Proceedings of Machine Learning Research **318** (2026)
The 39th Canadian Conference on Artificial Intelligence

Canadian AI 2026
25–29 May 2026, Vancouver, BC

Diffusion-based Long and Short Term Interest Sequence Recommendation

Xiaowen Wang [†,*], Thomas Tran[†]
[†] University of Ottawa, Ottawa, ON, Canada

Abstract

Sequential recommendation requires modeling both stable long-term preferences and dynamic short-term intents. However, most existing methods rely on static fusion strategies, which cannot adaptively balance these signals. To address this, we propose DiffLSRec, a diffusion-based framework that performs progressive fusion of long- and short-term representations. The long-term embedding is treated as a prior, while short-term intent provides guidance during multi-step denoising, enabling dynamic and fine-grained integration. We further enhance short-term modeling with token-level contextual information and regulate the fusion process using SNR-adaptive guidance. Experiments on three Amazon datasets show that DiffLSRec consistently outperforms representative baselines across multiple metrics.

Keywords: Sequential Recommendation, Diffusion Models, Long- and Short-term Interest Modeling, Generative Modeling

1. Introduction

Recommender systems model user preferences from historical interactions, but user interests are dynamic and evolve over time, requiring the modeling of both long-term preferences and short-term intents. Sequential recommendation captures temporal dependencies, yet most existing methods rely on static fusion strategies or single representations, limiting their ability to adapt to rapid interest shifts.

To address this, we propose **DiffLSRec**, a diffusion-based framework that progressively fuses long-term and short-term representations. The long-term embedding serves as a stable prior, while short-term intent provides conditional guidance during multi-step denoising, enabling adaptive and fine-grained interest integration. We further enhance the model with **Token-level Contextual Enhancement** to capture recent interaction patterns via cross-attention, and **SNR-Adaptive Guidance** to dynamically regulate the balance between long- and short-term signals. Experiments on three benchmark datasets show that DiffLSRec consistently outperforms strong baselines.

2. Related Work

Conventional recommendation methods include collaborative filtering and matrix factorization, which capture global interactions but ignore temporal dynamics [1–3]. Sequential recommendation models user behavior with ordered interactions [4]. RNN-based methods (e.g., GRU4Rec [5–7]) capture short-term dependencies, while self-attention models (e.g., SASRec [8, 9]) model long-range patterns, but typically rely on a single representation. Recent works model multiple temporal scales using gating or multi-interest mechanisms, yet still rely on static fusion [10–13]. Diffusion-based approaches have been introduced for recommendation, leveraging iterative denoising for improved modeling [14–16]. However, they do not explicitly address adaptive fusion of long-term and short-term interests.

[*] xwang728@uottawa.ca

3. Methodology

DiffLSRec models long- and short-term user interests using a diffusion-based framework. It performs multi-step denoising, where short-term signals progressively refine long-term representations. We further introduce token-level contextual enhancement via cross-attention and SNR-adaptive guidance to regulate long–short balance.

As shown in Figure 1, interaction sequences are encoded into embeddings to derive long-term and short-term representations, which are fused through diffusion. The resulting representation is used for recommendation.

