Asymmetric Generative Recommendation via Multi-Expert Projection and Multi-Faceted Hierarchical Quantization

Bin Huang
DCST, Tsinghua University
Beijing, China
huangb23@mails.tsinghua.edu.cn

Xin Wang*
DCST, BNRist, Tsinghua University
Beijing, China
xin_wang@tsinghua.edu.cn

Junwei Pan
Tencent
Shenzhen, China
jonaspan@tencent.com

Yongqi Zhou
Tencent
Shenzhen, China
kolinzhou@tencent.com

Yifeng Zhou
Tencent
Shenzhen, China
joefzhou@tencent.com

Zhixiang Feng
Tencent
Shenzhen, China
lionelfeng@tencent.com

Shudong Huang
Tencent
Shenzhen, China
ericdhuang@tencent.com

Haijie Gu
Tencent
Shenzhen, China
jerrickgu@tencent.com

Wenwu Zhu*
DCST, BNRist, Tsinghua University
Beijing, China
wwzhu@tsinghua.edu.cn

Abstract

Generative Recommendation (GenRec) models reformulate recommendation as a sequence generation task, representing items as discrete Semantic IDs used symmetrically as both inputs and prediction targets. We identify a critical dual-stage information bottleneck in this design: (1) the Input Bottleneck, where lossy quantization degrades fine-grained semantics, while popularity bias skews the learned representations toward frequent items, and (2) the Output Bottleneck, where imprecise discrete targets limit supervision quality. To address these issues, we propose AsymRec, an asymmetric continuous-discrete framework that decouples input and output representations. Specifically, Multi-expert Semantic Projection (MSP) maps continuous embeddings into the Transformer's hidden space via expert-specialized projections, preserving semantic richness and improving generalization to infrequent items. Multi-faceted Hierarchical Quantization (MHQ) constructs high-capacity, structured discrete targets through multi-view and multi-level quantization with semantic regularization, preventing dimensional collapse while retaining fine-grained distinctions. Extensive experiments demonstrate that AsymRec consistently outperforms state-of-the-art generative recommenders by an average of 15.8%. The code will be released.

Keywords

Generative Recommendation, Semantic IDs, Vector Quantization

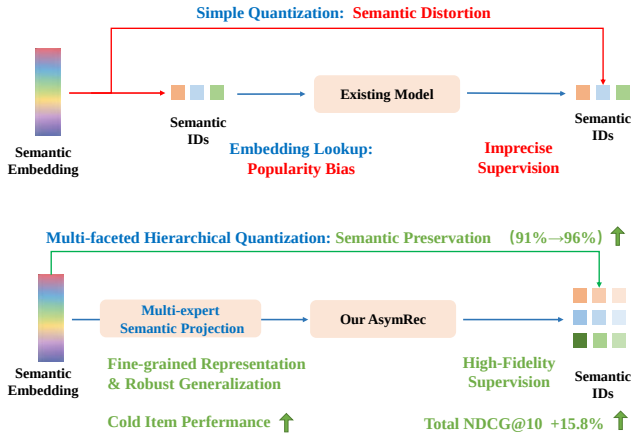

Figure 1: Existing generative recommenders rely on symmetric quantization of item embeddings, causing semantic distortion and popularity bias at the input, and imprecise supervision at the output. Our method decouples input and output representations via multi-expert semantic projection and multi-faceted hierarchical quantization, enabling high-fidelity generative recommendation.

1 Introduction

Recent advancements have given rise to Generative Recommendation (GenRec) models, which reformulate recommendation as a sequence-to-sequence generation task [3, 29, 31]. Drawing inspiration from the success of large language models [5, 28], these approaches represent items as sequences of discrete tokens—commonly referred to as *Semantic IDs* [7, 8, 20, 21]. This paradigm enables an auto-regressive Transformer to predict the next item in a unified generative manner [8, 21, 29], offering the potential for better capture of long-range dependencies and seamless integration of multi-modal item features [1, 7].

The efficacy of GenRec models hinges on the bridge between continuous item semantics and discrete generative tokens. Prevailing methods typically adopt a fully-discretized pipeline: they first quantize high-dimensional semantic embeddings (derived from text or visual encoders) into semantic IDs via RQ-VAE / vector-quantized autoencoding [15, 21, 26], and then use these discrete IDs as both the input representation and the prediction target [7, 8, 21]. This

*Corresponding authors. DCST is the abbreviation of Department of Computer Science and Technology. BNRist is the abbreviation of Beijing National Research Center for Information Science and Technology.

symmetric reliance on a single, lossy quantization mapping underpins many recent GenRec systems, spanning retrieval and ranking deployments [1, 3, 10, 21, 31].

The Input Bottleneck: Semantic Distortion and Popularity Bias. Traditional generative recommenders map discrete IDs into a learned embedding space via a lookup table before feeding them into the Transformer. This process introduces two key limitations. First, the initial quantization is inherently lossy, discarding fine-grained semantic nuances that cannot be recovered. Second, the learned ID embeddings tend to overemphasize frequently occurring "hot" items in the training set, limiting the model's ability to generalize to less frequent "cold" items.

The Output Bottleneck: Imprecise Supervision Signals. On the generation side, the model is trained to predict tokens generated by simplistic quantization. These methods often suffer from high reconstruction errors and "codebook collisions," where distinct items are represented by identical or highly similar ID sequences. This provides a noisy and imprecise learning target, limiting the model's capacity to learn accurate representations. While one might consider predicting continuous embeddings directly to avoid this, such a transition often leads to **"dimensional collapse,"** [9] where the model's output distribution shrinks to a narrow subspace, failing to distinguish between items with high precision. Thus, a high-capacity discrete target remains essential for effective supervision.

To address these challenges, we propose **AsymRec**, a high-fidelity generative framework that decouples the input and output representations to maximize semantic preservation and improve generalization. Specifically, we introduce **Multi-expert Semantic Projection (MSP)** for the input stage, which maps original continuous embeddings directly into the Transformer's hidden space via a lightweight Mixture-of-Experts mechanism, bypassing discrete ID lookup entirely. This continuous mapping preserves the original semantic embedding space, enabling better generalization to less frequent "cold" items, while different experts specialize in distinct semantic facets, enhancing prediction accuracy across all items. For the output stage, we propose **Multi-faceted Hierarchical Quantization (MHQ)**. MHQ first applies a learnable projection to reorganize embeddings into a structured latent space, explicitly regularized to balance semantic energy across subspaces and to reduce redundancy and correlation among them. Built upon this representation, MHQ performs hierarchical, multi-path quantization with an Exponential Moving Average (EMA) strategy to stabilize the discrete optimization process. This results in a multi-dimensional, multi-layer coordinate system that yields high-capacity discrete targets, effectively preventing dimensional collapse while retaining fine-grained semantic distinctions.

