau_{\text{lo}} \cdot \kappa^{\,u(t)}, \qquad u(t) \;=\; \frac{\lambda_{\max} - \lambda(t)}{\lambda_{\max} - \lambda_{\min}} \;\in\; [0, 1], \tag{7}$$

where $\kappa = \tau_{\text{hi}}/\tau_{\text{lo}} \geq 1$ controls the dynamic range.
This yields a power-law coupling between temperature and SNR,

$$\tau(t) \;\propto\; \text{SNR}(t)^{-\eta}. \tag{8}$$

As denoising progresses and the signal becomes more informative, $\tau(t)$ decreases, producing a sharper contrastive distribution that emphasizes finer distinctions within the semantic cloud.

Intrinsic density calibration. We anchor the base temperature to the intrinsic scale of the semantic embedding space:

$$\tau_{\text{lo}} \;=\; \mathbb{E}_{i \in \mathcal{V}}\, \mathbb{E}_{k \in \mathcal{N}_i(K)} \left[d_{\cos}(\mathbf{a}_i, \mathbf{a}_k)\right], \tag{9}$$

which equals the mean intra-cloud cosine distance and therefore aligns the temperature scale with the intrinsic geometry of the embedding space. As a result, τ_{lo} automatically adapts to the characteristic spread of semantic neighbors, while the upper bound $\tau_{\text{hi}} = \kappa\, \tau_{\text{lo}}$ is determined by the scaling factor κ.

3.2.3 SNR-GATED CLOUD-TO-POINT ALIGNMENT

We align the denoiser's intermediate representation $\mathbf{z}_t$ with the LLM semantic space, treating the semantic cloud as a set-valued supervisory target. We project $\mathbf{z}_t$ via a lightweight projection head g_ϕ, $\tilde{\mathbf{z}}_t = \text{norm}(g_\phi(\mathbf{z}_t)) \in \mathbb{R}^{d_\ell}$, and measure cosine similarity between the denoising representation $\tilde{\mathbf{z}}_t$ of target item i and the LLM embedding of item j:

$$S_{ij}(t) \;=\; \langle \tilde{\mathbf{z}}_{i,t},\, \mathbf{a}_j \rangle. \tag{10}$$

Let $\mathrm{Neg}(i) = \mathcal{B} \setminus (\{i\} \cup \mathcal{N}_i(K))$ denote in-batch negatives, excluding cloud members to avoid false negatives. Using $\tau(t)$ from equation 7 and w_{ik} from equation 5, define

$$M_i(t) = \exp(S_{ii}/\tau(t)), \quad P_i(t) = \sum_{k \in \mathcal{N}_i(K)} w_{ik} \exp(S_{ik}/\tau(t)), \quad N_i(t) = \sum_{j \in \mathrm{Neg}(i)} \exp(S_{ij}/\tau(t)). \tag{11}$$

To interpolate between region-level semantic supervision and exact-target supervision, we introduce a noise-dependent fade function

$$\Omega(t) = (1 - \bar{\alpha}_t)^p, \qquad p \geq \tfrac{1}{2}, \tag{12}$$

which controls the contribution of the semantic neighborhood at timestep t. At high noise, $\Omega(t)$ is large and the semantic cloud contributes strongly, while its effect gradually vanishes as denoising progresses. The objective is

$$\mathcal{L}_{\mathrm{SC}}(t) \;=\; -\log \frac{M_i(t) + \Omega(t) \cdot P_i(t)}{M_i(t) + \Omega(t) \cdot P_i(t) + N_i(t)}. \tag{13}$$

The numerator therefore acts as a set-valued positive whose support transitions from a broad semantic region at high noise to the exact target at low noise, yielding a coarse-to-fine supervisory structure.

3.3 Information-Theoretic Analysis

We revisit the design of §3.2 from an information-theoretic perspective. As the diffusion process progressively corrupts $\mathbf{x}_0$, the information retained in $\mathbf{x}_t$ decreases with the SNR, limiting the granularity of the target that can be reliably recovered at each timestep. We formalize this intuition through three results, with full proofs deferred to Appendix B.

Mutual information decay. Because $\mathbf{x}_t \mid \mathbf{x}_0$ forms a Gaussian channel (Kingma et al., 2021), the mutual information is upper bounded by the Gaussian channel capacity (Cover & Thomas, 2006). Let $\mathbf{x}_0 \sim p_{\text{data}}$ be the distribution of item embeddings with covariance Σ_0. Then

$$I(\mathbf{x}_0; \mathbf{x}_t) \;\leq\; \tfrac{1}{2} \log \det\!\big(\mathbf{I} + \mathrm{SNR}(t)\, \Sigma_0\big), \tag{14}$$

with equality when $\mathbf{x}_0$ is Gaussian (Cover & Thomas, 2006). Under standard noise schedules, $\mathrm{SNR}(t)$ decreases monotonically in t, leading to a reduction in the information that $\mathbf{x}_t$ retains about $\mathbf{x}_0$. In the low-SNR regime ($\bar{\alpha}_t \to 0$), the mutual information approaches zero, indicating that $\mathbf{x}_t$ contains limited information about the target. The interplay between the mutual information equation 14 and the posterior MSE in Proposition 1 is captured by the I-MMSE correspondence (Guo et al., 2005).

Proposition 1 (Posterior uncertainty upper bound). *Let $\mathbf{x}_0$ be any random vector with finite second moments and covariance $\Sigma_0 \succ 0$. Then the posterior mean-squared error satisfies*

$$\mathcal{E}(t) \;:=\; \mathbb{E}\,\|\mathbf{x}_0 - \mathbb{E}[\mathbf{x}_0 \mid \mathbf{x}_t]\|^2 \;\leq\; \mathrm{tr}\Big[\big(\Sigma_0^{-1} + \mathrm{SNR}(t)\,\mathbf{I}\big)^{-1}\Big] \;=:\; V(t), \tag{15}$$

with equality when $\mathbf{x}_0$ is Gaussian. Moreover, $V(t)$ is monotonically non-decreasing in t, with $V(t) \to 0$ as $\bar{\alpha}_t \to 1$ and $V(t) \to \mathrm{tr}\,\Sigma_0$ as $\bar{\alpha}_t \to 0$.

Implication. The bound equation 15 depends on $\mathbf{x}_0$ only through its covariance. As the SNR decreases, $V(t)$ increases toward the prior variance $\mathrm{tr}\,\Sigma_0$, indicating that $\mathbf{x}_t$ retains progressively less information about $\mathbf{x}_0$ at high noise. In this regime, point-level supervision induces *supervisory over-specification*: the model is asked to resolve distinctions beyond what the posterior can support. The set-valued target $\{i\} \cup \mathcal{N}_i(K)$ relaxes this demand by admitting semantically proximate items as valid supervision. A discrete-identity counterpart of this argument, formulated as a Fano-type lower bound on item-level error within a semantic neighborhood, is provided in Appendix B.4.

Proposition 2 (Posterior-deviation scaling). *Assume $\bar{\alpha}_t \geq \bar{\alpha}_{\min} > 0$. Then*

$$\mathcal{E}(t) \;\leq\; c_2\,(1 - \bar{\alpha}_t), \qquad \sqrt{\mathcal{E}(t)} \;\leq\; \sqrt{c_2}\,\sqrt{1 - \bar{\alpha}_t}, \tag{16}$$

with $c_2 = d/\bar{\alpha}_{\min}$.

Implication. The $\sqrt{1 - \bar{\alpha}_t}$ scaling in equation 16 prescribes a low-noise compatibility criterion: the fade function should vanish at least as fast as the posterior deviation as $\bar{\alpha}_t \to 1$, so that neighborhood supervision does not persist beyond the regime in which it is informative. The function $\Omega(t) = (1 - \bar{\alpha}_t)^p$ adopted in §3.2 satisfies this for any $p \geq 1/2$, and we fix $p = 0.5$ across datasets within this admissible range.

Proposition 3 (Log-temperature as a natural local coordinate). *Fix a candidate set $\mathcal{C}$ with deterministic scores $\{S_{ij}\}_{j\in\mathcal{C}}$ and define $p_\tau(j) \propto \exp(S_{ij}/\tau)$. With $\eta = 1/\tau$, the family $\{p_\tau\}_{\tau>0}$ forms a one-parameter exponential family (Wainwright & Jordan, 2008) in the natural parameter η, with Fisher information $\mathcal{I}(\eta) = \mathrm{Var}_{p_\tau}[S_{i,\cdot}]$. Moreover, for τ_2 in a neighborhood of τ_1, the local KL expansion, expressed in terms of $\log\tau$, is*

$$\mathrm{KL}(p_{\tau_1} \,\|\, p_{\tau_2}) = \tfrac{1}{2}\,\tau_1^{-2}\,\mathrm{Var}_{p_{\tau_1}}[S_{i,\cdot}](\log\tau_2 - \log\tau_1)^2 + \mathcal{O}\big((\log\tau_2 - \log\tau_1)^3\big)\,. \tag{17}$$

Implication. The Fisher-information geometry of the softmax family makes $\log\tau$ a natural local coordinate for controlling changes in the contrastive distribution. Since diffusion noise is also parameterized by log-SNR, coupling $\log\tau$ linearly to $\lambda(t)$ aligns the evolution of predictive sharpness with the diffusion trajectory. This geometric alignment motivates the form of $\tau(t)$ in §3.2.

Connection to the design. The three results jointly motivate the two SNR-coupled mechanisms of CloudDiff. Proposition 1 supports the use of set-valued supervision under high uncertainty, while Proposition 2 characterizes how the neighborhood contribution should contract as denoising progresses. Proposition 3 further shows that log-SNR provides a geometrically natural coordinate for controlling the sharpness of the contrastive distribution. Taken together, these results motivate a coarse-to-fine supervisory scheme in which both the support of the target and the sharpness of the predictive distribution evolve along the same log-SNR axis. We provide complementary empirical validation of these theoretical implications in Appendix F.

3.4 Training Objective and Inference

The full training objective of CloudDiff is

$$\mathcal{L} = \mathcal{L}_{\text{diff}} + \lambda_{\text{SC}}\,\mathcal{L}_{\text{SC}}. \tag{18}$$

The semantic cloud objective generalizes the standard InfoNCE loss (van den Oord et al., 2018) by replacing the single positive with a set-valued, SNR-gated target. The numerator of equation 13 aggregates the ground-truth and its semantic neighborhood into a *set-valued positive*, while the denominator normalizes over all candidates. As $\Omega(t) \to 0$, equation 13 recovers the standard InfoNCE with a single positive, whereas when $\Omega(t) \to 1$, all neighbors contribute to the positive signal proportional to their semantic weights. Together with the temperature schedule $\tau(t)$, which sharpens the contrastive distribution as denoising progresses, this structure distributes gradient mass across the semantic neighborhood, mitigating over-commitment to any single candidate in high-noise regimes. Its relation to standard contrastive objectives and limiting cases is detailed in Appendix C.

Inference. We follow the standard diffusion-based sampling procedure described in §2. Given a user history $\mathcal{H}_u$, we encode it into $\mathbf{h}_u = \Phi(\mathcal{H}_u)$ and condition the iterative reverse process on $\mathbf{h}_u$ to obtain the predicted item embedding $\hat{\mathbf{x}}_0$. This embedding is then used to score all candidate items via inner product, $y_v = \hat{\mathbf{x}}_0^\top \mathbf{r}_v$, and the top-$k$ items are returned as recommendations.

4 Experiments

In this section, we aim to answer the following questions: **(Q1)** Does CloudDiff improve recommendation performance over strong baselines? **(Q2)** How do the two SNR-coupled mechanisms contribute to the performance? **(Q3)** Are the theoretically motivated design choices empirically justified? **(Q4)** Does the proposed design induce the intended coarse-to-fine behavior along the denoising trajectory?

4.1 Experimental Setting

We evaluate on four widely used benchmark datasets: three Amazon review (Ni et al., 2019) domains (*Beauty*, *Toys*, *Sports*) and *Yelp*[1]. Following prior works (Yang et al., 2023b; Liu et al., 2024b; Hu et al., 2025), we adopt a user-wise chronological split with a ratio of 8:1:1 and a full-ranking protocol, reporting Hit Ratio (HR@K) and NDCG@K for $K \in \{5, 10, 20\}$. We compare against strong sequential (GRU4Rec (Hidasi et al., 2015), SASRec (Kang & McAuley, 2018), BERT4Rec (Sun

[1] `https://www.yelp.com/dataset`

Table 1: Overall performance comparison on four benchmark datasets, averaged over five seeds. **Bold** and underline indicate the best and second-best, respectively.