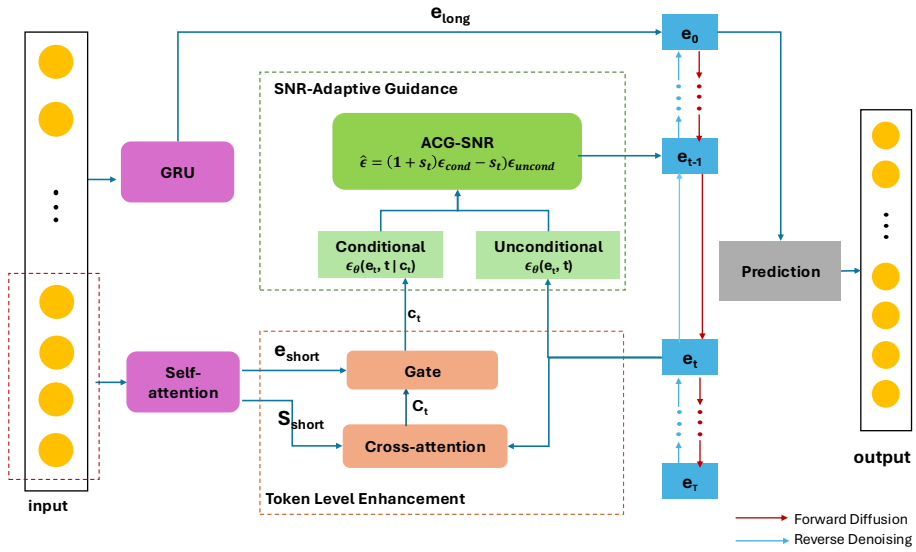

Figure 1. The overview framework of DiffLSRec model.

3.1. Diffusion

The long-term representation is progressively perturbed with Gaussian noise, producing a sequence $\{e_t\}_{t=1}^{T}$. In the reverse stage, the model reconstructs the clean representation from e_T via iterative denoising.

$$q(e_t \mid e_{long}) = \mathcal{N}(\sqrt{\bar{\alpha}_t}\, e_{long},\, (1-\bar{\alpha}_t)I), \quad \bar{\alpha}_t = \prod_{s=1}^{t}(1-\beta_s) \tag{3.1}$$

$$p_\theta(e_{t-1} \mid e_t, e_{short}, S_{short}) = \mathcal{N}(\mu_\theta(e_t, t, e_{short}, S_{short}), \sigma_t^2 I) \tag{3.2}$$

where e_t are latent representations at timesteps t, and e_{short} provides conditional guidance. β_s denotes the noise schedule. σ_t^2 follows a predefined variance schedule.

3.2. Enhancement

Token-level Contextual Enhancement The aggregated short-term embedding e_{short} may lose fine-grained context. We incorporate the most recent k item embeddings $S_{\text{short}} = [e_{T-k+1}, \ldots, e_{T-1}]$ as conditional tokens during reverse denoising.

At step t, the latent e_t interacts with these tokens via cross-attention:

$$C_t(e_t, S_{short}) = \text{softmax}\left(\frac{(W_Q e_t)(W_K S_{short})^\top}{\sqrt{d}}\right)(W_V S_{short}) \tag{3.3}$$

where W_Q, W_K, and W_V are projection matrices, and C_t captures token-level context.

We then balance C_t and e_{short} via:

$$c_t = \sigma(W_g[e_t; e_{short}]) \cdot C_t + (1 - \sigma(W_g[e_t; e_{short}])) \cdot e_{short} \tag{3.4}$$

where W_g is learnable and $\sigma(\cdot)$ is sigmoid. c_t is used as guidance for ε_θ.

Monotonic SNR-Adaptive Guidance We propose an SNR-aware guidance mechanism (ACG-SNR) that adjusts guidance strength during diffusion. Using SNR, it applies weaker short-term guidance in early stages and stronger refinement in later steps.

At each step, ACG-SNR interpolates between conditional and unconditional predictions:

$$\hat{\epsilon}_\theta^{\text{ACG-SNR}} = (1 + s_t)\,\epsilon_\theta(e_t, t \,|\, c_t) - s_t\,\epsilon_\theta(e_t, t \,|\, \varnothing) \tag{3.5}$$

where e_t is the noisy latent and $\epsilon_\theta(\cdot)$ is the noise prediction network.

The adaptive scale is:

$$s_t = s_{\max} \cdot \sigma\Big(-a\big(\tfrac{t}{T} - b\big) + \mathbf{u}^\top[\,\|e_t\|,\, \|e_{short}\|,\, \log(1 + \text{SNR}_t)\,] + b_0\Big) \tag{3.6}$$

where $\sigma(\cdot)$ bounds s_t, $\text{SNR}_t = \frac{\bar{\alpha}_t}{1-\bar{\alpha}_t}$, t/T denotes diffusion progress, a controls the rate, b determines the shift, and $\mathbf{u}^\top$ is the learnable projection. This keeps s_t small under high uncertainty and increases it as denoising progresses.