Our key contributions are summarized as follows:

- We identify and analyze the **dual-stage information bottleneck** in generative recommendation, highlighting how discrete inputs bias the model toward frequently occurring "hot" items and how standard quantized outputs limit prediction precision.
- We propose **Multi-expert Semantic Projection (MSP)**, which replaces traditional ID lookup with a continuous, expert-specialized projection, preserving fine-grained semantic topology to enhance generalization to less frequent "cold" items and improve prediction accuracy.
- We develop **Multi-faceted Hierarchical Quantization (MHQ)**, a structured discretization framework that integrates learnable projection, structural regularization, and EMA-stabilized hierarchical quantization to provide high-fidelity discrete supervision while maintaining semantic and hierarchical consistency.
- Extensive experiments on the Amazon public benchmarks demonstrate that **AsymRec** consistently outperforms state-of-the-art generative recommenders by an average of **15.8%**. Beyond offline evaluation, we further deploy **AsymRec** in a production pCVR system on one of the world's largest advertising platforms, where it achieves a 1.4% lift in total consumption and a 1.9% GMV uplift in online A/B tests, validating its effectiveness at industrial scale.

2 Related Works

2.1 Generative Recommendation

Sequential recommendation (SR) aims to capture the dynamic evolution of user preferences by modeling historical interaction sequences. Traditional discriminative approaches, from early Markov Chains [22] to modern Transformer-based models like SASRec [14] and BERT4Rec [24], primarily frame recommendation as a ranking task. They operate by scoring items from a fixed corpus, treating item IDs as independent, atomic tokens. This paradigm faces intrinsic challenges: it struggles with cold-start scenarios [27, 33] due to the sparsity and lack of inherent meaning in random IDs, and it fails to explicitly leverage the rich semantic correlations between items, limiting generalization.

To overcome these limitations, Generative Recommendation (GR) [16, 21] has recently emerged as a promising alternative paradigm. Instead of relying on learned item ID embeddings, GR models first encode item-side semantic content—such as titles, descriptions, or other multi-modal attributes—into continuous semantic embeddings. These embeddings are then discretized into a set of semantic IDs that capture high-level semantic attributes of items. Given a user's interaction history, the GR model takes the semantic IDs of previously interacted items as input and autoregressively generates the semantic IDs corresponding to the target item for recommendation.

2.2 Semantic ID Generation

A key component of Generative Recommendation is the discretization of continuous semantic embeddings into semantic IDs (SIDs), which enables item representation and generation in a discrete token space. Existing methods for SID generation can be broadly categorized into residual quantization (RQ)–based methods and product quantization (PQ)–based methods, each offering distinct advantages and limitations.

Residual quantization [13, 18] is the most widely adopted approach for semantic ID generation. By iteratively quantizing the residual between the original embedding and previously selected codewords, RQ constructs a hierarchical, coarse-to-fine representation. This multi-level structure aligns naturally with autoregressive

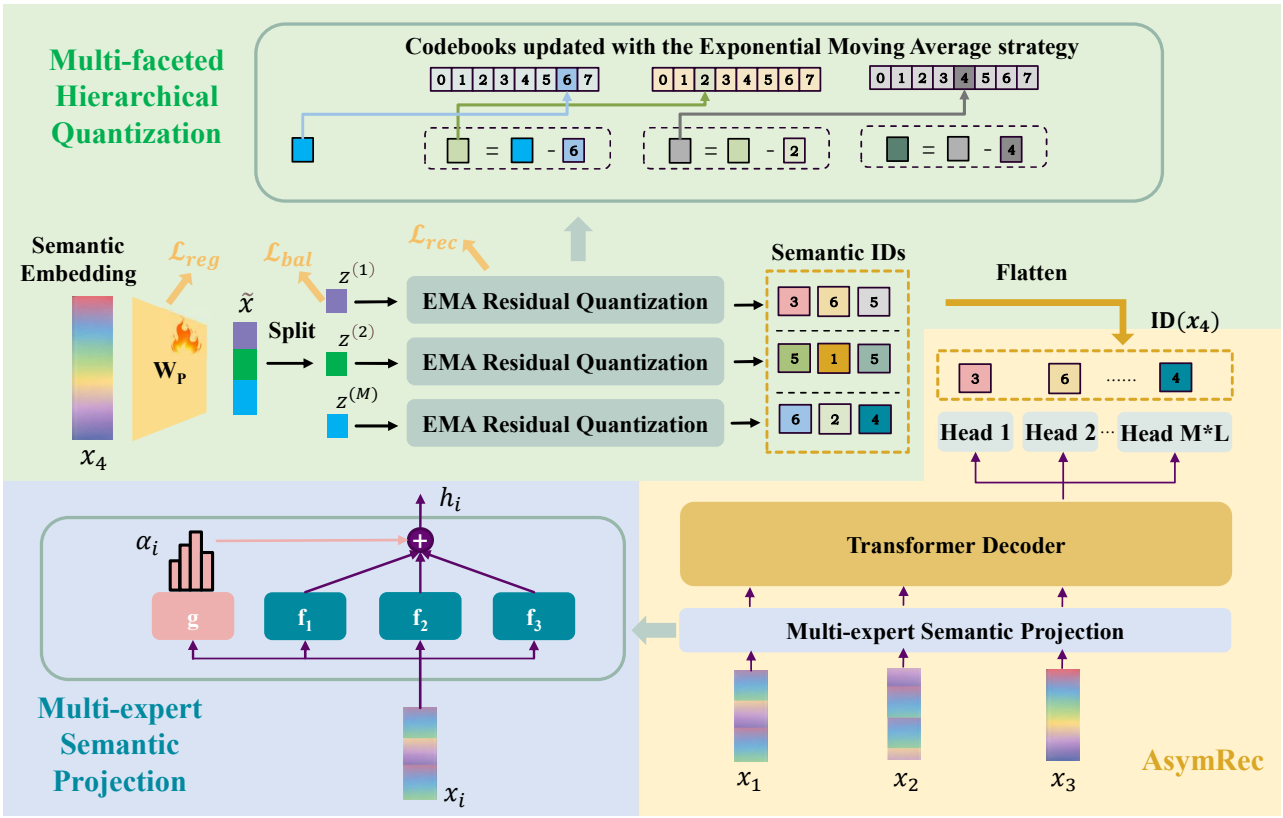

Figure 2: Overview of the proposed AsymRec framework. The input item is first encoded into a continuous semantic embedding, which is mapped by the Multi-expert Semantic Projection (MSP) module to produce fine-grained representations. These representations are then fed into the Transformer Decoder to predict the corresponding semantic IDs generated by the Multi-faceted Hierarchical Quantization (MHQ) module. MHQ maps each embedding into multiple subspaces and applies EMA Residual Quantization in each subspace, with codebooks updated via the Exponential Moving Average strategy, producing fine-grained discrete semantic IDs. This design enables the model to capture subtle differences between items while maintaining robust generalization across both popular and cold items.

generation, as it progressively narrows the candidate search space and captures semantic information at different levels of granularity. RQ-based methods have been extensively applied in generative recommender systems [3, 21]. However, a significant drawback of RQ is its tendency toward *semantic entanglement*. Because all residual levels are optimized along a single path, RQ often struggles to disentangle independent semantic facets (e.g., brand, category, and style), potentially conflating distinct item attributes into a coupled ID sequence.

