Guiding Generative Recommender Systems with Structured Human Priors via Multi-head Decoding

Yunkai Zhang
yunkai_zhang@berkeley.edu
University of California, Berkeley
Berkeley, United States

Qiang Zhang[*]
qiangzhang@meta.com
Meta
Menlo Park, United States

Diji Yang[*]
dyang39@ucsc.edu
University of California Santa Cruz
Santa Cruz, United States

Ryan Lin
ryanflin@uw.edu
University of Washington
Seattle, United States

Ruizhong Qiu
rq5@illinois.edu
University of Illinois
Urbana-Champaign
Champaign, United States

Benyu Zhang
byzhang@meta.com
Meta
Menlo Park, United States

Hanchao Yu
yhcece@gmail.com
Meta
Menlo Park, United States

Jason Liu
liujiayi@meta.com
Meta
Menlo Park, United States

Yinglong Xia
yxia@meta.com
Meta
Menlo Park, United States

Zhuokai Zhao
zhuokai@meta.com
Meta
Menlo Park, United States

Lizhu Zhang
lizhu@meta.com
Meta
Menlo Park, United States

Xiangjun Fan
ffvoyaged@gmail.com
Meta
Menlo Park, United States

Zhuoran Yu
yuzhuoran@meta.com
Meta
Menlo Park, United States

Abhishek Kumar
abi@meta.com
Meta
Menlo Park, United States

Zeyu Zheng
University of California, Berkeley
Berkeley, United States
zyzheng@berkeley.edu

Abstract

Optimizing recommender systems for objectives beyond accuracy, such as diversity, novelty, and personalization, is crucial for long-term user satisfaction. To this end, the industry has accumulated vast amounts of structured domain knowledge, which we term *human priors* (e.g., item taxonomies, temporal patterns). This knowledge is typically applied through post-hoc adjustments during ranking or post-ranking. However, this approach remains decoupled from the core model learning, which is particularly undesirable as the industry shifts to end-to-end generative recommendation foundation models. On the other hand, many methods targeting these beyond-accuracy objectives often require architecture-specific modifications and discard these valuable human priors by learning user intent in a fully unsupervised manner. Instead of discarding the human priors accumulated over years of practice, we introduce a backbone-agnostic framework that seamlessly integrates these human priors directly into the end-to-end training of generative recommenders. With lightweight, prior-conditioned adapter heads inspired by efficient LLM decoding strategies, our approach guides the model to disentangle user intent along human-understandable axes (e.g., interaction types, long- vs. short-term interests). We also introduce a hierarchical composition strategy for modeling complex interactions across different prior types. Extensive experiments on three large-scale datasets demonstrate that our method significantly enhances both accuracy and beyond-accuracy objectives. We also show that human priors allow the backbone model to more effectively leverage longer context lengths and larger model sizes.

CCS Concepts

• **Information systems → Language models**; **Recommender systems**; **Personalization**.

Keywords

Recommendation, Human Priors, Foundation Models

ACM Reference Format:
Yunkai Zhang, Qiang Zhang, Diji Yang, Ryan Lin, Ruizhong Qiu, Benyu Zhang, Hanchao Yu, Jason Liu, Yinglong Xia, Zhuokai Zhao, Lizhu Zhang, Xiangjun Fan, Zhuoran Yu, Abhishek Kumar, and Zeyu Zheng. 2026. Guiding Generative Recommender Systems with Structured Human Priors via Multi-head Decoding. In *Proceedings of the ACM Web Conference 2026 (WWW '26), April 13–17, 2026, Dubai, United Arab Emirates.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3774904.3792100

[*] Corresponding authors.

WWW '26, Dubai, United Arab Emirates

ACM ISBN 979-8-4007-2307-0/2026/04
https://doi.org/10.1145/3774904.3792100

1 Introduction

The goal of recommender systems extends beyond mere predictive accuracy. The importance of objectives such as novelty and diversity has long been recognized within the academic community [1], acknowledging that a successful system must balance relevance with discovery. Nevertheless, the metrics predominantly used to evaluate and optimize these systems have centered on accuracy and engagement [26]. This focus catalyzed significant algorithmic advancements, from collaborative filtering [13, 29] to deep sequential models [15, 37]. However, the prioritization of easily measurable signals has exposed a critical limitation, often termed the alignment problem: a model may predict the next interaction accurately, yet fail to align with the user's broader goals or well-being. This optimization imbalance has been shown to yield detrimental side effects, including polarization, addiction, and popularity bias, while discouraging the discovery of new user interests. This realization has accelerated a paradigm shift toward a human-centered approach. The critical question is evolving from "Is this recommendation accurate?" to "Is this recommendation worth your time?", which requires considering a richer set of objectives that extend beyond accuracy, such as diversity, novelty, and personalization [28].

To navigate these multifaceted objectives, industrial recommendation systems have accumulated a wide array of post-hoc adjustments applied during the ranking or post-ranking stage [22, 33]. We refer to this accumulated domain expertise as *human priors*. For example, diversity is often enforced by greedily selecting candidates that maximize a combined function of relevance and entropy (defined over manually tuned categories). To favor high-value interactions (e.g., purchases over clicks), practitioners typically build separate value models for each interaction type and apply heuristic weighting schemes [8, 23]. Similarly, balancing short-term engagement with long-term interests often involves temporal discounting heuristics or separate value models trained on different time horizons [31]. Furthermore, ensuring adequate personalization for minority users frequently relies on first identifying these minority users and then optimizing a separate value model, or boosting content based on demographic features [20].

Recently, the field is trending towards the development of end-to-end (E2E) generative recommendation foundation models [5, 9, 37]. While powerful, these models often attempt to learn user intent in an entirely unsupervised manner. Consequently, we still rely on the aforementioned post-hoc adjustments. However, these adjustments remain disconnected from the core representation learning process. As a result, the core model itself remains a black box, unaware of the crucial objectives. Additionally, to accommodate such adjustments, it is usually required to make specific changes to the model recommendations, which incurs additional cost. Alternative approaches attempt to explicitly address specific aspects, such as multi-interest networks for diversity [4, 18, 34] or disentanglement methods for interpretability [12, 21]. However, these methods typically require specialized architectures and their applicability in industry scenarios is still limited.

This dichotomy between complex post-hoc adjustments and unsupervised E2E models motivates a question: Instead of discarding the human priors accumulated over years of practice, can we integrate them directly into the learning process of generative recommender systems in a simple, effective, and interpretable manner? To this end, we propose a backbone-agnostic framework that seamlessly injects various human priors into the generative model training with lightweight adapters, by drawing inspiration from efficient decoding strategies in Large Language Models (LLMs) [3]. Unlike post-hoc filtering or architecture-specific modifications, these adapter heads guide the sequential model to learn user representations that are naturally disentangled. This renders the model inherently controllable, explainable, and better aligned with complex, real-world objectives.

Our main contributions are summarized as follows:

- We generalize the concept of "multi-interest" to "multi-faceted intent" by demonstrating the framework's effectiveness across diverse human priors, including semantic, behavioral, temporal, and graph priors.
- We propose a lightweight and backbone-agnostic framework that uses prior-conditioned adapter heads to disentangle multifaceted user intent in an end-to-end manner.
- We introduce a hierarchical composition strategy to model complex interactions across different prior types, providing a flexible inductive bias for learning compositional representations.
- Extensive experiments on three large-scale datasets demonstrate that our method not only improves standard accuracy metrics, but also yields significant improvements on other objectives, such as diversity, personalization, and user interest discovery[1].

2 Related Work

We position our work at the intersection of generative recommendation, multi-interest and disentangled representation learning, and the integration of structured knowledge, motivated by the broader shift toward human-centered recommendation.

2.1 Generative Recommenders

Modeling the temporal dynamics of user behavior is a fundamental challenge in recommender systems. Early approaches used Recurrent Neural Networks (e.g., GRU4Rec [14]). The field shifted significantly by adopting the Transformer architecture, which offers superior scalability and capacity for modeling long-range dependencies. SASRec [15] established a strong baseline using self-attention for next-item prediction, leading to variants such as BERT4Rec [30] (bidirectional modeling) and S3Rec [38] (self-supervised learning).

Recently, the focus has shifted to large-scale foundational models. Generative Recommenders, such as HSTU [37], frame recommendation as a sequential transduction task, demonstrating significant performance gains at scale. HLLM [5] introduces a hierarchical approach by stacking two large language models (LLMs): an item LLM to capture item content and a user LLM to model user behavior.

Despite these advances, the prevailing paradigm relies on encoding the user's history into a single, monolithic state vector. This representation bottleneck struggles to capture the heterogeneity and multi-faceted nature of user intent, often leading to suboptimal recommendations when interests conflict or evolve.

[1] Code: github.com/zhykoties/Multi-Head-Recommendation-with-Human-Priors

2.2 Modeling Multi-Faceted User Intent

To address the limitations of monolithic representations, works on multi-interest frameworks and disentangled representation learning emerged. They generally attempt to discover latent factors of user intent in an unsupervised manner and learn the preference distribution conditioned on these factors.

Multi-interest frameworks aim to extract multiple vectors representing distinct user preferences from a single sequence. Many prominent models adopt a "cluster-then-encode" paradigm, relying on algorithms to partition the user history before encoding. For instance, MIND [18] employed dynamic routing via capsule networks to group interactions. ComiRec [4] extended this with a controllable aggregation framework, and REMI [34] aimed to improve the stability of this process using regularization to prevent routing collapse.