To stabilize reverse denoising, the clean latent $\hat{e}_0$ is estimated by removing the predicted noise, providing a stable reference for progressively recovering the representation.

$$\hat{e}_0(e_t, t, e_{\text{short}}, S_{\text{short}}) = \frac{e_t - \sqrt{1-\bar{\alpha}_t}\,\hat{\epsilon}_\theta^{\text{ACG-SNR}}}{\sqrt{\bar{\alpha}_t}} \quad \bar{\alpha}_t = \prod_{s=1}^{t}(1 - \beta_s) \tag{3.7}$$

$$e_{t-1} = \sqrt{\bar{\alpha}_{t-1}}\,\hat{e}_0(e_t, t, e_{\text{short}}, S_{\text{short}}) + \sqrt{1 - \bar{\alpha}_{t-1} - \sigma_t^2}\,\hat{\epsilon}_\theta^{\text{ACG-SNR}} + \sigma_t z \tag{3.8}$$

where $\sigma_t z$ adds mild Gaussian noise, yielding a smooth reverse trajectory that converges to e_0, the final fused user representation.

The reverse process outputs e_0, a fused representation capturing both long-term preferences and short-term intents, which is used for final recommendation.

4. Experiment

We conduct experiments on the Amazon Product Dataset (2023) across three domains: "Cell Phones and Accessories", "Movies and TV", and "Clothing Shoes and Jewelry". We construct a representative subset of 10,000 user interaction sequences per domain for computational feasibility, and evaluate all models under the same setting for fair comparison. Performance is measured using four top-N metrics: **HR@10**, **NDCG@10**, **MRR@10**, and **Precision@10**. We compare DiffLSRec with representative methods, including Fossil [17], MARank [12], GRU4Rec [5], HGN [11], SASRec [8], and DiffRec [14].

As shown in Table 1, DiffLSRec achieves the best performance across all datasets and metrics, consistently outperforming DiffRec and other baselines. The improvements are more pronounced on sparse datasets (Cellphone and Clothes), demonstrating strong robustness. DiffLSRec also outperforms DiffRec by +7.65% HR@10 on Cellphone, +8.73% on Clothes, and +2.53% on Movies. These gains are attributed to the multi-step diffusion-based fusion, which enables adaptive integration of long- and short-term interests. Overall, DiffLSRec achieves consistent improvements across different data scenarios.

We analyze the effect of embedding dimension $K \in \{8, 16, 32, 64, 128\}$. As shown in Figure 2, smaller dimensions consistently achieve the best performance, while larger ones

Table 1. Result of experiment

Dataset	Method	HR@10	NDCG@10	MRR@10	Prec@10
Cellphone	Fossil	0.14200000	0.06752663	0.03854444	0.01420000
	MARank	0.16266666	0.07918197	0.04461111	0.01626667
	GRU4Rec	0.22333333	0.10486140	0.05764259	0.02233333
	HGN	0.19933334	0.09574587	0.05421772	0.01993333
	SASRec	0.22733334	0.12611172	0.04825476	0.02273333
	DiffRec	0.27866667	0.15196856	0.06194497	0.02786667
	DiffLSRec	**0.29999999**	**0.16195426**	**0.06867434**	**0.02999999**
	Improvement	**7.65%**	**6.57%**	**10.86%**	**7.65%**
Clothes	Fossil	0.15733333	0.09125891	0.03220820	0.01573333
	MARank	0.16400000	0.09664540	0.03752249	0.01640000
	GRU4Rec	0.17333333	0.12531130	0.03099180	0.01733333
	HGN	0.16599999	0.11025910	0.03642751	0.01660000
	SASRec	0.18000001	0.12487048	0.03665503	0.01799999
	DiffRec	0.19066666	0.12571591	0.03780634	0.01906666
	DiffLSRec	**0.20733332**	**0.13277971**	**0.04104312**	**0.02073333**
	Improvement	**8.73%**	**5.62%**	**8.57%**	**8.73%**
Movies	Fossil	0.57002109	0.45017547	0.08733845	0.05700212
	MARank	0.62702322	0.52075344	0.08343806	0.06270233
	GRU4Rec	0.69950742	0.65578878	0.08368687	0.06995074
	HGN	0.72836030	0.67007812	0.08389436	0.07283604
	SASRec	0.74876845	0.67094975	0.09523251	0.07487685
	DiffRec	0.75158340	0.67806209	0.08949795	0.07515835
	DiffLSRec	**0.77058411**	**0.68713196**	**0.09086464**	**0.07705842**
	Improvement	**2.53%**	**1.34%**	**1.53%**	**2.53%**

