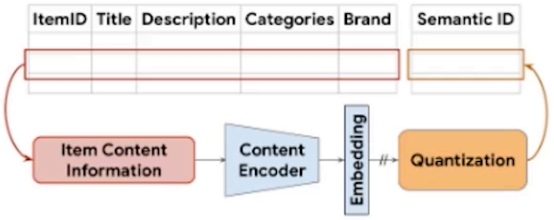
我们将语义ID定义为长度为m的token元组。

元组中的每个token来自不同的codebook。因此，语义id可以唯一表示的项数等于每层codebook大小的乘积。虽然生成语义ID的不同技术导致ID具有不同的语义属性，但我们希望它们至少具有以下属性：

相似的项（具有相似内容特征或语义嵌入接近的项）应该具有重叠的语义id。



(a) Semantic ID generation for items using quantization of content embeddings.

例如，语义ID〔10,21,35〕的项应该与语义ID〔10,21,40〕的项更相似，而不是具有ID〔10,23,32〕的项。

语义ID与大语言模型的tokenizer的区别

1. 它是离散化的语义表示，不依赖具体物品ID，而是基于内容语义生成；
2. 它的组成单位（codewords）来自固定大小的代码簿，大小可控、可重复利用；
3. 它支持组合式表达：比如每个ID由3个token组成，每个token选自1000个候选，那总共可表达10亿个组合，足以覆盖大规模推荐系统中的物品空间。

通过语义ID的设计，使得推荐系统能够像语言模型那样“输出”目标物品，但又规避了推荐领域中item ID/token空间过大、实体变化频繁等根本难题。可以说，语义ID是TIGER成功的核心支柱之一。

RQ-VAE：多层次量化机制

残差量化变分自动编码器（RQ-VAE）是一个多级矢量量化器，它对残差进行量化以生成语义id。通过更新量化codebook和DNN编码器参数对自编码器进行联合训练。

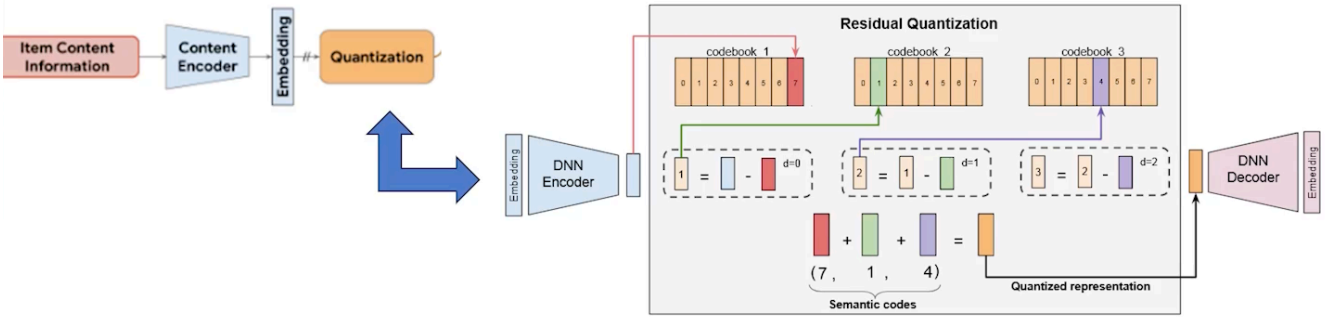


Figure 3: RQ-VAE: In the figure, the vector output by the DNN Encoder, say r_0 (represented by the blue bar), is fed to the quantizer, which works iteratively. First, the closest vector to r_0 is found in the first level codebook. Let this closest vector be e_{c_0} (represented by the red bar). Then, the residual error is computed as $r_1 := r_0 - e_{c_0}$. This is fed into the second level of the quantizer, and the process is repeated: The closest vector to r_1 is found in the second level, say e_{c_1} (represented by the green bar), and then the second level residual error is computed as $r_2 = r_1 - e'_{c_1}$. Then, the process is repeated for a third time on r_2. The semantic codes are computed as the indices of e_{c_0}, e_{c_1}, and e_{c_2} in their respective codebooks. In the example shown in the figure, this results in the code $(7, 1, 4)$.