Disentangled representation learning focuses on separating the underlying factors of variation in user behavior, often using Variational Autoencoders (VAEs). For example, MacridVAE [21] sought to separate high-level intentions from low-level preferences, while DualVAE [12] learns disentangled multi-aspect representations for both users and items, and ensures a correspondence between each aspect of the user representation and the item representation.

While valuable, these unsupervised approaches share critical limitations. First, they primarily focus on disentangling topic interests (e.g., "electronics" vs. "apparel"), often conflating other critical dimensions such as temporality or co-engagement structures. Second, the "cluster-then-encode" paradigm often relies on computationally intensive or potentially unstable discovery processes (e.g., dynamic routing, clustering). Third, the learned interest vectors often lack explicit semantic meaning. This lack of interpretability severely limits controllability, making it difficult to steer recommendations to align with business objectives, such as promoting more educational videos in order to comply with regulations.

2.3 Integration of Human Priors and Structure

There is growing recognition that integrating structured, human-understandable knowledge, or *human priors*, can enhance model performance and interpretability (e.g., expert-defined in-domain taxonomy [35]).

In recommender systems, human priors have traditionally been incorporated through rigid structures or post-hoc adjustments. Hierarchical models, such as HieRec [25], use fixed, predefined item taxonomies to create a static interest hierarchy. While effective for taxonomy-based disentanglement, such methods cannot easily accommodate diverse, orthogonal priors (e.g., temporal dynamics) that do not fit neatly into item categories. Alternatively, industrial systems often rely on brittle post-hoc heuristic rules, which are decoupled from the core learning process.

Knowledge and Adaptation in LLMs. In language models, there is significant work on enhancing models with external knowledge and structural biases. Methods like KnowBert [24] inject entity embeddings from knowledge bases to improve factual recall. Furthermore, introducing structural inductive biases, such as the Tree of Thoughts (ToT) framework [36], has been shown to improve reasoning abilities.

Table 1: Examples of human priors supported by our framework.

Prior Type	Description and Examples
Item	Semantic item attributes, such as product categories, content genres, or learned topic clusters.
Temporal	Evolution of user interests (e.g., short-term vs. long-term).
Event	The modality of the user-item interaction (e.g., *click*, *like*, *purchase*, *subscribe*).
Graph	Community-based item clusters derived from co-engagement or knowledge graphs.
User	User attributes such as demographics, subscription status, or clusters from a user co-interaction graph.

Drawing inspiration from these trends and efficient LLM adaptation techniques like Medusa [3], our work diverges from previous approaches by proposing an "encode-then-project" paradigm. We integrate diverse human priors directly into the end-to-end learning process using lightweight, prior-conditioned adapter heads. This bypasses the need for expensive unsupervised discovery or explicit history clustering, avoids rigid taxonomies, and yields representations that are inherently disentangled along interpretable and controllable axes.

3 Model

3.1 Problem Formulation

Let a user's interaction history be a sequence of items $x_{1:T} = (x_1, \ldots, x_T)$, where T is the context length, representing the number of item interactions in the history. The objective is to predict the user's future engagement over the next τ items, denoted as $\mathcal{Y} = \{y_{T+1}, \ldots, y_{T+\tau}\}$.

First, a sequential encoder f_θ (e.g., a decoder-only transformer) is used to map the interaction history $x_{1:T}$ into a latent user state representation $\mathbf{h}_T \in \mathbb{R}^d$. Let $\mathcal{V}$ be the set of all candidate items, and each item $i \in \mathcal{V}$ is represented by an embedding $\mathbf{e}_i \in \mathbb{R}^d$. These item embeddings can either be ID-based (e.g., HSTU) or semantic-based (e.g., HLLM). The conventional approach computes a relevance score for each candidate item i using the dot product between the user state and the item embedding:

$$s(i \mid \mathbf{h}_T) = \mathbf{h}_T^\top \mathbf{e}_i. \tag{1}$$

The top-K items with the highest scores are then recommended to the user. This approach relies on a single representation h_T to capture all facets of user intent, which may be suboptimal when interests are diverse, context-dependent, or evolving over time.

3.2 Incorporating Human Priors via Conditioned Query Heads

Real-world user behavior is often characterized by specific factors that can be formalized as "human priors". These priors provide a structured and interpretable way to partition the interaction space along meaningful dimensions, such as item semantics, temporal dynamics, or interaction modalities (see Table 1). To effectively incorporate these priors without modifying the backbone model

f_θ, we introduce a multi-head framework that employs multiple lightweight, prior-conditioned adapter heads to generate a set of specialized query embeddings, instead of relying on a single representation $\boldsymbol{h}_T$. Let $\mathcal{K}$ be the index set of the prior heads. With each head $k \in \mathcal{K}$ corresponding to a specific prior group (e.g., the "Sports" category), we can project the backbone's output $\mathbf{h}_T$ into different specialized query vectors $\mathbf{q}_1, \cdots \mathbf{q}_{|\mathcal{K}|}$. Inspired by the multi-head decoding structure of Medusa [3], we implement the projection through a residual adapter :

$$\mathbf{q}_k = \mathbf{h}_T + \mathrm{SiLU}\big(\mathbf{W}^{(k)}\mathbf{h}_T\big), \tag{2}$$

where $\mathbf{W}^{(k)} \in \mathbb{R}^{d\times d}$ is a learnable transformation matrix and SiLU is the activation function [11]. We initialize each $\mathbf{W}^{(k)}$ with zeros, ensuring that all heads output the same representation as the original user state $\mathbf{h}_T$ at the beginning of training. As training progresses, each individual head specializes only when supported by the training signal, whereas the backbone model is shared among all prior heads. This design allows the backbone to process a user's entire interaction history, while each prior head is dedicated to modeling a specific subset of interactions.

Compatibility masking. In our design, each head k is restricted to retrieve only items compatible with its associated prior group, with the set of such items denoted by $\Omega_k \subseteq \mathcal{V}$, where the definition of Ω_k depends on the prior type. For example, for *item-based* priors (e.g., categories), an item i belongs to Ω_k if it is labeled with category k, and for *event-based* priors, Ω_k includes items accessible through event type k. To enforce this specialization in inference, we define a score through the following compatibility masking:

$$s_k(i|\boldsymbol{h}_T) \;=\; \begin{cases} \mathbf{q}_k^\top \mathbf{e}_i, & i \in \Omega_k, \\ -\infty, & i \notin \Omega_k. \end{cases} \tag{3}$$

This masking approach filters out all the incompatible items for the prior heads and ensures each head can focus exclusively on the subset of items aligned with its prior group. As a result, in contrast to the score in Eq. (1), the resulting score in Eq. (3) is tailored to different items with their prior information, which leads to an explicit decomposition of user intent. Unlike conventional unsupervised approaches built on implicit latent factors [12, 18, 21], our method allows for more model interpretability as it guarantees that the learned representation $\boldsymbol{q}_k$ is identifiable. In addition, while conventional approaches suffer from the inherent uncertainty arising from the entanglement of user preferences and their underlying latent factors, our method mitigates this issue by disentangling this complexity into a set of more tractable sub-tasks, thereby enhancing the computational efficiency.

With compatibility masking, different query heads are specialized with distinct functional roles determined by the specified priors. Thus, when predicting for one prior group, our method can largely reduce the reliance on irrelevant features, which minimizes the mutual interference among these different objectives. As a result, the model can show stronger predictive capability for each prior group, and thus result in a performance improvement with the cooperation of the heads.

3.3 Hierarchical Composition of Priors

Practical recommendation settings often involve multiple, potentially interacting priors (e.g., combining item categories with temporal interests). Given D distinct sets of priors $\{\mathcal{P}^{(1)}, \dots, \mathcal{P}^{(D)}\}$ with cardinalities $C^{(1)}, \dots, C^{(D)}$ respectively, a key challenge when generalizing the adapter mechanism (proposed previously in Eq. (2)) is how to effectively capture the interactions between different prior sets while mitigating data sparsity for rare combinations.

We introduce a hierarchical composition strategy that organizes the adapters sequentially into a tree structure. This architecture enforces a coarse-to-fine specialization process, encouraging the model to first learn robust, shared intermediate representations at the upper levels before refining for specific prior combinations.

This design is motivated by Bayesian hierarchical modeling [2], which has the "shrinkage" effect, where group-level estimates are pulled towards a common mean as an effective form of regularization, preventing overfitting in rare prior combinations. Furthermore, this structural inductive bias mirrors recent advances in Large Language Models (LLMs), where hierarchical structures are employed to enhance reasoning, such as in Tree of Thoughts [36].

