GRAPH-TO-SEQUENCE KNOWLEDGE DISTILLATION METHOD FOR SEQUENTIAL RECOMMENDATION

Anonymous authors
Paper under double-blind review

ABSTRACT

The graph structure of user-item interactions and users' sequential behaviors are two fundamental signals in sequential recommendation (SR). Graph-based models excel at capturing high-order connectivity in the interaction graph, while sequential models are effective in modeling dynamic user preferences. However, the two embedding models learn different representation spaces, and naive fusion strategies often suffer from representation conflicts. To bridge this gap, we design **FRO**m **G**raph to **S**equence Knowledge Distillation (FROGS), a theoretically grounded method that transfers structural knowledge from a graph-based teacher to a sequential student model. Under common assumptions on embeddings, noise, and sample size, we derive upper bounds on the prediction error of three schemes, sequential-only, direct fusion, and graph-to-sequence distillation. Theoretically, we establish a distillability criterion for when knowledge distillation improves the prediction. We also analyze the bias–variance trade-off in FROGS distillation. Empirically, experiments on publicly available datasets and real-world data from an aggregated payment platform demonstrate consistent improvements over the evaluated baselines.

1 INTRODUCTION

Sequential recommendation (SR), which models users' sequential behaviors for next-item prediction, has become an essential component in real-world applications such as e-commerce platforms (Guo et al., 2025; Jansen et al., 2024) and food delivery services (Wang et al., 2025; Wan et al., 2024). In recent years, two major research lines have attracted interest for modeling user preferences in SR: graph-based recommendation and sequential recommendation.

Graph-based recommendation serves as a fundamental paradigm for modeling user–item interactions, where users and items are represented as nodes and collaborative signals are captured through global propagation (Wu et al., 2022a; Wang et al., 2021). Representative models such as NGCF (Wang et al., 2019c) and LightGCN (He et al., 2020) have shown significant efficacy in recommendation. Despite their success, graph-based methods still face the critical challenge of graph sparsity (Lei, 2016), which can significantly limit the quality of the embedding in recommendation scenarios with sparse interaction data (Yang & Toni, 2018; Idrissi & Zellou, 2020). Although recent studies have explored higher-order connectivity to alleviate this limitation (Liu et al., 2020; Zheng et al., 2024b), these methods still fail to capture users' dynamic behaviors (Xu et al., 2020).

Sequential recommendation has also achieved remarkable advancements by explicitly modeling the temporal order of historical interactions (Li et al., 2024). From GRU4Rec (Hidasi et al., 2016b;a) to recurrent neural network (RNN) variants (Hochreiter & Schmidhuber, 1997; Cho et al., 2014; Li et al., 2017; Wang et al., 2019a), this line of research has become another major paradigm for capturing dynamic user preferences. However, sequential models primarily rely on local interaction sequences and do not explicitly model global collaborative information among users and items (Xu et al., 2019), which may limit their ability to capture consistent global preferences or long-range dependencies, especially in sparse scenarios and for long-tail items (Wang et al., 2020).

Given the complementary strengths of graph-based and sequential recommendation models, a natural question is whether these strengths can be unified. Recent studies have attempted to integrate the two by directly fusing their learned representations (Wang et al., 2020; Wu et al., 2022b; Zhou

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.

& Lee, 2024). However, graph and sequential models operate in heterogeneous embedding spaces, so directly aligning or merging their representations can cause conflicts and diminish predictive accuracy (Chaudhuri et al., 2025). Even within a sequential recommender, using intermediate-layer representations has been shown to yield poor performance (Xu et al., 2025).

Knowledge distillation (KD) provides a natural solution for transferring soft logit-level knowledge across heterogeneous models (Hinton et al., 2015; Chaudhuri et al., 2025). Existing KD-based recommenders primarily focus on model compression within a single paradigm, such as ranking matching distillation (Reddi et al., 2021; Lee & Kim, 2021; Kang et al., 2020; 2023) and representation-level distillation (Xu et al., 2025), or extracting sequential patterns from ensemble logits (Du et al., 2023). However, few studies explore prediction-distribution-level KD from graph and sequence perspectives.

Our contributions. We formalize the representation conflict between graph and sequential models as an alignment residual that does not vanish with more data. This conflict arises when graph and sequential models operate in heterogeneous spaces. We prove that direct embedding fusion inherits this residual, while knowledge distillation avoids it because both models share item-indexed output coordinates. We derive high-probability prediction error bounds for sequential-only, direct-fusion, and graph-to-sequence distillation. We obtain a distillability criterion that predicts, prior to distillation, when it improves over both alternatives. We further characterize the bias–variance trade-off controlled by the graph neighborhood. Experiments on nine datasets, including publicly available benchmarks and real-world data from an aggregated payment platform, confirm the theory and the resulting method.

2 Related Work

Graph-based Collaborative Filtering Models. Recent advancements have focused on refining graph propagation efficiency and robustness. For instance, Zhang et al. (2024) optimize the architecture by performing GCN propagation only at prediction time with continuous relaxation. To further address the aforementioned data sparsity, several specialized strategies have emerged: Wang et al. (2019b) and Zhou et al. (2020) incorporate external knowledge graphs to augment user-item signals, while Sun et al. (2024) introduce sentiment analysis as auxiliary information. In parallel, contrastive learning (CL) has been introduced to alleviate sparsity by constructing augmented views through graph perturbation (Jaiswal et al., 2020; Bachman et al., 2019). To avoid explicit augmentation, Yu et al. (2023) inject noise into embeddings during propagation, enabling effective CL with minimal convolution operations.

Sequential Recommendation Models. Since user behaviors in recommender systems (RecSys) naturally exhibit temporal and sequential characteristics, a large body of sequential models has been proposed to capture dynamic user preferences (Li et al., 2024). Modern sequential models have shifted towards attention-based architectures to overcome the limitations of RNN architectures in modeling long-range dependencies. Building on attention-based architectures, ranging from lightweight attention (e.g., STAMP (Liu et al., 2018)) to full Transformer models (e.g., BERT4Rec (Sun et al., 2019)), various models have been proposed to achieve more precise weighting of historical behaviors (Li et al., 2017; Yuan et al., 2021; Xie et al., 2022). More recently, the focus has further evolved towards harnessing Large Language Models (LLMs) and their multimodal variants. These approaches, such as TALLRec (Bao et al., 2023) and other tuning frameworks (Zheng et al., 2024a; Ye et al., 2025), leverage collaborative semantics and pre-trained knowledge to better capture behavioral semantics.

Knowledge Distillation Approaches for RecSys. Knowledge distillation (KD), originally introduced by Hinton et al. (2015), is a machine learning technique that aims to transfer the acquired knowledge of a large pre-trained model to a smaller one. This paradigm was later applied to RecSys to develop compact yet effective models (Song et al., 2025). In recommendation settings, a mainstream paradigm is response-based distillation, where the student learns from the teacher's predicted distribution. Current methodologies are distinguished by how the distillation signal is constructed. Tang & Wang (2018b) emphasize transferring ranking information beyond labeled interactions. Lee et al. (2019) focus on preserving relative preference structures through sampling strategies, while Lee & Kim (2021) introduce correction mechanisms to refine distillation signals. Moving beyond output-level supervision, Kang et al. (2022) explore transferring richer user-specific preference in-

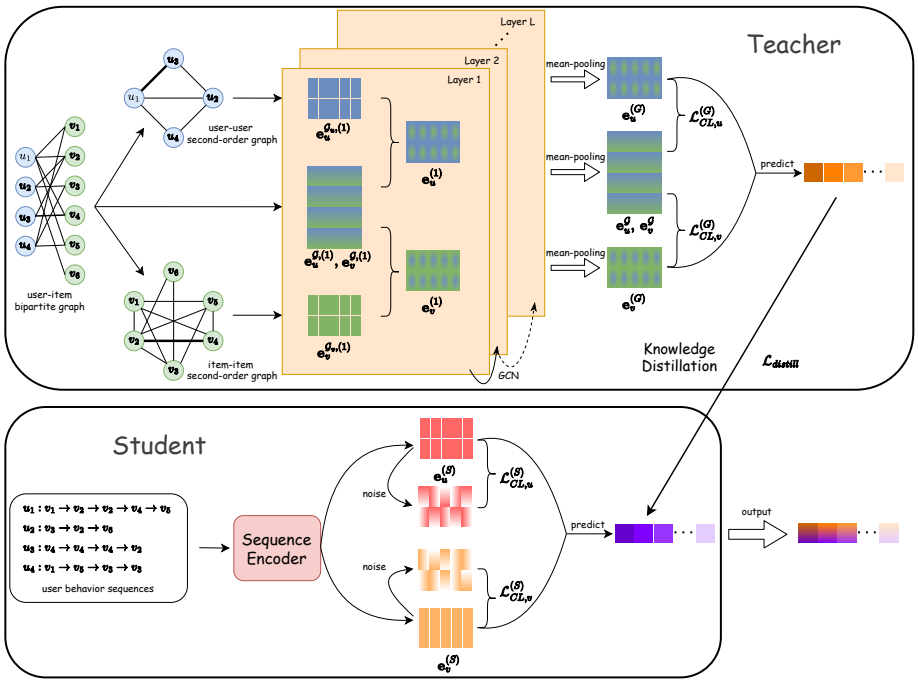


Figure 1: Overview of our proposed FROGS method.

formation. In addition, Chen et al. (2023) analyze distillation bias and propose an adjusted objective to mitigate it.

3 FROGS

In this section, we first define the recommendation problem formally. Then we present the proposed FROGS in detail, which consists of a graph-based teacher module, a sequential student module, and a distillability criterion and distillation objective function. The overall method of FROGS is illustrated in Figure 1.

Let $\mathcal{U}$ denote the set of users and $\mathcal{V}$ denote the set of items. Let N be the number of users and M be the number of items, i.e., $N = |\mathcal{U}|$ and $M = |\mathcal{V}|$. For each user $u_i, i \in \{1, 2, \ldots, N\}$, we define the behavior sequence as $\mathcal{S}_i = (v_{i1}, v_{i2}, \ldots, v_{is_i})$, where $v_{ik} \in \mathcal{V}$ for all $k \in \{1, 2, \ldots, s_i\}$, and s_i denotes the length of sequence $\mathcal{S}_i$. Given the behavior sequences $\mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_N$, the goal of recommendation is to predict the next-clicked item $v_{i,s_i+1} \in \mathcal{V}$ for each user u_i. Let d denote the shared embedding dimension. For variable-length sequences, set $n = \min_i s_i$ for conservative pre-distillation.

3.1 Representation Conflict

Graph and sequential models encode different signals. Let u_i^* and v_j^* denote the true user and item embeddings. Let $\hat{u}_i^G$ denote the learned graph-side user embedding. Let $\hat{v}_j^S$ and $\hat{v}_j^G$ denote the learned sequence-side and graph-side item embeddings. Let Q^* be the optimal orthogonal map from the graph space to the sequential space. Motivated by the empirical analysis in Appendix G, we define the alignment residual as

$$\Delta_{\text{align}} = \frac{1}{N} \sum_{i=1}^{N} \left\| u_i^* - Q^* \mathbb{E}[\hat{u}_i^G] \right\|_2 . \tag{1}$$

The expectation is over training randomness. A positive residual can persist as the sample size grows. Prediction distributions remain comparable because both models use the same item indices.

3.2 Graph-based Teacher Module

Following graph-based recommendation (Berg et al., 2017; He et al., 2020; Wang et al., 2019c; Wu et al., 2020; Zheng et al., 2018), we construct a user–item bipartite graph $\mathcal{G} = \langle \mathcal{W}, \mathcal{E} \rangle$, where $\mathcal{W} = \mathcal{U} \cup \mathcal{V}$, and let $\mathbf{R} \in \mathbb{R}^{N \times M}$ record interaction frequencies. To capture dependencies that a sparse bipartite graph and shallow propagation may fail to capture (Li & Chen, 2013; Zhang et al., 2024), we additionally construct second-order user and item graphs following Zhou et al. (2007). Their adjacency matrices are $\tilde{\mathbf{A}}^{\mathcal{G}} = \left[\begin{smallmatrix} \mathbf{0} & \mathbf{R} \\ \mathbf{R}^\top & \mathbf{0} \end{smallmatrix}\right]$, $\tilde{\mathbf{A}}^{\mathcal{G}_u} = \mathbf{R}\mathbf{R}^\top$, and $\tilde{\mathbf{A}}^{\mathcal{G}_v} = \mathbf{R}^\top\mathbf{R}$, where the latter two matrices encode user–user and item–item relations. For each $x \in \{\mathcal{G}, \mathcal{G}_u, \mathcal{G}_v\}$, symmetric normalization (Kipf, 2016) gives $\mathbf{A}^x = (\mathbf{D}^x)^{-1/2}\,\tilde{\mathbf{A}}^x\,(\mathbf{D}^x)^{-1/2}$, where $\mathbf{D}^x$ is the degree matrix of $\tilde{\mathbf{A}}^x$.

For pre-distillation, we use row-normalized user-graph weights: $w_{ii'} = \frac{\tilde{A}^{\mathcal{G}_u}_{ii'}}{\sum_{k=1}^{N} \tilde{A}^{\mathcal{G}_u}_{ik}}$, $\sum_{i'=1}^{N} w_{ii'} = 1$. We apply LightGCN (He et al., 2020) to all three graphs and combine their layer-wise representations using the adaptive fusion of Zheng et al. (2024b). For user u at layer l, $\mathbf{e}_u^{(l)} = \mathbf{e}_u^{\mathcal{G},(l)} + \alpha_u^{(l)} \mathbf{e}_u^{\mathcal{G}_u,(l)}$, with $\alpha_u^{(l)} = \left[l^2 + s^2(\mathbf{e}_u^{\mathcal{G},(l)}, \mathbf{e}_u^{\mathcal{G}_u,(l)})\right]^{-1/2}$ for $l = 1, 2, \ldots, L$, where $s(\cdot,\cdot)$ is cosine similarity; item representations are fused analogously. Following He et al. (2020), mean pooling over layers produces the final embeddings $\mathbf{E}^{(G),u}$ and $\mathbf{E}^{(G),v}$. We also retain the bipartite-only pooled embeddings $\mathbf{E}_{\mathrm{CL}}^{(G),u}$ and $\mathbf{E}_{\mathrm{CL}}^{(G),v}$ as auxiliary views. We align the fused and auxiliary views with InfoNCE (Oord et al., 2018), using $\mathcal{L}_{CL}^{(G)} = \mathrm{InfoNCE}(\mathbf{E}^{(G),u}, \mathbf{E}_{\mathrm{CL}}^{(G),u}, \tau_{\mathrm{CL}}) + \gamma\,\mathrm{InfoNCE}(\mathbf{E}^{(G),v}, \mathbf{E}_{\mathrm{CL}}^{(G),v}, \tau_{\mathrm{CL}})$, where τ_{CL} is the contrastive-learning temperature parameter and γ controls the contribution of the item-side contrastive term. The teacher computes $s_{u,v}^{(G)} = (\mathbf{e}_u^{(G)})^\top \mathbf{e}_v^{(G)}$. Its distribution over all candidate items is $\hat{\mathbf{y}}_u^{(G)} = \mathrm{softmax}(s_{u,v_1}^{(G)}/T, \ldots, s_{u,v_M}^{(G)}/T) \in \mathbb{R}^M$, where $T > 0$ controls distribution smoothness.