degrade results. This trend is more pronounced on sparse datasets (Cellphone and Clothes), where increasing K leads to clear performance drops, indicating overfitting. A similar pattern is observed on Movies, despite its denser interactions. Overall, DiffLSRec is not highly sensitive to K within a reasonable range, and compact embeddings are sufficient for effective modeling.

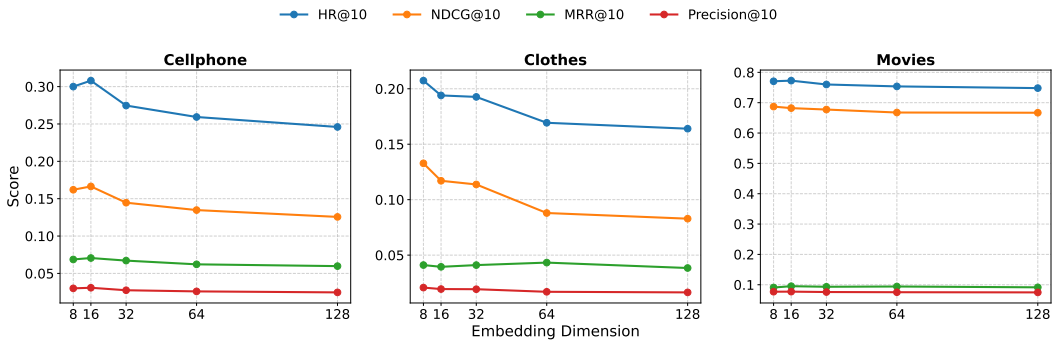

Figure 2. Effect of the embedding dimension K

We conduct an ablation study to evaluate the contribution of each component, including removing Token-level Contextual Enhancement (w/o Token), replacing SNR-adaptive guidance (w/o SNR), removing both (w/o Both), and a gate-based baseline. As shown in Figure 3, the full model achieves the best performance across all datasets and metrics. Removing either component degrades performance, while removing both leads to the largest drop. The gate-based baseline also underperforms the diffusion-based fusion. These results

demonstrate that token-level contextual modeling and SNR-adaptive guidance are complementary, and that multi-step diffusion provides more effective long–short interest fusion.