Product quantization [4] decomposes the embedding space into multiple independent subspaces and quantizes each separately, enabling fine-grained modeling of different semantic aspects [8]. While PQ effectively captures multi-faceted information, it lacks the *hierarchical depth* inherent in RQ.

3 Method

In this section, we introduce our model, AsymRec, as illustrated in Fig. 2. We first formally define the generative recommendation problem and describe the representation of items in both continuous and discrete spaces. Then, we present the two core components of our framework: Multi-expert Semantic Projection (MSP), which preserves rich semantic information by mapping continuous embeddings into the model's feature space, and Multi-faceted Hierarchical Quantization (MHQ), which produces high-fidelity, structured discrete targets for supervision. Finally, we describe the overall architecture of AsymRec, which integrates MSP and MHQ within a Transformer-based generative recommendation model to effectively capture fine-grained user preferences and generate coherent recommendations.

3.1 Problem Definition

Sequential recommendation aims to model user preferences based on historical interactions and predict the next item a user is likely to interact with. Formally, given a user interaction sequence

$$\mathcal{S}_u = [I_1, I_2, \ldots, I_T],$$

where I_i denotes the i-th item, the task is to predict the next item I_{T+1}.

In the generative recommendation setting, each item is represented by a continuous semantic embedding $x_i \in \mathbb{R}^d$. To enable generation in a discrete space, each embedding is quantized into a set of semantic IDs, denoted as $\mathbf{ID}(x_i)$. The task objective is then to predict the semantic IDs of the next item, denoted as $\mathbf{ID}(x_{T+1})$, based on the preceding items in the sequence.

3.2 Multi-expert Semantic Projection (MSP)

For a user interaction sequence $\mathcal{S}_u$, each item is represented by a continuous semantic embedding x_i as well as multiple quantized tokens $\mathbf{ID}(x_i)$. In prior generative recommendation methods, only the discrete token embeddings are used as input: the tokens are typically mapped to embeddings via a lookup table and then either concatenated along the sequence dimension [21] or averaged [8] to form the input to the recommendation model, while the original continuous embedding x_i is never utilized.

We identify two limitations of this approach: (1) *quantization is inherently lossy*, discarding fine-grained semantic nuances that cannot be recovered, which prevents the model from distinguishing cold or subtly different items; (2) *learning bias toward popular items*, as embeddings of frequent IDs are updated far more often during training, while rare items receive insufficient supervision and remain under-trained.

To address these issues, we propose the **Multi-expert Semantic Projection (MSP)** module, which directly maps the original item embedding x_i into the recommendation model's feature space via learnable MLPs. Unlike discrete ID lookup, this continuous projection avoids lossy quantization and allows semantically similar items to be mapped to nearby representations. As a result, the topological structure of the item space is largely preserved, enabling effective generalization from frequent to infrequent items and supporting robust modeling of both hot and cold items.

Furthermore, MSP employs a Mixture-of-Experts [11, 12] mechanism, allowing different experts to specialize in capturing distinct aspects of item features. This expertized decomposition facilitates the representation of fine-grained semantic information, enhancing the model's ability to generate precise and personalized recommendations.

Specifically, given an input item embedding $x_i \in \mathbb{R}^d$, MSP maps it into the recommendation model's feature space:

$$h_i = \mathrm{MSP}(x_i) \in \mathbb{R}^{d_m}. \tag{1}$$

Concretely, the MSP consists of E expert functions $\{f_e(\cdot)\}_{e=1}^{E}$ and a gating network $g(\cdot)$ that assigns dynamic weights to each expert. The mapping can be written as

$$h_i = \sum_{e=1}^{E} \alpha_{i,e}\, f_e(x_i), \quad \text{with} \quad \boldsymbol{\alpha}_i = g(x_i), \quad \sum_{e=1}^{E} \alpha_{i,e} = 1,\ \alpha_{i,e} \geq 0.$$

Here, $f_e(\cdot)$ is implemented as a 2-layer MLP that captures a specific aspect of the item's semantic representation. The gating function $g(x_i)$ dynamically determines the contribution of each expert for the given item, allowing different experts to specialize and capture complementary facets of the input. Through this mechanism, the resulting representation h_i preserves the topological structure of the original embedding x_i while enriching it with fine-grained, expert-specific features. This representation serves as the input to downstream generative recommendation modules.

3.3 Multi-faceted Hierarchical Quantization (MHQ)

By mapping input embeddings into a continuous feature space via MSP, we avoid input-side quantization loss. While predicting continuous vectors at the output could seem natural, this often causes dimension collapse (Sec. 4.3.2), resulting in suboptimal performance. Instead, predicting discrete semantic identifiers preserves the structure and improves generative recommendation quality.

Nevertheless, as pointed out in Sec. 2.2, existing quantization methods are insufficient to capture both multi-faceted and hierarchical semantic information simultaneously. To address this, we propose the **Multi-faceted Hierarchical Quantization (MHQ)** module. MHQ combines the strengths of residual quantization (RQ) and product quantization (PQ) by first partitioning the embedding into multiple orthogonal subspaces, and then applying hierarchical residual quantization within each subspace. This design ensures that the generated semantic IDs are both multi-dimensional in semantic coverage and progressively detailed within each dimension.

Given a semantic embedding vector $x \in \mathbb{R}^d$, the MHQ module first projects it into a latent space $\mathbb{R}^D$ via a learnable linear transformation $\tilde{x} = W_P x$, where $W_P \in \mathbb{R}^{D\times d}$. To extract diverse semantic facets, the projected vector $\tilde{x}$ is partitioned into M disjoint subspaces:

$$\tilde{x} = [z^{(1)}, z^{(2)}, \ldots, z^{(M)}], \quad z^{(m)} \in \mathbb{R}^{d_m} \tag{2}$$

where $d_m = D/M$ denotes the dimensionality of each subspace.