3.3 Sequential Student Module

We use the self-attention-based SASRec (Kang & McAuley, 2018) to encode order-sensitive user behavior. Given $\{\mathcal{S}_i\}_{i=1}^N$, it produces $\mathbf{E}^{(S),u}, \mathbf{E}^{(S),v} = \mathrm{SASRec}(\mathcal{S}_1, \ldots, \mathcal{S}_N)$, where $\mathbf{E}^{(S),u} = (\mathbf{e}_{u_1}^{(S)}, \mathbf{e}_{u_2}^{(S)}, \ldots, \mathbf{e}_{u_N}^{(S)})^\top \in \mathbb{R}^{N \times d}$ and $\mathbf{E}^{(S),v} = (\mathbf{e}_{v_1}^{(S)}, \mathbf{e}_{v_2}^{(S)}, \ldots, \mathbf{e}_{v_M}^{(S)})^\top \in \mathbb{R}^{M \times d}$ are the user and item embedding matrices.

For robustness, we perturb the user representations with Gaussian noise: $\mathbf{E}_{\mathrm{CL}}^{(S),u} = \mathbf{E}^{(S),u} + \boldsymbol{\epsilon}^u$, where $\boldsymbol{\epsilon}^u \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I})$ has the same shape as $\mathbf{E}^{(S),u}$. The contrastive loss is $\mathcal{L}_{CL}^{(S)} = \mathrm{InfoNCE}(\mathbf{E}^{(S),u}, \mathbf{E}_{\mathrm{CL}}^{(S),u}, \tau_{\mathrm{CL}})$. We regularize only user representations because they depend on the input sequence, whereas item embeddings are static parameters. The student computes scores by $\hat{s}_{u,v}^{(S)} = (\mathbf{e}_u^{(S)})^\top \mathbf{e}_v^{(S)}$. It then normalizes them as $\hat{\mathbf{y}}_u^{(S)} = \mathrm{softmax}(\hat{s}_{u,v_1}^{(S)}/T, \ldots, \hat{s}_{u,v_M}^{(S)}/T) \in \mathbb{R}^M$, using the same temperature T as the graph teacher.

3.4 Distillability Criterion

Train the graph and sequential modules independently first. Their embeddings enable pre-distillation screening. Let $\widehat{Q}$ be the orthogonal Procrustes map:

$$\widehat{Q} = \arg\min_{Q^\top Q = \mathbf{I}_d} \frac{1}{N} \sum_{i=1}^{N} \left\| \mathbf{e}_{u_i}^{(S)} - Q\mathbf{e}_{u_i}^{(G)} \right\|_2^2, \qquad \widehat{\Delta}_{\mathrm{align}} = \frac{1}{N} \sum_{i=1}^{N} \left\| \mathbf{e}_{u_i}^{(S)} - \widehat{Q}\mathbf{e}_{u_i}^{(G)} \right\|_2. \tag{2}$$

The residual $\widehat{\Delta}_{\mathrm{align}}$ measures graph–sequence mismatch. Define the effective neighborhood size and bias proxy $\widehat{\kappa} = \min_i \left(\sum_{i'=1}^{N} w_{ii'}^2\right)^{-1}$, $\widehat{b} = \frac{1}{N}\sum_{i=1}^{N} \left\| \sum_{i'=1}^{N} w_{ii'} \mathbf{e}_{u_{i'}}^{(S)} - \mathbf{e}_{u_i}^{(S)} \right\|_2$. Estimate $\widehat{\nu}$ from held-out residuals or bootstrap refits. For a planned fusion weight w, set $\tau = T$ and $\Lambda_\delta =$

$\sqrt{d+\log(NM/\delta)}$. Define $\widehat{\sigma}_S = \sqrt{\widehat{\nu}/n}$, $\widehat{\sigma}_G = \sqrt{\widehat{\nu}/(n\widehat{\kappa})}$, and $\widehat{\sigma}_w = \sqrt{(1-w)^2\widehat{\sigma}_S^2 + w^2\widehat{\sigma}_G^2}$. We then compare three approximate error bounds:

$$\widehat{\mathcal{B}}_S = \frac{2C\Lambda_\delta}{\tau}\widehat{\sigma}_S, \qquad \widehat{\mathcal{B}}_{\text{fuse}} = \frac{C}{\tau}\left(w\widehat{\Delta}_{\text{align}} + \widehat{\sigma}_w\Lambda_\delta\right), \qquad \widehat{\mathcal{B}}_{\text{KD}} = \frac{C}{\tau}\left(\widehat{b} + \widehat{\sigma}_G\Lambda_\delta\right) + \varepsilon_{\text{opt}}. \quad (3)$$

KD is promising when $\widehat{\mathcal{B}}_{\text{KD}} < \min\{\widehat{\mathcal{B}}_S, \widehat{\mathcal{B}}_{\text{fuse}}\}$. These are pre-distillation upper-bound proxies, not observed errors. The screening remains approximate and requires the assumptions below.

3.5 Distillation Loss and Final Objective

For each user u, let the graph-based and sequential prediction distributions be $\hat{\mathbf{y}}_u^{(G)} = (\hat{y}_{u,v_1}^{(G)}, \ldots, \hat{y}_{u,v_M}^{(G)})^\top$ and $\hat{\mathbf{y}}_u^{(S)} = (\hat{y}_{u,v_1}^{(S)}, \ldots, \hat{y}_{u,v_M}^{(S)})^\top$, respectively. Let $\mathcal{V}_u^+ = \{v \in \mathcal{V} : v \in \mathcal{S}_u\}$ and $\mathcal{V}_u^- = \mathcal{V} \setminus \mathcal{V}_u^+$ denote the observed and unobserved item sets. The distillation loss is

$$\mathcal{L}_{\text{distill}} = \frac{T^2}{N}\sum_{u\in\mathcal{U}}\left[\sum_{v\in\mathcal{V}_u^+}\hat{y}_{u,v}^{(G)}\log\frac{\hat{y}_{u,v}^{(G)}}{\hat{y}_{u,v}^{(S)}} + \beta\sum_{v\in\mathcal{V}_u^-}\hat{y}_{u,v}^{(G)}\log\frac{\hat{y}_{u,v}^{(G)}}{\hat{y}_{u,v}^{(S)}}\right]. \quad (4)$$

Here $\beta > 0$ controls the weight of the unobserved-item terms relative to the observed-item terms. The determination of β is evaluated empirically. The sequential student produces the final top-K recommendations and is supervised by the cross-entropy loss $\mathcal{L}_{\text{pred}} = -\sum_{u\in\mathcal{U}}\log\hat{y}_{u,v_u^+}^{(S)}$, where v_u^+ denotes the ground-truth next interacted item of user u. The student objective is $\mathcal{L}_{\text{student}} = \mathcal{L}_{\text{pred}} + \lambda_1\mathcal{L}_{\text{CL}}^{(S)} + \lambda_3\|\Theta_S\|_2^2$, where Θ_S denotes the regularized parameters of the sequential student. The teacher objective is $\mathcal{L}_{\text{teacher}} = \lambda_2\mathcal{L}_{\text{CL}}^{(G)}$, and the overall training objective is

$$\mathcal{L} = \mathcal{L}_{\text{student}} + \mathcal{L}_{\text{teacher}} + \lambda_4\mathcal{L}_{\text{distill}}. \quad (5)$$

4 Theoretical Analysis

Notation. Define $s_{ij}^* = \langle v_j^*, u_i^*\rangle$ and $p_{ij}^* = \text{softmax}(s_{i\cdot}^*/\tau)_j$. For a predictor $X \in \{S, \text{fuse}, \text{KD}\}$, let $\hat{p}_{i\cdot}^X$ be its output distribution and $\mathcal{E}_X = \frac{1}{N}\sum_i\|\hat{p}_{i\cdot}^X - p_{i\cdot}^*\|_1$.

Assumptions. We assume that the graph and sequential embedding spaces are related by an unknown orthogonal transformation Q, without restricting its form. Without loss of generality, we take the sequential embedding space as the reference space containing the true embeddings. The residual Δ_{align} is defined in Section 3.1 and may remain positive. We require $N \geq d$ and $M \geq d$, together with the output-layer constraint $\max\{\max_j\|v_j^S\|_2, \max_j\|v_j^G\|_2\} \leq \bar{c}_V$. Define the propagation on graphs as nonnegative weights $w_{ii'}$ satisfying $\sum_{i'} w_{ii'} = 1$. Let $\kappa_i = \frac{1}{\sum_{i'=1}^N w_{ii'}^2}, b_i = \left\|\sum_{i'=1}^N w_{ii'}u_{i'}^* - u_i^*\right\|_2, b = \frac{1}{N}\sum_{i=1}^N b_i$. For simplicity, we assume $\kappa = \kappa_i$ is the same for all users. For a distribution q over items, define $\text{Cov}_q(\hat{v}^S) = \sum_j q_j\hat{v}_j^S\hat{v}_j^{S\top} - (\sum_j q_j\hat{v}_j^S)(\sum_j q_j\hat{v}_j^S)^\top$. We assume that there exists $\eta > 0$ such that, on the event of Lemma 3, the following covariance stability condition holds for every user i: $\lambda_{\min}\left(\text{Cov}_{\hat{p}_{i\cdot}^S}(\hat{v}^S)\right) \geq \frac{\eta}{2d}$. The minimum-eigenvalue lower bound is a common condition in matrix computation theory.

Theorem 1 (Prediction error bounds for distillation) *Let* $\Lambda_\delta = \sqrt{d+\log(NM/\delta)}$, $\sigma_S = \sqrt{\nu/n}$, $\sigma_G = \sqrt{\nu/(n\kappa)}$, $\sigma_w = \sqrt{(1-w)^2\sigma_S^2 + w^2\sigma_G^2}$, *and* $\Delta_{\text{align}} \geq 2C(\sigma_S + \sigma_G)\Lambda_\delta$, *where* ν *is the single-observation noise scale;* ε_{opt} *is the student's residual error in matching the teacher distribution;* $w \in (0,1]$ *is the graph-side weight in fusion. There exists a constant* C *s.t.* $\forall\delta \in (0,1)$, *with probability at least* $1-3\delta$, *the following inequalities hold simultaneously*

$$\mathcal{E}_S \leq \frac{2C\Lambda_\delta}{\tau}\sqrt{\frac{\nu}{n}}, \quad (6)$$

$$\frac{c\eta w}{\tau d}e^{-2R_{\max}/\tau}\Delta_{\text{align}} - \frac{C\sigma_S\Lambda_\delta}{\tau} \leq \mathcal{E}_{\text{fuse}} \leq \frac{C}{\tau}\left(w\Delta_{\text{align}} + \sigma_w\Lambda_\delta\right), \quad (7)$$

$$\mathcal{E}_{\mathrm{KD}} \le \frac{C}{\tau}\Big[\, b \;+\; \sigma_G \Lambda_\delta \Big] \;+\; \varepsilon_{\mathrm{opt}}. \tag{8}$$

where $R_{\max} = 2w\bar{c}_V(r_u + \max_i \|\mathbb{E}[\hat{u}_i^G]\|_2 + C(\sigma_S + \sigma_G)\Lambda_\delta)$.

Corollary 1 (Distillability Criterion) *The KD upper bound is smaller than the sequential upper bound whenever*

$$\frac{\tau}{2}\varepsilon_{\mathrm{opt}} + b < C\Lambda_\delta \sqrt{\frac{\nu}{n}} \left(1 - \kappa^{-1/2}\right). \tag{9}$$

Equivalently, this holds for $n < n^*$*, where*

$$n^* = \left\lceil \frac{C\Lambda_\delta \sqrt{\nu}(1 - \kappa^{-1/2})}{\frac{\tau}{2}\varepsilon_{\mathrm{opt}} + b} \right\rceil^2. \tag{10}$$

Moreover, for a fixed $w \in (0, 1]$*, a sufficient condition for KD to outperform direct fusion is*

$$\Delta_{\mathrm{align}} > \frac{Cd}{\eta w} e^{2R_{\max}/\tau} \left[\tau\varepsilon_{\mathrm{opt}} + \Lambda_\delta(\sigma_S + \sigma_G) + b\right]. \tag{11}$$

Corollary 2 (Convergence rates of Prediction Error) *The leading errors are*

$$\mathcal{E}_S = O(n^{-1/2}), \qquad \mathcal{E}_{\mathrm{fuse}} = O(n^{-1/2}) + O(w\Delta_{\mathrm{align}}), \qquad \mathcal{E}_{\mathrm{KD}} = O((n\kappa)^{-1/2}) + O(b). \tag{12}$$

If κ *is bounded, KD preserves the* $n^{-1/2}$ *order but improves its variance constant by* $\kappa^{-1/2}$*; a faster order is possible only when* κ *grows with data size.*

Corollary 3 (Bias–Variance Trade-off in FROGS Distillation) *For a family of propagation weights such that* $b(\kappa)$ *is continuous and non-decreasing, the leading KD term*

$$F(\kappa) = b(\kappa) + C\Lambda_\delta \sqrt{\frac{\nu}{n\kappa}} \tag{13}$$

attains a minimum on any admissible compact interval of κ*.*

Theorem 1 compares the prediction errors of sequential-only, direct fusion, and knowledge distillation. These bounds follow from explicit assumptions and yield a criterion that could be evaluated before distillation in Corollary 1. They thus guide the method design and predict when distillation helps, rather than explain the gains after the fact. Corollary 2 characterizes how prediction error scales with sample size. Corollary 3 characterizes the bias–variance trade-off in FROGS distillation, showing that larger graph neighborhoods reduce variance but may increase aggregation bias. Together, these results distinguish the costs of transferring embeddings from those of transferring output distributions. Direct fusion can reduce statistical variance, but when the two learned embedding spaces are mismatched it also introduces an alignment term that does not vanish with additional observations. Distillation avoids cross-model alignment because both outputs use the same item-indexed coordinates, but its benefit depends on the graph-induced variance reduction being large enough to offset the propagation bias b and residual $\varepsilon_{\mathrm{opt}}$. Theoretically, the upper-bound comparison predicts the clear gains for users with small n and informative graph neighborhoods κ.

5 Experiments

In this section, we conduct extensive experiments to evaluate the proposed **FROGS** method from multiple perspectives, including overall effectiveness, the contribution of knowledge distillation, and the impact of key hyper-parameters. Specifically, we aim to answer the following research questions:

RQ1: Does FROGS outperform state-of-the-art recommendation models?

RQ2: Does the knowledge distillation mechanism contribute to the performance gain?

RQ3: How robust is FROGS under different hyper-parameter settings?

5.1 DATASETS

We evaluate FROGS on four publicly available benchmark datasets and five real-world datasets provided by Shouqianba, one of the largest aggregated payment platforms in China. The public benchmarks are *Yelp*[1], a business review dataset; *Amazon-Video-Game* (*Amazon*), a product category from the dataset introduced by Hou et al. (2024a); *Gowalla* (Cho et al., 2011), a location-based check-in dataset; and *Steam* (Kang & McAuley, 2018), a video game review dataset.

The five real-world datasets contain anonymized food delivery transactions from five universities in China, denoted as *campus 13*, *campus 38*, *campus 107*, *campus 151*, and *campus 18*. Each interaction represents an actual purchase. Their sparse interactions and short user sequences reflect the small-sample regime in which our theory predicts the greatest distillation gains.

Data Filtering. For computational efficiency, we retain the most recent data in each dataset according to the time windows specified in Appendix D.2. We further remove users and items with fewer than 10 interactions to reduce extreme sparsity and improve model reliability.