Method	Metric	Beauty		Toys		Sports		Yelp	
		NDCG	HR	NDCG	HR	NDCG	HR	NDCG	HR
GRU4Rec	@5	0.0187	0.0231	0.0121	0.0158	0.0055	0.0072	0.0025	0.0039
	@10	0.0210	0.0301	0.0129	0.0184	0.0060	0.0089	0.0030	0.0055
	@20	0.0225	0.0362	0.0143	0.0238	0.0070	0.0126	0.0038	0.0083
SASRec	@5	0.0231	0.0302	0.0237	0.0297	0.0105	0.0142	0.0063	0.0081
	@10	0.0262	0.0385	0.0249	0.0325	0.0124	0.0189	0.0076	0.0116
	@20	0.0289	0.0497	0.0261	0.0367	0.0131	0.0242	0.0095	0.0179
BERT4Rec	@5	0.0095	0.0157	0.0084	0.0118	0.0046	0.0074	0.0044	0.0072
	@10	0.0119	0.0233	0.0098	0.0165	0.0059	0.0114	0.0061	0.0125
	@20	0.0150	0.0355	0.0110	0.0209	0.0072	0.0167	0.0076	0.0183
ICLRec	@5	0.0267	0.0385	0.0222	0.0332	0.0101	0.0140	0.0067	0.0097
	@10	0.0322	0.0427	0.0265	0.0411	0.0130	0.0171	0.0084	0.0138
	@20	0.0384	0.0527	0.0308	0.0471	0.0160	0.0232	0.0103	0.0192
TIGER	@5	0.0189	0.0238	0.0130	0.0159	0.0042	0.0059	0.0015	0.0024
	@10	0.0210	0.0305	0.0146	0.0209	0.0052	0.0089	0.0019	0.0036
	@20	0.0242	0.0431	0.0165	0.0283	0.0065	0.0142	0.0027	0.0065
LLM-ESR	@5	0.0122	0.0193	0.0144	0.0225	0.0085	0.0136	0.0079	0.0106
	@10	0.0162	0.0334	0.0192	0.0339	0.0117	0.0180	0.0091	0.0129
	@20	0.0219	0.0529	0.0248	0.0448	0.0158	0.0236	0.0115	0.0203
DiffuRec	@5	0.0199	0.0247	0.0188	0.0231	0.0055	0.0073	0.0071	0.0092
	@10	0.0214	0.0291	0.0203	0.0280	0.0061	0.0091	0.0080	0.0117
	@20	0.0228	0.0349	0.0217	0.0337	0.0069	0.0122	0.0086	0.0143
DreamRec	@5	0.0294	0.0368	0.0276	0.0347	0.0100	0.0122	0.0059	0.0079
	@10	0.0318	0.0444	0.0288	0.0386	0.0105	0.0138	0.0066	0.0100
	@20	0.0332	0.0497	0.0297	0.0422	0.0111	0.0161	0.0073	0.0126
PreferDiff	@5	0.0310	0.0382	0.0320	0.0401	0.0114	0.0144	0.0060	0.0083
	@10	0.0334	0.0456	0.0332	0.0429	0.0125	0.0180	0.0068	0.0106
	@20	0.0344	0.0498	0.0345	0.0453	0.0134	0.0215	0.0076	0.0140
CloudDiff	@5	**0.0351**	**0.0450**	**0.0325**	**0.0414**	**0.0139**	**0.0190**	**0.0101**	**0.0147**
	@10	**0.0384**	**0.0560**	**0.0347**	**0.0484**	**0.0158**	**0.0249**	**0.0118**	**0.0209**
	@20	**0.0423**	**0.0713**	**0.0370**	**0.0575**	**0.0176**	**0.0321**	**0.0134**	**0.0269**

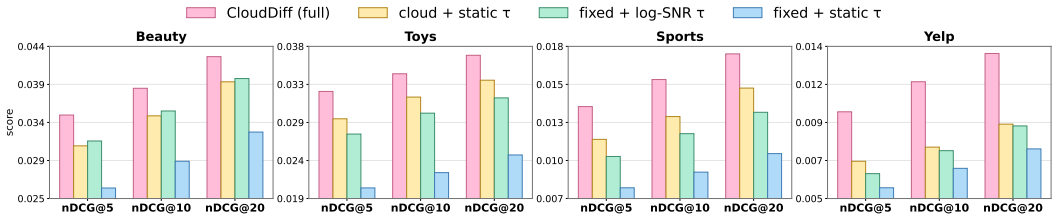


Figure 3: Ablation study of CloudDiff. Results are reported in terms of NDCG.

et al., 2019), ICLRec (Chen et al., 2022)), semantic generative (TIGER (Rajput et al., 2023), LLM-ESR (Liu et al., 2024a)), and diffusion-based (DiffuRec (Li et al., 2023b), DreamRec (Yang et al., 2023b), PreferDiff (Liu et al., 2024b)) recommenders. Additional details on the datasets, baselines, implementation, and HR results are provided in Appendix E.1–E.4 and Appendix I, respectively. Additional analyses of neighborhood construction, neighborhood corruption, and semantic priors are provided in Appendix G.

4.2 Main Results (Q1)

Table 1 reports overall performance across diverse strong baselines. CloudDiff achieves the best results on all datasets and metrics. Among diffusion baselines, PreferDiff is the strongest, likely because its BPR-based objective injects ranking signals directly into training. CloudDiff, in contrast, performs supervision entirely in the embedding space without an explicit ranking objective, yet still outperforms PreferDiff across all datasets, with particularly large gains on top-ranked metricss.

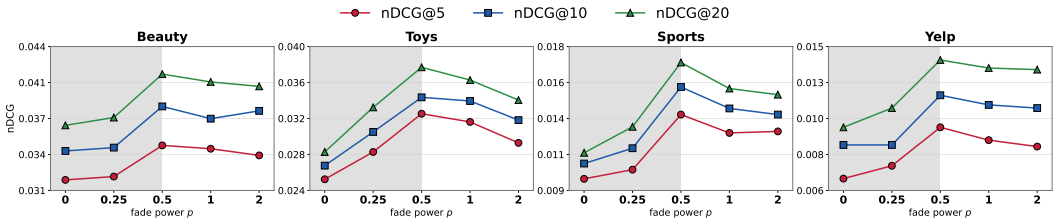


Figure 4: Sensitivity analysis of fade power p. Results are reported in terms of NDCG.

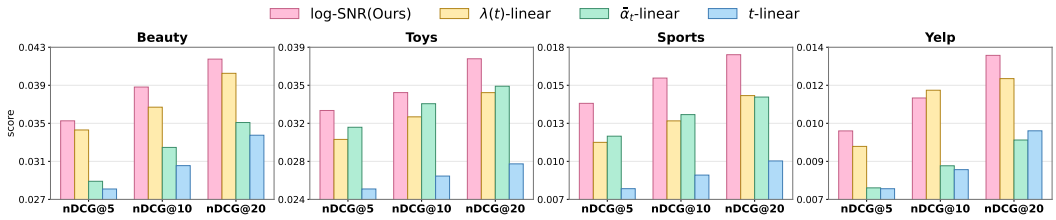


Figure 5: Comparison of temperature schedule variants. Results are reported in terms of NDCG.

This suggests that aligning supervisory resolution to the information available at each diffusion step improves the quality of the denoising signal and helps the model learn fine-grained ranking structure.

The gains over the second-best baseline are most pronounced on *Sports* and *Yelp*, which are relatively sparse and large-scale datasets. On *Yelp*, CloudDiff and LLM-ESR are the only methods that achieve competitive performance, both leveraging expressive representations from LLMs. This suggests that under high sparsity, semantic representations provide a useful prior to compensate for limited interaction signals. While LLM-ESR uses LLM embeddings for embedding table initialization, CloudDiff further exploits them at the supervisory level by treating semantic neighbors as soft targets. This encourages representations to be learned over semantic regions rather than isolated point targets.

4.3 Ablation Study of SNR-Coupled Mechanisms (Q2)

To isolate the contribution of each SNR-coupled mechanism, we compare the full model against three ablated variants: **(i)** *cloud + static* τ, which keeps the semantic cloud loss but uses a fixed temperature τ_{lo}; **(ii)** *fixed + log-SNR* τ, which keeps the log-SNR-coupled temperature but reverts to point-wise supervision; and **(iii)** *fixed + static* τ, which uses point-wise supervision with a fixed temperature. Results are reported in Figure 3.

The benefit of the semantic cloud is not just from using set-valued targets, but depends on how their contribution evolves along the diffusion trajectory. In particular, performance degrades under a fixed temperature even when the semantic cloud is retained, showing the importance of adjusting supervisory sharpness. As the diffusion state becomes more informative, supervision should progressively focus on the target by sharpening the contribution of neighbors. These results suggest that both the support and sharpness of supervision must be adapted. Modulating only one while keeping the other fixed leads to a mismatch with the time-varying information in the denoising process.

4.4 Empirical Validation of the Theoretical Design Choices (Q3)

We empirically validate the two design choices motivated by our theoretical analysis: the fade exponent $p \geq 1/2$ and the log-SNR coupling of the temperature schedule.

Fade exponent p**.** Proposition 2 prescribes a low-noise compatibility criterion: the fade function $\Omega(t) = (1 - \bar{\alpha}_t)^p$ should decay at least as fast as $\sqrt{1 - \bar{\alpha}_t}$, which corresponds to $p \geq 1/2$. We evaluate $p \in \{0, 0.25, 0.5, 1, 2\}$ across datasets. As shown in Figure 4, performance collapses for $p < 1/2$ and stabilizes for $p \geq 1/2$, with the optimum consistently at $p = 0.5$. The transition point empirically coincides with the theoretical lower bound, indicating that insufficient decay of the neighborhood contribution interferes with fine-grained supervision precisely in the regime where the theory predicts incompatibility.

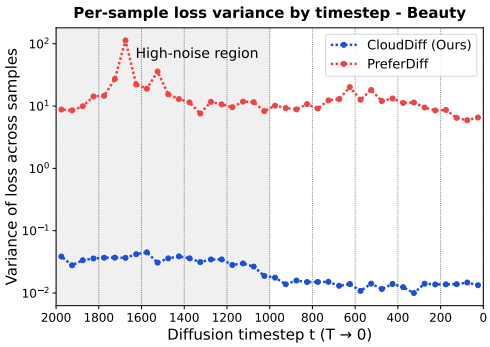


Figure 6: Timestep-wise Loss Variance.

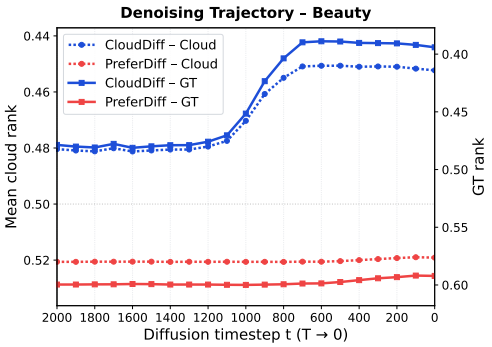


Figure 7: Denoising Trajectory Analysis.

Temperature schedule. We compare the proposed log-SNR-coupled temperature against three alternatives: a timestep t-linear schedule, an $\bar{\alpha}_t$-linear schedule, and a $\lambda(t)$-linear schedule. As shown in Figure 5, SNR-based schedules generally outperform the others, and the log-SNR-coupled schedule achieves the best performance on three of four datasets while remaining competitive on Yelp. This pattern indicates that the benefit arises not from varying the temperature alone, but from aligning it with the SNR-driven information structure of the diffusion process. The advantage of the log-SNR parameterization is consistent with Proposition 3: it provides a coordinate on which changes in temperature correspond to comparable local divergences in the contrastive distribution. Additional details are provided in Appendix E.5.

4.5 Analysis of Training Dynamics and Denoising Trajectory (Q4)

We compare CloudDiff against PreferDiff, the strongest diffusion-based baseline in our experiments.

Stability of the training signal. Figure 6 shows the per-sample loss variance across timesteps, with details provided in Appendix E.5. PreferDiff exhibits high variance across the entire denoising trajectory, with particularly pronounced peaks in the high-noise region. This is consistent with supervisory over-specification, where point-wise supervision induces inconsistent optimization behavior across samples. In contrast, CloudDiff shows a smooth decrease in variance from high-noise to low-noise regimes, indicating stable training dynamics across all timesteps. This suggests that the semantic cloud provides a more consistent supervisory signal when the target is not yet identifiable.

Emergence of coarse-to-fine behavior. Figure 7 tracks the rank of the ground-truth item and the mean rank of its semantic neighborhood. For CloudDiff, the two ranks are similar at high noise but diverge during denoising, with the ground-truth clearly outperforming the cloud at low noise. This indicates a transition from broad semantic supervision to exact-target discrimination as SNR increases. In contrast, PreferDiff shows nearly flat trajectories, suggesting that it neither captures meaningful semantic neighborhoods nor improves discrimination toward the ground-truth item. This result highlights the necessity of adapting supervisory granularity along the denoising process to achieve coarse-to-fine refinement in diffusion-based recommenders.