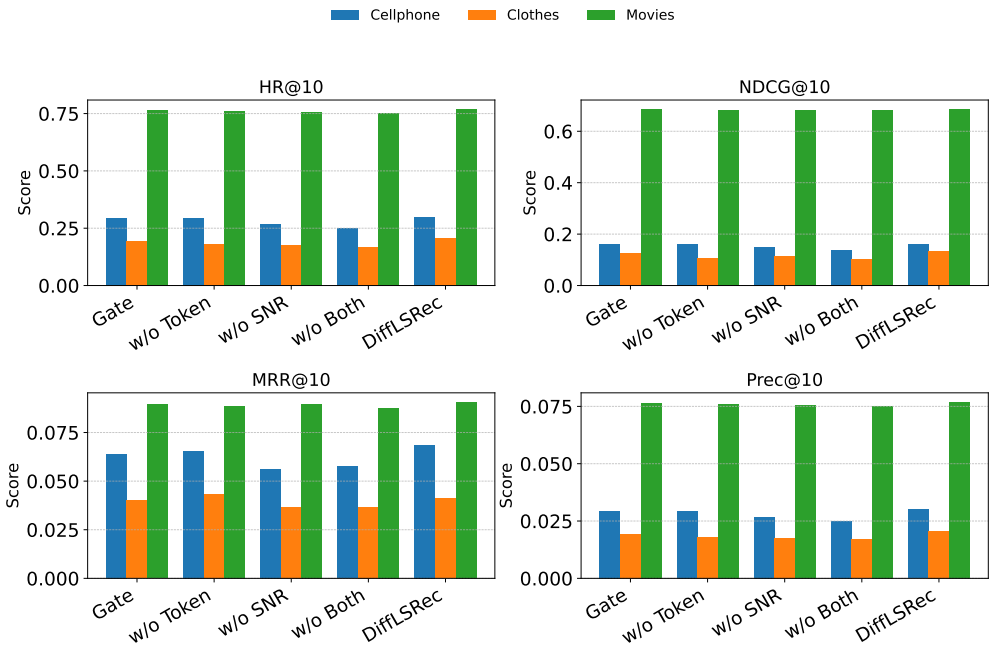

Figure 3. Ablation study of DiffLSRec and its variants.

5. Conclusion and Future Work

DiffLSRec introduces a diffusion-based long–short interest fusion framework that progressively refines user representations through multi-step denoising. By allowing short-term signals to guide long-term preferences at each step, it enables more effective user modeling. Experiments show consistent improvements over sequential and diffusion-based baselines.

Despite these gains, performance is still affected by data sparsity. Future work includes data-efficient strategies such as diffusion-based augmentation, LLM-driven item representations, and collaborative token modeling [18–21], as well as cross-domain learning [22].

Overall, DiffLSRec provides a generative refinement perspective for sequential recommendation, enabling more adaptive and robust modeling in sparse settings.

Acknowledgements

The authors thank reviewers for their valuable feedback.

References

[1] L. Zhang, L. Peng, et al. “Novel Recommendation of User-Based Collaborative Filtering”. In: *Journal of Digital Information Management* 12.3 (2014), pp. 165–175.

[2] D. Cai, X. He, et al. “Graph Regularized Nonnegative Matrix Factorization for Data Representation”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33.8 (2011), pp. 1548–1560.

[3] S. Rendle, C. Freudenthaler, et al. "BPR: Bayesian Personalized Ranking from Implicit Feedback". In: *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*. 2009, pp. 452–461.

[4] S. Rendle, C. Freudenthaler, et al. "Factorizing Personalized Markov Chains for Next-Basket Recommendation". In: *Proceedings of the 19th International Conference on World Wide Web*. 2010, pp. 811–820.

[5] B. Hidasi, A. Karatzoglou, et al. "Session-based Recommendations with Recurrent Neural Networks". In: *Proceedings of the 4th International Conference on Learning Representations*. 2016, pp. 1–10.

[6] B. Hidasi and A. Karatzoglou. "Recurrent Neural Networks with Top-k Gains for Session-based Recommendations". In: *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*. 2018, pp. 843–852.

[7] R. Pascanu, T. Mikolov, et al. "On the Difficulty of Training Recurrent Neural Networks". In: *Proceedings of the 30th International Conference on Machine Learning*. 2013, pp. 1310–1318.

[8] W. Kang and J. McAuley. "Self-Attentive Sequential Recommendation". In: *Proceedings of the 18th IEEE International Conference on Data Mining*. 2018, pp. 197–206.

[9] J. Li, Y. Wang, et al. "Time Interval Aware Self-Attention for Sequential Recommendation". In: *Proceedings of the 13th ACM International Conference on Web Search and Data Mining*. 2020, pp. 322–330.

[10] Y. Song, A. M. Elkahky, et al. "Multi-Rate Deep Learning for Temporal Recommendation". In: *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2016, pp. 909–912.

[11] C. Ma, P. Kang, et al. "Hierarchical Gating Networks for Sequential Recommendation". In: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2019, pp. 825–833.

[12] L. Yu, C. Zhang, et al. "Multi-Order Attentive Ranking Model for Sequential Recommendation". In: *Proceedings of the 33rd AAAI Conference on Artificial Intelligence*. 2019.