Within each subspace m, we implement a Residual Quantization process of depth L. For each level $l \in \{1, \ldots, L\}$, a codebook $C^{(m,l)} = \{c_k^{(m,l)}\}_{k=1}^{K}$ is maintained, where K is the codebook size. The quantizer iteratively identifies the optimal centroid index $i_{m,l}$ by minimizing the L_2 distance between the current residual $r_l^{(m)}$ and the codebook entries:

$$i_{m,l} = \arg\min_{k\in\{1,\ldots,K\}} \|r_l^{(m)} - c_k^{(m,l)}\|_2^2 \tag{3}$$

where $r_1^{(m)} = z^{(m)}$, and the residual for the subsequent level is updated as $r_{l+1}^{(m)} = r_l^{(m)} - c_{i_{m,l}}^{(m,l)}$. The reconstructed representation in the m-th subspace is thus $\hat{z}^{(m)} = \sum_{l=1}^{L} c_{i_{m,l}}^{(m,l)}$.

To stabilize the discrete optimization, we employ the Exponential Moving Average (EMA) strategy for codebook updates instead of standard backpropagation. For a given centroid $c_k^{(m,l)}$, the update rules are:

$$N_k^{(m,l)} \leftarrow \gamma N_k^{(m,l)} + (1-\gamma) \sum_{j=1}^{B} \mathbb{1}[i_{m,l}^{(j)} = k] \tag{4}$$

$$m_k^{(m,l)} \leftarrow \gamma m_k^{(m,l)} + (1-\gamma) \sum_{j=1}^{B} \mathbb{1}[i_{m,l}^{(j)} = k] r_l^{(m,j)} \tag{5}$$

$$c_k^{(m,l)} = \frac{m_k^{(m,l)}}{N_k^{(m,l)}} \tag{6}$$

where γ is the decay factor and B is the batch size. Finally, each item is represented by a flattened sequence of indices $\mathbf{ID}(x) = \{i_{1,1}, i_{1,2}, i_{1,3}, \ldots, i_{M,L}\}$, forming a structured semantic codeword of length $M \times L$.

The training objective of MHQ is formulated as a multi-task loss function. The primary component is the reconstruction loss:

$$\mathcal{L}_{rec} = \|\tilde{x} - \text{concat}(\hat{z}^{(1)}, \ldots, \hat{z}^{(M)})\|_2^2, \tag{7}$$

which ensures high fidelity of the quantized IDs. To prevent information from collapsing into a subset of subspaces, we introduce a subspace energy balance loss that explicitly penalizes uneven energy allocation across different facets. Specifically, let $\mathbb{E}[\|z^{(m)}\|_2^2]$ denote the expected energy of the m-th subspace. We first compute the mean energy across all M subspaces:

$$\bar{E} = \frac{1}{M} \sum_{m=1}^{M} \mathbb{E}\big[\|z^{(m)}\|_2^2\big], \tag{8}$$

and define the balance loss as the mean absolute deviation from this average:

$$\mathcal{L}_{bal} = \frac{1}{M} \sum_{m=1}^{M} \Big| \mathbb{E}\big[\|z^{(m)}\|_2^2\big] - \bar{E} \Big|. \tag{9}$$

This formulation encourages an equitable distribution of information across all M facets.

In addition, to reduce redundancy and correlation among different subspaces, we impose an orthogonality regularization on the projection matrix W_P, defined as

$$\mathcal{L}_{reg} = \left\| W_P W_P^\top - I \right\|_F, \tag{10}$$

where I denotes the identity matrix.

The overall training objective is given by

$$\mathcal{L}_{MHQ} = \mathcal{L}_{rec} + \lambda_{bal}\mathcal{L}_{bal} + \lambda_{reg}\mathcal{L}_{reg}. \tag{11}$$

This loss is applied only during the training of MHQ and is not used in the subsequent training of the recommendation model. After training, the discrete tokens $\mathbf{ID}(x_i)$ is assigned to each x_i based on the learned quantization.

3.4 AsymRec Architecture

In this section, we present the overall architecture of AsymRec, as illustrated in Fig. 2. The framework adopts an asymmetric continuous-discrete design: continuous embeddings are mapped into the model's feature space via MSP, while high-fidelity, multi-faceted discrete targets are produced by MHQ for supervision. These two complementary components are integrated within a Transformer-based generative model.

Given a sequence of item embeddings corresponding to a user's interactions $[x_1, x_2, \ldots, x_T]$ each item embedding x_i is first mapped into the recommendation feature space via the Multi-expert Semantic Projection (MSP) module as $h_i = \text{MSP}(x_i)$.

Positional encodings are then added:

$$\mathbf{H}^0 = [h_1 + p_1, h_2 + p_2, \ldots, h_T + p_T]. \tag{12}$$

The resulting sequence $\mathbf{H}^0$ is then fed into L_T Transformer decoder layers. Each layer utilizes multi-head self-attention and feed-forward networks to model the complex transitions between user interests:

Table 1: Statistics of the processed datasets. "Avg. t" denotes the average number of interactions per input sequence.

Datasets	#Users	#Items	#Interactions	Avg. t
Sports	18,357	35,598	260,739	8.32
Beauty	22,363	12,101	176,139	8.87
Toys	19,412	11,924	148,185	8.63
CDs	75,258	64,443	1,022,334	14.58

$$H^i = \text{Decoder}(H^{i-1}), \quad i = 1, \ldots, L_T, \tag{13}$$

We extract the hidden state of the last item from the final decoder layer, denoted as $\mathbf{H}_T^{L_T} \in \mathbb{R}^{d_m}$, and feed it into $M \times L$ parallel prediction heads to predict the structured semantic IDs of the next item, $\mathbf{ID}(x_{T+1})$. Each prediction head is implemented as a two-layer MLP that maps $\mathbf{H}_T^{L_T}$ to a K-way categorical distribution over the corresponding quantized codebook entries. Formally, we optimize the cross-entropy loss over all heads:

$$\mathcal{L}_{\text{CE}} = -\frac{1}{ML} \sum_{m=1}^{M} \sum_{l=1}^{L} \log p\big(i_{m,l}^{T+1} \,\big|\, \text{model}(x_{\le T})\big), \tag{14}$$

which encourages accurate semantic ID predictions across all facets and hierarchical layers. At inference, we employ a graph-constrained decoding strategy [8] to ensure that only valid codewords are generated.

This asymmetric design naturally integrates the continuous input mapping from MSP with the multi-faceted, hierarchical quantization of MHQ, allowing AsymRec to capture fine-grained item semantics while producing coherent and structured recommendations.