基于RQ-VAE的semantic ID

RQ-VAE first encodes the input x via an encoder $\mathcal{E}$ to learn a latent representation $z := \mathcal{E}(x)$.

At the zero-th level ($d = 0$), the initial residual is simply defined as $r_0 := z$. At each level d, we have a codebook $\mathcal{C}_d := \{e_k\}_{k=1}^{K}$, where K is the codebook size.

Then, r_0 is quantized by mapping it to the nearest embedding from that level's codebook. The index of the closest embedding e_{c_d} at $d = 0$, i.e., $c_0 = \arg\min_k \|r_0 - e_k\|$, represents the zero-th codeword.

For the next level $d = 1$, the residual is defined as $r_1 := r_0 - e_{c_0}$.

Then, similar to the zero-th level, the code for the first level is computed by finding the embedding in the codebook for the first level which is nearest to r_1.

This process is repeated recursively m times to get a tuple of m codewords that represent the Semantic ID.

基于RQ-VAE的语义ID生成

Once we have the Semantic ID $(c_0, \ldots, c_{m-1})$, a quantized representation of $\mathbf{z}$ is computed as $\hat{\mathbf{z}} := \sum_{d=0}^{m-1} e_{c_i}$.

Then $\hat{\mathbf{z}}$ is passed to the decoder, which tries to recreate the input $\mathbf{x}$ using $\hat{\mathbf{z}}$. The RQ-VAE loss is defined as

$$\mathcal{L}(\mathbf{x}) := \mathcal{L}_{\text{recon}} + \mathcal{L}_{\text{rqvae}}$$

where $\mathcal{L}_{\text{recon}} := \|\mathbf{x} - \hat{\mathbf{x}}\|^2$, and $\mathcal{L}_{\text{rqvae}} := \sum_{d=0}^{m-1} \|\text{sg}[r_i] - e_{c_i}\|^2 + \beta\|r_i - \text{sg}[e_{c_i}]\|^2$.

Here $\hat{\mathbf{x}}$ is the output of the decoder, and sg is the stop-gradient operation. This loss jointly trains the encoder, decoder, and the codebook.

To prevent RQ-VAE from a codebook collapse, where most of the input gets mapped to only a few codebook vectors, we use k-means clustering-based initialization for the codebook. Specifically, we apply the k-means algorithm on the first training batch and use the centroids as initialization.

语义ID生成

其它量化方法：

LSH：LSH（Locality Sensitive Hashing）：一种简单但粗糙的离散化方式，速度快但精度低消融实验证实RQ-VAE性能更优

VQ-VAE：不具有语义层次粒度

基于k-means层次聚类：丢失ID之间的语义关系；

冲突处理：

附加标识位：增加一位token 表示语义重复。

查找表：维护一个映射表，将语义ID映射回具体的物品ID，避免模型误判。

只需训练后处理一次：冲突检测和修复仅在训练好RQ-VAE之后执行一次，不影响训练效率。

推荐多样性

Table 3: The entropy of the category distribution predicted by the model for the Beauty dataset. A higher entropy corresponds more diverse items predicted by the model.

Temperature	Entropy@10	Entropy@20	Entropy@50
T = 1.0	0.76	1.14	1.70
T = 1.5	1.14	1.52	2.06
T = 2.0	1.38	1.76	2.28

Table 4: Recommendation diversity with temperature-based decoding.

Target Category	Most-common Categories for top-10 predicted items	
	T = 1.0	T = 2.0
Hair Styling Products	Hair Styling Products	Hair Styling Products, Hair Styling Tools, Skin Face
Tools Nail	Tools Nail	Tools Nail, Makeup Nails
Makeup Nails	Makeup Nails	Makeup Nails, Skin Hands & Nails, Tools Nail
Skin Eyes	Skin Eyes	Hair Relaxers, Skin Face, Hair Styling Products, Skin Eyes
Makeup Face	Tools Makeup Brushes,Makeup Face	Tools Makeup Brushes, Makeup Face,Skin Face, Makeup Sets, Hair Styling Tools
Hair Loss Products	Hair Loss Products,Skin Face, Skin Body	Skin Face, Hair Loss Products, Hair Shampoos,Hair & Scalp Treatments, Hair Conditioners