Data Splitting. For each user, let the purchase sequence be $\{i_1, i_2, \ldots, i_l\}$ with length l. We adopt a chronological leave-one-out strategy commonly used in sequential recommendation, using the first $t-1$ items as input to predict the t-th item for training ($t = 2, \ldots, l-2$), the first $l-2$ items to predict i_{l-1} for validation, and the first $l-1$ items to predict i_l for testing. This ensures strict chronological evaluation and prevents any information leakage from future interactions. Detailed statistics of each dataset are summarized in Table 4 in Appendix D.1.

5.2 EVALUATION METRICS AND BASELINES

Following prior surveys on SR (Wu et al., 2022a; Li et al., 2024), we adopt **Hit Rate** (H@K) and **Normalized Discounted Cumulative Gain** (N@K) as the primary evaluation metrics. The formal definitions of H@K and N@K are given in Appendix C.

To comprehensively evaluate the effectiveness of FROGS, we compare it with 14 state-of-the-art recommendation models, including 2 traditional methods (S-POP(Adomavicius & Tuzhilin, 2005), FPMC(Rendle et al., 2010)), 4 sequential methods (GRU4Rec(Hidasi et al., 2016b), Caser(Tang & Wang, 2018a), SASRec(Kang & McAuley, 2018), BERT4Rec(Sun et al., 2019)), 3 graph-based methods (MF(Koren et al., 2009), LightGCN(He et al., 2020), XSimGCL(Yu et al., 2023)), 3 KD methods (SimRec(Xia et al., 2023), UnKD(Chen et al., 2023), SLMRec(Xu et al., 2025)), 1 contrastive learning (CL) method (CL4SRec(Xie et al., 2022)), and 1 prompt-based zero-shot LLM ranker DeepSeek-V3.2(DS)(Hou et al., 2024b; Liu et al., 2025).

5.3 IMPLEMENTATION DETAILS

We implement all baseline models using their publicly available implementations and keep their default configurations except for the embedding dimension. For a fair comparison, the embedding size d of FROGS is set to 64. For FPMC and Caser, which maintain two separate embedding matrices for users and items, we use an embedding dimension of 32 so that their total parameter size remains comparable to other models. All remaining baseline models adopt an embedding dimension of 64. The hyperparameter settings for FROGS are provided in Appendix E. We use Adam optimizer with a learning rate of 10^{-3}. If the model's performance on the validation set does not improve for 10 consecutive epochs, the training process is terminated in advance. All the experiments are executed on a workstation with an Intel Xeon Platinum 8369B CPU, 29 GB memory, and an NVIDIA A10 GPU (24 GB).

5.4 OVERALL COMPARISON (RQ1)

We evaluate the proposed FROGS method on nine datasets introduced in Section 5.1, and compare it with all baseline models. For each dataset, we use H@5, N@5, H@10, N@10, H@20, and N@20 to comprehensively evaluate top-K ranking performance on the test set. Among them, the main results in terms of H@10 and N@10 are reported in Table 1, while the full results on all evaluation metrics are provided in Table 8 in Appendix H.

[1] https://business.yelp.com/data/resources/open-dataset/

Table 1: Performance comparison of different recommendation models. Bold indicates the best result, and underline indicates the second-best result.

Type	Dataset	Metrics	FROGS	Traditional		Sequential				Graph-Based			KD			CL	LLM
				S-POP	FPMC	GRU4Rec	Caser	SASRec	BERT4Rec	LightGCN	XSimGCL	MF	SimRec	UnKD	SLMRec	CL4SRec	DS
Publicly available datasets	*Yelp*	H@10	**0.0777**	0.0498	0.0558	0.0417	0.0579	0.0648	0.0641	0.0679	0.0722	0.0681	0.0704	0.0701	0.0257	0.0605	0.0478
		N@10	**0.0403**	0.0282	0.0274	0.0204	0.0300	0.0339	0.0332	0.0329	0.0346	0.0340	0.0386	0.0354	0.0149	0.0311	0.0261
	Amazon	H@10	**0.0922**	0.0031	0.0615	0.0528	0.0536	0.0687	0.0643	0.0495	0.0519	0.0407	0.0672	0.0576	0.0388	0.0649	0.0040
		N@10	**0.0432**	0.0009	0.0308	0.0274	0.0265	0.0304	0.0322	0.0257	0.0265	0.0188	0.0368	0.0297	0.0210	0.0308	0.0018
	Gowalla	H@10	**0.2094**	0.0002	0.1677	0.1119	0.1362	0.1642	0.1501	0.1508	0.1513	0.1303	0.1846	0.1515	0.1141	0.1532	0.0241
		N@10	**0.1084**	0.0001	0.0928	0.0621	0.0756	0.0878	0.0823	0.0843	0.0851	0.0728	0.0951	0.0859	0.0719	0.0822	0.0127
	Steam	H@10	**0.1788**	0.0663	0.1477	0.1672	0.1669	0.1636	0.1641	0.1446	0.1446	0.1418	0.1304	0.0786	0.1634	0.1546	0.0579
		N@10	**0.1062**	0.0321	0.0731	0.0949	0.0823	0.0914	0.0907	0.0710	0.0709	0.0700	0.0616	0.0327	0.0948	0.0776	0.0285
Real-world transaction data	*campus 13*	H@10	**0.5617**	0.3080	0.5145	0.4342	0.5128	0.4488	0.5467	0.3029	0.2879	0.3114	0.1741	0.0525	0.2264	0.4711	0.2997
		N@10	**0.4117**	0.1743	0.3455	0.2793	0.3789	0.2619	0.3996	0.1821	0.1750	0.1855	0.1047	0.0268	0.1597	0.2681	0.1770
	campus 38	H@10	**0.5086**	0.4238	0.4717	0.3796	0.4618	0.4498	0.4845	0.4125	0.3884	0.4099	0.2141	0.0782	0.2777	0.4476	0.4172
		N@10	**0.3558**	0.2735	0.3208	0.2470	0.3259	0.2817	0.3398	0.2660	0.2529	0.2681	0.1264	0.0424	0.2120	0.2776	0.2731
	campus 107	H@10	**0.4541**	0.3504	0.4029	0.3093	0.4108	0.3795	0.4370	0.3403	0.2997	0.3440	0.1528	0.0465	0.2103	0.3592	0.3341
		N@10	**0.3131**	0.2072	0.2618	0.1946	0.2950	0.2169	0.3064	0.2048	0.1860	0.2026	0.0852	0.0241	0.1515	0.2065	0.1981
	campus 151	H@10	**0.6061**	0.5716	0.5841	0.4861	0.5610	0.5524	0.5975	0.5476	0.5312	0.5466	0.2968	0.0768	0.2843	0.5514	0.5629
		N@10	**0.4348**	0.3619	0.4100	0.3268	0.4165	0.3429	0.4280	0.3624	0.3494	0.3644	0.1825	0.0398	0.2217	0.3457	0.3570
	campus 18	H@10	**0.4472**	0.2557	0.3769	0.3140	0.4023	0.3680	0.4403	0.2406	0.2201	0.2443	0.1092	0.0569	0.2274	0.3588	0.2518
		N@10	**0.3086**	0.1540	0.2362	0.1823	0.2817	0.2042	0.3039	0.1485	0.1364	0.1481	0.0562	0.0265	0.1587	0.2014	0.1566

FROGS consistently achieves the best performance across all datasets and evaluation metrics. Compared with 14 baselines, FROGS demonstrates clear and stable improvements in both hit-based and ranking-sensitive metrics. These results validate the effectiveness of our proposed multi-view knowledge distillation method, highlighting its ability to integrate global structural information and sequential dynamics into a unified and more expressive representation. The consistent gains across different datasets further confirm the robustness and generalization capability of FROGS. A 50-run Monte Carlo study with paired-sample t-tests against BERT4Rec on the five campus datasets further confirms the statistical significance of FROGS's improvements in most settings (Appendix I.1).

5.5 Ablation Study (RQ2)

Table 2: Ablation study on FROGS.

Metric	Method	Publicly available datasets				Real-world transaction data				
		Yelp	*Amazon*	*Gowalla*	*Steam*	*campus 13*	*campus 38*	*campus 107*	*campus 151*	*campus 18*
H@10	FROGS	**0.0777**	**0.0922**	**0.2094**	**0.1788**	**0.5617**	**0.5086**	**0.4541**	**0.6061**	**0.4472**
	Only S	0.0733	0.0894	0.2073	0.1762	0.5524	0.4940	0.4432	0.5764	0.4387
	Only G	0.0755	0.0434	0.1361	0.1411	0.2775	0.3906	0.3149	0.5082	0.2244
	S+G	0.0758	0.0907	0.1975	0.1757	0.5516	0.4965	0.4440	0.5841	0.4396
	MLP	0.0585	0.0613	0.1506	0.1669	0.4967	0.4136	0.3800	0.5043	0.3721
N@10	FROGS	**0.0403**	**0.0432**	**0.1084**	**0.1062**	**0.4117**	**0.3558**	**0.3131**	**0.4348**	**0.3086**
	Only S	0.0372	0.0430	0.1077	0.0980	0.4047	0.3416	0.3086	0.4209	0.3023
	Only G	0.0374	0.0214	0.0764	0.0696	0.1699	0.2553	0.1905	0.3430	0.1392
	S+G	0.0382	0.0419	0.1043	0.1036	0.4018	0.3518	0.3099	0.4296	0.3053
	MLP	0.0278	0.0309	0.0819	0.0967	0.3648	0.2953	0.2722	0.3669	0.2553

Note. S denotes Sequential Module. G denotes Graph-based Module. S+G and MLP denote the adaptive fusion and multilayer-perceptron-based fusion of representations generated by the Graph-based Module and the Sequential Module, respectively.

To verify the effectiveness of the FROGS design, particularly the role of knowledge distillation, we conduct an ablation study, with results reported in Table 2. The distillability criterion does not yield a single numeric threshold. Instead, it produces three upper-bound curves Eq.3 that vary with unknown quantities such as ε_{opt}. We compare jointly on the actual data, to determine whether distillation was expected to be worthwhile. We compare the proposed method with the sequential-only, graph-only, and direct-fusion baselines. Specifically, we remove the KD component and evaluate

two simplified variants that rely exclusively on the sequential module or the graph-based module for prediction. We further examine two variants that directly combine the representations of the two modules through adaptive fusion or multilayer-perceptron-based (MLP-based) fusion. The adaptive fusion variant follows the fusion formulation analyzed in Theorem 1, with the weight w treated as a trainable parameter.

Across all nine datasets, FROGS achieves the highest H@10 and N@10 among the ablation variants in Table 2. The graph-only variant falls well behind FROGS on most datasets, while Only S provides a stronger baseline but lacks the graph teacher's complementary structural information. Directly fusing the two representations does not close this gap: neither fusion variant surpasses FROGS, and MLP fusion underperforms Only S on both reported metrics for every dataset. Together, these results support the contribution of knowledge distillation, which transfers graph-derived predictive information to the sequential student without directly combining the two representation spaces. Nevertheless, the numerical advantage of FROGS over Only S is relatively modest in several cases. We therefore examine whether this advantage is stable when the evaluated users vary, while keeping the train/validation/test split and random seed fixed. On four selected datasets, we repeatedly draw paired bootstrap samples of test users and compare both models using the same sampled users and ranking metrics. The result in Table 10 in Appendix I.2 shows that FROGS significantly outperforms Only S, confirming the robustness of our conclusions.

5.6 Robustness to Hyper-parameters (RQ3)

We further analyze the sensitivity of FROGS with respect to the hyper-parameter β in Eq. equation 4. Specifically, we vary β from 0 to 30 and evaluate the model on four representative datasets. Taken together, Figure 2 and Table 5 suggests that datasets with more complex user behavior sequences tend to show stronger robustness to β and may benefit from a relatively larger value. This is theoretically intuitive, since when user sequences are more diverse and uncertain, the sequential student requires more complementary knowledge from the graph-based teacher to improve recommendation quality. In contrast, for datasets with lower sequence complexity, the model is generally more robust when β is within 10, and the optimal value should be searched within a smaller range. When β becomes excessively large, the model may overemphasize complementary knowledge and impede the acquisition of primary sequential knowledge, leading to degraded performance. In addition, we also conduct sensitivity analyses of FROGS to the temperature T and the KD loss weight λ_4, with results reported in Appendix F.

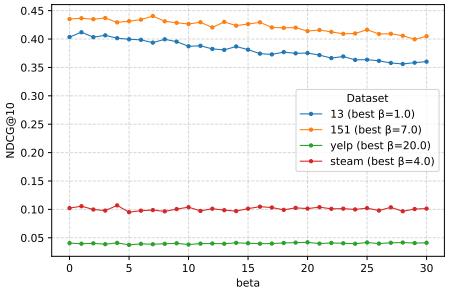


Figure 2: Effect of β on N@10.

6 Conclusions

In this work, we propose FROGS, a graph-to-sequence recommendation method that enhances sequential recommendation by distilling global structural knowledge from a graph-based teacher into a sequential student. Rather than directly fusing representations from heterogeneous architectures, our method transfers knowledge at the prediction level, which avoids potential representation conflict while still preserving the complementary strengths of both modules.

Our method has several limitations. First, when the interaction graph is severely sparse, the teacher may provide unreliable soft targets, which can weaken the effectiveness of distillation. Second, maintaining a parallel graph module introduces additional training overhead; before full deployment, one could use unbiased small-sample estimation to pre-evaluate the potential performance gain. Third, the behavior of our method on extremely short sequences (e.g., cold-start users with fewer than 3 interactions) remains underexplored. Additionally, comparisons with additional multi-view fusion methods remain for future work.

AI USE STATEMENT

In this work, we used generative AI tools for assisting with translation. We have not used generative AI tools for helping develop theoretical models or conceptual frameworks, assisting in the writing of proofs, proposing or refining hypotheses, designing or proposing feedback on research methodology or experiments, formulating mathematical claims, providing critical ingredients for proving mathematical claims, cleaning and reformatting dataset, supporting qualitative and thematic data analysis, interpreting results, implementing methods, and generating synthetic data sets is not applicable to this work. Additionally, we used generative AI tools for editing a research paper to improve readability. We have reviewed all AI-assisted work. We have checked all LLM-generated content, including text translated or polished by LLMs. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

In our experiments, five out of nine real-world datasets are provided by Shouqianba (Shanghai Shouqianba Internet Technology Co., Ltd.). The company has authorized us to use these datasets for academic research. All personally identifiable information, including but not limited to users' names or nicknames, campus names, has been removed and replaced by unique ID numbers. Moreover, we will not release these five datasets in accordance with the company's requirements.

REPRODUCIBILITY STATEMENT

We have made great efforts to ensure reproducibility. For the theoretical results, we state all the assumptions in Section 4 and Appendix B. Complete proofs of all the claims are included in Appendix B. For the datasets we use in this paper, we describe the data processing steps in Section 5.1 and provide descriptive statistics in Appendix D. For the four public datasets, we either cite the corresponding literature or provide a link in Section 5.1. For the five private datasets provided by Shouqianba, we cannot release them due to the company's data-sharing restrictions. Our code will be released upon acceptance.

REFERENCES

Gediminas Adomavicius and Alexander Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE transactions on knowledge and data engineering*, 17(6):734–749, 2005.

Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. *Advances in neural information processing systems*, 32, 2019.

Keqin Bao, Jizhi Zhang, Yang Zhang, Wenjie Wang, Fuli Feng, and Xiangnan He. Tallrec: An effective and efficient tuning framework to align large language model with recommendation. In *Proceedings of the 17th ACM conference on recommender systems*, pp. 1007–1014, 2023.