4 Experiment

To evaluate the effectiveness of AsymRec and validate our hypotheses regarding the dual-stage bottleneck, we aim to answer the following research questions:

- **RQ1: Overall Performance.** How does AsymRec perform compared to state-of-the-art generative and sequential recommendation baselines across various benchmarks?
- **RQ2: Impact of Continuous Input on Optimization and Generalization.** Does the Multi-expert Semantic Projection truly alleviate the input bottleneck? Specifically, how does it affect the model's ability to maintain semantic topology and generalize to cold items compared to discrete ID inputs ?
- **RQ3: Necessity of Discrete Output vs. Continuous Output.** Why not adopt a fully continuous pipeline? What are the empirical consequences when using continuous embeddings as the generation target ?
- **RQ4: Effectiveness of Multi-faceted Hierarchical Quantization (MHQ).** Does MHQ provide a higher-fidelity supervision signal than traditional quantization methods? How does the subspace decomposition affect recommendation precision?

Table 2: Performance comparison among baselines and AsymRec. The best performance score is denoted in bold. The second-best performance score is denoted in underline.

Model	Sports and Outdoors				Beauty				Toys and Games				CDs and Vinyl			
	R@5	N@5	R@10	N@10	R@5	N@5	R@10	N@10	R@5	N@5	R@10	N@10	R@5	N@5	R@10	N@10
Item ID-based																
Caser [25]	0.0116	0.0072	0.0194	0.0097	0.0205	0.0131	0.0347	0.0176	0.0166	0.0107	0.0270	0.0141	0.0116	0.0073	0.0205	0.0101
GRU4Rec [6]	0.0129	0.0086	0.0204	0.0110	0.0164	0.0099	0.0283	0.0137	0.0097	0.0059	0.0176	0.0084	0.0195	0.0120	0.0353	0.0171
HGN [17]	0.0189	0.0120	0.0313	0.0159	0.0325	0.0206	0.0512	0.0266	0.0321	0.0221	0.0497	0.0277	0.0259	0.0153	0.0467	0.0220
BERT4Rec[24]	0.0115	0.0075	0.0191	0.0099	0.0203	0.0124	0.0347	0.0170	0.0116	0.0071	0.0203	0.0099	0.0326	0.0201	0.0547	0.0271
SASRec [14]	0.0233	0.0154	0.0350	0.0192	0.0387	0.0249	0.0605	0.0318	0.0463	0.0306	0.0675	0.0374	0.0351	0.0177	0.0619	0.0263
FDSA [30]	0.0182	0.0122	0.0288	0.0156	0.0267	0.0163	0.0407	0.0208	0.0228	0.0140	0.0381	0.0189	0.0226	0.0137	0.0378	0.0186
S^3-Rec [32]	0.0251	0.0161	0.0385	0.0204	0.0387	0.0244	0.0647	0.0327	0.0443	0.0294	0.0700	0.0376	0.0213	0.0130	0.0375	0.0182
Semantic ID-based																
RecJPQ [20]	0.0141	0.0076	0.0220	0.0102	0.0311	0.0167	0.0482	0.0222	0.0331	0.0182	0.0484	0.0231	0.0075	0.0046	0.0138	0.0066
VQ-Rec [7]	0.0208	0.0144	0.0300	0.0173	0.0457	0.0317	0.0664	0.0383	0.0497	0.0346	0.0737	0.0423	0.0352	0.0238	0.0520	0.0292
TIGER [21]	0.0264	0.0181	0.0400	0.0225	0.0454	0.0321	0.0648	0.0384	0.0521	0.0371	0.0712	0.0432	0.0492	0.0329	0.0748	0.0411
HSTU [29]	0.0258	0.0165	0.0414	0.0215	0.0469	0.0314	0.0704	0.0389	0.0433	0.0281	0.0669	0.0357	0.0417	0.0275	0.0638	0.0346
RPG [8]	0.0314	0.0216	0.0463	0.0263	0.0550	0.0381	0.0809	0.0464	0.0592	0.0401	0.0869	0.0490	0.0498	0.0338	0.0735	0.0415
AsymRec	**0.0371**	**0.0250**	**0.0550**	**0.0308**	**0.0618**	**0.0424**	**0.0901**	**0.0516**	**0.0658**	**0.0450**	**0.0971**	**0.0551**	**0.0614**	**0.0415**	**0.0902**	**0.0508**

4.1 Experimental Setup

Dataset. We conduct experiments on four widely-used categories from the Amazon Review benchmark [19]: **Sports and Outdoors** (**Sports**), **Beauty**, **Toys and Games** (**Toys**), and **CDs and Vinyl** (**CDs**). Following previous studies [8, 14, 21, 32], we treat user reviews as interactions and organize them chronologically to construct interaction sequences. We follow the standard "5-core" filtering, ensuring each user and item has at least five interactions. The detailed statistics of the datasets are summarized in Table 1.

Evaluation Protocol. We adopt the widely used leave-last-out evaluation protocol, reserving the last item in each sequence for testing, the second-to-last item for validation, and the remaining prefix for training. To measure recommendation performance, we employ two widely-adopted ranking metrics: Recall@K and Normalized Discounted Cumulative Gain (NDCG@K), with $K \in \{5, 10\}$. We report the test performance corresponding to the best results on the validation set.

Implementation details. We use OpenAI's text-embedding-3-large as the semantic encoder following Hou et al. [8], with output dimension $d = 3072$. When training MHQ, we set the quantized embedding dimension $D = 512$, the loss weight $\lambda_{bal} = 0.01$, $\lambda_{reg} = 0.01$, the decay factor $\gamma = 0.99$, the learning rate to 0.001, and train for 50 epochs. For training AsymRec, we set the number of experts $E = 3$, and employ a Transformer decoder with $L_T = 2$ layers and an embedding dimension $d_m = 448$. We train for a maximum of 100 epochs with a batch size of 256 and a learning rate of 0.003. An early stopping strategy is adopted to halt training if validation performance does not improve for 20 consecutive epochs. We tune the number of codebooks $M \in \{8, 16, 32\}$, the number of layers $L \in \{2, 3\}$, and the codebook size $K \in \{256, 512, 1024\}$. On the Beauty dataset, among the 12,101 items, 12,099 have unique codes; therefore, no additional collision handling is applied. Training and evaluation on the Beauty dataset complete within one hour using an NVIDIA GeForce RTX 3090 GPU.

Table 3: Ablation study on the Beauty dataset.

Row	Variant	N@10
1	**AsymRec**	**0.0516**
2	w/ discrete codes as inputs	0.0491
3	w/ only one expert as input	0.0508
4	w/ continuous embeddings as outputs	0.0406
5	w/o MHQ	0.0494

4.2 Overall Performance (RQ1)

We compare AsymRec with item ID-based and semantic ID-based baselines across four datasets. The results are shown in Table 2.

Compared to all baselines, the proposed AsymRec achieves the best overall performance, ranking first in all metrics. It outperforms the strongest baseline by an average of 15.8% on the NDCG@10 metric.

4.3 Ablation Study

We conduct a series of ablation experiments to validate the effectiveness of the design choices in AsymRec. The primary results are summarized in Table 3.