Starting with the base representation $\mathbf{z}^{(0)} = \mathbf{h}_T$, we recursively apply prior-specific residual adapters. At depth d, the representation corresponding to the path $(g_1, \dots, g_d)$ is:

$$\mathbf{z}^{(d)}_{g_1,\dots,g_d} = \mathbf{z}^{(d-1)}_{g_1,\dots,g_{d-1}} + \mathcal{A}^{(d)}_{g_1,\dots,g_d}\big(\mathbf{z}^{(d-1)}_{g_1,\dots,g_{d-1}}\big). \tag{4}$$

Here, $\mathcal{A}^{(d)}_{g_1,\dots,g_d}$ denotes *path-dependent adapters*, where the parameters at depth d are conditioned on the entire upstream path $(g_1, \dots, g_d)$. It is defined as:

$$\mathcal{A}^{(d)}_{g_1,\dots,g_d}(\mathbf{z}) = \mathrm{SiLU}\big(\mathbf{W}^{(d)}_{g_1,\dots,g_d}\mathbf{z}\big) + \mathbf{e}_{g_{d-1}}, \tag{5}$$

where $\mathbf{W}^{(d)}_{g_1,\dots,g_d} \in \mathbb{R}^{d\times d}$ are path-dependent weights. The final queries are corresponding to the leaf nodes $\mathbf{q}_{g_1,\dots,g_D} = \mathbf{z}^{(D)}_{g_1,\dots,g_D}$ in the tree structure. We also incorporate a learned group embedding $\mathbf{e}_{g_{d-1}} \in \mathbb{R}^d$. This embedding is shared across all branches that include g_{d-1} (e.g., across all sub-trees rooted at "short-term interest"), encouraging information sharing among related heads.

Hierarchical Compatibility. The eligible item set for a hierarchical head is defined by the intersection of the compatibilities across all involved priors:

$$\Omega_{g_1,\dots,g_D} = \bigcap_{d=1}^{D} \Omega^{(d)}_{g_d}.$$

An item is eligible for the head $(g_1, \dots, g_D)$ if and only if it is compatible with *all* involved priors along the path.

3.4 Training Objective

For a specific head k, the set of positive examples is defined as the subset of future ground-truth items $\mathcal{Y}$ compatible with that head:

$$\mathcal{Y}_k = \{\, y \in \mathcal{Y} : y \in \Omega_k \,\}. \tag{6}$$

Heads for which $\mathcal{Y}_k = \emptyset$ in a batch are excluded from the loss computation for that batch.

We optimize the parameters of each head using a unified loss framework, which can be instantiated as either a next-token prediction loss (for ID-based embeddings) or a contrastive learning loss (for semantic-based embeddings):

$$\mathcal{L}_{k,t} \;=\; -\mathbb{1}_{y_{T+t} \in \mathcal{Y}_k} \log \frac{\exp(s_k(y_{T+t}))}{\sum_{j \in \tilde{\Omega}_k} \exp(s_k(j))}. \tag{7}$$

Here, $\mathcal{L}_{k,t}$ is the loss for head k with y_{T+t} as the positive item. $\tilde{\Omega}_k \subseteq \Omega_k$ is the set of items over which the softmax is computed. For next-token prediction, $\tilde{\Omega}_k$ can be all the compatible items Ω_k. For contrastive learning, it is typically a subset containing the positives and some sampled negatives.

We restrict $\tilde{\Omega}_k$ to be only from Ω_k. This *in-group negative sampling* forces the head to discriminate among items within the same prior group. This naturally exposes the model to harder negatives (semantically or contextually similar items), leading to improved representations [27].

To properly balance the contributions from different heads and prioritize near-future predictions, we introduce a reweighting scheme. The final loss is a sum over the forecast horizon, with each step weighted by a temporal discount factor:

$$\mathcal{L} \;=\; \sum_{t=1}^{\tau} \gamma^{t-1} \sum_{k \in \mathcal{K}} w_k \, \mathcal{L}_{k,t}, \tag{8}$$

where $\mathcal{L}_{k,t}$ is defined in Eq. (7). This formulation incorporates two mechanisms:

1. **Frequency Balancing:** To mitigate the impact of data imbalance across heads and prevent common priors from dominating the loss, we normalize by the relative frequency of each combination of priors: $w_k^{\text{freq}} = \frac{|\mathcal{Y}_k|}{\sum_{j \in \mathcal{K}} |\mathcal{Y}_j|}$.
2. **Temporal Discounting:** We apply a discount factor $\gamma \in (0, 1]$ to prioritize near-future predictions, since predicting the very next item is often more critical than the more distant items, and the labels are also less noisy.

3.5 Inference and Score Fusion

During inference, given a user state $\mathbf{h}_T$, we compute all prior-conditioned queries $\{\mathbf{q}_k\}_{k \in \mathcal{K}}$. For a candidate item i, we identify the set of eligible heads $\mathcal{H}(i) = \{k : i \in \Omega_k\}$. We then fuse the scores $\{s_k(i)\}_{k \in \mathcal{H}(i)}$ to obtain a final relevance score $S(i)$, which is then used to rank the candidates.

We adopt a *maximum fusion* strategy: $S_{\max}(i) \;=\; \max_{k \in \mathcal{H}(i)} s_k(i)$, which allows the most relevant specialist to dominate[2]. Not only is it computationally simple, but it also enhances interpretability by providing a clear explanation for the recommendation (e.g., "this item was recommended because it strongly matches your short-term interest in electronics").

3.6 Implementation Details

In Figure 1, we illustrate an instantiation of our hierarchical framework with Temporal (LT/ST_2) and Graph Priors (C_1, C_2). The forecast horizon ($\tau = 6$) is split into short-term (ST) and long-term (LT) segments. In the first layer, adapters specialize on these temporal segments, yielding the intermediate representations $\mathbf{z}_{ST}^{(1)}$ and $\mathbf{z}_{LT}^{(1)}$. The adapter $\mathcal{A}_{ST}^{(1)}$ is trained using only ground-truth items in the ST segment, while $\mathcal{A}_{LT}^{(1)}$ is trained using items in the LT segment. The second layer further refines this specialization based on Graph Prior. For example, the adapter $\mathcal{A}_{ST,C_1}^{(2)}$ is optimized specifically to predict items that belong to the cluster C_1 (green) and the ST segment. In other words, each leaf head is responsible only for the intersection of its associated priors.

Crucially, the backbone processes the full context, regardless of the item's cluster. Specialization occurs only in the adapter heads. During training, we use in-group negative sampling (Section 3.4). Graph heads sample negatives from compatible cluster sets (e.g., Ω_{C_1}), whereas Temporal heads restrict training signals based on the item's position in the forecast horizon.

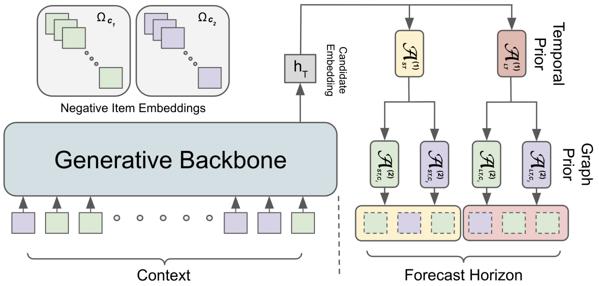

Figure 1: An instantiation of the hierarchical composition strategy with Temporal (LT/ST$_2$) and Graph Priors.

The proposed framework is model-agnostic and can be integrated with any generative recommender system that produces a dense user representation $\mathbf{h}_T$. The efficiency of the approach stems from the lightweight nature of the adapters[3] and the parallelizability of the query computations. Formally, let K denote the total number of heads and d the hidden dimension. The added parameter complexity is $O(K \cdot d^2)$. The value of K depends on the structure: it is the sum of categories (e.g., taxonomies), or the product for independent priors (e.g., Temporal x Graph). During inference, adapters at the same depth level are executed in parallel, ensuring the latency overhead remains low. Per-group indices $\{\Omega_k\}$ are pre-computed and cached, enabling efficient computation of the masked scores (Eq. (3)) via batched matrix multiplication.

4 Experiments

4.1 Datasets and Prior Instantiation

To instantiate and evaluate different types of human priors (Table 1), we select three real-world datasets from various domains: video (Pixel8M), e-Commerce (MerRec), and news (EB-NeRD). Detailed statistics are provided in Appendix A (Table 4).

4.1.1 Pixel8M (Video). Pixel8M [7] is a large-scale dataset from an online video sharing platform, featuring rich multimodal item content with text and images. As the industry trend is to incorporate more modalities, this allows us to test whether our framework provides additional benefits even when the backbone model can leverage these modalities to recommend diverse contents.

Semantic Item Prior: To create a structured semantic prior, we consolidated the dataset's 111 highly unbalanced and sometimes

[2] We explore alternative fusion strategies in Section A.1.3.

[3] We show each head only incurs an extra 0.14% of overall parameters in Section 4.3.

redundant tags into eight high-level categories: "Real Life", "Informational & Educational", "Fictional Character", "Music", "Science & Technology", "Entertainment", "Gaming", "Performance & Arts". To perform this task easily, consistently, and at scale, we follow the practical approach [10] and prompted ChatGPT to assign each original tag to one or more categories.

4.1.2 MerRec (E-commerce). MerRec [19] is derived from the Mercari C2C marketplace. It is characterized by exceptionally long interaction sequences (there are 119756 users who have at least 2000 interactions) and diverse user behaviors.

Event Prior: Users interact with items with one of six event types: "item view", "item like", "add to cart", "offer make", "Buy start", and "Buy complete". These events represent different levels of user intent. A key challenge here is that "offer make", "Buy start", and "Buy complete" only occur less than 1% of the time, but they are also most directly related to monetization. We use event types as priors to investigate the framework's ability to specialize on sparse, high-value signals.

4.1.3 EB-NeRD (News). EB-NeRD [16] is a news recommendation dataset with high-quality textual content. This dataset is less sparse than the previous two datasets, and the interaction patterns in news consumption often reflect the underlying community structures.