5 Conclusion

We introduced **CloudDiff**, a diffusion framework for sequential recommendation that addresses the mismatch between static point-wise supervision and the time-varying information structure of the diffusion process. Rather than treating the supervisory target as fixed throughout denoising, CloudDiff adapts both the support and sharpness of supervision according to SNR, yielding a coarse-to-fine refinement process from broad semantic regions to exact-target discrimination. Our theoretical analysis connects these two design axes through a shared log-SNR coordinate, motivated by the contraction of posterior uncertainty and the geometry of the contrastive distribution. Empirically, CloudDiff consistently outperforms strong diffusion-based and non-diffusion baselines, while diagnostic analyses confirm the emergence of the intended coarse-to-fine behavior throughout the denoising process. We view SNR-coupled supervisory granularity as a promising direction for extending coarse-to-fine supervision to diffusion-based generative modeling beyond recommendation.

AI USE STATEMENT

In this work, we used generative AI tools to assist with code implementation and debugging. We also used generative AI tools for language editing and polishing to improve the clarity and readability of the manuscript. All AI-assisted code was reviewed and verified by the authors, and all AI-assisted edits were manually reviewed. We take responsibility for the final content of this work, including text and code produced with the aid of generative AI.

ETHICS STATEMENT

This work uses publicly available benchmark datasets and does not involve human-subject experiments or the collection of new personal data. As with recommender systems more broadly, deployment may be affected by biases present in historical interaction data and pretrained representations. The proposed method is intended for research purposes, and appropriate evaluation of fairness, privacy, and potential biases should be conducted before deployment in real-world applications.

REPRODUCIBILITY STATEMENT

We provide detailed descriptions of the datasets, preprocessing, baselines, implementation, and hyperparameter settings in Appendix E.1–E.4. Complete derivations and proofs of the theoretical results are provided in Appendix B. Additional robustness analyses, including alternative neighborhood constructions and related analyses, are provided in Appendices G and F. The implementation code will be made publicly available upon acceptance.

REFERENCES

Rajendra Bhatia. *Matrix Analysis*. Graduate Texts in Mathematics. Springer, 1997.

Yukuo Cen, Jianwei Zhang, Xu Zou, Chang Zhou, Hongxia Yang, and Jie Tang. Controllable multi-interest framework for recommendation. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 2942–2951, 2020.

Jialei Chen, Yuanbo Xu, and Yiheng Jiang. Unlocking the power of diffusion models in sequential recommendation: A simple and effective approach. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pp. 155–166, 2025.

Yongjun Chen, Zhiwei Liu, Jia Li, Julian McAuley, and Caiming Xiong. Intent contrastive learning for sequential recommendation. In *Proceedings of the ACM web conference 2022*, pp. 2172–2182, 2022.

Jooyoung Choi, Jungbeom Lee, Chaehun Shin, Sungwon Kim, Hyunwoo Kim, and Sungroh Yoon. Perception prioritized training of diffusion models. In *CVPR*, 2022.

Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley-Interscience, 2nd edition, 2006.

Katherine Crowson, Stefan Andreas Baumann, Alex Birch, Tanishq Mathew Abraham, Daniel Z Kaplan, and Enrico Shippole. Scalable high-resolution pixel-space image synthesis with hourglass diffusion transformers. In *Forty-first International Conference on Machine Learning*, 2024.

Ziqiang Cui, Yunpeng Weng, Xing Tang, Xiaokun Zhang, Shiwei Li, Peiyang Liu, Bowei He, Dugang Liu, Weihong Luo, Chen Ma, et al. Semantic retrieval augmented contrastive learning for sequential recommendation. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*, 2024.

Kawin Ethayarajh. How contextual are contextualized word representations? comparing the geometry of bert, elmo, and gpt-2 embeddings. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*, pp. 55–65, 2019.

Ziwei Fan, Zhiwei Liu, Lei Zheng, Shen Wang, and Philip S Yu. Modeling sequences as distributions with uncertainty for sequential recommendation. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management (CIKM)*, 2021.

Ziwei Fan, Zhiwei Liu, Yu Wang, Alice Wang, Zahra Nazari, Lei Zheng, Hao Peng, and Philip S Yu. Sequential recommendation via stochastic self-attention. In *Proceedings of the ACM Web Conference (WWW)*, pp. 2036–2047, 2022.

Dongning Guo, Shlomo Shamai, and Sergio Verdú. Mutual information and minimum mean-square error in Gaussian channels. *IEEE Transactions on Information Theory*, 51(4):1261–1282, 2005.

Tiankai Hang, Shuyang Gu, Chen Li, Jianmin Bao, Dong Chen, Han Hu, Xin Geng, and Baining Guo. Efficient diffusion training via min-snr weighting strategy. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 7441–7451, 2023.

Tiankai Hang, Shuyang Gu, Jianmin Bao, Fangyun Wei, Dong Chen, Xin Geng, and Baining Guo. Improved noise schedule for diffusion training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4796–4806, 2025.

Jesse Harte, Wouter Zorgdrager, Panos Louridas, Asterios Katsifodimos, Dietmar Jannach, and Marios Fragkoulis. Leveraging large language models for sequential recommendation. In *Proceedings of the 17th ACM Conference on Recommender Systems*, pp. 1096–1102, 2023.

Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. Session-based recommendations with recurrent neural networks. *arXiv preprint arXiv:1511.06939*, 2015.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Guoqing Hu, An Zhang, Shuchang Liu, Wenyu Mao, Jiancan Wu, Xun Yang, Xiang Li, Lantao Hu, Han Li, Kun Gai, and Xiang Wang. Fading to grow: Growing preference ratios via preference fading discrete diffusion for recommendation. *NeurIPS*, 2025.

Wang-Cheng Kang and Julian McAuley. Self-attentive sequential recommendation. In *2018 IEEE international conference on data mining (ICDM)*, pp. 197–206. IEEE, 2018.

Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *Advances in neural information processing systems*, 35:26565–26577, 2022.

Steven M. Kay. *Fundamentals of Statistical Signal Processing: Estimation Theory*. Prentice Hall, 1993.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Jin-Young Kim, Hyojun Go, Soonwoo Kwon, and Hyun-Gyoon Kim. Denoising task difficulty-based curriculum for training diffusion models. In *The Thirteenth International Conference on Learning Representations*, 2025.

Diederik Kingma, Tim Salimans, Ben Poole, and Jonathan Ho. Variational diffusion models. *Advances in neural information processing systems*, 34:21696–21707, 2021.

Walid Krichene and Steffen Rendle. On sampled metrics for item recommendation. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 1748–1757, 2020.

Sean Lee, Aamir Shakir, Darius Koenig, and Julius Lipp. Open source strikes bread - new fluffy embeddings model, 2024. URL https://www.mixedbread.ai/blog/mxbai-embed-large-v1.

Xingjian Leng, Jaskirat Singh, Yunzhong Hou, Zhenchang Xing, Saining Xie, and Liang Zheng. Repa-e: Unlocking vae for end-to-end tuning of latent diffusion transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 18262–18272, 2025.

Omer Levy and Yoav Goldberg. Neural word embedding as implicit matrix factorization. *Advances in neural information processing systems*, 27, 2014.

Jiacheng Li, Ming Wang, Jin Li, Jinmiao Fu, Xin Shen, Jingbo Shang, and Julian McAuley. Text is all you need: Learning language representations for sequential recommendation. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 1258–1267, 2023a.

Wuchao Li, Rui Huang, Haijun Zhao, Chi Liu, Kai Zheng, Qi Liu, Na Mou, Guorui Zhou, Defu Lian, Yang Song, et al. Dimerec: a unified framework for enhanced sequential recommendation via generative diffusion models. In *Proceedings of the Eighteenth ACM International Conference on Web Search and Data Mining*, pp. 726–734, 2025.

Xianming Li and Jing Li. Angle-optimized text embeddings. *arXiv preprint arXiv:2309.12871*, 2023.

Yiheng Li, Heyang Jiang, Akio Kodaira, Masayoshi Tomizuka, Kurt Keutzer, and Chenfeng Xu. Immiscible diffusion: Accelerating diffusion training with noise assignment. *Advances in neural information processing systems*, 37:90198–90225, 2024.

Zihao Li, Aixin Sun, and Chenliang Li. Diffurec: A diffusion model for sequential recommendation. *ACM Transactions on Information Systems*, 42(3):1–28, 2023b.

Shanchuan Lin, Bingchen Liu, Jiashi Li, and Xiao Yang. Common diffusion noise schedules and sample steps are flawed. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pp. 5404–5411, 2024.

Qidong Liu, Xian Wu, Yejing Wang, Zijian Zhang, Feng Tian, Yefeng Zheng, and Xiangyu Zhao. Llm-esr: Large language models enhancement for long-tailed sequential recommendation. *Advances in Neural Information Processing Systems*, 37:26701–26727, 2024a.

Qidong Liu, Xian Wu, Wanyu Wang, Yejing Wang, Yuanshao Zhu, Xiangyu Zhao, Feng Tian, and Yefeng Zheng. Llmemb: Large language model can be a good embedding generator for sequential recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 12183–12191, 2025.

Shuo Liu, An Zhang, Guoqing Hu, Hong Qian, and Tat-Seng Chua. Preference diffusion for recommendation. In *The Thirteenth International Conference on Learning Representations*, Singapore, 2024b.

Haokai Ma, Ruobing Xie, Lei Meng, Xin Chen, Xu Zhang, Leyu Lin, and Zhanhui Kang. Plug-in diffusion model for sequential recommendation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pp. 8886–8894, 2024.

Wenyu Mao, Jiancan Wu, Guoqing Hu, Zhengyi Yang, Wei Ji, and Xiang Wang. On efficiency-effectiveness trade-off of diffusion-based recommenders. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

Jianmo Ni, Jiacheng Li, and Julian McAuley. Justifying recommendations using distantly-labeled reviews and fine-grained aspects. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*, pp. 188–197, 2019.

Ruihong Qiu, Zi Huang, Hongzhi Yin, and Zijian Wang. Contrastive learning for representation degeneration problem in sequential recommendation. In *Proceedings of the fifteenth ACM international conference on web search and data mining*, pp. 813–823, 2022.

Shashank Rajput, Nikhil Mehta, Anima Singh, Raghunandan Hulikal Keshavan, Trung Vu, Lukasz Heldt, Lichan Hong, Yi Tay, Vinh Tran, Jonah Samost, et al. Recommender systems with generative retrieval. *Advances in Neural Information Processing Systems*, 36:10299–10315, 2023.

Steffen Rendle, Christoph Freudenthaler, and Lars Schmidt-Thieme. Factorizing personalized markov chains for next-basket recommendation. In *Proceedings of the 19th international conference on World wide web*, pp. 811–820, 2010.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10684–10695, 2022.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494, 2022.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*.

Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. Bert4rec: Sequential recommendation with bidirectional encoder representations from transformer. In *Proceedings of the 28th ACM international conference on information and knowledge management*, pp. 1441–1450, 2019.

Qiaoyu Tan, Jianwei Zhang, Jiangchao Yao, Ninghao Liu, Jingren Zhou, Hongxia Yang, and Xia Hu. Sparse-interest network for sequential recommendation. In *Proceedings of the 14th ACM international conference on web search and data mining*, pp. 598–606, 2021.

Gemma Team. Gemma 3. 2025. URL `https://goo.gle/Gemma3Report`.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Martin J. Wainwright and Michael I. Jordan. Graphical models, exponential families, and variational inference. *Foundations and Trends in Machine Learning*, 1(1–2):1–305, 2008.

Liyan Wang, Qinyu Yang, Cong Wang, Wei Wang, Jinshan Pan, and Zhixun Su. Learning a coarse-to-fine diffusion transformer for image restoration. *arXiv preprint arXiv:2308.08730*, 2023a.

Wenjie Wang, Yiyan Xu, Fuli Feng, Xinyu Lin, Xiangnan He, and Tat-Seng Chua. Diffusion recommender model. In *Proceedings of the 46th international ACM SIGIR conference on research and development in information retrieval*, pp. 832–841, 2023b.

Ting-Ruen Wei and Yi Fang. Diffusion models in recommendation systems: A survey. *arXiv preprint arXiv:2501.10548*, 2025.

Wenjia Xie, Hao Wang, Luankang Zhang, Rui Zhou, Defu Lian, and Enhong Chen. Breaking determinism: Fuzzy modeling of sequential recommendation using discrete state space diffusion model. *Advances in Neural Information Processing Systems*, 37:22720–22744, 2024.

Xu Xie, Fei Sun, Zhaoyang Liu, Shiwen Wu, Jinyang Gao, Jiandong Zhang, Bolin Ding, and Bin Cui. Contrastive learning for sequential recommendation. In *2022 IEEE 38th international conference on data engineering (ICDE)*, pp. 1259–1273. IEEE, 2022.