4.3.1 Impact of Input (RQ2). To investigate the necessity of the proposed *Multi-expert Semantic Projection (MSP)*, we compare AsymRec (Row 1) with a variant that directly uses discrete semantic IDs (SIDs) as model inputs (Row 2). Specifically, in this variant, each SID token is mapped to a learnable embedding via a lookup table, and the embeddings of all tokens corresponding to an item are

averaged to form its input representation [8]:

$$h_i = \frac{1}{|\mathrm{ID}(x_i)|} \sum_{i_{m,l} \in \mathrm{ID}(x_i)} \mathrm{Emb}(i_{m,l})$$

As shown in Table 3, replacing continuous embeddings with discrete codes leads to a performance drop.

To further isolate and analyze the impact of input representations, we conduct a frequency-aware similarity analysis at the representation level. For each user sequence, we compute the mean-pooled input representation of the historical items, $\bar{h} = \frac{1}{T}\sum_{i=1}^{T} h_i$, and measure its similarity to the representation of the ground-truth next item h_{T+1} as well as 99 randomly sampled negative items. Based on these similarity scores, we compute Recall@10 across different item frequency bins.

The results are shown in Fig. 3. We observe that models using discrete SID inputs suffer from substantial performance degradation on low-frequency (cold) items, indicating poor generalization beyond popular items. In contrast, our model consistently achieves higher Recall@10 across almost all frequency bins, with particularly significant gains in the low- and mid-frequency regimes. This suggests that directly learning continuous input representations via MSP better preserves the topological structure of the original embedding space, enabling the model to capture fine-grained semantic relationships even for infrequent items. Notably, while the discrete-input variant performs better on the highest-frequency bin, it exhibits a clear bias toward popular items, whereas AsymRec maintains a more balanced performance profile across the long-tail distribution.

Furthermore, we observed that incorporating a Reciprocal Rank Fusion (RRF) [2] strategy to combine the recommendation lists from Row 1 and Row 2 leads to a substantial performance gain, reaching an NDCG@10 of 0.0540. Specifically, the fusion is performed by accumulating scores as $\sum 1/(50 + \mathrm{rank})$. While this late-fusion approach demonstrates the complementary nature of the two variants, we leave the comprehensive optimization of this mechanism for future work.

To further verify the effectiveness of the proposed multi-expert design, we introduce a stronger baseline that replaces MSP with a *single* expert whose capacity is scaled to match the total parameter budget of the multi-expert setting. Specifically, this variant employs one expert with an E-times larger projection dimension as the input module, and the resulting configuration is reported in Row 3 of Table 3.

As shown in the results, the single-expert variant performs slightly worse than our full **Multi-expert Semantic Projection**, while still substantially outperforming the discrete-input baseline (Row 2). This comparison indicates that the asymmetric design of using *continuous mappings as model inputs* is the primary factor driving performance gains, whereas the multi-expert architecture further enhances representation quality by capturing diverse and complementary semantic subspaces.

4.3.2 Discrete Output vs. Continuous Output(RQ3). The comparison between Row 1 and Row 4 in Table 3 reveals that adopting a fully continuous pipeline (i.e., predicting continuous embeddings instead of discrete IDs) leads to the most significant performance degradation.

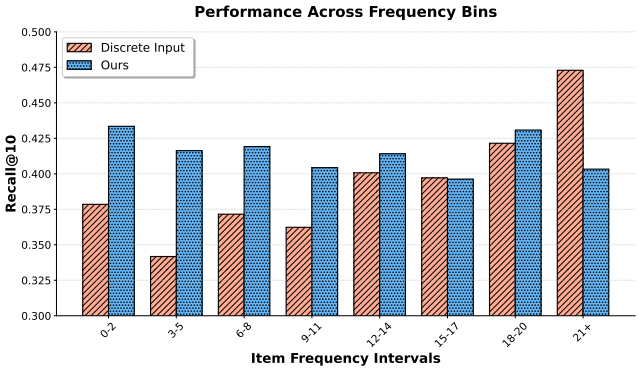

Figure 3: Retrieval performance at the input stage using Mean Pooling. Results are based on a 1-of-100 sampled ranking (1 positive target vs. 99 random negatives. 40% of the items have a frequency of 6 or less, while 80% of the items interact no more than 15 times.

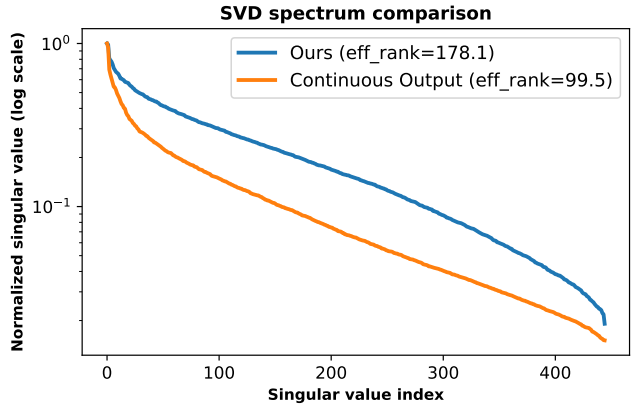

Figure 4: Normalized Singular Spectrum of Transformer Output. We observe that Continuous Embedding Token leads to collapsed singular values, while the Discrete Token leads more dimensionally robust representations.

We hypothesize that this is due to Representation Collapse in the continuous output space. To verify this, we compute the SVD-rank (Effective Rank) of the Transformer's output representations. As illustrated in Fig. 4, the effective rank of the continuous output is only 99.5, while our discrete SID output maintains a significantly higher rank of 178.1. The singular spectrum of the continuous variant decays rapidly, indicating that the model's predictions are trapped in a narrow, low-dimensional manifold. By supervising the model with discrete classification targets (SIDs), AsymRec forces the Transformer to preserve a more dimensionally robust and discriminative representation space.

To further investigate the performance discrepancy between continuous and discrete outputs, we hypothesize that the continuous output space suffers from representation collapse, a phenomenon where the learned hidden states are restricted to a low-dimensional manifold, thereby limiting the model's expressive capacity. We quantify the dimensionality of the representation space using the Effective Rank (ER) [23]. Specifically, given a matrix of output representations $\mathbf{Z} \in \mathbb{R}^{N \times d}$ for all N predictions, we perform Singular Value Decomposition (SVD) to obtain its singular values $\{\sigma_1, \sigma_2, \ldots, \sigma_d\}$. These singular values are normalized into a probability-like distribution $p_i = \sigma_i / \sum_{j=1}^{d} \sigma_j$, and the Effective Rank is defined as the exponential of the Shannon entropy of this distribution, $ER(\mathbf{Z}) = \exp(-\sum_{i=1}^{k} p_i \ln p_i)$. As illustrated in Fig. 4, the empirical results verify our hypothesis. The normalized singular value spectrum of the continuous variant exhibits a precipitous, power-law decay, where the first few components dominate the variance and the remaining dimensions provide negligible contributions. This leads to a significantly lower effective rank of only 99.5.