Graph Prior: We construct an item co-engagement graph, where an edge exists if two items are interacted with by the same user. To discover underlying structural priors, for simplicity, we apply the off-the-shelf Leiden algorithm [32] from the `igraph` package, an established method for community detection that optimizes modularity and guarantees that the resulting communities are well connected. We control the influence of highly active users and merge very small clusters (details in Appendix A). The resulting item clusters are used as graph-based priors, testing the framework's ability to leverage community structures[4].

4.1.4 General. **Temporal Prior.** In addition to domain-specific priors, we instantiate temporal priors, applicable across all datasets, to capture the evolution of user interests. Given a forecast horizon τ, we divide it into n contiguous segments. Each segment corresponds to a prior head (e.g., short-term vs. long-term), trained only on ground-truth items falling within that specific temporal segment.

4.2 Experiment Setup

We integrate our framework with two recent generative recommender architectures to demonstrate its generalizability:

- **HSTU** [37]: A scalable, Transformer-based architecture representing the state-of-the-art in **ID-based** modeling. It uses learned item ID embeddings, and is trained with a next-item prediction objective. We experiment with five sizes, from 12.42M to 1B parameters. We report based on size 3 by default.
- **HLLM** [5]: A hierarchical LLM-based architecture representing the state-of-the-art in **semantic-based** modeling. It uses an Item LLM to derive item embeddings from text and visual content, and is trained with a contrastive learning objective for next item prediction.

We compare the performance of these backbone models against their counterparts enhanced with our prior-conditioned adapter framework. We also compare against the following baselines:

- **ComiRec** [4] is a representative *multi-interest* network that outputs multiple embeddings as the different interests for each user, and uses a controllable aggregation framework to balance diversity and accuracy.
- **REMI** [34] improves *multi-interest* networks like ComiRec by introducing Interest-aware Hard Negative Mining and a Routing Regularization term to mitigate routing collapse.
- **DualVAE** [12] learns *disentangled* multi-aspect representations for both users and items, and uses neighborhood-enhanced contrastive learning to ensure a direct correspondence between each aspect of item representations and user representations.

Architectural details and hyperparameters are provided in Appendix A.[5] We evaluate the recommendation accuracy using standard retrieval metrics: **Recall@K** and **NDCG@K** (Normalized Discounted Cumulative Gain).

4.3 Main Results

Table 2 summarizes the overall performance across the three datasets and two backbone architectures. The integration of human priors consistently improves both Recall and NDCG over different settings. Furthermore, combining multiple priors (e.g., Item + Temporal, Graph + Temporal) can lead to additional performance gains, demonstrating that our framework can effectively capture multi-faceted user intents. On EB-NeRD, with 8 temporal segments and 11 clusters, our method is able to scale to a total of 88 heads. Notably, our adapter heads are very lightweight. In the HSTU case, a single head only takes up 0.14% of the overall model parameters.

Both ComiRec and REMI underperform their counterparts when we inject human prior into HSTU. The reason is that instead of relying on purely unsupervised methods to discover latent interests as in multi-interest networks, our approach uses these explicit priors as a form of weak supervision to guide the model in learning different user intents as well as disentangled representations. As for DualVAE, we observe it to be very unstable when implemented based on the deep HSTU backbone. With its original shallow encoders, it only marginally outperforms HSTU on Pixel8M and even slightly lags behind HSTU on MerRec and EB-NeRD.

4.4 Benefits Beyond Standard Metrics

4.4.1 A Better Accuracy-Diversity Trade-off. We demonstrate that our method not only improves traditional ranking metrics such as Recall and NDCG, but also promotes recommendation diversity. To quantify this, we define an entropy-based metric in terms of the eight binary item categories on the Pixel8M dataset, $H@\text{K} = -\sum_{j=1}^{8}\left(\frac{n_j}{K}\log_2\frac{n_j}{K}\right)$, where n_j is the number of top-K items for

[4] While this algorithmic clustering is a practical choice for our implementation, its success also underscores the framework's tolerance for noisy or approximated priors, suggesting it can derive benefits even from imperfect structural information.

[5] For a fair comparison, we implement ComiRec and REMI on top of the HSTU backbone, and we validated that they achieve better performances compared to the original dense layers. However, DualVAE becomes very unstable once we switch to deep encoders, instead of the shallow encoders in its codebase, so we stick with its original implementation.

Table 2: Overall performance comparison. Human priors consistently lead to improvements over the backbone models (HSTU and HLLM). The backbones and baselines are highlighted in gray. Note that HLLM and HSTU results are not directly comparable due to different context lengths used (See Appendix A), and all the baselines are run under the HSTU settings and should only be compared to HSTU.

		Recall (%)		NDCG (%)	
Dataset	Model	@5	@10	@5	@10
Pixel8M	HLLM	0.84	1.37	1.46	1.42
	+Item	0.91	1.48	1.57	1.54
	+LT/ST	0.88	1.44	1.52	1.50
	+Both	**0.92**	**1.50**	**1.59**	**1.56**
	ComiRec	1.04	1.70	1.80	1.77
	REMI	1.13	1.81	1.99	1.92
	DualVAE	0.95	1.49	1.67	1.60
	HSTU	0.90	1.45	1.56	1.53
	+Item	1.08	1.75	1.88	1.83
	+LT/ST	1.15	1.90	1.98	1.96
	+Both	**1.23**	**2.00**	**2.12**	**2.09**
MerRec	HLLM	33.83	42.03	24.38	27.05
	+Event	**35.85**	**43.48**	**26.87**	**29.36**
	ComiRec	40.74	49.49	30.07	33.01
	REMI	41.46	49.27	31.55	33.15
	DualVAE	38.36	48.20	27.29	31.03
	HSTU	40.37	49.30	29.71	32.61
	+Event	**42.61**	**50.33**	**33.49**	**35.99**
EB-NeRD	HLLM	18.14	31.23	26.40	29.47
	+Graph	21.09	36.05	32.19	35.05
	+LT/ST	19.76	34.05	28.17	31.57
	+Both	**21.54**	**36.24**	**32.38**	**35.16**
	ComiRec	21.47	35.71	32.18	34.75
	REMI	21.61	34.79	33.11	34.88
	DualVAE	18.54	31.24	28.28	30.61
	HSTU	19.77	32.54	30.36	32.30
	+Graph	20.78	34.48	31.59	33.84
	+LT/ST	21.50	36.28	32.20	34.99
	+Both	**22.36**	**37.05**	**33.87**	**36.24**

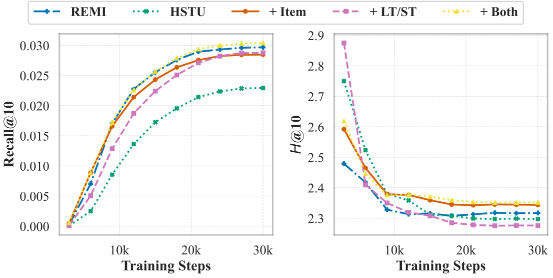

Figure 2: Evolution of entropy as training progresses on the validation set. Here HSTU is the backbone model.

which the j-th binary feature is active. Intuitively, a higher entropy means that the top-K recommended items are semantically more diverse.

As shown in Figure 2, all model variants begin training with a high entropy, which gradually declines as the training progresses. Interestingly, higher entropy typically corresponds to a lower NDCG due to the accuracy-diversity trade-off. However, we observe that injecting item priors can partially break this constraint: the variants using Item Prior and both priors simultaneously achieve a higher NDCG while maintaining a higher entropy. This balance between relevance and diversity is also observed in HLLM (Appendix Table 8). We also include a strong multi-interest network baseline, REMI. Without explicit supervision from item priors, it only marginally improves diversity over the HSTU backbone.

4.4.2 User Interest Exploration. Exploration in recommender systems aims to uncover content a user may like but has not yet engaged with. Although it improves long-term user engagement, it is often believed to negatively affect near-term experience [6], similar to the exploration versus exploitation dilemma in reinforcement learning. We argue that adding multiple heads for human priors can strike a good balance between the two. To evaluate this, we analyze the performance of our method on a targeted subset of users in the Pixel8M dataset using HLLM. Specifically, we identify users who interacted with item features during the forecast horizon of the test split, but who had never engaged with those features in their prior history. In these cases, the model's success hinges on its ability to recommend items from entirely new categories. Applying this criterion, we identified a total of 283,497 such users.

Table 3 compares the relative improvement over the HLLM without priors baseline achieved by different prior configurations on the standard evaluation set (*All Users*) versus this subset (*New Interest*). We also quantify the *relative boost*, which measures how much more the method improves performance on the *New Interest* subset compared to *All Users*.

We observe that the relative improvements are consistently higher for this *New Interest* subset across all configurations. Moreover, incorporating Item Prior yields substantially more benefits compared to using only Temporal Prior (LT/ST). For example, HLLM with both LT/ST$_1$ and Item Prior achieves a remarkable relative boost of +15.76% for NDCG@10, compared to only +3.58% for LT/ST$_1$ alone. This supports the intuition that when learning the different categories together, the user embedding will be biased against the minority items, while dedicating specific heads to the minority items allows us to retain the capacity and encourage exploration toward novel categories. Case studies in Section A.1.1 further illustrate this insight.

4.4.3 Personalization. Another common issue in recommender systems is popularity bias. In some cases, it means that the behaviors of some users, whose interests are different from the majority, might receive inferior recommendations. To address this issue, we instantiated User Prior on EB-NeRD, constructed similar to Graph Prior. But here, the nodes are the users in the co-engagement graph, and an edge exists between the two users if they interacted with the

Table 3: The relative improvement over the HLLM baseline for *All Users* vs. *New Interest*. Both priors help with user interest exploration, and Item Prior brings a even larger gain compared to LT/ST interest due to more direct supervision.