Ling Yang, Zhilong Zhang, Yang Song, Shenda Hong, Runsheng Xu, Yue Zhao, Wentao Zhang, Bin Cui, and Ming-Hsuan Yang. Diffusion models: A comprehensive survey of methods and applications. *ACM computing surveys*, 56(4):1–39, 2023a.

Zhengyi Yang, Jiancan Wu, Zhicai Wang, Xiang Wang, Yancheng Yuan, and Xiangnan He. Generate what you prefer: Reshaping sequential recommendation via guided diffusion. *Advances in Neural Information Processing Systems*, 36:24247–24261, 2023b.

Sihyun Yu, Sangkyung Kwak, Huiwon Jang, Jongheon Jeong, Jonathan Huang, Jinwoo Shin, and Saining Xie. Representation alignment for generation: Training diffusion transformers is easier than you think. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 87400–87442, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/d9e42b4d7163931f3689d6d6fbaa11d0-Paper-Conference.pdf.

Wayne Xin Zhao, Junhua Chen, Pengfei Wang, Qi Gu, and Ji-Rong Wen. Revisiting alternative experimental settings for evaluating top-n item recommendation algorithms. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pp. 2329–2332, 2020.

Yunqin Zhu, Chao Wang, Qi Zhang, and Hui Xiong. Graph signal diffusion model for collaborative filtering. In *Proceedings of the 47th International ACM SIGIR conference on research and development in information retrieval*, pp. 1380–1390, 2024.

A Related Work

CloudDiff draws on three lines of research: diffusion-based recommendation, representation learning for sequential recommendation, and SNR-aware design in diffusion models. We organize the discussion accordingly and clarify our positioning within each.

A.1 Sequential Recommendation

Sequential recommendation has evolved from Markov-chain models Rendle et al. (2010) and RNN-based encoders Hidasi et al. (2015) to attention-based architectures. SASRec Kang & McAuley (2018) and BERT4Rec Sun et al. (2019) establish transformer-based paradigms that form the backbone of modern sequential recommenders and provide the user representation $\mathbf{h}_u$ used in our framework. Beyond architectural advances, prior work has explored richer supervision beyond point-wise next-item prediction. Contrastive learning approaches construct multiple views of user interaction sequences and align them to learn robust and consistent sequence representations Xie et al. (2022); Qiu et al. (2022); Cui et al., while intent-aware methods explicitly model latent user interests to capture diverse preference patterns in user behavior Chen et al. (2022); Tan et al. (2021). These approaches highlight the value of richer supervisory signals in sequential modeling. Another line of work incorporates pretrained language model embeddings to inject semantic information into item representations, improving generalization under sparsity Harte et al. (2023); Liu et al. (2024a; 2025). While these methods primarily use LLM features as enhanced representations Li et al. (2023a), our approach instead exploits them as a *semantic structure* to define set-valued supervisory targets whose granularity adapts along the diffusion process.

A.2 Diffusion Models for Recommendation

Diffusion models Ho et al. (2020); Song et al.; Wei & Fang (2025) have recently emerged as a generative paradigm for recommendation, with existing approaches falling into two categories. *Discriminative diffusion recommenders* such as DiffRec Wang et al. (2023b) and its extensions Ma et al. (2024); Zhu et al. (2024) treat the user interaction vector as a noisy latent state and reconstruct it through reverse diffusion, but mainly target collaborative filtering rather than sequential prediction. More closely related are *generative diffusion recommenders for SR*. DiffuRec Li et al. (2023b) introduced diffusion into SR by representing items through a diffusion-based generative process. DreamRec Yang et al. (2023b) reframes SR as a learning-to-generate task and applies classifier-free guidance to generate an "oracle" item embedding conditioned on user history. PreferDiff Liu

et al. (2024b) reformulates the BPR objective into a log-likelihood ranking loss with multiple negatives and cosine error, while subsequent works address embedding collapse in diffusion-based sequential recommenders Chen et al. (2025). More recent efforts have further extended this line in diverse directions, including multi-interest modeling Li et al. (2025), discrete-space diffusion formulations Xie et al. (2024); Hu et al. (2025), and balancing computational efficiency and modeling capacity Mao et al. (2025). CloudDiff belongs to this generative branch but focuses on an aspect that has received less attention: the interaction between the diffusion noise level and the supervisory target. Existing methods typically apply a fixed-resolution objective across timesteps, whereas CloudDiff adapts supervisory granularity according to the SNR-dependent information available during denoising.

A.3 SNR-Aware Design in Diffusion Models

The role of the signal-to-noise ratio in diffusion training has received increasing attention in the generative-modeling community. Min-SNR Hang et al. (2023) treats diffusion training as multi-task learning and clamps per-timestep loss weights using SNR to balance conflicting gradients across noise levels. HDiT Crowson et al. (2024) introduces a soft variant of this weighting, and improved noise schedules Esser et al. (2024); Hang et al. (2025) further refine per-timestep emphasis toward informative noise levels. On the architectural side, REPA Yu et al. (2025) shows that aligning intermediate diffusion features with pretrained vision encoders accelerates training and improves sample quality, and follow-up work extends this alignment to end-to-end VAE tuning Leng et al. (2025). These methods adapt the diffusion process to the noise structure, either by reweighting losses or by injecting external semantic information, but they keep the supervisory target fixed. CloudDiff introduces a complementary SNR-coupled axis by adapting the *resolution of supervision*. Instead of reweighting a fixed point target, we jointly modulate its spatial support, via the semantic cloud and fade function $\Omega(t)$, and its predictive sharpness, via a log-SNR-coupled temperature $\tau(t)$. Our use of LLM embeddings for intermediate representations $\mathbf{z}_t$ is related to representation alignment methods such as REPA, but differs in formulation: we cast alignment as a contrastive objective with set-valued, SNR-gated targets.

A.4 Positioning of CloudDiff

In summary, CloudDiff is differentiated along three axes: (i) Within *diffusion-based SR*, we move beyond fixed-resolution supervision and introduce SNR-coupled supervisory granularity. (ii) Within *soft-label and contrastive supervision* Khosla et al. (2020), we propose a set-valued target whose support and sharpness are jointly scheduled along the diffusion log-SNR axis. (iii) Within *SNR-aware diffusion training*, we shift the SNR coupling from loss weighting to the supervisory target itself, providing a complementary mechanism that can be combined with existing reweighting strategies.

B Proofs

Throughout this appendix we use the notation of §2 and §3. The forward process is variance-preserving,

$$q(\mathbf{x}_t \mid \mathbf{x}_0) \;=\; \mathcal{N}\big(\mathbf{x}_t;\, \sqrt{\bar\alpha_t}\,\mathbf{x}_0,\, (1-\bar\alpha_t)\mathbf{I}\big)\,, \tag{19}$$

with $\mathrm{SNR}(t) = \bar\alpha_t/(1-\bar\alpha_t)$ and $\lambda(t) = \log \mathrm{SNR}(t)$. Expectations are taken over the joint law of $(\mathbf{x}_0, \mathbf{x}_t)$.

B.1 Proof of Proposition 1

Step 1: Gaussian case attains $V(t)$. Suppose $\mathbf{x}_0 \sim \mathcal{N}(\boldsymbol{\mu}_0, \Sigma_0)$. Then $(\mathbf{x}_0, \mathbf{x}_t)$ is jointly Gaussian by equation 19, and $\mathbf{x}_0 \mid \mathbf{x}_t$ is Gaussian. Treating the likelihood $p(\mathbf{x}_t \mid \mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t;\, \sqrt{\bar\alpha_t}\mathbf{x}_0,\, (1-\bar\alpha_t)\mathbf{I})$ as an affine-Gaussian observation with gain $A = \sqrt{\bar\alpha_t}\,\mathbf{I}$ and noise covariance $R = (1-\bar\alpha_t)\,\mathbf{I}$, the standard Gaussian posterior identity gives

$$\mathrm{Cov}(\mathbf{x}_0 \mid \mathbf{x}_t)^{-1} \;=\; \Sigma_0^{-1} + A^\top R^{-1} A \;=\; \Sigma_0^{-1} + \mathrm{SNR}(t)\,\mathbf{I}. \tag{20}$$

Since the posterior covariance does not depend on the realization of $\mathbf{x}_t$,

$$\mathcal{E}(t) \;=\; \mathrm{tr}\,\mathbb{E}[\mathrm{Cov}(\mathbf{x}_0 \mid \mathbf{x}_t)] \;=\; \mathrm{tr}\big[(\Sigma_0^{-1} + \mathrm{SNR}(t)\,\mathbf{I})^{-1}\big] \;=\; V(t), \tag{21}$$

attaining equation 15 with equality.

Step 2: General case. For arbitrary $\mathbf{x}_0$ with covariance $\Sigma_0 \succ 0$, the minimum mean-squared-error estimator Kay (1993) $\mathbb{E}[\mathbf{x}_0 \mid \mathbf{x}_t]$ dominates the linear MMSE estimator $\hat{\mathbf{x}}_0^{\mathrm{lin}}(\mathbf{x}_t)$ in MSE:

$$\mathcal{E}(t) \;=\; \mathbb{E}\,\|\mathbf{x}_0 - \mathbb{E}[\mathbf{x}_0 \mid \mathbf{x}_t]\|^2 \;\le\; \mathbb{E}\,\|\mathbf{x}_0 - \hat{\mathbf{x}}_0^{\mathrm{lin}}(\mathbf{x}_t)\|^2. \tag{22}$$

The linear MMSE estimator depends only on the first two moments of $(\mathbf{x}_0, \mathbf{x}_t)$. With $\mathrm{Cov}(\mathbf{x}_t) = \bar{\alpha}_t\Sigma_0 + (1-\bar{\alpha}_t)\mathbf{I}$ and $\mathrm{Cov}(\mathbf{x}_0, \mathbf{x}_t) = \sqrt{\bar{\alpha}_t}\,\Sigma_0$, direct computation together with the Woodbury identity yields

$$\mathbb{E}\,\|\mathbf{x}_0 - \hat{\mathbf{x}}_0^{\mathrm{lin}}(\mathbf{x}_t)\|^2 \;=\; \mathrm{tr}\Big[\big(\Sigma_0^{-1} + \mathrm{SNR}(t)\,\mathbf{I}\big)^{-1}\Big] \;=\; V(t). \tag{23}$$

Combining the two displays gives $\mathcal{E}(t) \le V(t)$, with equality when $\mathbf{x}_0$ is Gaussian (Step 1).

Step 3: Monotonicity and limits. By VP construction $\bar{\alpha}_t$ is monotonically non-increasing in t, hence $\mathrm{SNR}(t)$ is non-increasing. Consequently $\Sigma_0^{-1} + \mathrm{SNR}(t)\,\mathbf{I}$ is non-increasing in the Loewner order on symmetric positive-definite matrices Bhatia (1997), so its inverse is non-decreasing. Since the trace is a monotone linear functional on positive-semidefinite matrices, $V(t)$ is non-decreasing in t. The limits follow by continuity: as $\bar{\alpha}_t \to 1$, $\mathrm{SNR}(t) \to \infty$ and the inverse tends to $\mathbf{0}$, so $V(t) \to 0$; as $\bar{\alpha}_t \to 0$, $\mathrm{SNR}(t) \to 0$ and the inverse tends to Σ_0, so $V(t) \to \mathrm{tr}\,\Sigma_0$. □

B.2 Proof of Proposition 2

From Proposition 1, $\mathcal{E}(t) \le V(t)$. Since $\Sigma_0^{-1} \succeq 0$, we have $\Sigma_0^{-1} + \mathrm{SNR}(t)\,\mathbf{I} \succeq \mathrm{SNR}(t)\,\mathbf{I}$. Loewner monotonicity of the matrix inverse on the positive-definite cone Bhatia (1997) gives

$$\big(\Sigma_0^{-1} + \mathrm{SNR}(t)\,\mathbf{I}\big)^{-1} \;\preceq\; \frac{1}{\mathrm{SNR}(t)}\,\mathbf{I}, \tag{24}$$

so

$$V(t) \;\le\; \frac{d}{\mathrm{SNR}(t)} \;=\; \frac{d\,(1-\bar{\alpha}_t)}{\bar{\alpha}_t}. \tag{25}$$

Using $\bar{\alpha}_t \ge \bar{\alpha}_{\min}$,

$$\mathcal{E}(t) \;\le\; V(t) \;\le\; \frac{d}{\bar{\alpha}_{\min}}\,(1-\bar{\alpha}_t) \;=\; c_2\,(1-\bar{\alpha}_t), \tag{26}$$

establishing equation 16. Taking square roots of both sides preserves the inequality since both $\mathcal{E}(t)$ and $c_2(1-\bar{\alpha}_t)$ are non-negative. □

B.3 Proof of Proposition 3

Fix a reference item i and candidate set $\mathcal{C} \ni i$ with deterministic scores $s_j := S_{ij}$ for $j \in \mathcal{C}$. Define

$$p_\tau(j) \;=\; \frac{\exp(s_j/\tau)}{\sum_{k\in\mathcal{C}}\exp(s_k/\tau)}, \qquad \tau > 0. \tag{27}$$

Exponential family and Fisher information. Introducing $\eta = 1/\tau$ and writing $p_\eta(j) = \exp(\eta\, s_j - A(\eta))$ with log-partition

$$A(\eta) \;=\; \log\sum_{k\in\mathcal{C}}\exp(\eta\, s_k), \tag{28}$$

the family $\{p_\eta\}_{\eta>0}$ is a one-parameter exponential family in the natural parameter η with scalar sufficient statistic s. Standard exponential-family identities give

$$A'(\eta) = \mathbb{E}_{j\sim p_\eta}[s_j], \qquad A''(\eta) = \mathrm{Var}_{j\sim p_\eta}[s_j], \tag{29}$$

so $\mathcal{I}(\eta) = A''(\eta) = \mathrm{Var}_{j\sim p_\eta}[s_j]$.