[13] Q. Pi, W. Bian, et al. "Practice on Long Sequential User Behavior Modeling for Click-Through Rate Prediction". In: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2019, pp. 2671–2679.

[14] W. Wang, Y. Xu, et al. "Diffusion Recommender Model". In: *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2023, pp. 832–841.

[15] Z. Li, A. Sun, et al. "DiffuRec: A Diffusion Model for Sequential Recommendation". In: *ACM Transactions on Information Systems* 42.3 (2024), 66:1–66:28.

[16] H. Lee and J. Kim. "EDiffuRec: An Enhanced Diffusion Model for Sequential Recommendation". In: *Mathematics* 12.12 (2024), p. 1795.

[17] R. He and J. McAuley. "Fusing Similarity Models with Markov Chains for Sparse Sequential Recommendation". In: *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining*. 2016, pp. 191–200.

[18] Q. Liu, F. Yan, et al. "Diffusion Augmentation for Sequential Recommendation". In: *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*. 2023, pp. 1576–1586.

[19] B. Zheng, Y. Hou, et al. "Adapting Large Language Models by Integrating Collaborative Semantics for Recommendation". In: *Proceedings of the 40th IEEE International Conference on Data Engineering*. 2024, pp. 1435–1448.

[20] J. Tan, S. Xu, et al. "IDGenRec: LLM-RecSys Alignment with Textual ID Learning". In: *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2024, pp. 355–364.

[21] Y. Wang, Z. Ren, et al. "Content-Based Collaborative Generation for Recommender Systems". In: *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*. 2024, pp. 2420–2430.

[22] C. Zhao, H. Zhao, et al. "Cross-domain Recommendation via User Interest Alignment". In: *Proceedings of the ACM Web Conference 2023*. 2023, pp. 887–896.

Appendix A. Embedding

To prepare user interaction data for modeling, DiffLSRec embeds raw inputs such as users, items, and temporal information into a low-dimensional latent space while preserving their underlying patterns.

Let U be the set of users and I the set of items. Each interaction is a tuple (u, i, t), with a user $u \in U$, item $i \in I$, and timestamp t. For each user u, the interaction history is the time-ordered item sequence $\{i_1, \ldots, i_{|I_u|}\}$ with timestamps $\{t_1, \ldots, t_{|I_u|}\}$. Each item i is assigned an embedding $e_i \in \mathbb{R}^d$ stored in a shared matrix E.

Temporal patterns are modeled through time intervals $\Delta t_j = |t_j - t_{j-1}|$ where $j \geq 2$. Intervals are normalized via Min-Max normalization to stabilize training and get $\Delta t'_j$.

Item frequency f_i is normalized to $[0, 1]$ and mapped to a trainable embedding e_{f_i}, providing additional information on item occurrence.

The user sequence is divided into long-term and short-term parts. The long-term sequence $I_{\text{long}} = \{i_1, \ldots, i_{|I_u|-2}\}$ captures stable interests, while the short-term sequence $I_{\text{short}} = \{i_{|I_u|-2-n}, \ldots, i_{|I_u|-2}\}$ focuses on the most recent n interactions (left-padded if shorter). Time-interval and frequency features are partitioned in the same way.

Appendix B. Long-term and Short-term Encoders

Long-term Encoder The long-term encoder $F_{long}(\cdot)$ is a GRU enhanced with time- and frequency-aware gates, capturing temporal decay and popularity effects,

$$g_t^{(\text{time})} = \sigma(W_\tau \, \Delta t_t + b_\tau) \tag{B.1}$$

$$g_t^{(\text{freq})} = \sigma(W_f \, f_t + b_f) \tag{B.2}$$

The two gates jointly determine an adaptive decay factor

$$\delta_t = \text{softplus}\Big(W_\delta [g_t^{(\text{time})} \| g_t^{(\text{freq})}] + b_\delta\Big) \tag{B.3}$$

which attenuates historical states before the GRU update. Frequency information further provides a directional modulation when forming the candidate state. The final long-term representation is taken as the last hidden state:

$$e_{\text{long}} = h_{u_l - 2} \tag{B.4}$$

Short-term Encoder The short-term encoder $F_{short}(\cdot)$ focuses on recent interactions. A GRU is first applied to extract local sequential dependencies, and a self-attention layer is then used to highlight the most relevant recent behaviors. Given the GRU hidden states H, the short-term representation is obtained via standard scaled dot-product attention:

$$e_{\text{short}} = \text{Attn}(H). \tag{B.5}$$

The resulting embeddings e_{long} and e_{short} capture user interests at different temporal scales. e_{long} summarizes stable, long-term preferences accumulated over the full interaction history, such as favored categories or consistently preferred item types. In contrast, e_{short} represents recent and rapidly changing intents shaped by recency or session-specific goals. Together, these complementary embeddings form the inputs to the diffusion-based fusion module.

Appendix C. Final Prediction

Overall, the diffusion-based fusion module defines a function F_θ that maps long- and short-term interests into a fused representation. After T denoising steps, integrating both stable long-term preferences and rapidly evolving short-term cues, the final user embedding

is:

$$e_{user} = e_0 = F_\theta(e_{\text{long}}, e_{\text{short}}, S_{\text{short}}) \tag{C.1}$$

The resulting e_{user} reflects a progressive refinement process in which noise is removed and short-term signals are injected via classifier-free guidance. Unlike static fusion methods, this multi-step refinement flexibly balances long- and short-term interests, while the guidance scale s allows shifting emphasis depending on context. The fused representation is then used for recommendation. In the prediction stage, e_{user} is projected into the item embedding space to compute relevance scores. Each item i has a trainable embedding e_i, and the interaction score is defined by:

$$\hat{y}_{u,i} = e_{user}^\top e_i \tag{C.2}$$

A higher score indicates stronger alignment between the user and item embeddings. Applying softmax over all items yields a probability distribution:

$$P(i \mid u) = \frac{\exp(\hat{y}u, i)}{\sum j \in \mathcal{I} \exp(\hat{y}_{u,j})} \tag{C.3}$$

The model is optimized by maximizing the likelihood of observed interactions through the cross-entropy loss:

$$\mathcal{L} = -\sum_{(u,i)\in\mathcal{D}} \log P(i \mid u) \tag{C.4}$$

which encourages e_{user} to align with interacted items while separating it from irrelevant ones. This prediction layer links the diffusion-based fused representation to next-item ranking, enabling efficient computation and fair comparison with sequential recommendation baselines while highlighting the adaptability of DiffLSRec.

The model is trained by minimizing the mean squared error between the true noise ϵ and the predicted noise ϵ_θ, allowing the network to accurately estimate noise at each step and progressively refine the long-term embedding under short-term guidance:

$$\mathcal{L}_{denoise} = \mathbb{E}_{t,e_{long},\epsilon,u}\left[\left\|\epsilon - \epsilon_\theta\Big(\sqrt{\bar{\alpha}_t}\, e_{long} + \sqrt{1-\bar{\alpha}_t}\,\epsilon,\ t \,\Big|\, c_t\Big)\right\|^2\right] \tag{C.5}$$

where $\mathbb{E}$ denotes expectation and $\|\cdot\|$ is the Euclidean norm. The conditioning variable c_t is applied with probability $1-p$ and replaced with $\varnothing$ with probability p.

To enhance stability and generalization, L2 regularization is applied to the projection matrices W_Q, W_K, W_V, the gating parameters W_g, and the feed-forward layers in the denoising network. The overall training objective is:

$$\mathcal{L}_{diff} = \mathcal{L}_{denoise} + \lambda_{reg}\big(\|W_Q\|_2^2 + \|W_K\|_2^2 + \|W_V\|_2^2 + \|W_g\|_2^2\big) \tag{C.6}$$

where λ_{reg} controls the strength of weight decay.

As conditioning is applied repeatedly across multiple steps, fusion becomes deeper and more flexible than single-step operators, enabling the model to emphasize short-term cues when needed and rely on long-term signals in more stable contexts. The reverse process outputs e_0 as the denoised representation, which is then used by the prediction layer.