In contrast, our AsymRec employing discrete SID output maintains a much flatter and more robust singular spectrum. The singular values decay far more gradually, resulting in a substantially higher effective rank of 178.1. This contrast indicates that while direct continuous regression often leads the model toward a "lazy" solution—predicting mean-like vectors that lack discriminative power—supervising the model with discrete classification targets across $M \times L$ subspaces acts as a strong regularizer. By forcing the Transformer to distinguish between diverse semantic clusters defined by MHQ, AsymRec effectively prevents the representations from collapsing into a narrow manifold. Consequently, the model preserves a high-dimensional and discriminative feature space, which is essential for capturing the complex, fine-grained item relations required for accurate generative recommendation.

4.3.3 Impact of MHQ (RQ4). Finally, we evaluate the contribution of MHQ by replacing it with a standard PQ approach (Row 5). To further investigate the impact of the number of subspaces M and the number of residual layers per subspace L, we visualize the corresponding NDCG@10 scores in a heatmap, as shown in Figure 5. Here, we only consider configurations where $M \cdot L \leq 128$, as increasing $M \cdot L$ beyond this range does not lead to further performance gains.

From the heatmap, it is evident that increasing the number of subspaces M generally improves performance, especially when moving from $M = 4$ to $M = 32$. The effect of adding more residual layers L is more nuanced: moderate increases in L (from $L = 1$ to $L = 3$) tend to improve NDCG@10, while further increases show diminishing returns. This indicates that M and L play complementary roles in capturing the embedding structure.

Notably, our MHQ design demonstrates significant advantages over standard PQ. For example, with $M = 8$ and $L = 3$, MHQ achieves NDCG@10= 0.0514 using only 24 tokens, surpassing the best PQ configuration ($M = 64, L = 1$) which requires 64 tokens but only reaches 0.0494. Overall, these results highlight that MHQ can more efficiently leverage subspace and residual-layer structures to achieve higher recommendation quality with fewer tokens.

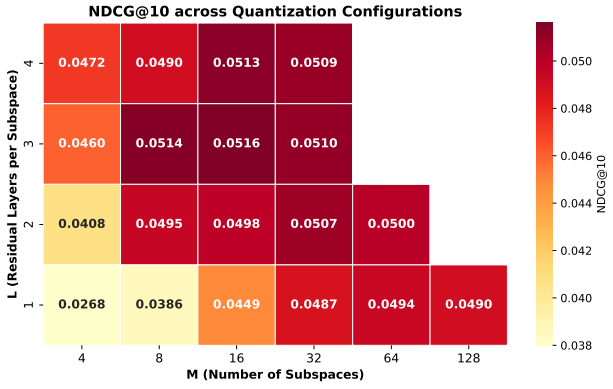

Figure 5: NDCG@10 under different quantization configurations on the Beauty dataset. The horizontal axis shows the number of subspaces M, and the vertical axis shows the number of residual layers per subspace L.

4.4 Online A/B Tests

To evaluate the practical effectiveness of the proposed method in real-world scenarios, we conducted online A/B tests within our post-click conversion rate (pCVR) prediction system deployed on one of the world's largest advertising platforms.

4.4.1 Feature Representation and Model Integration. Our approach leverages high-dimensional embeddings from two primary sources:

- **Cross-domain Latent Factor Model:** General-purpose embeddings generated by a large-scale cross-domain model, capturing broad user-item interactions.
- **Multimodal Alignment Embeddings:** Aligned multimodal features extracted from our internal Multimodal LLM via end-to-end contrastive learning, providing rich semantic information from diverse content signals.

The model is trained in an end-to-end fashion with a joint optimization objective. The total loss function consists of the primary pCVR prediction loss and a reconstruction loss:

$$\mathcal{L}_{total} = \mathcal{L}_{pCVR} + \lambda \mathcal{L}_{rec} \tag{15}$$

where $\mathcal{L}_{rec}$ ensures that the quantized SIDs retain the essential information of the original embeddings. These SIDs are then integrated as high-level categorical features into the downstream ranking network, significantly enhancing the model's ability to generalize across sparse conversion events.

4.4.2 Experimental Results and Analysis. We deployed the proposed method in a production environment and conducted an online A/B test on a 1% traffic slice over seven consecutive days. The experimental results demonstrate a significant business impact: compared to the production baseline, our method achieved a **1.4% increase in total consumption** and a **1.9% uplift in Gross Merchandise Volume (GMV)**.

These gains are statistically significant and underscore the advantage of integrating discretized SIDs from cross-domain and multimodal sources. The improvement in GMV specifically suggests that asymmetric encoding helps the model better capture

high-value conversion signals, thereby optimizing the trade-off between ad delivery volume and conversion quality.

5 Conclusion

In this paper, we have identified a critical yet often overlooked limitation in Generative Recommendation: the Dual-stage Information Bottleneck. Symmetric reliance on lossy quantization for both input and output leads to semantic distortion, popularity bias, and imprecise supervision, which collectively hinder the model's ability to generalize and capture fine-grained item relationships.

To address these challenges, we proposed AsymRec, an asymmetric continuous-discrete framework that decouples the representation paradigms of input and output. On the input side, Multi-expert Semantic Projection (MSP) directly maps continuous item embeddings into the model's feature space, preserving the underlying semantic topology and enabling better generalization to less frequent "cold" items. On the output side, Multi-faceted Hierarchical Quantization (MHQ) constructs high-capacity discrete targets through multi-path, hierarchical quantization, providing precise supervision while preventing dimensional collapse.

Extensive experiments across multiple benchmarks demonstrate that AsymRec consistently outperforms state-of-the-art generative recommenders, validating that asymmetric representation learning—combining continuous input mapping with structured discrete supervision—is key to achieving high-fidelity, fine-grained, and robust generative recommendations.

References

[1] Prabhat Agarwal, Anirudhan Badrinath, Laksh Bhasin, Jaewon Yang, Edoardo Botta, Jiajing Xu, and Charles Rosenberg. 2025. PinRec: Outcome-Conditioned, Multi-Token Generative Retrieval for Industry-Scale Recommendation Systems. arXiv:2504.10507 [cs.IR] doi:10.48550/arXiv.2504.10507 PinRec.

[2] Gordon V Cormack, Charles LA Clarke, and Stefan Buettcher. 2009. Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*. 758–759.

[3] 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. arXiv:2502.18965 [cs.IR] doi:10.48550/arXiv.2502.18965 OneRec.

[4] Robert Gray. 1984. Vector quantization. *IEEE Assp Magazine* 1, 2 (1984), 4–29.