Local KL expansion. KL divergence on an exponential family is the Bregman divergence of A Wainwright & Jordan (2008):

$$\mathrm{KL}(p_{\eta_1}\,\|\,p_{\eta_2}) \;=\; A(\eta_2) - A(\eta_1) - A'(\eta_1)(\eta_2-\eta_1). \tag{30}$$

Taylor-expanding A around η_1,

$$\mathrm{KL}(p_{\eta_1}\,\|\,p_{\eta_2}) \;=\; \tfrac{1}{2}\,A''(\eta_1)\,(\eta_2-\eta_1)^2 + \mathcal{O}\big((\eta_2-\eta_1)^3\big). \tag{31}$$

Change of variables $\eta \leftrightarrow \log \tau$. Let $\Delta = \log \tau_2 - \log \tau_1$. Since $\eta = 1/\tau = e^{-\log \tau}$, $\eta_2 = \eta_1 e^{-\Delta}$, giving

$$\eta_2 - \eta_1 \;=\; -\eta_1 \, \Delta + \tfrac{1}{2} \, \eta_1 \, \Delta^2 + \mathcal{O}(\Delta^3). \tag{32}$$

Squaring and retaining leading order,

$$(\eta_2 - \eta_1)^2 \;=\; \eta_1^2 \, \Delta^2 + \mathcal{O}(\Delta^3). \tag{33}$$

Substituting into equation 31 and using $\eta_1 = 1/\tau_1$,

$$\mathrm{KL}(p_{\tau_1} \,\|\, p_{\tau_2}) \;=\; \tfrac{1}{2} \, \tau_1^{-2} \, \mathrm{Var}_{j \sim p_{\tau_1}}[s_j] \cdot \Delta^2 + \mathcal{O}(\Delta^3), \tag{34}$$

which establishes equation 17. □

B.4 Discrete-Identity Counterpart: A Fano-Type Lemma

The posterior MSE bounds in Propositions 1–2 quantify *embedding-level* uncertainty. We complement them with an *identity-level* bound that more directly speaks to the recommendation language of selecting a discrete target item.

Setting. Fix a user history $\mathcal{H}_u$ and a candidate cluster $\mathcal{C} = \{v_1, \ldots, v_M\} \subset \mathcal{V}$ (e.g., the semantic cloud $\{i\} \cup \mathcal{N}_i(K)$ of the target). Let $V^\star \in \mathcal{C}$ denote the true target item, and assume for concreteness that $V^\star$ is uniformly distributed over $\mathcal{C}$ given $\mathcal{H}_u$. Let $\hat{V}(\mathbf{x}_t, \mathcal{H}_u)$ be any decoder that attempts to recover $V^\star$ from the noisy state $\mathbf{x}_t$ and the history.

Lemma (Fano-type lower bound on item identification). *Under the setting above,*

$$\Pr\big(\hat{V} \neq V^\star\big) \;\geq\; 1 - \frac{I(V^\star; \mathbf{x}_t \mid \mathcal{H}_u) + \log 2}{\log M}. \tag{35}$$

Proof. By the standard Fano inequality applied conditionally on $\mathcal{H}_u$ Cover & Thomas (2006),

$$H(V^\star \mid \mathbf{x}_t, \mathcal{H}_u) \;\leq\; H(P_e) + P_e \log(M-1) \;\leq\; \log 2 + P_e \log M,$$

where $P_e = \Pr(\hat{V} \neq V^\star)$. Under uniformity, $H(V^\star \mid \mathcal{H}_u) = \log M$, and the chain rule gives $H(V^\star \mid \mathbf{x}_t, \mathcal{H}_u) = \log M - I(V^\star; \mathbf{x}_t \mid \mathcal{H}_u)$. Combining the two yields equation 35. □

Connection to the diffusion channel. Item identity $V^\star$ deterministically determines its embedding $\mathbf{x}_0(V^\star)$, so by the data-processing inequality $I(V^\star; \mathbf{x}_t \mid \mathcal{H}_u) \leq I(\mathbf{x}_0; \mathbf{x}_t \mid \mathcal{H}_u)$, which is in turn upper-bounded by the Gaussian channel capacity in equation 14. Consequently, in the low-SNR regime $\bar{\alpha}_t \to 0$, the right-hand side of equation 35 approaches one, indicating that exact item discrimination within $\mathcal{C}$ becomes information-theoretically infeasible.

Remark on assumptions. The uniform-prior assumption is made only to obtain a clean closed-form lower bound; the same argument yields a qualitatively identical conclusion for any prior over $\mathcal{C}$, with $\log M$ replaced by $H(V^\star \mid \mathcal{H}_u)$. The lemma is therefore a *discrete-identity counterpart* to Proposition 1: it makes precise, in the recommendation language of item identification, why high-noise regimes admit set-valued supervision more naturally than a single point target. We do not invoke this lemma as a universal impossibility theorem; rather, it provides an SNR-controlled uncertainty scale that complements the embedding-level analysis of Propositions 1–2.

C Unified View and Connections to Prior Work

C.1 Derivation of the SemCloud Objective

We show that $\mathcal{L}_{\text{SemCloud}}$ generalizes InfoNCE to a set-valued positive under SNR-gating.

Standard InfoNCE with a single positive i is

$$\mathcal{L}_{\text{NCE}} = -\log \frac{\exp(S_{ii}/\tau)}{\exp(S_{ii}/\tau) + \sum_{j \in \text{Neg}(i)} \exp(S_{ij}/\tau)}. \tag{36}$$

CloudDiff replaces the numerator with a mixture over $\{i\} \cup \mathcal{N}_i(K)$, weighted by $\Omega(t)$:

$$\mathcal{L}_{\text{SemCloud}}(t) = -\log \frac{M_i(t) + \Omega(t) \cdot P_i(t)}{M_i(t) + \Omega(t) \cdot P_i(t) + N_i(t)}, \tag{37}$$

where $M_i = \exp(S_{ii}/\tau)$, $P_i = \sum_{k \in \mathcal{N}_i(K)} w_{ik} \exp(S_{ik}/\tau)$, and $N_i = \sum_{j \in \text{Neg}(i)} \exp(S_{ij}/\tau)$.

This formulation admits two limiting cases:

- $\Omega(t) \to 0$ (low noise): recovers standard InfoNCE against the ground-truth item only.
- $\Omega(t) \to 1$ (high noise): the numerator becomes $M_i + P_i$, treating the entire semantic neighborhood as a soft positive pool.

The gradient with respect to the representation has a *soft-OR* structure: at high noise, gradient mass is distributed across all neighbors proportional to their current scores and semantic weights w_{ik}, avoiding over-specification toward any single candidate. As the SNR increases and $\Omega(t)$ fades, supervision concentrates on the exact ground-truth item, consistent with the posterior contraction characterized in Propositions 1 and 2.

C.2 Connections to Prior Formulations

The unified formulation recovers several existing designs as limiting cases:

- **Pure diffusion.** When $\lambda_{\text{SC}} = 0$, the contrastive term vanishes and training reduces to standard $\mathbf{x}_0$-prediction diffusion, as in DiffuRec Li et al. (2023b) and DreamRec Yang et al. (2023b).
- **Static contrastive learning.** When $\Omega(t) \equiv 0$ and $\tau(t) \equiv \tau_0$, $\mathcal{L}_{\text{SC}}$ reduces to a standard InfoNCE loss with a single positive and fixed temperature.
- **Decoupled auxiliary losses.** Prior approaches that combine diffusion with contrastive learning typically use a contrastive objective that is independent of the diffusion timestep. In the present framework, this corresponds to constant Ω and τ, decoupling the contrastive signal from the noise level.
- **SNR-coupled formulation (ours).** CloudDiff introduces explicit coupling between the supervisory signal and the diffusion process via $\Omega(t)$ and $\tau(t)$, so that both the support and the sharpness of the target distribution evolve with SNR.

A continuous interpolation principle. The key structural property of CloudDiff is not that it reduces to either extreme, but that it interpolates between them along a single physically meaningful axis: SNR. As $\bar{\alpha}_t$ varies from 0 to 1, the objective continuously transitions from a fully set-valued, soft-positive regime to a single-positive contrastive regime. This transition is not a heuristic convex combination of two losses but emerges from modulating two scalar quantities $(\Omega(t), \tau(t))$ within a single InfoNCE-type objective. Standard contrastive learning and set-valued contrastive learning thus appear as the two boundary points of a one-dimensional family parameterized by SNR.

D Optimized Training and Inference Algorithms

Algorithm 1 Training Phase of CloudDiff

Require: Sequential encoder Φ, denoiser f_θ, SemCloud projector g_ϕ; training set $\mathcal{D}$; trainable item embeddings $\{\mathbf{r}_v\}_{v\in\mathcal{V}}$; frozen LLM anchor embeddings $\{\mathbf{a}_v\}_{v\in\mathcal{V}}$; semantic clouds $\mathcal{N}_i(K)$.
Ensure: Updated parameters.

Initialize τ_{lo} as the mean intra-cloud cosine distance.
Initialize log-SNR window $[\lambda_{\min}, \lambda_{\max}]$ from the noise schedule $\bar{\alpha}$.
repeat
  Sample mini-batch $\{(\mathcal{H}_u, i)\} \sim \mathcal{D}$.
  $\mathbf{h}_u \leftarrow \Phi(\mathbf{r}_{v_1}, \ldots, \mathbf{r}_{v_n})$ {Sequence encoding}
  $\mathbf{h}_u \leftarrow \text{dropout}(\mathbf{h}_u, p_u)$ {For CFG training}
  `// Diffusion Forward & Reconstruction`
  Sample $t \sim \text{U}(1, T)$ and $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$.
  $\mathbf{x}_0 \leftarrow \mathbf{r}_i$.
  $\mathbf{x}_t \leftarrow \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}$.
  $\hat{\mathbf{x}}_0, \mathbf{z}_t \leftarrow f_\theta(\mathbf{x}_t, \mathbf{h}_u, t)$.
  $\mathcal{L}_{\text{diff}} \leftarrow \|\mathbf{x}_0 - \hat{\mathbf{x}}_0\|^2$.
  `// Semantic Cloud Alignment`
  Compute $\tau(t)$ from Eq. equation 7 and $\Omega(t) = (1-\bar{\alpha}_t)^p$.
  $\tilde{\mathbf{z}}_{i,t} \leftarrow \text{norm}(g_\phi(\mathbf{z}_t))$.
  Construct $\text{Neg}(i) = \mathcal{B} \setminus (\{i\} \cup \mathcal{N}_i(K))$.
  $M_i(t) \leftarrow \exp(\langle \tilde{\mathbf{z}}_{i,t}, \mathbf{a}_i \rangle / \tau(t))$.
  $P_i(t) \leftarrow \sum_{k\in\mathcal{N}_i(K)} w_{ik} \exp(\langle \tilde{\mathbf{z}}_{i,t}, \mathbf{a}_k \rangle / \tau(t))$.
  $N_i(t) \leftarrow \sum_{j\in\text{Neg}(i)} \exp(\langle \tilde{\mathbf{z}}_{i,t}, \mathbf{a}_j \rangle / \tau(t))$.
  $\mathcal{L}_{\text{SC}}(t) \leftarrow -\log \frac{M_i(t)+\Omega(t)P_i(t)}{M_i(t)+\Omega(t)P_i(t)+N_i(t)}$.
  $\mathcal{L} \leftarrow \mathcal{L}_{\text{diff}} + \lambda_{\text{SC}}\mathcal{L}_{\text{SC}}(t)$.
  Update trainable parameters using AdamW.
until convergence
return trained parameters.

Algorithm 2 Inference Phase of CloudDiff

Require: Trained parameters; Test sequence $\mathcal{H}$; Guidance scale w; Candidate item pool $\mathcal{V}$.
Ensure: Top-K recommendations.