Rianne van den Berg, Thomas N Kipf, and Max Welling. Graph convolutional matrix completion. *arXiv preprint arXiv:1706.02263*, 2017.

Abhra Chaudhuri, Anjan Dutta, Tu Bui, and Serban Georgescu. A closer look at multimodal representation collapse. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=Vf9f7eNX6T.

Gang Chen, Jiawei Chen, Fuli Feng, Sheng Zhou, and Xiangnan He. Unbiased knowledge distillation for recommendation. In *Proceedings of the sixteenth ACM international conference on web search and data mining*, pp. 976–984, 2023.

Eunjoon Cho, Seth A Myers, and Jure Leskovec. Friendship and mobility: user movement in location-based social networks. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 1082–1090, 2011.

Kyunghyun Cho, Bart Van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder–decoder approaches. In *Proceedings of SSST-8, eighth workshop on syntax, semantics and structure in statistical translation*, pp. 103–111, 2014.

Hanwen Du, Huanhuan Yuan, Pengpeng Zhao, Fuzhen Zhuang, Guanfeng Liu, Lei Zhao, Yanchi Liu, and Victor S Sheng. Ensemble modeling with contrastive knowledge distillation for sequential recommendation. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 58–67, 2023.

Wenhao Guo, Jin Tian, and Minqiang Li. Reputation vs. price: sequential recommendations based on cue diagnosticity theory. *Journal of Retailing and Consumer Services*, 83:104157, 2025.

Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yongdong Zhang, and Meng Wang. Lightgcn: Simplifying and powering graph convolution network for recommendation. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pp. 639–648, 2020.

Balázs Hidasi, Massimo Quadrana, Alexandros Karatzoglou, and Domonkos Tikk. Parallel recurrent neural network architectures for feature-rich session-based recommendations. In *Proceedings of the 10th ACM conference on recommender systems*, pp. 241–248, 2016a.

Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. Session-based recommendations with recurrent neural networks. In *ICLR (Poster)*, 2016b. URL `http://arxiv.org/abs/1511.06939`.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8): 1735–1780, 1997.

Yupeng Hou, Jiacheng Li, Zhankui He, An Yan, Xiusi Chen, and Julian McAuley. Bridging language and items for retrieval and recommendation. *arXiv preprint arXiv:2403.03952*, 2024a.

Yupeng Hou, Junjie Zhang, Zihan Lin, Hongyu Lu, Ruobing Xie, Julian McAuley, and Wayne Xin Zhao. Large language models are zero-shot rankers for recommender systems. In *European conference on information retrieval*, pp. 364–381. Springer, 2024b.

Nouhaila Idrissi and Ahmed Zellou. A systematic literature review of sparsity issues in recommender systems. *Social Network Analysis and Mining*, 10(1):15, 2020.

Ashish Jaiswal, Ashwin Ramesh Babu, Mohammad Zaki Zadeh, Debapriya Banerjee, and Fillia Makedon. A survey on contrastive self-supervised learning. *Technologies*, 9(1):2, 2020.

Laura ZH Jansen, Kwabena E Bennin, Ellen van Kleef, and Ellen J Van Loo. Online grocery shopping recommender systems: Common approaches and practices. *Computers in human behavior*, 159:108336, 2024.

SeongKu Kang, Junyoung Hwang, Wonbin Kweon, and Hwanjo Yu. De-rrd: A knowledge distillation framework for recommender system. In *Proceedings of the 29th ACM international conference on information & knowledge management*, pp. 605–614, 2020.

SeongKu Kang, Dongha Lee, Wonbin Kweon, and Hwanjo Yu. Personalized knowledge distillation for recommender system. *Knowledge-Based Systems*, 239:107958, 2022.

SeongKu Kang, Wonbin Kweon, Dongha Lee, Jianxun Lian, Xing Xie, and Hwanjo Yu. Distillation from heterogeneous models for top-k recommendation. In *Proceedings of the ACM Web Conference 2023*, pp. 801–811, 2023.

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

TN Kipf. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.

Jae-woong Lee, Minjin Choi, Jongwuk Lee, and Hyunjung Shim. Collaborative distillation for top-n recommendation. In *2019 IEEE international conference on data mining (ICDM)*, pp. 369–378. IEEE, 2019.

Youngjune Lee and Kee-Eung Kim. Dual correction strategy for ranking distillation in top-n recommender system. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, pp. 3186–3190, 2021.

Jing Lei. A goodness-of-fit test for stochastic block models. *The Annals of Statistics*, pp. 401–424, 2016.

Jing Li, Pengjie Ren, Zhumin Chen, Zhaochun Ren, Tao Lian, and Jun Ma. Neural attentive session-based recommendation. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pp. 1419–1428, 2017.

Xin Li and Hsinchun Chen. Recommendation as link prediction in bipartite graphs: A graph kernel-based machine learning approach. *Decision Support Systems*, 54(2):880–890, 2013.

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

Aixin Liu, Aoxue Mei, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, et al. Deepseek-v3. 2: Pushing the frontier of open large language models. *arXiv preprint arXiv:2512.02556*, 2025.

Qiao Liu, Yifu Zeng, Refuoe Mokhosi, and Haibin Zhang. Stamp: short-term attention/memory priority model for session-based recommendation. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 1831–1839, 2018.

Yang Liu, Liang Chen, Xiangnan He, Jiaying Peng, Zibin Zheng, and Jie Tang. Modelling high-order social relations for item recommendation. *IEEE Transactions on Knowledge and Data Engineering*, 34(9):4385–4397, 2020.

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

Sashank Reddi, Rama Kumar Pasumarthi, Aditya Menon, Ankit Singh Rawat, Felix Yu, Seungyeon Kim, Andreas Veit, and Sanjiv Kumar. Rankdistil: Knowledge distillation for ranking. In *International Conference on Artificial Intelligence and Statistics*, pp. 2368–2376. PMLR, 2021.

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.

Haoyuan Song, Yibowen Zhao, Yixin Zhang, Hongxu Chen, and Lizhen Cui. Knowledge distillation based recommendation systems: A comprehensive survey. *Electronics*, 14(8):1538, 2025.

Bingli Sun, Xiao Song, Wenxin Li, Lu Liu, Guanghong Gong, and Yan Zhao. A user review data-driven supplier ranking model using aspect-based sentiment analysis and fuzzy theory. *Engineering Applications of Artificial Intelligence*, 127:107224, 2024.

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.

Jiaxi Tang and Ke Wang. Personalized top-n sequential recommendation via convolutional sequence embedding. In *Proceedings of the eleventh ACM international conference on web search and data mining*, pp. 565–573, 2018a.

Jiaxi Tang and Ke Wang. Ranking distillation: Learning compact ranking models with high performance for recommender system. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 2289–2298, 2018b.

Zhizhong Wan, Bin Yin, Junjie Xie, Fei Jiang, Xiang Li, and Wei Lin. Larr: Large language model aided real-time scene recommendation with semantic understanding. In *Proceedings of the 18th ACM Conference on Recommender Systems*, pp. 23–32, 2024.

Meirui Wang, Pengjie Ren, Lei Mei, Zhumin Chen, Jun Ma, and Maarten De Rijke. A collaborative session-based recommendation approach with parallel memory modules. In *Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval*, pp. 345–354, 2019a.

Shoujin Wang, Liang Hu, Yan Wang, Xiangnan He, Quan Z Sheng, Mehmet A Orgun, Longbing Cao, Francesco Ricci, and Philip S Yu. Graph learning based recommender systems: A review. *arXiv preprint arXiv:2105.06339*, 2021.

Xiang Wang, Xiangnan He, Yixin Cao, Meng Liu, and Tat-Seng Chua. Kgat: Knowledge graph attention network for recommendation. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 950–958, 2019b.

Xiang Wang, Xiangnan He, Meng Wang, Fuli Feng, and Tat-Seng Chua. Neural graph collaborative filtering. In *Proceedings of the 42nd international ACM SIGIR conference on Research and development in Information Retrieval*, pp. 165–174, 2019c.

Yuyan Wang, Long Tao, and Xian Xing Zhang. Recommending for a multi-sided marketplace: A multi-objective hierarchical approach. *Marketing Science*, 44(1):1–29, 2025.

Ziyang Wang, Wei Wei, Gao Cong, Xiao-Li Li, Xian-Ling Mao, and Minghui Qiu. Global context enhanced graph neural networks for session-based recommendation. In *Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval*, pp. 169–178, 2020.

Le Wu, Yonghui Yang, Kun Zhang, Richang Hong, Yanjie Fu, and Meng Wang. Joint item recommendation and attribute inference: An adaptive graph convolutional network approach. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pp. 679–688, 2020.

Shiwen Wu, Fei Sun, Wentao Zhang, Xu Xie, and Bin Cui. Graph neural networks in recommender systems: a survey. *ACM Computing Surveys*, 55(5):1–37, 2022a.

Yiqing Wu, Ruobing Xie, Yongchun Zhu, Xiang Ao, Xin Chen, Xu Zhang, Fuzhen Zhuang, Leyu Lin, and Qing He. Multi-view multi-behavior contrastive learning in recommendation. In *International conference on database systems for advanced applications*, pp. 166–182. Springer, 2022b.

Lianghao Xia, Chao Huang, Jiao Shi, and Yong Xu. Graph-less collaborative filtering. In *Proceedings of the ACM web conference 2023*, pp. 17–27, 2023.

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.

Chengfeng Xu, Pengpeng Zhao, Yanchi Liu, Victor S Sheng, Jiajie Xu, Fuzhen Zhuang, Junhua Fang, and Xiaofang Zhou. Graph contextualized self-attention network for session-based recommendation. In *Ijcai*, volume 19, pp. 3940–3946, 2019.

Da Xu, Chuanwei Ruan, Evren Korpeoglu, Sushant Kumar, and Kannan Achan. Inductive representation learning on temporal graphs. *arXiv preprint arXiv:2002.07962*, 2020.

Wujiang Xu, Qitian Wu, Zujie Liang, Jiaojiao Han, Xuying Ning, Yunxiao Shi, Wenfang Lin, and Yongfeng Zhang. SLMRec: Distilling large language models into small for sequential recommendation. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=G4wARwjF8M.

Kaige Yang and Laura Toni. Graph-based recommendation system. In *2018 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, pp. 798–802. IEEE, 2018.

Yuyang Ye, Zhi Zheng, Yishan Shen, Tianshu Wang, Hengruo Zhang, Peijun Zhu, Runlong Yu, Kai Zhang, and Hui Xiong. Harnessing multimodal large language models for multimodal sequential recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 13069–13077, 2025.

Junliang Yu, Xin Xia, Tong Chen, Lizhen Cui, Nguyen Quoc Viet Hung, and Hongzhi Yin. Xsimgcl: Towards extremely simple graph contrastive learning for recommendation. *IEEE Transactions on Knowledge and Data Engineering*, 36(2):913–926, 2023.

Jiahao Yuan, Zihan Song, Mingyou Sun, Xiaoling Wang, and Wayne Xin Zhao. Dual sparse attention network for session-based recommendation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pp. 4635–4643, 2021.

Weizhi Zhang, Liangwei Yang, Zihe Song, Henry Peng Zou, Ke Xu, Liancheng Fang, and Philip S Yu. Do we really need graph convolution during training? light post-training graph-ode for efficient recommendation. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pp. 3248–3258, 2024.

Bowen Zheng, Yupeng Hou, Hongyu Lu, Yu Chen, Wayne Xin Zhao, Ming Chen, and Ji-Rong Wen. Adapting large language models by integrating collaborative semantics for recommendation. In *2024 IEEE 40th International Conference on Data Engineering (ICDE)*, pp. 1435–1448. IEEE, 2024a.

Lei Zheng, Chun-Ta Lu, Fei Jiang, Jiawei Zhang, and Philip S Yu. Spectral collaborative filtering. In *Proceedings of the 12th ACM conference on recommender systems*, pp. 311–319, 2018.

ZhenZhong Zheng, Jianxin Li, Xiaoming Wu, Xiangzhi Liu, and Lili Pei. Higher-order graph contrastive learning for recommendation. In *International Conference on Database Systems for Advanced Applications*, pp. 35–51. Springer, 2024b.

Sijin Zhou, Xinyi Dai, Haokun Chen, Weinan Zhang, Kan Ren, Ruiming Tang, Xiuqiang He, and Yong Yu. Interactive recommender system via knowledge graph-enhanced reinforcement learning. In *Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval*, pp. 179–188, 2020.

Tao Zhou, Jie Ren, Matúš Medo, and Yi-Cheng Zhang. Bipartite network projection and personal recommendation. *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics*, 76(4): 046115, 2007.

Xiaofan Zhou and Kyumin Lee. Id and graph view contrastive learning with multi-view attention fusion for sequential recommendation. In *2024 IEEE International Conference on Big Data (BigData)*, pp. 690–699. IEEE, 2024.

A Notation Summary

Table 3 shows the key notations in this paper.

Table 3: Notations and descriptions

Notation	Description
$\mathcal{U}, \mathcal{V}$	Sets of users and items
$\mathcal{S}_i$	Behavior sequence of user u_i
N, M, d, n	Numbers of users, items, embedding dimensions, and observations per user
$u_i^*, v_j^*, s_{ij}^*, p_{ij}^*$	True user/item embeddings, score, and choice probability; $\|v_j^*\|_2 = 1$
τ	Temperature of the population choice distribution
$\hat{u}_i^S, \hat{u}_i^G, \hat{v}_j^S, \hat{v}_j^G$	Learned user and item embeddings of the sequential(S) and graph(G) models
$\hat{s}_{ij}^S, \hat{s}_{ij}^G, \hat{p}_{i\cdot}^S, \hat{p}_{i\cdot}^G$	Predicted scores and output distributions
$\mathcal{E}_S, \mathcal{E}_{\text{fuse}}, \mathcal{E}_{\text{KD}}$	Average L_1 distribution errors of the three strategies
b_i, b	User-level and average graph-propagation biases
$Q^*, \hat{Q}$	Population-optimal and empirical orthogonal alignment maps
$\bar{u}_i^G, \mu_i^G, r_i$	Mean graph embedding, its aligned version, and alignment residual
$\Delta_{\text{align}}, \hat{\Delta}_{\text{align}}$	Population and empirical mean alignment residuals
ζ_i^S, ζ_i^G	Sequential and aligned graph estimation noises
$\nu, \sigma_S, \sigma_G, \sigma_w$	Base, sequential, graph, and fused noise scales
Λ_δ	Uniform concentration factor $\sqrt{d + \log(NM/\delta)}$
w	Graph-side share in direct embedding fusion
ε_{opt}	Student residual in matching the teacher distribution
η	Non-degeneracy constant of the learned item covariance
$r_u, \bar{c}_V, R_{\max}$	Maximum true-user norm, learned-item norm bound, and fusion perturbation bound
CS	Finite-sample covariance stability condition
$\mathbf{A}^{\mathcal{G}}$	Normalized adjacency matrix of graph $\mathcal{G}$
$\mathcal{N}^{\mathcal{G}}(u)$	The set of neighbors of node u in graph $\mathcal{G}$
T	Temperature parameter in knowledge distillation
$\mathcal{V}_u^+, \mathcal{V}_u^-$	Observed and unobserved item sets for user u
β	Complementary knowledge weight in distillation

B Proofs for Theoretical Analysis

B.1 Additional Assumptions

Estimation noise. Let $\bar{u}_i^G = \mathbb{E}[\hat{u}_i^G]$, $\mu_i^G = Q^* \bar{u}_i^G$, $r_i = u_i^* - \mu_i^G$, and $r_u = \max_i \|u_i^*\|_2$. For a fixed data-generating process and training algorithm, write

$$\hat{u}_i^S = u_i^* + \zeta_i^S, \qquad Q^* \hat{u}_i^G = \mu_i^G + \zeta_i^G. \tag{14}$$

Conditional on the population parameters and graph structure, the error coordinates are independent, centered, and sub-Gaussian. The sequential variance proxy is ν/n, while the graph-side proxy is induced by the weighted propagation defined below; the two errors are conditionally independent. We further assume $\mathbb{E}[\hat{s}_{ij}^S] = s_{ij}^*$ and that $\hat{s}_{ij}^S - \mathbb{E}[\hat{s}_{ij}^S]$ and $\hat{s}_{ij}^G - \mathbb{E}[\hat{s}_{ij}^G]$ obey the corresponding sub-Gaussian tail bounds.