Variant	Split	N@10	R@10	N@200	R@200
LT/ST$_1$	All	7.16%	7.80%	8.10%	8.50%
	New	7.41%	8.50%	8.90%	9.70%
	Rel. Boost	*+3.58%*	*+9.00%*	*+10.30%*	*+14.40%*
LT/ST$_2$	All	5.23%	6.20%	8.00%	9.20%
	New	5.37%	6.30%	8.40%	9.80%
	Rel. Boost	*+2.68%*	*+1.70%*	*+5.50%*	*+6.80%*
LT/ST$_1$ + Item	All	8.53%	9.00%	7.90%	7.50%
	New	9.87%	10.40%	9.20%	9.00%
	Rel. Boost	***+15.76%***	***+16.50%***	***+17.50%***	***+19.50%***
LT/ST$_2$ + Item	All	9.63%	10.40%	10.70%	11.20%
	New	10.22%	11.20%	11.80%	12.50%
	Rel. Boost	*+6.10%*	*+7.50%*	*+9.60%*	*+11.80%*
LT/ST$_4$ + Item	All	8.02%	8.90%	9.80%	10.60%
	New	8.49%	9.50%	10.50%	11.50%
	Rel. Boost	*+5.94%*	*+7.30%*	*+6.90%*	*+8.30%*

same item in the train set. To avoid popular items from making the graph too dense, for each item, we random sample a maximum of 2000 users that have interacted with it before generating the edges. We then employ the Leiden algorithm [32] with the modularity objective to cluster the graph into a total of 9 user groups. We define *User Prior* by assigning an adapter to each group.

As shown in Figure 3, before introducing any prior, the minority groups with fewer users generally suffer from lower recommendation quality. However, by dedicating a separate head to each user group, we effectively lower the impact of the majority users from dominating the query representation, which allows us to better personalize the recommendations for the minority users. This results in bigger improvements for groups with fewer users, and recommendation quality looks more balanced after *User Prior* is introduced.

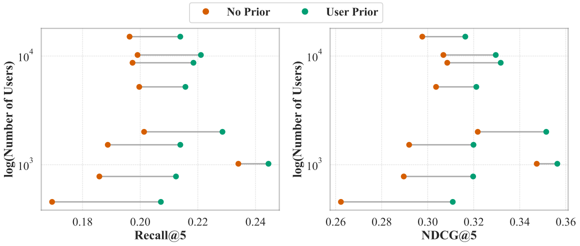

Figure 3: *User Prior* leads to more personalized recommendations, especially on the minority user groups.

4.5 Scalability

A key trend in recommender systems is the use of longer context lengths to enhance personalization. While scaling laws typically require more training data to improve larger models, our experimental setup with a fixed dataset introduces a trade-off: increasing the context length reduces the number of available training windows. To investigate this, we test HSTU with longer context lengths and larger model sizes on Pixel8M. In Figure 4, the left subplot shows that for size 4 and 1b, the base model (LT/ST$_1$, equivalent to the discounted loss) struggles to benefit from increasing context as we increase the context length beyond 20 items. However, in the right subplot, when guided by human priors, the same model architecture continues to extract performance gains from longer contexts and larger model sizes, although the magnitude of increase slowly plateaus. This finding suggests that the structural information imposed by human priors facilitates more efficient learning, allowing the model to better benefit from increasing context and larger model sizes when the amount of training data is fixed.

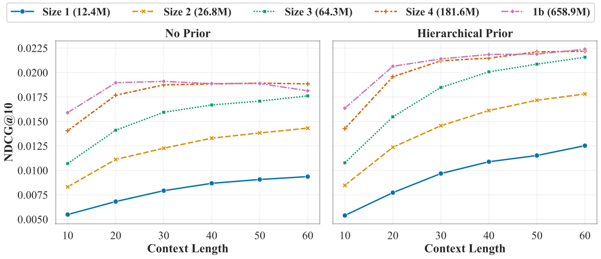

Figure 4: Scaling by context lengths and sizes for HSTU.

5 Conclusion

By integrating human priors directly into generative recommenders' learning process, our proposed framework offers a principled approach to aligning recommendations with multifaceted human objectives. We view the capacity to leverage human priors not as a dependency, but as a strategic advantage tailored to the unique landscape of recommender systems. Unlike domains where human priors are scarce, the recommendation field possesses decades of market-validated expertise, resulting in many high-quality priors. Our experiments demonstrate that prior-conditioned adapter heads not only enhance accuracy, but also improve diversity, novelty, and personalization, which are key dimensions of user experience that are often overlooked. The framework's backbone-agnostic design and hierarchical composition strategy further enable flexible modeling of complex user intent, making it broadly applicable.

While this work showcases the efficacy of using pre-defined priors, we view this as a fundamental but not the final step. A promising future direction is to establish a formal methodology for what constitutes a "good" prior, guiding the prior curation whether through human-in-the-loop or automated discovery of salient priors directly from data. Advancing the architectural fusion of these priors with more dynamic, context-aware mechanisms will also be critical. As the first attempt towards this goal, this work aims to strengthen the bridge between abstract human knowledge and the end-to-end optimization of large-scale generative recommenders.

References

[1] Gediminas Adomavicius and Alexander Tuzhilin. 2005. 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 (2005), 734–749.

[2] Greg M. Allenby, Peter E. Rossi, and Robert E. McCulloch. 2005. Hierarchical Bayes Models: A Practitioners Guide. *Econometrics eJournal* (2005).

[3] Tianle Cai, Yuhong Li, Zhengyang Geng, Hongwu Peng, Jason D Lee, Deming Chen, and Tri Dao. 2024. Medusa: Simple LLM Inference Acceleration Framework with Multiple Decoding Heads. In *International Conference on Machine Learning*. PMLR, 5209–5235.

[4] Yukuo Cen, Jianwei Zhang, Xu Zou, Chang Zhou, Hongxia Yang, and Jie Tang. 2020. Controllable Multi-Interest Framework for Recommendation. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2942–2951.

[5] Junyi Chen, Lu Chi, Bingyue Peng, and Zehuan Yuan. 2024. HLLM: Enhancing Sequential Recommendations via Hierarchical Large Language Models for Item and User Modeling. arXiv:2409.12740 [cs.IR] https://arxiv.org/abs/2409.12740

[6] Minmin Chen, Yuyan Wang, Can Xu, Ya Le, Mohit Sharma, Lee Richardson, Su-Lin Wu, and Ed Chi. 2021. Values of User Exploration in Recommender Systems. In *Proceedings of the 15th acm Conference on recommender systems*. 85–95.

[7] Yu Cheng, Yunzhu Pan, Jiaqi Zhang, Yongxin Ni, Aixin Sun, and Fajie Yuan. 2024. An image dataset for benchmarking recommender systems with raw pixels. In *Proceedings of the 2024 SIAM International Conference on Data Mining*. 418–426.

[8] Alvise De Biasio, Andrea Montagna, Fabio Aiolli, and Nicolò Navarin. 2023. A systematic review of value-aware recommender systems. *Expert Systems with Applications* 226 (2023), 120131.

[9] Jiaxin Deng, Shiyao Wang, Kuo Cai, Lejian Ren, Qigen Hu, Weifeng Ding, Qiang Luo, and Guorui Zhou. 2025. Onerec: Unifying retrieve and rank with generative recommender and iterative preference alignment.

[10] Bosheng Ding, Chengwei Qin, Linlin Liu, Yew Ken Chia, Boyang Li, Shafiq Joty, and Lidong Bing. 2023. Is GPT-3 a Good Data Annotator?. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 11173–11195.

[11] Stefan Elfwing, Eiji Uchibe, and Kenji Doya. 2018. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *Neural networks* 107 (2018), 3–11.

[12] Zhiqiang Guo, Guohui Li, Jianjun Li, Chaoyang Wang, and Si Shi. 2024. Dualvae: Dual disentangled variational autoencoder for recommendation. In *Proceedings of the 2024 SIAM International Conference on Data Mining (SDM)*. SIAM, 571–579.

[13] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. In *Proceedings of the 26th international conference on world wide web*. 173–182.

[14] B Hidasi. 2015. Session-based Recommendations with Recurrent Neural Networks. *arXiv preprint arXiv:1511.06939* (2015).

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

[16] Johannes Kruse, Kasper Lindskow, Saikishore Kalloori, Marco Polignano, Claudio Pomo, Abhishek Srivastava, Anshuk Uppal, Michael Riis Andersen, and Jes Frellsen. 2024. EB-NeRD a large-scale dataset for news recommendation. In *Proceedings of the Recommender Systems Challenge 2024*. 1–11.

[17] Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. 2020. Conservative q-learning for offline reinforcement learning. *Advances in neural information processing systems* 33 (2020), 1179–1191.

[18] Chao Li, Zhiyuan Liu, Mengmeng Wu, Yuchi Xu, Huan Zhao, Pipei Huang, Guoliang Kang, Qiwei Chen, Wei Li, and Dik Lun Lee. 2019. Multi-interest network with dynamic routing for recommendation at Tmall. In *Proceedings of the 28th ACM international conference on information and knowledge management*. 2615–2623.

[19] Lichi Li, Zainul Abi Din, Zhen Tan, Sam London, Tianlong Chen, and Ajay Daptardar. 2025. Merrec: A large-scale multipurpose mercari dataset for consumer-to-consumer recommendation systems. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 1*. 2371–2382.