$\mathbf{h}_u \leftarrow \Phi(\mathcal{H}_u)$ {Encode history}
$\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ {Start from pure noise}
for $t = T, T-1, \ldots, 1$ (via DDIM sub-steps) **do**
  $\hat{\mathbf{x}}_0 \leftarrow (1+w)f_\theta(\mathbf{x}_t, \mathbf{h}_u, t) - w f_\theta(\mathbf{x}_t, \varnothing, t)$ {CFG with $\mathbf{x}_0$-prediction}
  $\mathbf{x}_{t-1} \leftarrow \text{DDIM_Step}(\mathbf{x}_t, \hat{\mathbf{x}}_0)$ {Deterministic update}
end for
Compute scores $y_v = \hat{\mathbf{x}}_0^\top \mathbf{r}_v,\ \forall v \in \mathcal{V}$
$\mathcal{R} \leftarrow \text{TopK}(\mathbf{y}, K)$
return $\mathcal{R}$

E Experiments

E.1 Datasets and Evaluation Protocol

We follow the experimental setup described in Section 4.1. Table 2 summarizes the statistics of the processed datasets. We follow the full-ranking evaluation protocol Krichene & Rendle (2020); Zhao et al. (2020), ranking the ground-truth item against the entire item set rather than a sampled candidate set. We adopt a user-wise chronological split, where interaction sequences are partitioned by timestamp for each user. This protocol prevents information leakage across splits and results in

a more challenging evaluation setting, as the model is tested on future interactions of unseen user histories.

Table 2: Statistics of the processed datasets.

Dataset	#Users	#Items	#Reviews	Density (%)
Sports	35,598	18,357	296,337	0.0453
Beauty	22,363	12,101	198,502	0.0734
Toys	19,412	11,924	167,597	0.0724
Yelp	30,431	20,033	316,354	0.0519

E.2 BASELINES

We compare CloudDiff against a diverse set of representative baselines spanning sequential, semantic, and diffusion-based recommendation paradigms.

Conventional Sequential Recommenders.

- **GRU4Rec** Hidasi et al. (2015): A GRU-based recurrent encoder for session-based recommendation, representing the standard recurrent baseline for sequential modeling.
- **SASRec** Kang & McAuley (2018): A unidirectional Transformer that uses self-attention over the user's interaction history to predict the next item, and serves as the dominant attention-based backbone in sequential recommendation.
- **BERT4Rec** Sun et al. (2019): A bidirectional Transformer trained with a Cloze-style masked-item objective, representing the masked-prediction counterpart of SASRec.
- **ICLRec** Chen et al. (2022): An intent-aware sequential recommender that infers latent user intents via clustering and incorporates them through a contrastive self-supervised objective. We include it as a representative method that enriches the training signal beyond next-item point supervision.

Semantic and Generative Recommenders.

- **TIGER** Rajput et al. (2023): A generative retrieval framework that represents each item as a tuple of discrete semantic tokens (Semantic IDs) obtained via residual quantization of content embeddings, and predicts the next item by autoregressively decoding its Semantic ID. We include it as a representative non-diffusion generative recommender.
- **LLM-ESR** Liu et al. (2024a): A semantic-enhanced sequential recommender that augments a conventional backbone with item and user embeddings derived from a frozen large language model. It serves as the closest baseline that, like CloudDiff, leverages frozen language-model embeddings, but uses them at the representation level.

Diffusion-based Recommenders.

- **DiffuRec** Li et al. (2023b): An early diffusion model for sequential recommendation that represents each item as a distribution and learns to denoise the target item embedding conditioned on the user history.
- **DreamRec** Yang et al. (2023b): A guided diffusion recommender that frames sequential recommendation as a learning-to-generate task and applies classifier-free guidance to generate an oracle item embedding conditioned on the user history.
- **PreferDiff** Liu et al. (2024b): A diffusion recommender with a tailored ranking objective derived from BPR, integrating multiple negative samples and a cosine error to better align the generative objective with personalized ranking. It is the strongest diffusion baseline in our setting and the one with an explicit ranking-aware objective.

E.3 Implementation Details

Training Settings. We implement CloudDiff in Python 3.10 with PyTorch 2.6.0 on a single NVIDIA RTX A6000 GPU. CloudDiff is trained with AdamW using a batch size of 256 for training and 32 for evaluation, for up to 200 epochs with early stopping patience 20. Unless otherwise noted, we report results averaged over five random seeds. For each user, the model is conditioned on the 10 most recent interactions in the history. The denoising network is a two-layer MLP that takes as input the concatenation of the noisy item embedding, the user representation, and a sinusoidal timestep embedding. The forward diffusion process uses a linear β schedule with $\beta_{\text{start}} = 10^{-4}$, $\beta_{\text{end}} = 2 \times 10^{-2}$, and $T = 2000$ steps, while inference uses DDIM with 100 steps. We adopt classifier-free guidance with condition-dropout probability $p_u = 0.1$ during training and guidance scale w at inference. In the main experiments, we use centered 1024-dimensional item-preference embeddings derived from frozen LLM representations.

LLM-Based Semantic Item Descriptions. To construct semantically enriched item representations, we first generate a preference-oriented textual description for each item using a LLM. Rather than directly embedding raw metadata, the LLM is prompted to infer the types of users who would prefer the item based on its metadata and reviews. The prompt encourages the model to summarize favored attributes, usage scenarios, and functional benefits into a concise preference-centric description. Rather than merely encoding metadata in a structured form, this prompting strategy encourages the LLM to leverage its semantic reasoning and world knowledge to infer latent user preferences and functional item characteristics.

Specifically, given the item title, brand, description, and associated reviews, the LLM generates a concise preference-oriented paragraph with a maximum length of 512 tokens beginning with the template:

Prompt for LLM-based Item Description

You are an intelligent assistant that analyzes item metadata to generate personalized item recommendations. Analyze the provided metadata to determine what kinds of users are most likely to prefer the target item. The response should begin with:

"Users who prefer [common themes/preferences] would find this item suitable."

Incorporate favored features, usage scenarios, and functional benefits inferred from the metadata.
Target item metadata:

- Title: {Title}
- Brand: {Brand}
- Description: {Description}
- Begin with the sentence: "Users who prefer ..."

Response:

We use `Gemma-3-27B-it` Team (2025) for description generation, and encode the resulting text using `mxbai-embed-large-v1` Li & Li (2023); Lee et al. (2024) to obtain centered 1024-dimensional semantic item embeddings $\mathbf{a}_i$. These embeddings are used to construct semantic neighborhoods $\mathcal{N}_i(K)$ for the semantic cloud objective.

E.4 Hyperparameter Details

E.4.1 Common Search Space

For all competitive baselines and the proposed **CloudDiff**, we perform a grid search over a common set of foundational hyper-parameters: the learning rate $lr \in \{1\text{e-}2, 5\text{e-}3, 1\text{e-}3, 5\text{e-}4, 1\text{e-}4, 1\text{e-}5\}$, the hidden embedding dimension $d \in \{64, 128, 256, 512, 1024\}$, and the number of in-batch negatives $|\text{Neg}(i)| \in \{128, 256\}$. The batch size is fixed at 256, and the weight decay is kept at 0 unless otherwise specified.

Model	Specific Hyperparameter Search Space
BERT4Rec	*mask_p* $\in \{0.2, 0.4, 0.6, 0.8\}$
ICLRec	*num_intent_clusters* $\in \{64, 128, 256\}$, $\lambda_{\text{cf}} \in \{0.1, 0.3, 0.5\}$
LLM-ESR	Distillation weight $\alpha \in \{0.1, 0.3, 0.5, 1.0\}$
DiffuRec	$T \in \{5, 20, 50, 100\}$
DreamRec	$T \in \{500, 1000, 1500, 2000\}$, $w \in \{1, 2, 4, 6, 8\}$, $p \in \{0.1, 0.2, 0.3\}$
TIGER	Weight decay *wd* $\in \{0, 1\text{e-}1, 1\text{e-}2, 1\text{e-}3\}$
PreferDiff	Alignment weight $\lambda \in \{0.2, 0.4, 0.6, 0.8\}$, $w \in \{1, 2, 4, 6, 8\}$
CloudDiff	$\lambda_{\text{SC}} \in \{0.1, 0.2, 0.5\}$, $w \in \{1, 2, 4, 6, 8\}$, $\kappa \in \{1.5, 2, 3, 4\}$

E.4.2 Best Hyperparameter Configurations

We fix the shared hyperparameters across all datasets, setting the hidden size to $d = 512$, the learning rate to $lr = 5 \times 10^{-4}$, and the fade power to $p = 0.5$, and report dataset-specific configurations in Table 3.

Table 3: Dataset-specific hyperparameter configurations of CloudDiff.

Hyperparameter	Beauty	Toys	Sports	Yelp
κ	1.5	3.0	1.5	3.0
# Negatives	128	128	256	256
Guidance scale (w)	2	1	2	2
λ_{SC}	0.1	0.2	0.2	0.2

We observe that larger and sparser datasets (e.g., Sports and Yelp) generally benefit from a larger negative pool ($|\text{Neg}(i)| = 256$), while relatively smaller datasets are sufficiently modeled with 128 negatives. This suggests that harder retrieval settings require a broader contrastive context to maintain effective discrimination. Interestingly, the optimal guidance scale remains consistently small ($w \in \{1, 2\}$) across all datasets. In recommendation settings, strong classifier-free guidance is often used to enforce highly personalized predictions from noisy intermediate states. In contrast, CloudDiff achieves strong performance with only mild guidance, suggesting that the SNR-coupled semantic supervision helps maintain stable denoising trajectories and preserve user-specific preference structure throughout the reverse process. As a result, accurate personalized recommendations can be obtained without relying on excessively large guidance scales.

E.4.3 Sensitivity to κ

The hyperparameter $\kappa = \tau_{\text{hi}}/\tau_{\text{lo}}$ controls the dynamic range of the SNR-coupled temperature schedule. Figure 8 reports performance across different choices of κ on all datasets. Overall, CloudDiff is relatively robust to the choice of κ. Across datasets, performance varies smoothly and the gap between different settings remains moderate, suggesting that the proposed temperature schedule does not require excessive tuning. Interestingly, the preferred value of κ differs across datasets. Beauty and Sports achieve the best performance with a relatively small dynamic range ($\kappa = 1.5$), whereas other catalogs benefit from a wider range. While a deeper analysis of this dataset-dependent behavior is left for future work, the consistent within-range robustness suggests that $\kappa \in [1.5, 3]$ is a safe operating range across diverse domains. These results suggest that κ acts as a lightweight dataset-level adaptation parameter for controlling the sharpness transition of the supervisory signal.

E.5 Additional Implementation Details

$\tau(t)$ Ablation. To validate the role of the proposed log-SNR-coupled temperature schedule, we compare CloudDiff against several alternative temperature parameterizations while keeping all other training configurations fixed. The corresponding results are reported in Figure 5. Specifically, we evaluate:

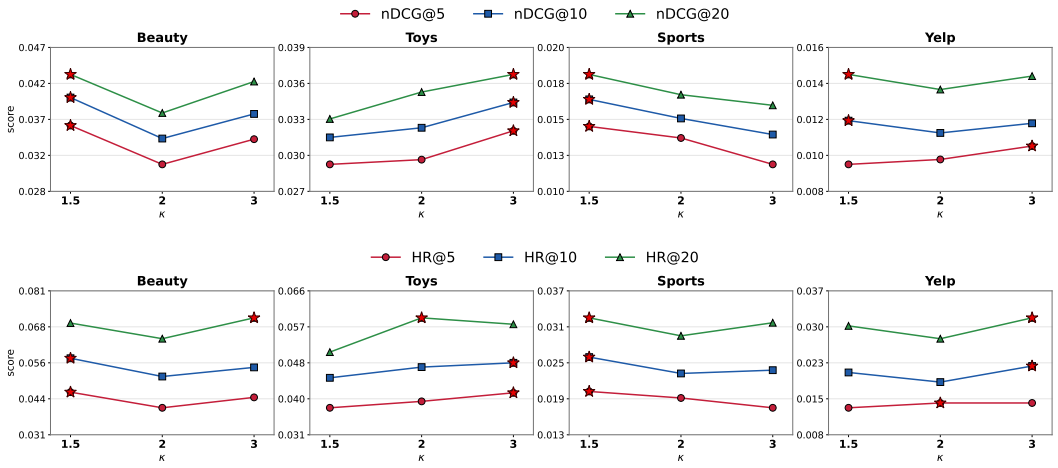


Figure 8: Sensitivity analysis of κ. Results are reported in terms of NDCG (top) and HR (bottom).