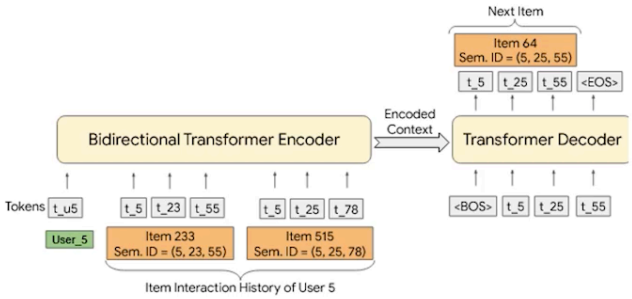
- Entropy@K，计算top-K物品的ground-truth的类别分布；
- 解码器过程中基于温度系数的采样 temperature-based sampling可以应用于任何现有的推荐模型来控制结果多样性，但由于RQ-VAE语义id的特性，TIGER允许跨不同层次的采样。

基于语义ID的生成式检索

我们通过按时间顺序排序的用户历史交互的物品来为每个用户构建物品序列。

然后，给定一个形式为（$item_1$，…，$item_n$）的序列，推荐系统的任务是预测下一个物品$item_{n+1}$。我们提出了一种直接预测下一个物品的语义ID的生成方法。

形式上，设（$c_{i,0}$，…，$c_{i,m-1}$）为$item_i$的m长度的语义ID。然后,我们将物品序列转换为序列$(c_{1,0} \ldots, c_{1,m-1}, c_{2,0}, \ldots, c_{2,m-1}, \ldots, c_{n,0}, \ldots, c_{n,m-1})$。然后训练序列到序列模型来预测物品的语义ID，即（$c_{n+1,0}, \ldots, c_{n+1,m-1}$）。考虑到我们框架的生成特性，从解码器生成的语义ID可能与推荐语料库中的物品不匹配，但此类事件发生的概率很低。



(b) Transformer based encoder-decoder setup for building the sequence-to-sequence model used for generative retrieval.

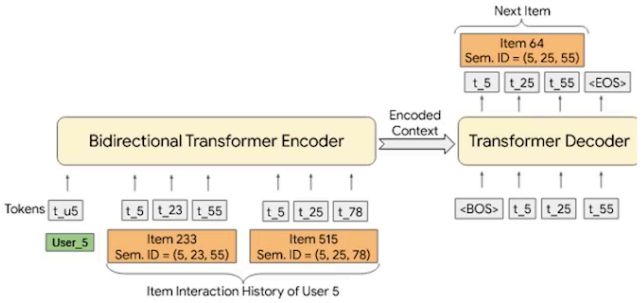
基于语义ID的生成式检索

我们通过按时间顺序排序的用户历史交互的物品来为每个用户构建物品序列。

然后，给定一个形式为（item_1，…，item_n）的序列，推荐系统的任务是预测下一个物品item_{n+1}。我们提出了一种直接预测下一个物品的语义ID的生成方法。

形式上，设（$c_{i,0}$， …， $c_{i,m-1}$）为item_i的m长度的语义ID。然后,我们将物品序列转换为序列$(c_{1,0} \ldots, c_{1,m-1}, c_{2,0}, \ldots, c_{2,m-1}, \ldots, c_{n,0}, \ldots, c_{n,m-1})$。然后训练序列到序列模型来预测物品的语义ID，即（$c_{n+1,0}, \ldots, c_{n+1,m-1}$）。考虑到我们框架的生成特性，从解码器生成的语义ID可能与推荐语料库中的物品不匹配，但此类事件发生的概率很低。

通过语义ID的设计，TIGER框架成功地将推荐系统引入了一个生成式检索的新范式，使得系统能够像语言模型那样直接“输出”目标物品。



(b) Transformer based encoder-decoder setup for building the sequence-to-sequence model used for generative retrieval.