[20] Feng Lin, Chaoyue Zhao, Xiaoning Qian, Kendra Vehik, and Shuai Huang. 2025. Fair Collaborative Learning (FairCL): A Method to Improve Fairness amid Personalization. *INFORMS Journal on Data Science* 4, 1 (2025), 67–84.

[21] Jianxin Ma, Chang Zhou, Peng Cui, Hongxia Yang, and Wenwu Zhu. 2019. *Learning disentangled representations for recommendation.*

[22] Thomas M McDonald, Lucas Maystre, Mounia Lalmas, Daniel Russo, and Kamil Ciosek. 2023. Impatient bandits: Optimizing recommendations for the long-term without delay. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 1687–1697.

[23] Changhua Pei, Xinru Yang, Qing Cui, Xiao Lin, Fei Sun, Peng Jiang, Wenwu Ou, and Yongfeng Zhang. 2019. Value-aware recommendation based on reinforcement profit maximization. In *The World Wide Web Conference*. 3123–3129.

[24] Matthew E Peters, Mark Neumann, Robert Logan, Roy Schwartz, Vidur Joshi, Sameer Singh, and Noah A Smith. 2019. Knowledge enhanced contextual word representations. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*. 43–54.

[25] Tao Qi, Fangzhao Wu, Chuhan Wu, Peiru Yang, Yang Yu, Xing Xie, and Yongfeng Huang. 2021. HieRec: Hierarchical User Interest Modeling for Personalized News Recommendation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. 5446–5456.

[26] Francesco Ricci, Lior Rokach, and Bracha Shapira. 2021. Recommender systems: Techniques, applications, and challenges. *Recommender systems handbook* (2021), 1–35.

[27] Joshua David Robinson, Ching-Yao Chuang, Suvrit Sra, and Stefanie Jegelka. 2021. Contrastive Learning with Hard Negative Samples. In *International Conference on Learning Representations*. https://openreview.net/forum?id=CR1XOQ0UTh-

[28] Alan Said, Maria Soledad Pera, and Michael D. Ekstrand. 2025. We're Still Doing It (All) Wrong: Recommender Systems, Fifteen Years Later. arXiv:2509.09414 [cs.IR] https://arxiv.org/abs/2509.09414

[29] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. 2001. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web*. 285–295.

[30] Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. 2019. BERT4Rec: Sequential recommendation with bidirectional encoder representations from transformer. In *Proceedings of the 28th ACM international conference on information and knowledge management*. 1441–1450.

[31] Jiaxi Tang, Francois Belletti, Sagar Jain, Minmin Chen, Alex Beutel, Can Xu, and Ed H. Chi. 2019. Towards neural mixture recommender for long range dependent user sequences. In *The World Wide Web Conference*. 1782–1793.

[32] Vincent A Traag, Ludo Waltman, and Nees Jan Van Eck. 2019. From Louvain to Leiden: guaranteeing well-connected communities. *Scientific Reports* 9, 1 (2019), 1–12.

[33] Yihan Wu, Mingze Gao, Haoran Liu, Weiwei Li, Kevin Han, Junfeng Pan, Xinyi Zhang, Jiawei Wen, and Gedi Zhou. 2025. GAS: Large-Scale Heterogeneous Personalization in Social Network Applications at Meta. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*. 5049–5058.

[34] Yueqi Xie, Jingqi Gao, Peilin Zhou, Qichen Ye, Yining Hua, Jae Boum Kim, Fangzhao Wu, and Sunghun Kim. 2023. Rethinking multi-interest learning for candidate matching in recommender systems. In *Proceedings of the 17th ACM Conference on Recommender Systems*. 283–293.

[35] Diji Yang and Omar Alonso. 2024. A Bespoke Question Intent Taxonomy for E-commerce. In *Proceedings of the ACM SIGIR Workshop on eCommerce 2024 co-located with the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2024), Washington D.C., USA, July 18, 2024 (CEUR Workshop Proceedings, Vol. 3843)*, Surya Kallumadi, Yubin Kim, Tracy Holloway King, Maarten de Rijke, and Vamsi Salaka (Eds.). https://ceur-ws.org/Vol-3843/paper_2.pdf

[36] Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik R Narasimhan. 2023. Tree of Thoughts: Deliberate Problem Solving with Large Language Models. In *Thirty-seventh Conference on Neural Information Processing Systems*. https://openreview.net/forum?id=5Xc1ecxO1h

[37] Jiaqi Zhai, Lucy Liao, Xing Liu, Yueming Wang, Rui Li, Xuan Cao, Leon Gao, Zhaojie Gong, Fangda Gu, Jiayuan He, et al. 2024. Actions Speak Louder than Words: Trillion-Parameter Sequential Transducers for Generative Recommendations. In *International Conference on Machine Learning*. PMLR, 58484–58509.

[38] Kun Zhou, Hui Wang, Wayne Xin Zhao, Yutao Zhu, Sirui Wang, Fuzheng Zhang, Zhongyuan Wang, and Ji-Rong Wen. 2020. S3-rec: Self-supervised learning for sequential recommendation with mutual information maximization. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. 1893–1902.

A Experiment Details

Table 4: Statistics of the datasets after preprocessing. Filtering criteria vary slightly depending on the backbone model.

Dataset	Backbone	# Users	# Items	# Interactions
Pixel8M	HSTU	561,737	398,261	57M
	HLLM	2,220,506	404,182	102M
MerRec	Both	119,754	1,255,665	521M
EB-NeRD	Both	44,968	25,216	30M

	Size 1	Size 2	Size 3	Size 4	1B
num_layers	4	8	12	16	22
num_heads	4	8	8	16	32
d_model	128	256	512	1024	2048
dropout	0.1	0.1	0.2	0.2	0.4
Total Params	12.4 M	26.8 M	64.3 M	181.6 M	658.9 M

Table 5: The list of hyperparameters in the five model sizes of HSTU, along with the total parameter counts. For *1B*, we use the same hyperparameters as TinyLlama as in the original HLLM paper. However, the total number of parameters is less than 1B due to the simplification of the feed-forward blocks in HSTU.

Pixel8M. For HSTU, we filter out users with fewer than 50 interactions. For HLLM, we filter out users with fewer than 20 interactions.

- Settings: For HSTU, we use a context length (T) of 50 and a forecast horizon (τ) of 8. For HLLM, we use $T = 10$ and $\tau = 4$ due to its higher computation cost.
- Modalities: Text inputs are "title", "tag", and "description". Images are rescaled to 224 × 224 pixels.
- Item Prior Details: Over 5% of all items are labeled with the *Miscellaneous* tag, which appear to be unlabeled data that can randomly come from any of the other tags. We have to simply set all 8 binary features to be True for this tag. However, this also demonstrates the robustness of our method to noise in the human prior. The resulting normalized frequency distribution of the eight features is: *Entertainment*: 24.95%, *Real life*: 21.10%, *Performance & Arts*: 15.30%, *Informational & Educational*: 12.69%, *Fictional character*: 9.00%, *Music*: 6.29%, *Gaming*: 6.89%, and *Science & technology*: 3.77%. For example, the original tag "Food Production" is labeled with *Real life*, *Informational & Educational*, and *Entertainment*, while the original tag "Celebrities Mix" is labeled with *Real life*, *Entertainment*, *Performance & Arts*.

MerRec. We select the user subset with over 2,000 interactions.

- Settings: For HSTU, we use a context length of $T = 400$, and for HLLM, we use $T = 50$, both to predict one item ahead.
- Modalities: Text inputs are "c2_name" and "brand_name".

EB-NeRD. We select the 44,968 users with over 512 interactions.

- Settings: $T = 50, \tau = 8$ for HSTU; $T = 24, \tau = 4$ for HLLM.
- Modalities: Text inputs are "title", "subtitle", and "topics".
- Graph Construction Details: For the co-engagement graph, we cap the contribution of highly active users by considering only their last 1000 interactions to prevent them from dominating the graph structure. When running the Leiden algorithm [32] (implemented via the `igraph` package), we use modularity as the optimization objective and tune the resolution parameter based on the desired number of clusters. Small clusters falling below a size threshold are merged into a single larger cluster.

For both HSTU and HLLM backbones, we use a discount factor $\gamma = 0.99$. However, we observed that a smaller γ might lead to higher accuracy at the cost of lower diversity (Table 9).

HLLM Configurations. The choice of LLMs for HLLM depends on the available modalities in the dataset. For Pixel8M, which contains both text and images, we use Qwen2-VL-2B-Instruct as the Item LLM and Qwen2.5-1.5B as the User LLM. For MerRec and EB-NeRD, which only contain text, we use TinyLlama-1.1B-Chat-v1.0 as the Item LLM and TinyLlama_v1.1 as the User LLM.

HSTU Configurations. We experiment with various sizes of the HSTU model, detailed in Table 5. For all datasets, we remap product IDs based on the ones that still have been interacted with after filtering. We report the results based on size 3 unless noted.

Baseline Configurations. For ComiRec and REMI, we use the self-attention version, which demonstrates performance comparable to the dynamic routing version, but with stabler and faster training. We use a learning rate of 0.001, search the number of interest from $\{1, 2, 4, 8, 16, 32\}$, and report the best performance. For REMI, we tune the additional β parameter for interest-aware hard negative mining from $\{0.1, 1, 4, 10\}$ and set routing regularization weight λ to 100 following the original paper.