- **linear**: temperature varies linearly with the normalized diffusion timestep,

$$\tau(t) = \tau_{\min} + \tilde{t}\,(\tau_{\max} - \tau_{\min}), \qquad \tilde{t} = \frac{t}{T-1}.$$

- **alpha-bar linear**: temperature varies linearly with the cumulative noise schedule $\bar{\alpha}_t$,

$$\tau(t) = \tau_{\text{lo}} + (\tau_{\text{hi}} - \tau_{\text{lo}})\bar{\alpha}_t.$$

- **lambda linear**: temperature varies linearly with the normalized log-SNR coordinate,

$$\tau(t) = \tau_{\text{lo}} + u(t)(\tau_{\text{hi}} - \tau_{\text{lo}}),$$

  where $u(t)$ is the normalized log-SNR defined in Eq. equation 7.

- **snr increasing**: temperature follows a geometric schedule directly parameterized by SNR,

$$\tau(t) = \tau_{\text{lo}} \kappa^{w(t)},$$

  where $w(t)$ denotes normalized SNR.

The proposed **log-SNR** schedule differs from these variants by introducing a multiplicative power-law coupling between temperature and the normalized log-SNR coordinate, rather than an additive linear interpolation, yielding the power-law relation described in Eq. equation 7. All schedule variants share the same backbone architecture, optimization settings, diffusion configuration, and evaluation protocol as the full model.

Training Dynamics Analysis. For the loss-variance analysis (Figure 6), we evaluate per-sample reconstruction errors across diffusion timesteps using the training split. For each sequence, a timestep t is sampled uniformly, and the denoiser prediction $\hat{\mathbf{x}}_0$ is evaluated against the ground-truth embedding using a common post-hoc reconstruction metric (ℓ_2 distance), independent of the model's original training objective. The same noisy input construction and timestep sampling procedure are applied to both models. We group samples by timestep bins and compute the variance of the loss within each bin. The plotted curves show the variance as a function of t on a logarithmic scale. This analysis is intended to measure timestep-wise prediction stability under a unified metric rather than compare the original optimization objectives directly.

Denoising Trajectory Analysis. For the trajectory analysis (Figure 7), we track the ranking behavior of the model during the reverse diffusion process on the test set. Starting from Gaussian noise, we perform DDIM sampling and record intermediate predictions at each step. At every timestep, we compute scores over the full item set via inner product with the predicted embedding. We report two metrics: (i) the rank of the ground-truth item, and (ii) the mean rank over the set $\{i\} \cup \mathcal{N}_i(K)$, where $\mathcal{N}_i(K)$ denotes the semantic neighborhood. We randomly sample 200 users from the test set and average ranks across users at each timestep. We then visualize their evolution along the denoising trajectory from high-noise to low-noise regimes.

Implementation Consistency. Unless otherwise specified, all analyses use the same trained checkpoints as the main experiments, with identical best hyperparameters and data splits. This ensures that the observed behaviors are attributable to the supervision design rather than differences in training configuration.

F Empirical Validation of the Theoretical Analysis

F.1 Empirical Analysis of Uncertainty Contraction

As SNR increases, posterior entropy decreases from 9.304 to 6.506, the probability assigned to the ground-truth item increases from 0.0001 to 0.3715, and the effective candidate set contracts from 10,978 to 679. The learned item space therefore exhibits an uncertainty-contraction pattern consistent with the motivation of our analysis: at low SNR, the plausible candidate region is broad and the target is difficult to identify, whereas increasing SNR progressively reduces uncertainty and permits finer discrimination. These observations provide empirical support for adapting supervisory granularity along the diffusion trajectory.

Table 4: Candidate uncertainty at different SNR levels. Higher SNR corresponds to a less corrupted diffusion state.

Noise regime (SNR)	Posterior entropy	GT probability	Effective candidates
High noise (0.00)	9.304	0.0001	10,978
Mid noise (1.64)	8.186	0.1686	3,647
Low noise (4.10)	6.506	0.3715	679

As SNR increases, posterior entropy decreases from 9.304 to 6.506, the probability assigned to the ground-truth item increases from 0.0001 to 0.3715, and the effective candidate set contracts from 10,978 to 679. The learned item space therefore exhibits an uncertainty-contraction pattern consistent with the motivation of our analysis: at low SNR, the plausible candidate region is broad and the target is difficult to identify, whereas increasing SNR progressively reduces uncertainty and permits finer discrimination. These observations provide empirical support for adapting supervisory granularity along the diffusion trajectory without assuming that the empirical item-embedding distribution is Gaussian.

F.2 Empirical Validation of the Log-SNR Temperature Coupling

Proposition 3 motivates log-temperature as a natural coordinate for controlling predictive sharpness, leading to the log-SNR-coupled temperature schedule in equation 7. While the analysis motivates this coupling, it does not uniquely determine the specific parameterization. We therefore examine whether the temperature preferred empirically follows the log-linear relationship assumed in equation 7.

We partition the diffusion trajectory into B log-SNR bins and estimate the optimal temperature $\tau^\star(b)$ for each bin by minimizing the held-out set-valued InfoNCE loss in equation 13. To avoid evaluating a relationship already induced by the proposed temperature schedule, we primarily use frozen representations from the *cloud + static* τ checkpoint, which is trained without log-SNR-dependent temperature scheduling. We then regress $\log \tau^\star(b)$ on the mean log-SNR λ_b within each bin. For completeness, we also report the same analysis using representations from the full CloudDiff model.

Across datasets, checkpoints, and binning resolutions, the resulting R^2 values range from 0.867 to 0.904. In particular, the consistently high fit obtained from the *cloud + static* τ checkpoint indicates that the relationship is also observed without training under the proposed temperature schedule. These results provide empirical support for the log-linear coupling between temperature and log-SNR adopted in equation 7.

Table 5: Goodness of fit (R^2) between the empirically estimated optimal log-temperature $\log \tau^\star(b)$ and mean log-SNR λ_b across different numbers of log-SNR bins.

Dataset	**Checkpoint**	$B = 5$	$B = 10$	$B = 20$
Beauty	Cloud + static τ	0.889	0.870	0.867
Beauty	Full	0.904	0.878	0.873
Sports	Cloud + static τ	0.880	0.876	0.872
Sports	Full	0.872	0.885	0.875

F.3 Alternative Coarse-to-Fine Objective Formulations

We further examine whether the additive positive-mass formulation in equation 13 is important for realizing coarse-to-fine supervision. We compare CloudDiff with two alternative formulations that use the same SNR-dependent fade $\Omega(t)$ but interpolate supervision at different levels.

Logit interpolation. We interpolate between the ground-truth similarity and the weighted neighborhood similarity:

$$\widetilde{S}_i(t) = (1 - \Omega(t))S_{ii}(t) + \Omega(t) \sum_{k \in \mathcal{N}_i} w_{ik} S_{ik}(t). \tag{38}$$

Loss interpolation. We instead interpolate between a target-only contrastive loss and a cloud-based contrastive loss:

$$\mathcal{L}_{\text{interp}}(t) = (1 - \Omega(t))\mathcal{L}_{\text{target}}(t) + \Omega(t)\mathcal{L}_{\text{cloud}}(t). \tag{39}$$

Table 6: Comparison with alternative coarse-to-fine formulations. Results are NDCG@10 averaged over five random seeds.

Variant	**Beauty**	**Toys**	**Sports**	**Yelp**
CloudDiff (Eq. 13)	**0.0384 (0.0010)**	**0.0347 (0.0011)**	**0.0158 (0.0006)**	**0.0118 (0.0005)**
Logit interpolation	0.0355 (0.0005)	0.0321 (0.0008)	0.0131 (0.0022)	0.0100 (0.0011)
Loss interpolation	0.0347 (0.0009)	0.0316 (0.0015)	0.0129 (0.0015)	0.0073 (0.0008)

The formulation in equation 13 consistently outperforms both interpolation-based alternatives across all four datasets. These results empirically support the additive positive-mass formulation used by CloudDiff as an effective realization of coarse-to-fine supervision, beyond simply interpolating either similarity scores or contrastive losses.

G Additional Experimental Analysis

G.1 Sensitivity to Semantic Cloud Size

We analyze the sensitivity of CloudDiff to the semantic cloud size K. Figure 9 reports performance across different choices of K on all datasets. Overall, we observe that CloudDiff is robust across a moderate range of neighborhood sizes, while exhibiting a clear performance degradation at both extremes. When K is too small (e.g., $K = 3$ or 5), the semantic cloud contains only a few neighbors, and the supervision becomes overly concentrated on a limited set of items. In this regime, the benefit of set-valued supervision diminishes, as the model effectively reduces to near point-wise alignment, limiting the advantage of semantic smoothing. Interestingly, Yelp shows the best NDCG performance at a relatively small cloud size ($K = 5$). We hypothesize that this behavior is related to the high sparsity and semantic diversity of Yelp interactions, where a sharper and more selective neighborhood may provide more reliable supervisory signals than broader semantic support. On the other hand, when K is too large (e.g., $K = 50$), the semantic cloud includes many weakly related items, introducing noise into the supervisory signal. This broad support weakens the alignment toward truly relevant items and can hinder effective representation learning. Across all datasets, we find that moderate values (e.g., $K \in [10, 20]$) consistently achieve the best performance. This suggests that an

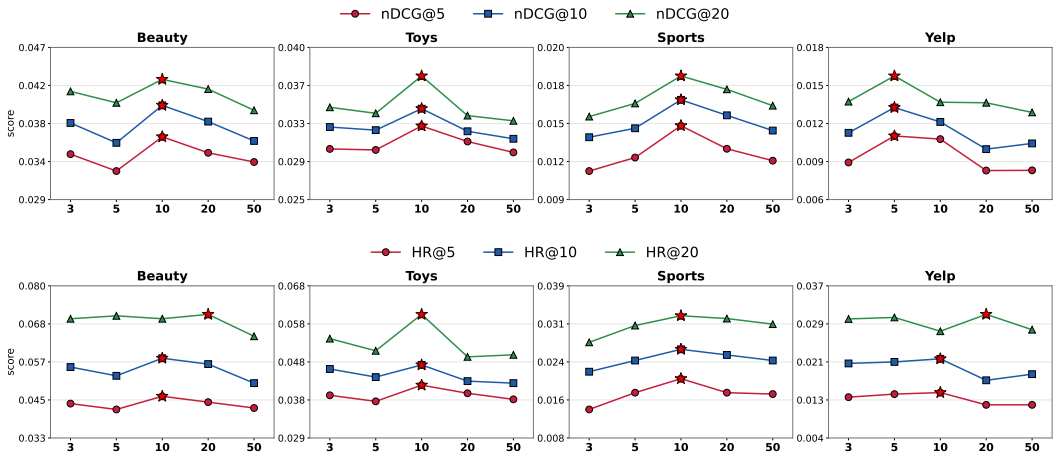


Figure 9: Sensitivity analysis of semantic cloud size K. Results are reported in terms of NDCG (top) and HR (bottom).

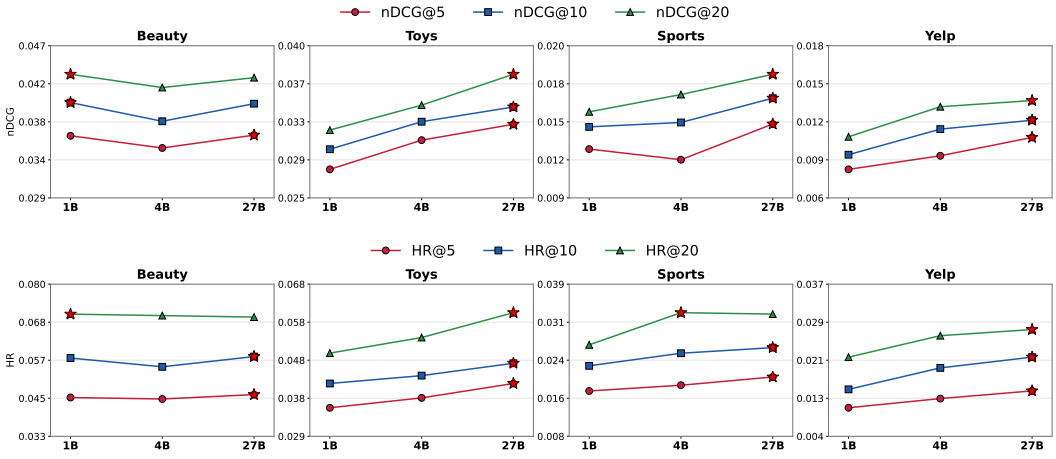


Figure 10: Effect of LLM backbone size. Results are reported in terms of NDCG (top) and HR (bottom).

appropriately sized semantic neighborhood strikes a balance between providing sufficient diversity for robust supervision and maintaining focus on semantically meaningful targets. Based on these results, we set $K = 5$ for Yelp and $K = 10$ for the remaining datasets in all experiments.