实验设置

- 基座模型: Setence-T5 (获得物品的语义嵌入：768维）
- 数据集: Amazon Product Reviews （1996.5-2014.7）
  - Beauty
  - Sports and Outdoors
  - Toys and Games
- 评估指标
  - Recall@k
  - NDCG@k （K=5,or 10)

Table 6: Dataset statistics for the three real-world benchmarks.

Dataset	# Users	# Items	Sequence Length	
			Mean	Median
Beauty	22,363	12,101	8.87	6
Sports and Outdoors	35,598	18,357	8.32	6
Toys and Games	19,412	11,924	8.63	6

基线模型

- GRU[2015] ：首个将定制化GRU（门控循环单元）用于序列推荐任务的基于RNN的方法
- Caser[2018] ：该模型采用CNN架构，通过应用横向和垂直卷积操作来捕捉高阶马尔可夫链，以实现序列推荐。
- HGN[2019] ：层次门控网络利用一种新的门控架构以捕捉用户的长期与短期兴趣。
- SASRec[2018] ：自注意力序列推荐模型采用因果掩码Transformer，为用户的序列交互行为建模
- BERT4Rec[2019] ： BERT4Rec采用双向自注意力Transformer，克服了单向模型的缺陷，并将其应用于推荐任务
- FDSA[2019] ：特征级深度自注意力网络将物品特征与物品嵌入一同引入，作为Transformer的输入序列
- S^3–Rec[2020] ：序列推荐的自监督学习技术引入基于自监督任务的双向Transformer预训练，以提升模型性能。
- P5[2022] ： P5是一项新近提出的技术，它通过使用预训练大语言模型，将多种推荐任务整合到一个统一的框架中。

实验结果

Table 1: Performance comparison on sequential recommendation. The last row depicts % improvement with TIGER relative to the best baseline. Bold (underline) are used to denote the best (second-best) metric.

Methods	Sports and Outdoors				Beauty				Toys and Games			
	Recall @5	NDCG @5	Recall @10	NDCG @10	Recall @5	NDCG @5	Recall @10	NDCG @10	Recall @5	NDCG @5	Recall @10	NDCG @10
P5 [8]	0.0061	0.0041	0.0095	0.0052	0.0163	0.0107	0.0254	0.0136	0.0070	0.0050	0.0121	0.0066
Caser [33]	0.0116	0.0072	0.0194	0.0097	0.0205	0.0131	0.0347	0.0176	0.0166	0.0107	0.0270	0.0141
HGN [25]	0.0189	0.0120	0.0313	0.0159	0.0325	0.0206	0.0512	0.0266	0.0321	0.0221	0.0497	0.0277
GRU4Rec [11]	0.0129	0.0086	0.0204	0.0110	0.0164	0.0099	0.0283	0.0137	0.0097	0.0059	0.0176	0.0084
BERT4Rec [32]	0.0115	0.0075	0.0191	0.0099	0.0203	0.0124	0.0347	0.0170	0.0116	0.0071	0.0203	0.0099
FDSA [42]	0.0182	0.0122	0.0288	0.0156	0.0267	0.0163	0.0407	0.0208	0.0228	0.0140	0.0381	0.0189
SASRec [17]	0.0233	0.0154	0.0350	0.0192	0.0387	0.0249	0.0605	0.0318	0.0463	0.0306	0.0675	0.0374
S^3-Rec [44]	0.0251	0.0161	0.0385	0.0204	0.0387	0.0244	0.0647	0.0327	0.0443	0.0294	0.0700	0.0376
TIGER [Ours]	**0.0264**	**0.0181**	**0.0400**	**0.0225**	**0.0454**	**0.0321**	**0.0648**	**0.0384**	**0.0521**	**0.0371**	**0.0712**	**0.0432**
	+5.22%	+12.55%	+3.90%	+10.29%	+17.31%	+29.04%	+0.15%	+17.43%	+12.53%	+21.24%	+1.71%	+14.97%