B.2 Preliminary Lemmas

Lemma 1 (Comparability of the two representation spaces) (i) *For every orthogonal matrix Q and vectors v, u, $\langle Qv, Qu\rangle = \langle v, u\rangle$. Thus, if $(\hat{V}, \hat{U})$ is a solution, $(\hat{V}Q, \hat{U}Q)$ gives identical scores and loss.*

(ii) *The output distributions $\hat{p}_{i\cdot}^S$ and $\hat{p}_{i\cdot}^G$ both lie in Δ^{M-1}, and their j-th coordinates refer to the same item j; hence, they are directly comparable without a cross-model transformation.*

Proof. Part (i) follows from $\langle Qv, Qu\rangle = v^\top Q^\top Q u = v^\top u$. For part (ii), both output layers apply softmax over the same indexed item set $\{1, \dots, M\}$, so their coordinates correspond one-to-one. □

Lemma 2 (Lipschitz continuity of softmax) *For any $s, \hat{s} \in \mathbb{R}^M$ and $\tau > 0$, let $\Delta = \hat{s} - s$. Then*

$$\|\text{softmax}(\hat{s}/\tau) - \text{softmax}(s/\tau)\|_1 \le \frac{\max_j \Delta_j - \min_j \Delta_j}{\tau} \le \frac{2}{\tau}\|\hat{s} - s\|_\infty. \tag{15}$$

Proof. Let $g(t) = \text{softmax}((s + t\Delta)/\tau)$. Then $\|g(1) - g(0)\|_1 \le \int_0^1 \|g'(t)\|_1 \, dt$, and

$$g_j'(t) = \frac{1}{\tau} g_j(t) \left(\Delta_j - \mathbb{E}_{g(t)}[\Delta]\right). \tag{16}$$

Therefore,

$$\|g'(t)\|_1 = \frac{1}{\tau}\sum_j g_j(t)\left|\Delta_j - \mathbb{E}_{g(t)}[\Delta]\right| \\ \leq \frac{1}{\tau}\max_j\left|\Delta_j - \mathbb{E}_{g(t)}[\Delta]\right| \leq \frac{\max_j \Delta_j - \min_j \Delta_j}{\tau} \leq \frac{2\|\Delta\|_\infty}{\tau}. \tag{17}$$

Integrating over t proves the result. □

Lemma 3 (Uniform high-probability estimation bounds) *Let* $\Lambda_\delta = \sqrt{d + \log(NM/\delta)}$, $\sigma_S = \sqrt{\nu/n}$, $\sigma_G = \sqrt{\nu/(n\kappa)}$. *There exists an absolute constant* $C > 0$, *independent of* n, d, N, M, *and* δ, *such that, with probability at least* $1 - \delta$, *the following inequalities hold simultaneously for all* $i \in [N]$ *and* $j \in [M]$:

$$\left|\hat{s}_{ij}^S - \mathbb{E}[\hat{s}_{ij}^S]\right| \leq C\sigma_S\Lambda_\delta, \qquad \left|\hat{s}_{ij}^G - \mathbb{E}[\hat{s}_{ij}^G]\right| \leq C\sigma_G\Lambda_\delta, \tag{18}$$

$$\|\zeta_i^S\|_2 \leq C\sigma_S\Lambda_\delta, \qquad \|\zeta_i^G\|_2 \leq C\sigma_G\Lambda_\delta. \tag{19}$$

Moreover, for any fixed $w \in [0,1]$, *if* ζ_i^S *and* ζ_i^G *are conditionally independent, then the same event can be chosen so that, simultaneously for all* $i \in [N]$,

$$\left\|(1-w)\zeta_i^S + w\zeta_i^G\right\|_2 \leq C\sigma_w\Lambda_\delta, \qquad \sigma_w = \sqrt{(1-w)^2\sigma_S^2 + w^2\sigma_G^2}. \tag{20}$$

Proof. Throughout the proof, $C > 0$ denotes a universal constant whose value may change from line to line. At the end of the proof, $C > 0$ is chosen as a fixed universal constant to make all the stated bounds hold simultaneously.

Let $\zeta \in \mathbb{R}^d$ have independent, centered, σ-sub-Gaussian coordinates. The standard concentration inequality for the Euclidean norm of a sub-Gaussian random vector gives

$$\mathbb{P}\left(\|\zeta\|_2 \geq C\sigma(\sqrt{d} + t)\right) \leq e^{-t^2}, \qquad t \geq 0. \tag{21}$$

Applying equation 21 to ζ_i^S with $\sigma = \sigma_S$, choosing

$$t = \sqrt{\log(5N/\delta)},$$

and taking a union bound over $i \in [N]$, we obtain, with failure probability at most $\delta/5$,

$$\max_{i\in[N]}\|\zeta_i^S\|_2 \leq C\sigma_S\left(\sqrt{d} + \sqrt{\log(5N/\delta)}\right).$$

The same argument applied to ζ_i^G with $\sigma = \sigma_G$ gives, with failure probability at most $\delta/5$,

$$\max_{i\in[N]}\|\zeta_i^G\|_2 \leq C\sigma_G\left(\sqrt{d} + \sqrt{\log(5N/\delta)}\right).$$

By assumptions, the centered score errors $\hat{s}_{ij}^S - \mathbb{E}[\hat{s}_{ij}^S]$ and $\hat{s}_{ij}^G - \mathbb{E}[\hat{s}_{ij}^G]$ are respectively σ_S- and σ_G-sub-Gaussian. Hence, for every $x \geq 0$,

$$\mathbb{P}\left(\left|\hat{s}_{ij}^S - \mathbb{E}[\hat{s}_{ij}^S]\right| \geq x\right) \leq 2\exp\left(-\frac{x^2}{2\sigma_S^2}\right), \tag{22}$$

$$\mathbb{P}\left(\left|\hat{s}_{ij}^G - \mathbb{E}[\hat{s}_{ij}^G]\right| \geq x\right) \leq 2\exp\left(-\frac{x^2}{2\sigma_G^2}\right). \tag{23}$$

Choosing

$$x_S = \sigma_S\sqrt{2\log(10NM/\delta)}, \qquad x_G = \sigma_G\sqrt{2\log(10NM/\delta)},$$

and taking a union bound over all $(i,j) \in [N]\times[M]$, the two score bounds each fail with probability at most $\delta/5$.

For the fusion bound, fix $w \in [0,1]$. Conditional independence of ζ_i^S and ζ_i^G implies that each coordinate of

$$(1-w)\zeta_i^S + w\zeta_i^G$$

is centered sub-Gaussian with variance proxy

$$\sigma_w^2 = (1-w)^2\sigma_S^2 + w^2\sigma_G^2.$$

Applying equation 21 with $\sigma = \sigma_w$ and taking a union bound over $i \in [N]$ gives equation 20 with failure probability at most $\delta/5$.

Finally, a union bound over the five families of events above shows that all the stated inequalities hold simultaneously with probability at least $1-\delta$. Since

$$\sqrt{a} + \sqrt{b} \le \sqrt{2(a+b)}$$

and fixed numerical terms such as $\log 5$ and $\log 10$ can be absorbed into a universal multiplicative constant, all the bounds can be written using

$$\Lambda_\delta = \sqrt{d + \log(NM/\delta)}.$$

Choosing C large enough to absorb these universal constants completes the proof. □

Lemma 4 (Variance-proxy equivalence of graph propagation) *Conditional on the graph, suppose that the user-level observation errors $\{\bar{\epsilon}_{i'}\}_{i'=1}^N$ are independent and that each coordinate is centered sub-Gaussian with variance proxy ν/n. Then each coordinate of the centered graph-side estimator $\sum_{i'} w_{ii'}\bar{\epsilon}_{i'}$ is centered sub-Gaussian with variance proxy $\nu/(n\kappa)$. Thus, in terms of the sub-Gaussian variance proxy, the graph-side estimator has the same noise scale as an estimator based on $n\kappa$ independent observations of the target user.*

Proof. Fix a user i and a coordinate $\ell \in [d]$, and write $\zeta_{i\ell}^G = \sum_{i'} w_{ii'}\bar{\epsilon}_{i'\ell}$. By conditional independence and the sub-Gaussian moment-generating-function bound, for every $\lambda \in \mathbb{R}$,

$$\mathbb{E}\big[\exp\big(\lambda\zeta_{i\ell}^G\big) \mid \mathcal{G}\big] = \prod_{i'} \mathbb{E}[\exp(\lambda w_{ii'}\bar{\epsilon}_{i'\ell}) \mid \mathcal{G}] \tag{24}$$

$$\le \prod_{i'} \exp\left(\frac{\lambda^2 w_{ii'}^2 \nu}{2n}\right) = \exp\left(\frac{\lambda^2\nu}{2n}\sum_{i'} w_{ii'}^2\right) = \exp\left(\frac{\lambda^2\nu}{2n\kappa}\right), \tag{25}$$

where $\mathcal{G}$ denotes the given graph structure. Hence, each coordinate of ζ_i^G is centered sub-Gaussian with variance proxy

$$\sigma_G^2 = \frac{\nu}{n}\sum_{i'} w_{ii'}^2 = \frac{\nu}{n\kappa}. \tag{26}$$

Defining the variance-proxy-equivalent additional sample size as $m = n(\kappa-1)$ gives $n+m = n\kappa$ and $\sigma_G = \sqrt{\nu/(n+m)} = \sqrt{\nu/(n\kappa)}$. Since $w_{ii'} \ge 0$ and $\sum_{i'} w_{ii'} = 1$, we have $\sum_{i'} w_{ii'}^2 \le 1$, and hence $\kappa \ge 1$ and $m \ge 0$. The quantity $n+m$ is a variance-proxy-equivalent sample size and need not be an integer. □

Lemma 5 (Orthogonal alignment objective) *The population alignment objective attains its minimum over $\{Q : Q^\top Q = I_d\}$. Define*

$$\hat{Q} \in \arg\min_{Q^\top Q = I_d} \frac{1}{N}\sum_{i=1}^N \|\hat{u}_i^S - Q\hat{u}_i^G\|_2^2. \tag{27}$$

Then, for every orthogonal Q,

$$\frac{1}{N}\sum_i \|\hat{u}_i^S - \hat{Q}\hat{u}_i^G\|_2^2 \le \frac{1}{N}\sum_i \|\hat{u}_i^S - Q\hat{u}_i^G\|_2^2. \tag{28}$$

Proof. The orthogonal group is compact and both objectives are continuous, so their minima exist. The inequality follows directly from the optimality of $\hat{Q}$. □

Lemma 6 (Population and empirical alignment residuals) *Define*

$$\hat{\Delta}_{\text{align}} = \frac{1}{N}\sum_{i=1}^N \|\hat{u}_i^S - \hat{Q}\hat{u}_i^G\|_2. \tag{29}$$

With probability at least $1-\delta$,

$$\left|\hat{\Delta}_{\text{align}} - \Delta_{\text{align}}\right| \le C(\sigma_S + \sigma_G)\Lambda_\delta. \tag{30}$$

Proof. Using the optimality of $\hat{Q}$ and the uniform bounds in Lemma 3,

$$\begin{aligned}\hat{\Delta}_{\text{align}} &\le \frac{1}{N}\sum_i \|\hat{u}_i^S - Q^*\hat{u}_i^G\|_2 \\ &\le \frac{1}{N}\sum_i \left(\|r_i\|_2 + \|\zeta_i^S\|_2 + \|\zeta_i^G\|_2\right) \le \Delta_{\text{align}} + C(\sigma_S + \sigma_G)\Lambda_\delta. \end{aligned} \tag{31}$$

Conversely, the optimality of Q^* gives

$$\begin{aligned}\Delta_{\text{align}} &\le \frac{1}{N}\sum_i \|u_i^* - \hat{Q}\bar{u}_i^G\|_2 \\ &\le \frac{1}{N}\sum_i \left(\|\hat{u}_i^S - \hat{Q}\hat{u}_i^G\|_2 + \|\zeta_i^S\|_2 + \|\hat{Q}(\hat{u}_i^G - \bar{u}_i^G)\|_2\right) \\ &\le \hat{\Delta}_{\text{align}} + C(\sigma_S + \sigma_G)\Lambda_\delta. \end{aligned} \tag{32}$$

Orthogonality implies $\|\hat{Q}(\hat{u}_i^G - \bar{u}_i^G)\|_2 = \|\zeta_i^G\|_2$. Combining the two directions proves equation 30. The maps $\hat{Q}$ and Q^* need not coincide; only their optimality for their respective objectives is used. □

Lemma 7 (Anti-concentration of softmax perturbations) *Let* $q = \mathrm{softmax}(s/\tau)$ *and* $q' = \mathrm{softmax}((s+\Delta)/\tau)$. *Define*

$$R = \max_j \Delta_j - \min_j \Delta_j, \qquad \sigma_\Delta^2 = \mathrm{Var}_{j\sim q}(\Delta_j). \tag{33}$$

If $R > 0$, *then*

$$\|q' - q\|_1 \ge \frac{1}{2}e^{-2R/\tau}\frac{\sigma_\Delta^2}{\tau R}. \tag{34}$$

For $R = 0$, $q' = q$.