[5] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948* (2025).

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

[7] Yupeng Hou, Zhankui He, Julian McAuley, and Wayne Xin Zhao. 2023. Learning vector-quantized item representation for transferable sequential recommenders. In *Proceedings of the ACM Web Conference 2023*. 1162–1171.

[8] Yupeng Hou, Jiacheng Li, Ashley Shin, Jinsung Jeon, Abhishek Santhanam, Wei Shao, Kaveh Hassani, Ning Yao, and Julian McAuley. 2025. Generating long semantic ids in parallel for recommendation. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*. 956–966.

[9] Tianyu Hua, Wenxiao Wang, Zihui Xue, Sucheng Ren, Yue Wang, and Hang Zhao. 2021. On feature decorrelation in self-supervised learning. In *Proceedings of the IEEE/CVF international conference on computer vision*. 9598–9608.

[10] Yanhua Huang, Yuqi Chen, Xiong Cao, Rui Yang, Mingliang Qi, Yinghao Zhu, Qingchang Han, Yaowei Liu, Zhaoyu Liu, Xuefeng Yao, Yuting Jia, Leilei Ma, Yinqi Zhang, Taoyu Zhu, Liujie Zhang, Lei Chen, Weihang Chen, Min Zhu, Ruiwen Xu, and Lei Zhang. 2025. Towards Large-scale Generative Ranking. arXiv:2505.04180 [cs.IR] doi:10.48550/arXiv.2505.04180 GenRank.

[11] Robert A Jacobs, Michael I Jordan, Steven J Nowlan, and Geoffrey E Hinton. 1991. Adaptive mixtures of local experts. *Neural computation* 3, 1 (1991), 79–87.

[12] Michael I Jordan and Robert A Jacobs. 1994. Hierarchical mixtures of experts and the EM algorithm. *Neural computation* 6, 2 (1994), 181–214.

[13] Biing-Hwang Juang and A Gray. 1982. Multiple stage vector quantization for speech coding. In *ICASSP'82. IEEE International Conference on Acoustics, Speech, and Signal Processing*, Vol. 7. IEEE, 597–600.

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

[15] Doyup Lee, Chiheon Kim, Saehoon Kim, Minsu Cho, and Wook-Shin Han. 2022. Autoregressive Image Generation Using Residual Quantization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 11523–11532.

[16] Xiaopeng Li, Bo Chen, Junda She, Shiteng Cao, You Wang, Qinlin Jia, Haiying He, Zheli Zhou, Zhao Liu, Ji Liu, et al. 2025. A survey of generative recommendation from a tri-decoupled perspective: Tokenization, architecture, and optimization. (2025).

[17] Chen Ma, Peng Kang, and Xue Liu. 2019. Hierarchical gating networks for sequential recommendation. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*. 825–833.

[18] Julieta Martinez, Holger H Hoos, and James J Little. 2014. Stacked quantizers for compositional vector compression. *arXiv preprint arXiv:1411.2173* (2014).

[19] Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton Van Den Hengel. 2015. Image-based recommendations on styles and substitutes. In *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval*. 43–52.

[20] Aleksandr V. Petrov and Craig Macdonald. 2024. RecJPQ: Training Large-Catalogue Sequential Recommenders. In *Proceedings of the ACM International Conference on Web Search and Data Mining (WSDM)*. RecJPQ.

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

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

[23] Olivier Roy and Martin Vetterli. 2007. The effective rank: A measure of effective dimensionality. In *2007 15th European signal processing conference*. IEEE, 606–610.

[24] 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.

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

[26] Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. 2017. Neural Discrete Representation Learning. In *Advances in Neural Information Processing Systems*, Vol. 30. 6306–6315.

[27] Xiaolong Xu, Hongsheng Dong, Lianyong Qi, Xuyun Zhang, Haolong Xiang, Xiaoyu Xia, Yanwei Xu, and Wanchun Dou. 2024. Cmclrec: Cross-modal contrastive learning for user cold-start sequential recommendation. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1589–1598.

[28] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388* (2025).

[29] Jiaqi Zhai, Lucy Liao, Xing Liu, Yueming Wang, Rui Li, Xuan Cao, Leon Gao, Zhaojie Gong, Fangda Gu, Michael He, et al. 2024. Actions speak louder than words: Trillion-parameter sequential transducers for generative recommendations. *arXiv preprint arXiv:2402.17152* (2024).

[30] Tingting Zhang, Pengpeng Zhao, Yanchi Liu, Victor S Sheng, Jiajie Xu, Deqing Wang, Guanfeng Liu, Xiaofang Zhou, et al. 2019. Feature-level deeper self-attention network for sequential recommendation.. In *IJCAI*. 4320–4326.

[31] Guorui Zhou, Hengrui Hu, Hongtao Cheng, Huanjie Wang, Jiaxin Deng, Jinghao Zhang, Kuo Cai, Lejian Ren, Lu Ren, Liao Yu, Pengfei Zheng, Qiang Luo, Qianqian Wang, Qigen Hu, Rui Huang, Ruiming Tang, Shiyao Wang, Shujie Yang, Tao Wu, Wuchao Li, Xinchen Luo, Xingmei Wang, Yi Su, Yunfan Wu, Zexuan Cheng, Zhanyu Liu, Zixing Zhang, Bin Zhang, Boxuan Wang, Chaoyi Ma, Chengru Song, Chenhui Wang, Chenglong Chu, Di Wang, Dongxue Meng, Dunju Zang, Fan Yang, Fangyu Zhang, Feng Jiang, Fuxing Zhang, Gang Wang, Guowang Zhang, Han Li, Honghui Bao, Hongyang Cao, Jiaming Huang, Jiapeng Chen, Jiaqiang Liu, Jinghui Jia, Kun Gai, Lantao Hu, Liang Zeng, Qiang Wang, Qidong Zhou, Rongzhou Zhang, Shengzhe Wang, Shihui He, Shuang Yang, Siyang Mao, Sui Huang, Tiantian He, Tingting Gao, Wei Yuan, Xiao Liang, Xiaoxiao Xu, Xugang Liu, Yan Wang, Yang Zhou, Yi Wang, Yiwu Liu, Yue Song, Yufei Zhang, Yunfeng Zhao, Zhixin Ling, and Ziming Li. 2025. OneRec-V2 Technical Report. arXiv:2508.20900 [cs.IR] doi:10.48550/arXiv.2508.20900 OneRec-V2.

[32] 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.

[33] Yongchun Zhu, Ruobing Xie, Fuzhen Zhuang, Kaikai Ge, Ying Sun, Xu Zhang, Leyu Lin, and Juan Cao. 2021. Learning to warm up cold item embeddings for cold-start recommendation with meta scaling and shifting networks. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1167–1176.