For DualVAE, we reimplement it under the window-wise sequential recommendation setting. We search the aspect number from $\{4, 5, 10\}$ and set the VAE latent dimension to 32. We search dropout from $\{0.1, 0.15, 0.2\}$, γ (for contrastive loss) and β (for KL-divergence) from $\{1e-3, 1e-2, 1e-1, 1\}$.

A.1 Additional Results

A.1.1 Case Studies. Figure 5 illustrates the qualitative benefits of human priors using the Pixel8M dataset. We analyze the predictions of the HLLM backbone, which leverages both text and image modalities[6], configured with LT/ST$_2$ and Item Prior, compared against the baseline HLLM.

The user history (Rows 1-2) lacks videos in "Informational & Educational", but indicates a latent user interest in military and geopolitics, since they include clips from the WWII movie Downfall, a game on evolution under nuclear waste, and a meme on "atomic egg explosion". The baseline HLLM (Row 4) struggles to synthesize this nuanced intent. Even after filtering to the "Informational & Educational" category, it produces generic recommendations. In contrast, our proposed model demonstrates effective specialization (Row 5). The adapter head dedicated to the *Long-Term* interest within the "Informational & Educational" prior successfully captures the user's latent interest, recommending the revival of the Soviet Union and Russian territorial waters. The better recommendation quality is validated by the target items (Row 3). Starting from the second target item, the user engaged with content on the Soviet Union, which was successfully predicted by our model, and the Second Sino-Japanese War. This shows that dedicating a specialized head to model underrepresented categories effectively discovers the user's hidden interests, which the baseline overlooked.

A.1.2 How to Structure? When multiple priors are available, the composition strategy significantly impacts performance. We compare three strategies: Additive, Multiplicative, and Hierarchical.

[6] Video descriptions are omitted in Figure 5 due to space constraints.

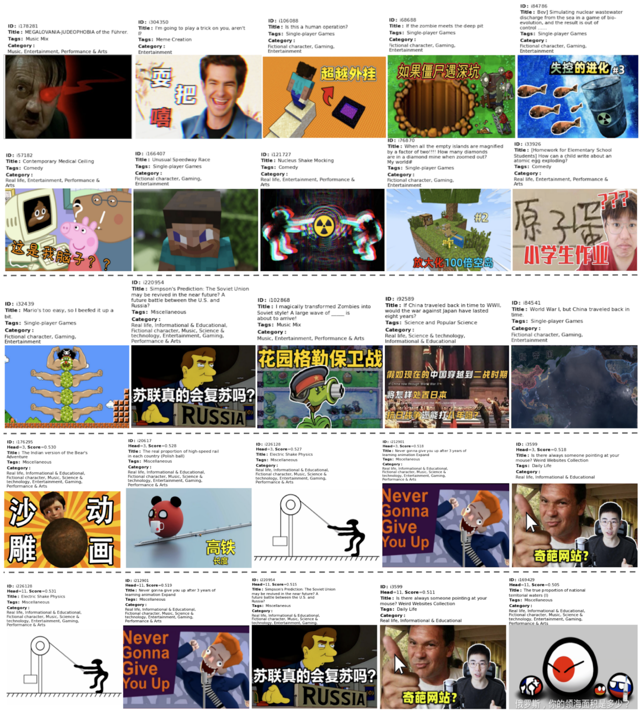
ID: i178281
Title: MEGALOVANIA-JUDEOPHOBIA of the Führer.
Tags: Music Mix
Category: Music, Entertainment, Performance & Arts
ID: i304350
Title: I'm going to play a trick on you, aren't I?
Tags: Meme Creation
Category: Entertainment
耍 把 嘻
ID: i106088
Title: Is this a human operation?
Tags: Single-player Games
Category: Fictional character, Gaming, Entertainment
超越外挂
ID: i68688
Title: If the zombie meets the deep pit
Tags: Single-player Games
Category: Fictional character, Gaming, Entertainment
如果僵尸遇深坑
ID: i84786
Title: Bev] Simulating nuclear wastewater discharge from the sea in a game of bio-evolution, and the result is out of control ......
Tags: Single-player Games
Category: Fictional character, Gaming, Entertainment
失控的进化 #3
ID: i57182
Title: Contemporary Medical Ceiling
Tags: Comedy
Category: Real life, Entertainment, Performance & Arts
这是我脑子？？
ID: i166407
Title: Unusual Speedway Race
Tags: Single-player Games
Category: Fictional character, Gaming, Entertainment
ID: i121727
Title: Nucleus Shake Mocking
Tags: Comedy
Category: Real life, Entertainment, Performance & Arts
ID: i76870
Title: When all the empty islands are magnified by a factor of two!!!! How many diamonds are in a diamond mine when zoomed out? My world#
Tags: Single-player Games
Category: Fictional character, Gaming, Entertainment
#2
放大化100倍空岛
ID: i33926
Title: [Homework for Elementary School Students] How can a child write about an atomic egg exploding?
Tags: Comedy
Category: Real life, Entertainment, Performance & Arts
原子蛋 ???
小学生作业
ID: i32439
Title: Mario's too easy, so I beefed it up a bit.
Tags: Single-player Games
Category: Fictional character, Gaming, Entertainment
ID: i220954
Title: Simpson's Prediction: The Soviet Union may be revived in the near future? A future battle between the U.S. and Russia?
Tags: Miscellaneous
Category: Real life, Informational & Educational, Fictional character, Music, Science & technology, Entertainment, Gaming, Performance & Arts
苏联真的会复苏吗？
RUSSIA
ID: i102868
Title: I magically transformed Zombies into Soviet style! A large wave of _____ is about to arrive!
Tags: Music Mix
Category: Music, Entertainment, Performance & Arts
花园格勒保卫战
ID: i92589
Title: If China traveled back in time to WWII, would the war against Japan have lasted eight years?
Tags: Science and Popular Science
Category: Real life, Science & technology, Informational & Educational
假如现在的中国穿越到二战时期
将怎样处置日本
抗日战争还能打八年吗？
ID: i84541
Title: World War I, but China traveled back in time.
Tags: Single-player Games
Category: Fictional character, Gaming, Entertainment
ID: i176295
Head=3, Score=0.530
Title: The Indian version of the Bear's Adventure
Tags: Miscellaneous
Category: Real life, Informational & Educational, Fictional character, Music, Science & technology, Entertainment, Gaming, Performance & Arts
沙 动 雕 画
ID: i20617
Head=3, Score=0.528
Title: The real proportion of high-speed rail in each country (Polish ball)
Tags: Miscellaneous
Category: Real life, Informational & Educational, Fictional character, Music, Science & technology, Entertainment, Gaming, Performance & Arts
高铁
长度
ID: i226128
Head=3, Score=0.527
Title: Electric Shake Physics
Tags: Miscellaneous
Category: Real life, Informational & Educational, Fictional character, Music, Science & technology, Entertainment, Gaming, Performance & Arts
ID: i212901
Head=3, Score=0.518
Title: Never gonna give you up after 3 years of learning animation Expand
Tags: Miscellaneous
Category: Real life, Informational & Educational, Fictional character, Music, Science & technology, Entertainment, Gaming, Performance & Arts
Never Gonna Give You Up
ID: i3599
Head=3, Score=0.518
Title: Is there always someone pointing at your mouse? Weird Websites Collection
Tags: Daily Life
Category: Real life, Informational & Educational
奇葩网站？
ID: i226128
Head=11, Score=0.531
Title: Electric Shake Physics
Tags: Miscellaneous
Category: Real life, Informational & Educational, Fictional character, Music, Science & technology, Entertainment, Gaming, Performance & Arts
ID: i212901
Head=11, Score=0.519
Title: Never gonna give you up after 3 years of learning animation Expand
Tags: Miscellaneous
Category: Real life, Informational & Educational, Fictional character, Music, Science & technology, Entertainment, Gaming, Performance & Arts
Never Gonna Give You Up
ID: i220954
Head=11, Score=0.515
Title: Simpson's Prediction: The Soviet Union may be revived in the near future? A future battle between the U.S. and Russia?
Tags: Miscellaneous
Category: Real life, Informational & Educational, Fictional character, Music, Science & technology, Entertainment, Gaming,
苏联真的会复苏吗？
RUSSIA
ID: i3599
Head=11, Score=0.511
Title: Is there always someone pointing at your mouse? Weird Websites Collection
Tags: Daily Life
Category: Real life, Informational & Educational
奇葩网站？
ID: i169429
Head=11, Score=0.505
Title: The true proportion of national territorial waters (I)
Tags: Miscellaneous
Category: Real life, Informational & Educational, Fictional character, Music, Science & technology, Entertainment, Gaming, Performance & Arts
俄罗斯，你的领海面积是多少？

Figure 5: A case study on Pixel8M. Rows 1-2: browsing history. Row 3: ground truth target items. Row 4: HLLM baseline predictions (filtered to "Informational & Educational"). Row 5: predictions from our model's long-term "Informational & Educational" head.

Additive Composition learns heads for each prior dimension independently. A head specializes in a single prior while remaining agnostic to others (e.g., a category-specific head optimizes for that category regardless of time horizon). *Multiplicative Composition* defines a distinct head for every element in the Cartesian product of the priors. Each head is derived independently from $\mathbf{h}_T$ using Eq. (2). We evaluate these strategies on Pixel8M using Item and Temporal Priors, applied to the HSTU backbone across five model sizes.