Proof. Let $X_j = \Delta_j/\tau$, $\rho = R/\tau$, $\bar{X} = \mathbb{E}_q[X]$, and $\widetilde{X}_j = X_j - \bar{X}$. Then $|\widetilde{X}_j| \le \rho$ and

$$q'_j = \frac{q_j e^{\widetilde{X}_j}}{Z}, \qquad Z = \mathbb{E}_q[e^{\widetilde{X}}] \in [1, e^\rho]. \tag{35}$$

Thus,

$$\|q' - q\|_1 = \frac{1}{Z}\mathbb{E}_q\left|e^{\widetilde{X}} - \mathbb{E}_q[e^{\widetilde{X}}]\right|. \tag{36}$$

For any random variable Y and an independent copy Y',

$$\mathbb{E}|Y - \mathbb{E}Y| \le \mathbb{E}|Y - Y'| \le 2\mathbb{E}|Y - \mathbb{E}Y|, \tag{37}$$

so $\mathbb{E}|Y - \mathbb{E}Y| \ge \frac{1}{2}\mathbb{E}|Y - Y'|$. Taking $Y = e^{\widetilde{X}}$ and applying the mean value theorem,

$$\begin{aligned}\mathbb{E}\left|e^{\widetilde{X}} - \mathbb{E}[e^{\widetilde{X}}]\right| &\ge \frac{1}{2}e^{-\rho}\mathbb{E}_{j,k}|\widetilde{X}_j - \widetilde{X}_k| \\ &\ge \frac{1}{2}e^{-\rho}\mathbb{E}|\widetilde{X}|. \end{aligned} \tag{38}$$

Because $\widetilde{X}^2 \le \rho|\widetilde{X}|$,

$$\mathbb{E}|\widetilde{X}| \ge \frac{\mathbb{E}[\widetilde{X}^2]}{\rho} = \frac{\sigma_\Delta^2}{\tau R}. \tag{39}$$

Finally, $1/Z \ge e^{-\rho}$ yields equation 34. □

Lemma 8 (Finite-sample conditions for the fusion lower bound) *Let*

$$\hat{r}_i = \hat{u}_i^S - Q^*\hat{u}_i^G, \qquad \Delta_{ij} = -w\langle\hat{v}_j^S, \hat{r}_i\rangle, \qquad R_i = \max_j \Delta_{ij} - \min_j \Delta_{ij}. \tag{40}$$

Suppose $w \in (0,1]$, $\Delta_{\text{align}} \geq 2C(\sigma_S + \sigma_G)\Lambda_\delta$, *and, on the event of Lemma 3, the covariance stability condition*

$$\lambda_{\min}\Big(\mathrm{Cov}_{\hat{p}_{i\cdot}^S}(\hat{v}^S)\Big) \geq \frac{\eta}{2d}, \qquad \forall i \tag{41}$$

holds. Then

$$R_i \leq R_{\max} := 2w\bar{c}_V\left(r_u + \max_i \|\mu_i^G\|_2 + C(\sigma_S + \sigma_G)\Lambda_\delta\right), \tag{42}$$

$$\sigma_{\Delta,i}^2 := \mathrm{Var}_{j\sim\hat{p}_{i\cdot}^S}(\Delta_{ij}) \geq w^2\frac{\eta}{2d}\|\hat{r}_i\|_2^2, \qquad \frac{1}{N}\sum_i \|\hat{r}_i\|_2 \geq \frac{\Delta_{\text{align}}}{2}. \tag{43}$$

Proof. Work on the event of Lemma 3 where equation 41 holds. Since $\hat{r}_i = r_i + \zeta_i^S - \zeta_i^G$, the triangle inequality gives $\|\hat{r}_i\|_2 \leq \|r_i\|_2 + C(\sigma_S + \sigma_G)\Lambda_\delta$. By Cauchy–Schwarz and the item-norm constraint,

$$R_i \leq 2\max_j |\Delta_{ij}| \leq 2w\bar{c}_V\|\hat{r}_i\|_2 \leq R_{\max}, \tag{44}$$

where the last inequality uses $\|r_i\|_2 = \|u_i^* - \mu_i^G\|_2 \leq r_u + \max_k \|\mu_k^G\|_2$.

Next, the variance of the linear form defining Δ_{ij} satisfies

$$\begin{aligned}\sigma_{\Delta,i}^2 &= w^2\hat{r}_i^\top \mathrm{Cov}_{\hat{p}_{i\cdot}^S}(\hat{v}^S)\hat{r}_i \\ &\geq w^2\frac{\eta}{2d}\|\hat{r}_i\|_2^2,\end{aligned} \tag{45}$$

by the minimum-eigenvalue bound in equation 41.

Finally, the reverse triangle inequality and the same uniform noise bounds give

$$\begin{aligned}\frac{1}{N}\sum_i \|\hat{r}_i\|_2 &\geq \frac{1}{N}\sum_i \left(\|r_i\|_2 - \|\zeta_i^S\|_2 - \|\zeta_i^G\|_2\right) \\ &\geq \Delta_{\text{align}} - C(\sigma_S + \sigma_G)\Lambda_\delta \geq \frac{\Delta_{\text{align}}}{2}.\end{aligned} \tag{46}$$

The last step follows from the assumed lower bound on Δ_{align}. □

B.3 Proof of Theorem 1

Part (a): sequential model. By the score-unbiasedness assumption, $\mathbb{E}[\hat{s}_{i\cdot}^S] = s_{i\cdot}^*$. Lemma 3 gives

$$\|\hat{s}_{i\cdot}^S - s_{i\cdot}^*\|_\infty \leq C\sigma_S\Lambda_\delta. \tag{47}$$

Applying Lemma 2, averaging over users, and substituting $\sigma_S = \sqrt{\nu/n}$ yields

$$\mathcal{E}_S \leq \frac{2}{\tau N}\sum_i \|\hat{s}_{i\cdot}^S - s_{i\cdot}^*\|_\infty \leq \frac{2C\sigma_S\Lambda_\delta}{\tau}, \tag{48}$$

which proves equation 6.

Part (b): direct fusion. For simplicity, we consider only fusion of the user embeddings; fusion of the item embeddings follows analogously. We have

$$\hat{u}_i^S = u_i^* + \zeta_i^S, \qquad Q^*\hat{u}_i^G = \mu_i^G + \zeta_i^G = u_i^* - r_i + \zeta_i^G. \tag{49}$$

Therefore,

$$\hat{u}_i^{\text{fuse}} = (1-w)\hat{u}_i^S + wQ^*\hat{u}_i^G = u_i^* - wr_i + (1-w)\zeta_i^S + w\zeta_i^G, \tag{50}$$

and

$$\hat{s}_{ij}^{\text{fuse}} - \hat{s}_{ij}^S = \left\langle \hat{v}_j^S, w(Q^*\hat{u}_i^G - \hat{u}_i^S)\right\rangle = -w\langle \hat{v}_j^S, \hat{r}_i\rangle, \tag{51}$$

where $\hat{r}_i = \hat{u}_i^S - Q^*\hat{u}_i^G = r_i + \zeta_i^S - \zeta_i^G$.

The prediction error admits the exact decomposition

$$\hat{s}_{ij}^{\text{fuse}} - s_{ij}^* = \left\langle \hat{v}_j^S, \hat{u}_i^{\text{fuse}} - u_i^*\right\rangle + \left\langle \hat{v}_j^S - v_j^*, u_i^*\right\rangle.$$

Assuming that the item-embedding estimators are consistent in the sequential reference space, the second term converges to zero in probability for each fixed user–item pair; for brevity, we suppress this asymptotically vanishing remainder in the subsequent user-side analysis.

For the upper bound, equation 50, Cauchy–Schwarz, the item-norm constraint, and Lemma 3 imply

$$\begin{aligned}\|\hat{s}_{i\cdot}^{\text{fuse}} - s_{i\cdot}^*\|_\infty &\le \bar{c}_V \left[w\|r_i\|_2 + \|(1-w)\zeta_i^S + w\zeta_i^G\|_2\right] \\ &\le \bar{c}_V \left[w\|r_i\|_2 + C\sigma_w\Lambda_\delta\right]. \end{aligned} \tag{52}$$

Applying Lemma 2 and averaging over users gives

$$\mathcal{E}_{\text{fuse}} \le \frac{C}{\tau}\left[w\Delta_{\text{align}} + \sigma_w\Lambda_\delta\right], \tag{53}$$

for C sufficiently large, proving the first inequality in equation 7.

For the lower bound, let

$$D = \frac{1}{N}\sum_i \|\hat{p}_{i\cdot}^{\text{fuse}} - \hat{p}_{i\cdot}^S\|_1. \tag{54}$$

The triangle inequality gives

$$\mathcal{E}_{\text{fuse}} \ge D - \mathcal{E}_S. \tag{55}$$

By Lemma 8, $R_i \le R_{\max}$, $\sigma_{\Delta,i}^2 \ge w^2\eta\|\hat{r}_i\|_2^2/(2d)$, and $N^{-1}\sum_i \|\hat{r}_i\|_2 \ge \Delta_{\text{align}}/2$. Applying Lemma 7 to equation 51 gives

$$\begin{aligned}\|\hat{p}_{i\cdot}^{\text{fuse}} - \hat{p}_{i\cdot}^S\|_1 &\ge \frac{1}{2}e^{-2R_{\max}/\tau}\frac{w^2(\eta/(2d))\|\hat{r}_i\|_2^2}{\tau\cdot 2w\bar{c}_V\|\hat{r}_i\|_2} \\ &= \frac{\eta w}{8\tau d\bar{c}_V}e^{-2R_{\max}/\tau}\|\hat{r}_i\|_2. \end{aligned} \tag{56}$$

Averaging over users, substituting into equation 55, and using part (a) yields

$$\mathcal{E}_{\text{fuse}} \ge \frac{c\eta w}{\tau d}e^{-2R_{\max}/\tau}\Delta_{\text{align}} - \frac{C\sigma_S\Lambda_\delta}{\tau}, \tag{57}$$

with $c = 1/(32\bar{c}_V)$ after absorbing constants. This proves the second inequality in equation 7.

Part (c): graph-to-sequence distillation. The teacher and student distributions have the same item-indexed coordinates by Lemma 1. Hence, the triangle inequality gives

$$\mathcal{E}_{\text{KD}} \le \underbrace{\frac{1}{N}\sum_i \|\hat{p}_{\theta_S,i\cdot}^S - \hat{p}_{i\cdot}^G\|_1}_{\varepsilon_{\text{opt}}} + \underbrace{\frac{1}{N}\sum_i \|\hat{p}_{i\cdot}^G - p_{i\cdot}^*\|_1}_{\mathcal{E}_{\text{teacher}}}. \tag{58}$$

For the teacher term, Lemma 2 gives

$$\mathcal{E}_{\text{teacher}} \le \frac{2}{\tau N}\sum_i \|\hat{s}_{i\cdot}^G - s_{i\cdot}^*\|_\infty. \tag{59}$$

The score error satisfies

$$\begin{aligned}\|\hat{s}_{i\cdot}^G - s_{i\cdot}^*\|_\infty &\le \|\hat{s}_{i\cdot}^G - \mathbb{E}[\hat{s}_{i\cdot}^G]\|_\infty + \|\mathbb{E}[\hat{s}_{i\cdot}^G] - s_{i\cdot}^*\|_\infty \\ &\le C\sigma_G\Lambda_\delta + \|\mathbb{E}[\hat{s}_{i\cdot}^G] - s_{i\cdot}^*\|_\infty. \end{aligned} \tag{60}$$

As a theoretical abstraction of the graph teacher, we assume $\mathbb{E}[\hat{s}_{ij}^G] = \langle v_j^*, \sum_{i'} w_{ii'}u_{i'}^*\rangle$, which, together with $\|v_j^*\|_2 = 1$, yields

$$\frac{1}{N}\sum_i \|\mathbb{E}[\hat{s}_{i\cdot}^G] - s_{i\cdot}^*\|_\infty \le \frac{1}{N}\sum_i \left\|\sum_{i'} w_{ii'}u_{i'}^* - u_i^*\right\|_2 = b. \tag{61}$$

Combining these inequalities yields

$$\mathcal{E}_{\text{KD}} \le \frac{2}{\tau}\left(C\sigma_G\Lambda_\delta + b\right) + \varepsilon_{\text{opt}}, \tag{62}$$

which proves equation 8. A union bound over the required high-probability events completes the theorem. □

B.4 Proofs of Corollaries 1–3

Corollary 1. The KD upper bound is smaller than the sequential upper bound when

$$\frac{2}{\tau}\left(b + C\sigma_G\Lambda_\delta\right) + \varepsilon_{\text{opt}} < \frac{2C\sigma_S\Lambda_\delta}{\tau}. \tag{63}$$

Rearranging gives

$$\begin{aligned}\frac{\tau}{2}\varepsilon_{\text{opt}} + b &< C\Lambda_\delta(\sigma_S - \sigma_G) \\ &= C\Lambda_\delta\sqrt{\frac{\nu}{n}}\left(1 - \kappa^{-1/2}\right),\end{aligned} \tag{64}$$

which is equation 9. Solving this inequality for n gives $n < n^*$ with the threshold stated in the corollary.

For the comparison with direct fusion, a sufficient condition for $\mathcal{E}_{\text{KD}} < \mathcal{E}_{\text{fuse}}$ is

$$\frac{2}{\tau}\left(b + C\sigma_G\Lambda_\delta\right) + \varepsilon_{\text{opt}} < \frac{c\eta w}{\tau d}e^{-2R_{\max}/\tau}\Delta_{\text{align}} - \frac{C\sigma_S\Lambda_\delta}{\tau}. \tag{65}$$

Multiplying by τ, rearranging, and absorbing absolute constants gives equation 11. □

Corollary 2. Since $n + m = n\kappa$, Theorem 1 gives

- $\sigma_S = \sqrt{\nu/n}$ for the sequential model;
- $\sigma_w = \sqrt{\nu((1-w)^2/n + w^2/(n\kappa))}$ for fusion, together with the non-vanishing term $w\Delta_{\text{align}}$;
- $\sigma_G = \sqrt{\nu/(n\kappa)}$ for KD, together with the propagation term $b(\kappa)$.

Because $(n\kappa)^{-1/2} = \kappa^{-1/2}n^{-1/2}$, bounded κ leaves all stochastic terms at order $\Theta(n^{-1/2})$; KD changes only the constant by $\kappa^{-1/2}$. A faster order requires κ to grow with the data size. □

Corollary 3. If $b(\kappa)$ is continuous and non-decreasing, then

$$F(\kappa) = b(\kappa) + C\Lambda_\delta\sqrt{\frac{\nu}{n\kappa}} \tag{66}$$

is continuous. It therefore attains a minimum on every admissible compact interval. The minimizer may lie at the boundary and need not be unique; an interior or unique optimum requires additional conditions. □

C Evaluation Metrics Definitions

The formal definitions of H@K and N@K are given as

$$\text{H@K} = \frac{1}{N}\sum_{u\in\mathcal{U}} I(|R^K(u)\cap T(u)| > 0),\ \text{N@K} = \frac{1}{N}\sum_{u\in\mathcal{U}}\frac{\sum_{k=1}^{K}\frac{I(R_k^K(u)\in T(u))}{\log_2(k+1)}}{\sum_{k=1}^{\min(K,|T(u)|)}\frac{1}{\log_2(k+1)}}, \tag{67}$$

where $T(u)$ denotes the ground-truth item set, $R^K(u)$ denotes the top-K recommended item set, $R_k^K(u)$ denotes the k-th item in the recommended list $R^K(u)$, and $I(\cdot)$ is the indicator function.

In the general formulation above, the denominator corresponds to the ideal DCG, which represents the maximum possible discounted gain for user u. In our experiments, we adopt the leave-one-out evaluation protocol, where each user has exactly one ground-truth test item, i.e., $|T(u)| = 1$. Therefore, the ideal DCG reduces to

$$\text{IDCG@K}(u) = \frac{1}{\log_2(1+1)} = 1. \tag{68}$$

Accordingly, N@K in our setting is simplified as

$$\text{N@K} = \frac{1}{N}\sum_{u\in\mathcal{U}}\sum_{k=1}^{K}\frac{I(R_k^K(u) = v_u^{\text{test}})}{\log_2(k+1)}, \tag{69}$$

where v_u^{test} denotes the single ground-truth test item of user u.

D DATASETS INFORMATION

D.1 DATASET STATISTICS

Table 4 shows the summary of datasets we use after preprocessing.

Table 4: Dataset statistics (after preprocessing).