G.2 Effect of LLM Backbone Size

We examine the effect of the LLM backbone size used to construct the semantic embedding space. Specifically, we evaluate three variants (`Gemma-3-it` Team (2025)) with different model sizes (1B, 4B, and 27B), while keeping all other components fixed. Figure 10 reports the results across all datasets. As expected, larger LLM backbones generally yield improved performance, likely due to their stronger representational capacity, which enables more precise semantic distinction between items. Importantly, however, the performance does not degrade drastically when using smaller models. Even with significantly reduced model sizes (e.g., 1B or 4B), CloudDiff maintains competitive performance across all datasets. This suggests that the benefit of incorporating semantic priors is not solely dependent on the scale of the underlying LLM. Overall, these results indicate that while larger LLMs provide additional gains, the effectiveness of CloudDiff stems from the use of semantic structure itself, making the framework robust to the choice of backbone size.

G.3 Robustness to Neighborhood Construction

The main experiments instantiate the supervisory neighborhood using frozen language-model embeddings. However, the SNR-coupled supervisory principle does not require a particular representation space. To examine its dependence on neighborhood construction, we replace the LLM-derived space with two interaction-based alternatives: SPPMI-SVD (Levy & Goldberg, 2014) embeddings constructed from item co-occurrence and item embeddings obtained from a pretrained SASRec (Kang & McAuley, 2018) model. For each representation, we compare SNR-coupled supervision with a static counterpart while keeping all other settings fixed.

Table 7: Robustness to neighborhood construction. Results are NDCG@10 reported as mean (std) over five seeds.

Neighbor space	Scheduling	Beauty	Toys	Sports	Yelp
LLM	SNR-coupled	**0.0384 (0.0010)**	**0.0347 (0.0011)**	**0.0158 (0.0006)**	**0.0118 (0.0005)**
	Static	0.0350 (0.0023)	0.0318 (0.0020)	0.0132 (0.0014)	0.0081 (0.0004)
SPPMI-SVD	SNR-coupled	**0.0361 (0.0018)**	**0.0315 (0.0003)**	**0.0116 (0.0017)**	**0.0097 (0.0023)**
	Static	0.0315 (0.0026)	0.0286 (0.0034)	0.0107 (0.0004)	0.0072 (0.0014)
SASRec	SNR-coupled	**0.0359 (0.0016)**	**0.0342 (0.0006)**	**0.0130 (0.0005)**	**0.0107 (0.0001)**
	Static	0.0320 (0.0017)	0.0283 (0.0004)	0.0111 (0.0001)	0.0082 (0.0023)

SNR-coupled supervision consistently outperforms its static counterpart across all three representation spaces and all four datasets. The improvement therefore persists when the LLM-derived neighborhood is replaced by interaction-derived structure, indicating that the benefit of SNR-dependent supervisory adaptation is not specific to a particular semantic representation. At the same time, the LLM neighborhood achieves the strongest absolute performance across the four datasets, suggesting that its dense semantic relations provide an effective instantiation of the supervisory neighborhood. These results distinguish the SNR-coupled scheduling mechanism from the particular representation used to construct its relaxed target.

G.4 Robustness to Neighborhood Quality

While the preceding analysis shows that CloudDiff can operate over different item-relation spaces, the quality of the resulting neighborhood may still affect its effectiveness. We examine this sensitivity by randomly replacing either 50% or 100% of the LLM-derived neighbors while leaving the remaining training procedure unchanged.

Table 8: Robustness to neighborhood corruption. A fraction of the LLM-derived neighbors is replaced by randomly selected items. Results are NDCG@10 reported as mean (std) over five seeds.

Corruption ratio	Beauty	Toys	Sports	Yelp
0%	**0.0384 (0.0010)**	**0.0347 (0.0011)**	**0.0158 (0.0006)**	**0.0118 (0.0005)**
50%	0.0376 (0.0005)	0.0344 (0.0006)	0.0152 (0.0011)	0.0104 (0.0006)
100%	0.0349 (0.0011)	0.0321 (0.0001)	0.0120 (0.0007)	0.0086 (0.0019)

Performance decreases consistently as neighborhood corruption increases, confirming that informative neighbors provide a beneficial supervisory signal. Nevertheless, performance remains relatively stable under 50% corruption, whereas replacing the entire neighborhood produces a more pronounced degradation. Thus, neighborhood quality matters for obtaining the strongest performance, but moderate noise does not eliminate the benefit of the proposed training scheme. This behavior is consistent with the role of the cloud as auxiliary representation-level supervision: it shapes the denoiser representation without directly determining the final recommended item, while exact-target recovery remains governed by the diffusion objective.

G.5 Disentangling Semantic Priors from SNR-Coupled Supervision

We further examine whether the observed improvements can be explained by the use of pretrained language representations alone. We consider two controls. *LLM Init* uses the LLM embeddings only to initialize the trainable item representations, without semantic alignment during training. *LLM + Static* uses the same semantic neighborhood as CloudDiff but removes its SNR-dependent adaptation, corresponding to static semantic contrastive supervision.

Table 9: Comparison with simpler uses of LLM-derived representations. Results are NDCG@10 reported as mean (std) over five seeds.

Method	Beauty	Toys	Sports	Yelp
LLM + SNR (CloudDiff)	**0.0384 (0.0010)**	**0.0347 (0.0011)**	**0.0158 (0.0006)**	**0.0118 (0.0005)**
LLM Init	0.0076 (0.0013)	0.0124 (0.0018)	0.0054 (0.0006)	0.0026 (0.0007)
LLM + Static	0.0350 (0.0023)	0.0318 (0.0020)	0.0132 (0.0014)	0.0081 (0.0004)

LLM initialization alone yields substantially lower performance than CloudDiff across all datasets, showing that pretrained semantic representations by themselves do not account for the observed gains. Moreover, SNR-coupled supervision consistently improves over static semantic supervision. Together with the alternative-neighborhood results in Table 7, these controls indicate that CloudDiff benefits from both an informative neighborhood structure and its SNR-dependent adaptation, rather than from the external semantic prior alone.

H Computational Complexity

We compare the asymptotic complexity of representative recommendation models. Here, B denotes the batch size, L the maximum user sequence length, d the hidden embedding dimension, T the number of diffusion timesteps used in full reverse sampling, S the number of DDIM sampling steps during inference, and V the number of items. For TIGER, T_{tok} and V_{vocab} denote the token sequence length and vocabulary size, respectively. The term d_ℓ represents the dimensionality of the semantic embedding space used in CloudDiff.

Table 10: Asymptotic complexity comparison.

Method	**Training Complexity**	**Inference Complexity**
SASRec	$\mathcal{O}(BL^2d + BVd)$	$\mathcal{O}(BL^2d + BVd)$
TIGER	$\mathcal{O}(BT_{\text{tok}}^2d + BT_{\text{tok}}dV_{\text{vocab}})$	Autoregressive decoding
DiffuRec	$\mathcal{O}(BL^2d + BLd^2)$	$\mathcal{O}(TB(L^2d + Ld^2))$
DreamRec	$\mathcal{O}(BL^2d + Bd^2)$	$\mathcal{O}(BL^2d + TBd^2)$
PreferDiff	$\mathcal{O}(BL^2d + Bd^2)$	$\mathcal{O}(BL^2d + SBd^2 + BVd)$
CloudDiff (Ours)	$\mathcal{O}(BL^2d + Bd^2 + B^2d_\ell)$	$\mathcal{O}(BL^2d + SBd^2 + BVd)$

Unlike components that modify the inference pathway, CloudDiff introduces additional semantic regularization only during training. At inference, the model follows the same DDIM-style sampling procedure with a lightweight MLP denoiser, resulting in inference complexity comparable to other diffusion recommenders of this class. The semantic-cloud alignment objective introduces additional computation only during training. Specifically, the loss is computed over projected hidden representations and frozen LLM anchor embeddings, requiring pairwise similarity evaluation among the subset of samples participating in semantic supervision. Let $N \leq B$ denote the effective mini-batch size after timestep-based filtering, i.e., the number of samples whose diffusion timesteps fall within the active supervision region. The dominant additional cost is therefore the dense similarity computation, which scales as $\mathcal{O}(N^2d_\ell)$, where d_ℓ is the semantic embedding dimension. Importantly, this overhead remains moderate in practice for two reasons. First, timestep gating typically yields $N \ll B$, since only a subset of samples contributes to the semantic-cloud objective at a given timestep. Second, the semantic anchors are precomputed and frozen throughout training, avoiding any additional LLM

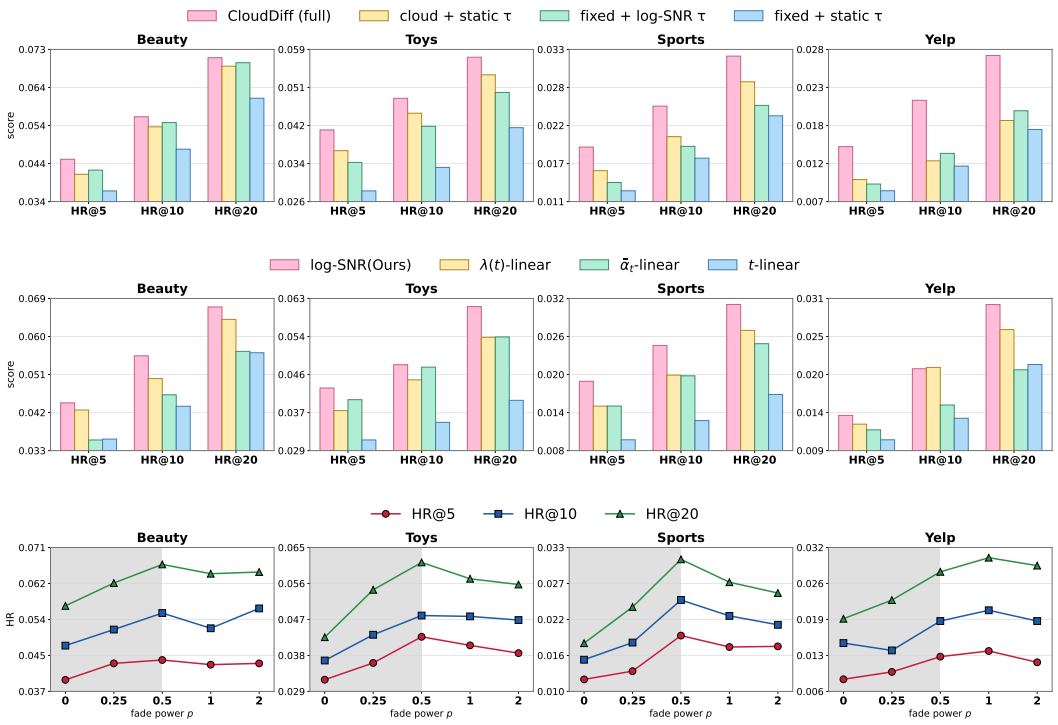


Figure 11: Additional HR results. Top: ablation study of CloudDiff. Middle: temperature-schedule variants. Bottom: sensitivity analysis of fade power p.

forward passes. As a result, CloudDiff adds only modest training overhead while leaving the inference pathway unchanged.

I Additional Results on Hit Ratio

We report HR@K results corresponding to the main-text experiments. While NDCG is presented in the main text as it reflects more fine-grained ranking quality, HR results are provided here due to space constraints. The top panel of Figure 11 corresponds to the ablation study in §4.3, while the middle and bottom panels correspond to the theoretical design validation in §4.4. Overall, the trends are consistent with those observed for NDCG. However, the performance gaps in the ablation and sensitivity analyses are slightly less pronounced under HR, suggesting that the proposed SNR-coupled mechanisms primarily improve fine-grained ranking quality, which is more directly captured by NDCG.

J Broader Impacts

CloudDiff is a sequential recommendation framework designed to better capture uncertainty in user preferences through SNR-coupled supervision. Potential positive and negative societal impacts are discussed below.

Positive impacts. By adapting supervisory granularity to the diffusion process, CloudDiff aims to produce more stable and semantically coherent recommendations. In particular, the framework explicitly moves beyond strict point-wise supervision by allowing semantically plausible alternatives to contribute during training, which may better reflect the ambiguity inherent in user preferences. The use of semantic neighborhoods derived from pretrained language model embeddings may also improve generalization for sparse or long-tail items, potentially reducing reliance on popularity-heavy collaborative signals.

Potential risks. Like recommender systems more broadly, CloudDiff may contribute to filter bubbles, popularity bias, or over-personalization that limits users' exposure to diverse content. In addition, because the semantic cloud is constructed from frozen language model embeddings, biases present in the underlying pretrained models may propagate into semantic neighborhoods and influence recommendation behavior.

Mitigation considerations. Existing mitigation strategies for recommender systems, including fairness-aware reranking, diversity regularization, and auditing of recommendation distributions, remain applicable to CloudDiff. We do not foresee dual-use risks specific to this work beyond those already associated with recommender systems and pretrained language models.