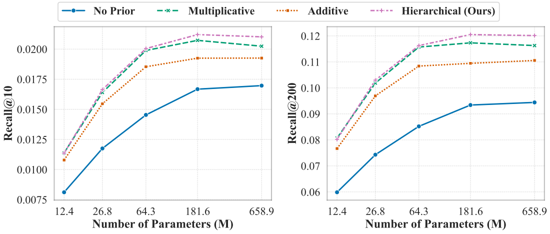

Figure 6: Comparison of composition strategies on Pixel8M across different HSTU model sizes.

Figure 6 shows all three strategies outperform the baseline (no priors). *Hierarchical Composition* consistently achieves the best results across all model sizes. Despite similar parameter complexity to Multiplicative, the Hierarchical structure yields superior performance. Specifically, Multiplicative treats the (A, B) head and the (A, C) head as completely independent entities, failing to leverage their shared "A" context. This causes data fragmentation and poor generalization, especially for rare prior combinations. This highlights the effectiveness of structural regularization and the inductive bias imposed by sequential adapters (Eq. (5)). Similar trends hold for the HLLM backbone (Table 8).

A.1.3 Why it works: priors or just more heads?

We answer two questions using Pixel8M while keeping the backbone and the number of heads fixed. First, do gains stem from human priors or simply more heads? We compare human priors against: (i) **Random** - items are assigned to heads uniformly at random, and (ii) **All** - each item is assigned to all heads.

Second, how to fuse the scores from different heads for each item candidate at inference time? Let $s_h(i)$ be the score of item i from head $h \in \mathcal{H}(i)$ (the heads that are responsible for item i). We compare average and maximum fusion:

$$S_{\text{avg}}(i) = \frac{1}{|\mathcal{H}(i)|} \sum_{h \in \mathcal{H}(i)} s_h(i), \qquad S_{\max}(i) = \max_{h \in \mathcal{H}(i)} s_h(i).$$

Table 6 shows **Random** underperforms the baseline, since simply adding more heads without meaningful partitions only decreases the amount of data to train each head. **All** improves modestly over LT/ST$_1$, confirming the benefits of simply having more heads, yet it trails human priors (LT/ST$_8$ or Item).

Regarding fusion, average fusion might seem more natural, as commonly done in an ensemble, whereas maximum fusion might lead to overly optimistic estimates, similar to the value overestimation issue in off-policy reinforcement learning [17]. In fact, average fusion on the random/all prior variant, or only Temporal Prior, performs better than maximum fusion. However, maximum fusion is better for Item Prior, in terms of both NDCG/recall and diversity. The reason might be each head evaluates whether the item candidate is a good fit under different criteria (e.g., item category or action type), similar to multi-interest networks [4, 18], while heads in random/all or Temporal Prior evaluate on more similar criteria. Future work may consider aggregating the scores hierarchically according to the nature of the prior type, and using inverse variance weighting to upweight more certain heads.

A.1.4 Long-Term vs. Short-Term Interests.

We study the generalization of LT/ST interests by training with $\tau = 4$ and evaluating on $\tau \in \{1, 4, 8\}$ (Table 7). For brevity, we report average gains over the baseline (next target prediction) across the eight metrics; full results are in Table 8. All variants surpass the baseline at $\tau = 4$, with benefits growing when we evaluate at $\tau = 8$. At $\tau = 1$, we only evaluate on the next one item as the target item, which is the same training objective as the baseline and only focuses on the short-term interest. Even under such a setting, modeling long-term interest causes only a minor performance drop, which diminishes with more segments and disappears once we add Item Prior. Finally, adding Item Prior consistently boosts performance over LT/ST alone, suggesting different human priors can be complementary.

Next, we show that the benefit of LT/ST interests is consistent across different model sizes of HSTU (Figure 7). Here, we train on $\tau = 8$ using only the LT/ST prior. The baseline is only trained to

Table 8: HLLM results on Pixel8M with evaluation horizon $\tau = 8$. With multiple human priors, enforcing a *hierarchical* composition (*Hier*) yields the best Recall (R@K) and NDCG (N@K) while maintaining diversity (H@K). A *multiplicative* composition (*Mult*) maximizes H but underperforms on Recall/NDCG as it lacks explicit structure.

HLLM Variants	R@10	N@10	H@10
No Prior	1.358	1.422	2.124
LT/ST$_1$	1.464	1.523	2.126
LT/ST$_1$ + Item	1.479	1.543	2.193
LT/ST$_2$	1.442	1.496	2.121
LT/ST$_2$ + Item (Mult)	1.441	1.496	**2.223**
LT/ST$_2$ + Item (Add)	1.467	1.524	2.193
LT/ST$_2$ + Item (Hier)	**1.499**	**1.559**	2.208

Table 9: Ablation studies on the training objective. We report Recall (R@10), NDCG (N@10), and Entropy (H@10). The best results within each section are highlighted in bold.

Ablation	R@10	N@10	H@10
Full Model ($\gamma = 0.99$)	**2.000**	**2.099**	2.3728
w/o in-group negative sampling	1.642	1.700	2.6395
w/o frequency balancing	1.928	2.005	2.4234
Discount Factor γ			
$\gamma = 1.0$	2.000	2.084	2.3773
$\gamma = 0.95$	2.033	2.113	2.3768
$\gamma = 0.9$	2.045	2.130	2.3727
$\gamma = 0.8$	2.062	2.143	2.3768
$\gamma = 0.7$	**2.065**	**2.155**	2.3773
$\gamma = 0.6$	2.025	2.110	2.3810
$\gamma = 0.5$	2.017	2.098	2.3818

Table 6: Head assignment vs. fusion on Pixel8M. Baseline here is LT/ST$_1$.

		Recall (%)		NDCG (%)		Entropy (H)	
Prior	Fusion	@10	@50	@10	@50	@10	@50
Baseline	—	1.655	4.534	1.726	2.977	2.303	2.461
Random	Max	1.586	4.377	1.648	2.861	**2.308**	2.462
	Avg	1.602	4.417	1.670	2.893	2.307	**2.463**
All	Max	1.671	4.559	1.741	2.996	2.296	2.457
	Avg	**1.672**	**4.560**	**1.743**	**2.998**	2.301	2.460
LT/ST$_8$	Max	1.878	5.251	1.937	3.403	2.296	2.455
	Avg	**1.950**	**5.338**	**2.028**	**3.501**	**2.303**	**2.457**
Item	Max	**1.749**	**4.695**	**1.834**	**3.115**	**2.371**	**2.515**
	Avg	1.738	4.663	1.825	3.097	2.335	2.489

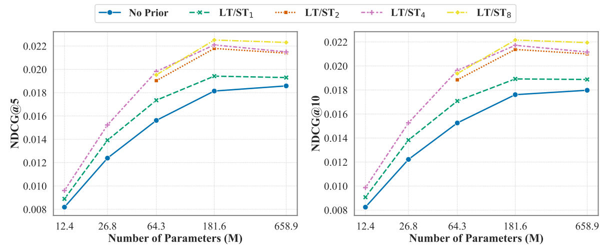

Figure 7: Sensitivity to the number of segments in Temporal Prior across different HSTU sizes.

Variant	$\tau = 1$	$\tau = 4$	$\tau = 8$
LT/ST$_1$	-3.07%	6.95%	7.72%
LT/ST$_1$ + Item	-1.23%	7.95%	8.34%
LT/ST$_2$	-2.68%	5.74%	6.71%
LT/ST$_2$ + Item	0.25%	9.72%	10.36%
LT/ST$_4$ + Item	0.04%	8.58%	9.07%

Table 7: Average gain over the baseline HLLM (w/o prior) across eight metrics. We train the model on $\tau = 4$, and show its robustness when evaluated on $\tau = 1, 4, 8$.

predict the next one item, while LT/ST$_1$ means that we only model one interest, and the objective reduces to the simple discounted loss over the next τ items. We see that explicitly assigning more different heads to model the long-term vs. short-term interests brings more improvements as we go from the baseline to seg = 2, and then the benefit from further increasing the number of segments slowly diminishes. We observe a slight dip from `size4` to `1b`, and our hypothesis is that the number of target items used to train each head decreases when the number of interests increases. Nevertheless, the benefit of separately modeling the long-term and short-term interests is still significant at the scale of `1b`, where the performance gain of the baseline model from increasing the model size has already flattened out at this scale.

A.1.5 Ablations on Training Objective. We ablate three key components of our training objective (Section 3.4): 1) in-group negative sampling, 2) frequency balancing, and 3) the temporal discount factor γ. Table 9 shows the results.

First, removing in-group negative sampling (i.e., sampling from all prior groups) and frequency balancing sharply reduces Recall and NDCG. Specifically, removing in-group sampling causes R@10 to drop significantly from 2.000 to 1.642. This validates the importance of these mechanisms for learning effective representations and handling data imbalance.

The impact of the temporal discount factor γ is more nuanced. Both Recall and NDCG peak at $\gamma = 0.7$. Interestingly, the entropy H initially decreases as γ decreases from 1.0 to 0.9, while Recall/NDCG increases. However, beyond $\gamma = 0.9$, the entropy H instead increases alongside Recall/NDCG until they reach their peak at $\gamma = 0.7$. This observation is intriguing because it suggests that, even for the same architecture, higher accuracy (Recall/NDCG) does not necessarily imply lower diversity (Entropy). Furthermore, the optimal $\gamma = 0.7$ is notably different from the values typically used in reinforcement learning, which range from 0.9 to 0.995.