Dataset	#user	#item	#interaction	#Train	#Valid	#Test	Density(%)
Yelp	7,242	5,026	91,182	69,456	7,242	7,242	0.25
Amazon	8,962	3,748	94,305	67,419	8,962	8,962	0.28
Gowalla	13,307	8,664	163,882	123,961	13,307	13,307	0.14
Steam	4,211	1,220	78,012	65,379	4,211	4,211	1.52
campus 13	4,721	2,504	135,575	121,412	4,721	4,721	1.15
campus 38	2,737	1,055	56,023	47,812	2,737	2,737	1.94
campus 107	3,547	2,241	118,775	108,134	3,547	3,547	1.49
campus 151	1,041	350	17,006	13,883	1,041	1,041	4.67
campus 18	5,660	2,500	207,569	190,589	5,660	5,660	1.47

In addition, two indicators are calculated to characterize the sequential complexity of each dataset: user-level sequence entropy, denoted as H_{user}, and next-item conditional entropy, denoted as H_{next}. User-level sequence entropy measures the diversity of items repeatedly interacted with by the same user. For user u with behavior sequence $\mathcal{S}_u = (v_{u1}, \ldots, v_{u|\mathcal{S}_u|})$, the empirical interaction probability of item i is defined as $p_u(i) = \frac{1}{|\mathcal{S}_u|} \sum_{t=1}^{|\mathcal{S}_u|} \mathbb{I}(v_{ut} = i)$, where $i \in \mathcal{V}_u^+$ and $\mathcal{V}_u^+$ denotes the set of items appearing in $\mathcal{S}_u$. The entropy of user u is computed as $H_u = -\sum_{i \in \mathcal{V}_u^+} p_u(i) \log p_u(i)$. Since users may interact with different numbers of unique items, the entropy is normalized by its maximum possible value, yielding $\widetilde{H}_u = H_u / \log |\mathcal{V}_u^+|$. The final dataset-level indicator is obtained through a sequence-length-weighted average over all users, i.e., $H_{\text{user}} = \sum_{u \in \mathcal{U}} \frac{|\mathcal{S}_u|}{\sum_{u' \in \mathcal{U}} |\mathcal{S}_{u'}|} \widetilde{H}_u$. A larger H_{user} indicates that users' historical behaviors are more evenly distributed over diverse items, suggesting weaker repetition patterns and higher intra-user preference uncertainty.

Next-item conditional entropy measures the uncertainty of item-to-item transition patterns in user sequences. For each item i, its empirical transition probability to item j is estimated as $P(j \mid i) = \frac{\sum_{u \in \mathcal{U}} \sum_{t=1}^{|\mathcal{S}_u|-1} \mathbb{I}(v_{ut}=i, v_{u,t+1}=j)}{\sum_{u \in \mathcal{U}} \sum_{t=1}^{|\mathcal{S}_u|-1} \mathbb{I}(v_{ut}=i)}$, where $j \in \mathcal{V}_i^{\text{next}}$ and $\mathcal{V}_i^{\text{next}}$ denotes the set of items that have appeared immediately after item i in the observed sequences. The conditional entropy of item i is defined as $H_{\text{next}}(i) = -\sum_{j \in \mathcal{V}_i^{\text{next}}} P(j \mid i) \log P(j \mid i)$, and its normalized form is given by $\widetilde{H}_{\text{next}}(i) = H_{\text{next}}(i) / \log |\mathcal{V}_i^{\text{next}}|$. The dataset-level indicator is then computed as $H_{\text{next}} = \sum_{i \in \mathcal{V}} w_i \widetilde{H}_{\text{next}}(i)$, where $w_i = \frac{\sum_{u \in \mathcal{U}} \sum_{t=1}^{|\mathcal{S}_u|-1} \mathbb{I}(v_{ut}=i)}{\sum_{i' \in \mathcal{V}} \sum_{u \in \mathcal{U}} \sum_{t=1}^{|\mathcal{S}_u|-1} \mathbb{I}(v_{ut}=i')}$. A larger H_{next} indicates that the next item following a given item is less deterministic, implying more complex and less predictable sequential transition patterns.

Table 5 shows H_{user} and H_{next} of each dataset.

D.2 TIME WINDOWS OF PUBLIC DATASETS

Table 6 shows the retained time window for each dataset.

Table 5: Supplementary dataset statistics.

Dataset	H_{user}	H_{next}
Yelp	0.9939	0.9945
Amazon	1.0000	0.9857
Gowalla	1.0000	0.9646
Steam	0.9900	0.9611
campus 13	0.9524	0.8312
campus 38	0.8811	0.8148
campus 107	0.9377	0.8706
campus 151	0.9045	0.8592
campus 18	0.9547	0.8375

Table 6: Time window of each dataset.

Dataset	Start Date	End Date
Yelp	August 5, 2020	January 19, 2022
Amazon	March 1, 2016	September 12, 2023
Gowalla	August 24, 2010	October 22, 2010
Steam	November 10, 2016	January 5, 2018
campus 13	December 1, 2024	December 31, 2024
campus 38	December 1, 2024	December 31, 2024
campus 107	December 1, 2024	December 31, 2024
campus 151	December 1, 2024	December 31, 2024
campus 18	December 1, 2024	December 31, 2024

E Additional Implementation Details

Additional implementation details are provided as follows. For FROGS, unless otherwise specified, the number of layers in graph-based teacher module is 2, the variance σ^2 of noise $\boldsymbol{\epsilon}^u$ is set to 1, the temperature parameter τ_{CL} is set to 0.2, and the temperature parameter T is chosen as 3. For the overall training objective in Eq. equation 5, the coefficients λ_1 and λ_2 are both set to 1, λ_3 is set to 10^{-4}, and λ_4 is set to 0.1. The complementary knowledge weight β is set to 1 by default. Figure 2 shows the sensitivity of performance to β. In the contrastive learning loss of the graph module $\mathcal{L}_{CL}^{(G)}$, the balancing coefficient γ is set to 1, indicating that user-side and item-side contrastive learning are treated equally.

Computational Complexity. The second-order adjacency matrices require $O(N^2 + M^2)$ space. The KD loss operates over all M items per user. For our datasets ($N, M \leq 2 \times 10^4$), this is tractable on a single GPU. For larger-scale deployment, sampling-based KD loss approximation can be applied. For all contrastive objectives, a single negative sample is used per positive pair. Regarding the sequential backbone SASRec, the maximum sequence length is limited to 50. The number of self-attention blocks is 2, the number of attention heads is 1, and the dropout rate is 0.2.

F Robustness to the KD loss weight λ_4 and the temperature T

We investigate the robustness of FROGS to parameters T in Eq. equation 4 and λ_4 in Eq. equation 5. Specifically, we first vary the distillation weight λ_4 on three representative datasets with different sparsity levels, and repeat each setting 10 times to report the mean performance together with error bars indicating the standard deviation. As shown in Figure 3, the performance remains stable across a reasonable range of λ_4, with relatively small fluctuations and largely overlapping error bars across different settings. This indicates that FROGS is not overly sensitive to the choice of the distillation weight, and that the knowledge distillation component does not require delicate tuning in practice. Combined with the ablation results, these findings further suggest that the benefit of knowledge distillation is consistent rather than relying on a narrowly tuned loss weight.

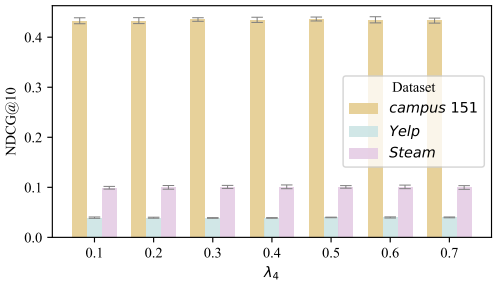


Figure 3: Performance of FROGS under different distillation weights λ_4.

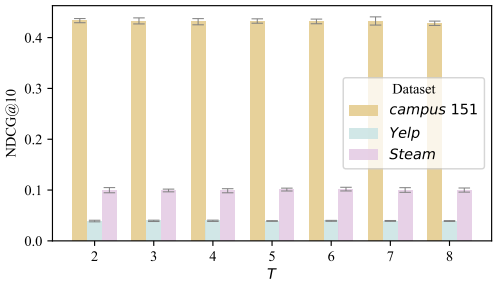


Figure 4: Robustness to the temperature T. Mean N@10 over 10 runs with standard deviation shown as error bars.

We further examine the robustness of FROGS to the temperature T in Eq. equation 4, which controls the smoothness of the distillation distributions. We conduct the experiments on three representative datasets with different sparsity levels, and repeat each setting 10 times to report the mean performance and standard deviation. As shown in Figure 4, the performance remains relatively stable across different values of T, with only limited fluctuations and largely overlapping error bars in most cases. This result suggests that FROGS is not overly sensitive to the choice of the temperature T, and can achieve consistent performance once T is selected within a reasonable range. Overall, these findings further support the robustness of FROGS under different hyper-parameter settings.

G Empirical Analysis of Embedding Mismatch

Embedding extraction and user split. On each dataset, we separately train the graph and sequential encoders with twenty random seeds, using independent training randomness across runs, identical training interactions, fixed data splits and hyper-parameters, and no cross-model alignment or distillation loss. We extract embeddings in evaluation mode from the same training-history cutoff. Write the resulting user matrices as $G^{(r)}, S^{(r)} \in \mathbb{R}^{N\times d}$ for $r \in \{1,\dots,20\}$, with identical user ordering and $d = 64$. Independently of the recommendation data split, we randomly assign 70% of users to alignment fitting and 30% to evaluation, using seed 2026 and the same partition throughout. Embeddings retain their original scales without centering or row normalization.

Disjoint run pairs. We group the twenty runs into ten fixed, non-overlapping pairs $(2k-1, 2k)$, $k \in \{1,\dots,10\}$, namely $(1,2),(3,4),\dots,(19,20)$. For each pair, we use its first run for the graph–sequence comparison and both runs for the within-model comparisons. Each run belongs to exactly one pair; all statistics indexed by k below are computed from that pair alone.

Orthogonal alignment and residuals. For source and target matrices $A, B \in \mathbb{R}^{N\times d}$, let $\mathcal{U}_{\text{fit}}$ and $\mathcal{U}_{\text{eval}}$ denote the fitting and evaluation users. We solve the orthogonal Procrustes problem

$$\hat{Q}(A,B) \in \arg\min_{Q^\top Q = I_d} \|A_{\mathcal{U}_{\text{fit}}} Q - B_{\mathcal{U}_{\text{fit}}}\|_F^2. \tag{70}$$

If $A_{\mathcal{U}_{\text{fit}}}^\top B_{\mathcal{U}_{\text{fit}}} = L\Sigma R^\top$, its solution is $\hat{Q} = LR^\top$. Embeddings are rows, so the map acts on the right. The normalized held-out residual is

$$D(A,B) = \frac{\sum_{i\in\mathcal{U}_{\text{eval}}} \|B_i - A_i\hat{Q}(A,B)\|_2}{\sum_{i\in\mathcal{U}_{\text{eval}}} \|B_i\|_2}. \tag{71}$$

The denominator expresses the error relative to the target representation scale. We obtain

$$D_{SG}^{(k)} = D(G^{(2k-1)}, S^{(2k-1)}), \qquad D_{\text{raw}}^{(k)} = \frac{\sum_{i\in\mathcal{U}_{\text{eval}}} \|S_i^{(2k-1)} - G_i^{(2k-1)}\|_2}{\sum_{i\in\mathcal{U}_{\text{eval}}} \|S_i^{(2k-1)}\|_2}. \tag{72}$$

Here, D_{raw} and D_{SG} measure discrepancies before and after alignment. Fitting minimizes squared error, whereas reporting averages Euclidean distances; this held-out diagnostic differs from the population-optimal, unsquared objective defining Δ_{align}.

Within-model variation. For each disjoint pair k, we compare the two sequential runs and the two graph runs:

$$D_{SS}^{(k)} = D(S^{(2k-1)}, S^{(2k)}), \qquad D_{GG}^{(k)} = D(G^{(2k-1)}, G^{(2k)}). \tag{73}$$

Each comparison fits its own map. Table 7 reports means and sample standard deviations over the ten pair-level observations. With different target scales, D_{SS} and D_{GG} are references for relative training variability, not calibrated bounds on graph–sequence estimation noise.

Significance assessment. For each disjoint pair, we take the larger within-model residual as the reference,

$$D_{\mathrm{noise}}^{(k)} = \max\{D_{SS}^{(k)}, D_{GG}^{(k)}\}. \tag{74}$$

We perform two one-sided paired Wilcoxon signed-rank tests on the ten pair-level differences:

$$g_{\mathrm{raw}}^{(k)} = D_{\mathrm{raw}}^{(k)} - D_{\mathrm{noise}}^{(k)}, \qquad g_{SG}^{(k)} = D_{SG}^{(k)} - D_{\mathrm{noise}}^{(k)}. \tag{75}$$

Conditional on the fixed dataset and user partition, independent training randomness and disjoint run pairs yield independent observations across k. Statistics within a pair may share embeddings; independence is required across pairs, and the sample size of each test is ten. For each test, the reference null is a zero-centered symmetric difference distribution, and the alternative is a positive shift. The first test examines whether the original graph–sequence discrepancy exceeds within-model variability; the second examines whether such an excess remains after orthogonal alignment. Table 7 reports the corresponding mean paired gaps

$$\bar{g}_{\mathrm{raw}} = \frac{1}{10}\sum_{k=1}^{10} g_{\mathrm{raw}}^{(k)}, \qquad \bar{g}_{SG} = \frac{1}{10}\sum_{k=1}^{10} g_{SG}^{(k)}. \tag{76}$$

Thus, $\bar{g}_{\mathrm{raw}}$ is the average amount by which the original mismatch exceeds the within-model reference, while $\bar{g}_{SG}$ is the corresponding excess remaining after orthogonal alignment. Significance stars for the two tests are attached to these two mean-gap columns: $^{*}p < 0.05$, $^{**}p < 0.01$, and $^{***}p < 0.001$.

Table 7: Graph–sequence embedding mismatch before and after orthogonal alignment over ten disjoint run pairs. Residuals are shown as mean ± sample standard deviation, and $\bar{g}_{\mathrm{raw}}$ and $\bar{g}_{SG}$ are mean paired gaps.

Type	Dataset	D_{raw}	D_{SG}	D_{SS}	D_{GG}	$\bar{g}_{\mathrm{raw}}$	$\bar{g}_{SG}$
Publicly available datasets	*Yelp*	1.011 ± 0.005	0.910 ± 0.009	0.735 ± 0.039	0.388 ± 0.097	0.276^{***}	0.175^{***}
	Amazon	1.004 ± 0.003	0.949 ± 0.005	0.761 ± 0.027	0.234 ± 0.029	0.243^{***}	0.188^{***}
	Gowalla	1.008 ± 0.003	0.935 ± 0.003	0.733 ± 0.022	0.328 ± 0.014	0.274^{***}	0.201^{***}
	Steam	1.009 ± 0.008	0.924 ± 0.006	0.535 ± 0.056	0.200 ± 0.066	0.474^{***}	0.389^{***}
Real-world transaction data	*campus 13*	1.014 ± 0.005	0.921 ± 0.003	0.743 ± 0.018	0.406 ± 0.010	0.271^{***}	0.177^{***}
	campus 38	1.012 ± 0.004	0.913 ± 0.002	0.705 ± 0.010	0.390 ± 0.026	0.307^{***}	0.208^{***}
	campus 107	1.017 ± 0.005	0.911 ± 0.004	0.728 ± 0.016	0.441 ± 0.015	0.289^{***}	0.183^{***}
	campus 151	1.012 ± 0.003	0.908 ± 0.004	0.732 ± 0.021	0.310 ± 0.016	0.280^{***}	0.176^{***}
	campus 18	1.016 ± 0.005	0.915 ± 0.003	0.715 ± 0.013	0.426 ± 0.022	0.301^{***}	0.200^{***}

Note. D_{SS} and D_{GG} measure within-model variation. Stars on $\bar{g}_{\mathrm{raw}}$ and $\bar{g}_{SG}$ refer to the separate one-sided tests of $D_{\mathrm{raw}} > D_{\mathrm{noise}}$ and $D_{SG} > D_{\mathrm{noise}}$, respectively, where $D_{\mathrm{noise}} = \max\{D_{SS}, D_{GG}\}$: $^{*}p < 0.05$, $^{**}p < 0.01$, $^{***}p < 0.001$.

Results and interpretation. Table 7 shows positive and significant values of $\bar{g}_{\mathrm{raw}}$, providing empirical evidence that the original mismatch between the graph and sequential embedding spaces exceeds their relative run-to-run variability. Furthermore, the positive and significant values of $\bar{g}_{SG}$ show that the aligned graph–sequence discrepancy remains above the same within-model reference. Thus, the observed mismatch persists after orthogonal alignment and cannot be fully explained by a rotation or reflection of the learned coordinates. These findings support the residual-mismatch motivation of our theoretical analysis and the use of knowledge distillation.

H OVERALL COMPARISON RESULT

Table 8: Performance comparison of different recommendation models. Bold indicates the best result, and underline indicates the second-best result.

Type	Dataset	Metrics	FROGS	Traditional		Sequential				Graph-Based			KD			CL	LLM
				S-POP	FPMC	GRU4Rec	Caser	SASRec	BERT4Rec	LightGCN	XSimGCL	MF	SimRec	UnKD	SLMRec	CL4SRec	DS
Publicly available datasets	*Yelp*	H@5	**0.0463**	0.0369	0.0323	0.0243	0.0348	0.0389	0.0377	0.0366	0.0394	0.0400	0.0434	0.0402	0.0174	0.0360	0.0322
		N@5	**0.0301**	0.0239	0.0199	0.0148	0.0226	0.0256	0.0247	0.0229	0.0241	0.0250	0.0268	0.0257	0.0123	0.0233	0.0210
		H@10	**0.0777**	0.0498	0.0558	0.0417	0.0579	0.0648	0.0641	0.0679	0.0722	0.0681	0.0704	0.0701	0.0257	0.0605	0.0478
		N@10	**0.0403**	0.0282	0.0274	0.0204	0.0300	0.0339	0.0332	0.0329	0.0346	0.0340	0.0386	0.0354	0.0149	0.0311	0.0261
		H@20	**0.1299**	0.0585	0.0950	0.0791	0.1025	0.1107	0.1069	0.1188	0.1236	0.1132	0.1227	0.1161	0.0458	0.1049	0.0599
		N@20	**0.0533**	0.0304	0.0373	0.0299	0.0412	0.0453	0.0439	0.0457	0.0475	0.0453	0.0517	0.0469	0.0200	0.0423	0.0291
	Amazon	H@5	**0.0579**	0.0000	0.0351	0.0319	0.0320	0.0363	0.0374	0.0295	0.0301	0.0206	0.0429	0.0362	0.0241	0.0377	0.0015
		N@5	**0.0322**	0.0000	0.0223	0.0208	0.0196	0.0201	0.0235	0.0192	0.0196	0.0123	0.0290	0.0228	0.0162	0.0221	0.0010
		H@10	**0.0922**	0.0031	0.0615	0.0528	0.0536	0.0687	0.0643	0.0495	0.0519	0.0407	0.0672	0.0576	0.0388	0.0649	0.0040
		N@10	**0.0432**	0.0009	0.0308	0.0274	0.0265	0.0304	0.0322	0.0257	0.0265	0.0188	0.0368	0.0297	0.0210	0.0308	0.0018
		H@20	**0.1423**	0.0135	0.0956	0.0863	0.0881	0.1089	0.0982	0.0799	0.0806	0.0648	0.1065	0.0896	0.0553	0.1057	0.0105
		N@20	**0.0559**	0.0036	0.0393	0.0359	0.0352	0.0405	0.0407	0.0332	0.0337	0.0249	0.0466	0.0377	0.0251	0.0410	0.0034
	Gowalla	H@5	**0.1387**	0.0000	0.1111	0.0753	0.0881	0.1082	0.1017	0.0993	0.1021	0.0852	0.1355	0.1005	0.0857	0.0993	0.0154
		N@5	**0.0857**	0.0000	0.0746	0.0504	0.0602	0.0699	0.0668	0.0678	0.0693	0.0583	0.0761	0.0695	0.0628	0.0649	0.0100
		H@10	**0.2094**	0.0002	0.1677	0.1119	0.1362	0.1642	0.1501	0.1508	0.1513	0.1303	0.1846	0.1515	0.1141	0.1532	0.0241
		N@10	**0.1084**	0.0001	0.0928	0.0621	0.0756	0.0878	0.0823	0.0843	0.0851	0.0728	0.0951	0.0859	0.0719	0.0822	0.0127
		H@20	**0.3025**	0.0011	0.2430	0.1667	0.2012	0.2490	0.2215	0.2236	0.2240	0.1888	0.2800	0.2234	0.1529	0.2253	0.0412
		N@20	**0.1319**	0.0003	0.1118	0.0759	0.0919	0.1091	0.1002	0.1027	0.1034	0.0875	0.1241	0.1039	0.0817	0.1004	0.0171
	Steam	H@5	**0.1206**	0.0370	0.0862	0.1062	0.0969	0.1069	0.1066	0.0810	0.0822	0.0805	0.0684	0.0328	0.1140	0.0829	0.0325
		N@5	**0.0877**	0.0225	0.0535	0.0754	0.0600	0.0734	0.0722	0.0509	0.0514	0.0503	0.0416	0.0185	0.0787	0.0545	0.0204
		H@10	**0.1788**	0.0663	0.1477	0.1672	0.1669	0.1636	0.1641	0.1446	0.1446	0.1418	0.1304	0.0786	0.1634	0.1546	0.0579
		N@10	**0.1062**	0.0321	0.0731	0.0949	0.0823	0.0914	0.0907	0.0710	0.0709	0.0700	0.0616	0.0327	0.0948	0.0776	0.0285
		H@20	**0.2553**	0.1000	0.2227	0.2515	0.2427	0.2477	0.2432	0.1971	0.1964	0.2282	0.2061	0.1510	0.2246	0.2465	0.0848
		N@20	**0.1255**	0.0407	0.0920	0.1160	0.1016	0.1127	0.1107	0.0843	0.0840	0.0915	0.0807	0.0514	0.1102	0.1009	0.0352
Real-world transaction data	*campus 13*	H@5	**0.4764**	0.2201	0.4162	0.3393	0.4385	0.3383	0.4620	0.2258	0.2150	0.2266	0.1260	0.0341	0.1822	0.3567	0.2269
		N@5	**0.3843**	0.1457	0.3136	0.2486	0.3549	0.2261	0.3721	0.1527	0.1516	0.1583	0.0894	0.0208	0.1454	0.2301	0.1535
		H@10	**0.5617**	0.3080	0.5145	0.4342	0.5128	0.4488	0.5467	0.3029	0.2879	0.3114	0.1741	0.0525	0.2264	0.4711	0.2997
		N@10	**0.4117**	0.1743	0.3455	0.2793	0.3789	0.2619	0.3996	0.1821	0.1750	0.1855	0.1047	0.0268	0.1597	0.2681	0.1770
		H@20	**0.6359**	0.3811	0.6031	0.5287	0.5876	0.5497	0.6261	0.3798	0.3645	0.3855	0.2457	0.0845	0.2804	0.5747	0.3711
		N@20	**0.4300**	0.1929	0.3680	0.3032	0.3980	0.2873	0.4198	0.2014	0.1943	0.2044	0.1226	0.0348	0.1732	0.2943	0.1951
	campus 38	H@5	**0.4293**	0.3420	0.3906	0.2886	0.3928	0.3553	0.4047	0.3292	0.3113	0.3339	0.1472	0.0482	0.2364	0.3434	0.3372
		N@5	**0.3301**	0.2468	0.2945	0.2176	0.3037	0.2507	0.3148	0.2389	0.2280	0.2435	0.1051	0.0329	0.1986	0.2437	0.2471
		H@10	**0.5086**	0.4238	0.4717	0.3796	0.4618	0.4498	0.4845	0.4125	0.3884	0.4099	0.2141	0.0782	0.2777	0.4476	0.4172
		N@10	**0.3558**	0.2735	0.3208	0.2470	0.3259	0.2817	0.3398	0.2660	0.2529	0.2681	0.1264	0.0424	0.2120	0.2776	0.2731
		H@20	**0.5864**	0.4717	0.5524	0.4695	0.5170	0.5462	0.5681	0.4841	0.4629	0.4750	0.3296	0.1323	0.3292	0.5440	0.4677
		N@20	**0.3756**	0.2856	0.3414	0.2698	0.3399	0.3061	0.3611	0.2841	0.2718	0.2846	0.1554	0.0559	0.2249	0.3020	0.2861
	campus 107	H@5	**0.3637**	0.2566	0.3172	0.2450	0.3533	0.2898	0.3612	0.2574	0.2326	0.2563	0.1029	0.0290	0.1706	0.2760	0.2518
		N@5	**0.2849**	0.1768	0.2344	0.1738	0.2764	0.1876	0.2818	0.1781	0.1642	0.1802	0.0691	0.0185	0.1387	0.1795	0.1717
		H@10	**0.4541**	0.3504	0.4029	0.3093	0.4108	0.3795	0.4370	0.3403	0.2997	0.3440	0.1528	0.0465	0.2103	0.3592	0.3341
		N@10	**0.3131**	0.2072	0.2618	0.1946	0.2950	0.2169	0.3064	0.2048	0.1860	0.2026	0.0852	0.0241	0.1515	0.2065	0.1981
		H@20	**0.5354**	0.4206	0.4750	0.3902	0.4728	0.4615	0.5114	0.4068	0.3767	0.4170	0.2148	0.0699	0.2599	0.4483	0.4085
		N@20	**0.3344**	0.2249	0.2801	0.2151	0.3108	0.2375	0.3253	0.2216	0.2055	0.2271	0.1008	0.0299	0.1641	0.2290	0.2170
	campus 151	H@5	**0.5130**	0.4793	0.4986	0.3862	0.4986	0.4448	0.4986	0.4726	0.4544	0.4678	0.2161	0.0490	0.2373	0.4419	0.4697
		N@5	**0.4047**	0.3313	0.3821	0.2947	0.3961	0.3078	0.3957	0.3379	0.3243	0.3381	0.1562	0.0309	0.2065	0.3101	0.3267
		H@10	**0.6061**	0.5716	0.5841	0.4861	0.5610	0.5524	0.5975	0.5476	0.5312	0.5466	0.2968	0.0768	0.2843	0.5514	0.5629
		N@10	**0.4348**	0.3619	0.4100	0.3268	0.4165	0.3429	0.4280	0.3624	0.3494	0.3644	0.1825	0.0398	0.2217	0.3457	0.3570
		H@20	**0.6792**	0.6254	0.6724	0.5927	0.6292	0.6427	0.6763	0.6158	0.6071	0.6042	0.4063	0.1230	0.3410	0.6551	0.6167
		N@20	**0.4533**	0.3754	0.4325	0.3536	0.4337	0.3656	0.4480	0.3798	0.3688	0.3789	0.2100	0.0515	0.2360	0.3721	0.3707
	campus 18	H@5	**0.3726**	0.1811	0.2928	0.2297	0.3373	0.2663	0.3666	0.1742	0.1610	0.1769	0.0670	0.0311	0.1800	0.2671	0.1875
		N@5	**0.2844**	0.1301	0.2090	0.1551	0.2608	0.1714	0.2799	0.1270	0.1174	0.1265	0.0427	0.0181	0.1434	0.1717	0.1361
		H@10	**0.4472**	0.2557	0.3769	0.3140	0.4023	0.3680	0.4403	0.2406	0.2201	0.2443	0.1092	0.0569	0.2274	0.3588	0.2518
		N@10	**0.3086**	0.1540	0.2362	0.1823	0.2817	0.2042	0.3039	0.1485	0.1364	0.1481	0.0562	0.0265	0.1587	0.2014	0.1566
		H@20	**0.5228**	0.3459	0.4611	0.3920	0.4625	0.4525	0.5087	0.3212	0.2943	0.3226	0.1027	0.0873	0.2850	0.4498	0.3438
		N@20	**0.3277**	0.1770	0.2575	0.2021	0.2969	0.2256	0.3211	0.1689	0.1552	0.1680	0.0697	0.0341	0.1731	0.2245	0.1800

I t-TESTS RESULT

I.1 t-TEST AGAINST FROGS AND BERT4REC

To further verify that the superiority of FROGS is not caused by sampling randomness, we conduct an additional Monte Carlo study against BERT4Rec, which achieves the second-best performance among all baselines on the five campus datasets. Specifically, for each dataset, we repeatedly sample

90% of the data, train both models on the same sampled set, and perform 50 runs. We then conduct paired-sample t-tests on all evaluation metrics. The random initialization, mini-batch sampling, and negative sampling in each run introduce requisite variability for reliable significance assessment. The results show that FROGS significantly outperforms BERT4Rec in most settings, which provides empirical validation evidence for the effectiveness and robustness of our method. The t-statistics and p-values of the t-tests comparing FROGS and BERT4Rec are reported in Table 9. Each entry in the table reports the corresponding t-statistic, rather than the raw performance value of the metric.

Table 9: t-statistics and p-values of the t-tests comparing FROGS and BERT4Rec.

Dataset	H@5	N@5	H@10	N@10	H@20	N@20
campus 13	14.2607***	10.8053***	17.8394***	13.4217***	20.1617***	14.7136***
campus 38	7.1245***	8.2164***	7.4168***	9.4852***	6.9503***	9.7071***
campus 107	5.9478***	3.8003***	13.5891***	8.3261***	11.4621***	8.5106***
campus 151	3.8020***	5.6251***	3.4005***	5.5458***	0.0491	4.3479***
campus 18	2.3505*	0.3859	7.1833***	2.5154*	13.1681***	4.3479***

Note. Significance levels: * $p < 0.05$, ** $p < 0.01$, and *** $p < 0.001$.

I.2 t-TEST COMPARING FROGS AND ONLY S

Due to computational constraints, we restrict this experiment to four datasets: campus 13, campus 18, campus 38, and Amazon, and report results only for these datasets. For each dataset, FROGS and Only S are trained with the same train/validation/test split and fixed random seed. We then evaluate both trained models on 10,000 paired bootstrap resamples of test users and apply one-sided paired t-tests to the resulting H@K and N@K scores. Table 10 reports the t-statistics and the significance levels. FROGS has positive differences on all six metrics for the three campus datasets and on five of the six Amazon metrics, which indicates that the modest improvements FROGS achieves over Only S on the evaluation metrics are statistically significant.

Table 10: Nominal t-statistics for FROGS versus Only S on the four selected datasets.

Dataset	H@5	N@5	H@10	N@10	H@20	N@20
Amazon	177.072***	185.668***	-56.575	29.272***	41.658***	80.914***
campus 13	159.249***	100.890***	126.815***	90.737***	153.080***	103.244***
campus 18	140.864***	150.696***	148.540***	176.274***	118.613***	174.982***
campus 38	171.222***	134.241***	102.028***	112.555***	48.933***	99.340***

Note. A positive t-statistic favors FROGS. Stars denote nominal one-sided p-values: * $p < 0.05$, ** $p < 0.01$, and *** $p < 0.001$.