- TIGER在各个数据集和评测指标上取得了最佳效果, 特别是在Beauty数据集上急NDCG指标上

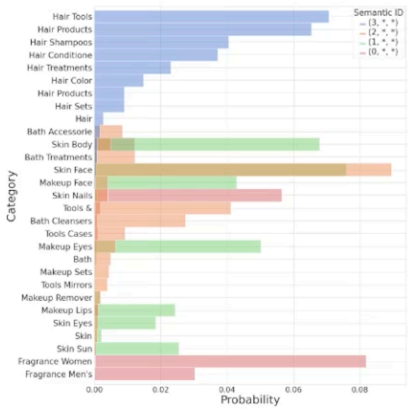
不同数据集误差对比结果

Table 9: The mean and stand error of the metrics for different dataset (computed using 3 runs with different random seeds)

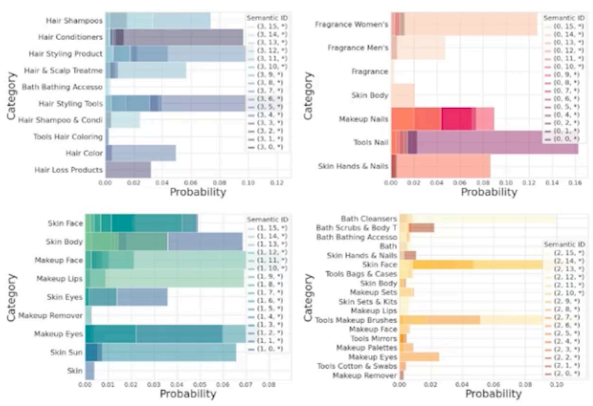
Datasets	Recall@5	NDCG@5	Recall@10	NDCG@10
Beauty	0.0441 ± 0.00069	0.0309 ± 0.00062	0.0642 ± 0.00092	0.0374 ± 0.00061
Sports and Outdoors	0.0278 ± 0.00069	0.0189 ± 0.00043	0.0419 ± 0.0010	0.0234 ± 0.00048
Toys and Games	0.0518 ± 0.00064	0.0375 ± 0.00039	0.0698 ± 0.0013	0.0433 ± 0.00047

- TIGER在三轮使用不同随机数种子的实验中，指标的均值和标准差的结果

商品表示：语义ID质量分析



(a) The ground-truth category distribution for all the items in the dataset colored by the value of the first codeword c_1.



(b) The category distributions for items having the Semantic ID as $(c_1, *, *)$, where $c_1 \in \{1, 2, 3, 4\}$. The categories are color-coded based on the second semantic token c_2.

Figure 4: Qualitative study of RQ-VAE Semantic IDs (c_1, c_2, c_3, c_4) on the Amazon Beauty dataset. We show that the ground-truth categories are distributed across different Semantic tokens. Moreover, the RQVAE semantic IDs form a hierarchy of items, where the first semantic token (c_1) corresponds to coarse-level category, while second/third semantic token (c_2/c_3) correspond to fine-grained categories.

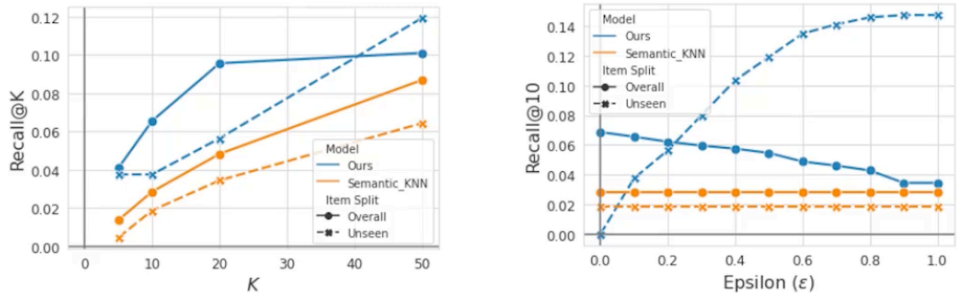
消融实验：不同ID生成方式

Table 2: Ablation study for different ID generation techniques for generative retrieval. We show that RQ-VAE Semantic ID (SID) perform significantly better compared to hashing SIDs and Random IDs.

Methods	Sports and Outdoors				Beauty				Toys and Games			
	Recall @5	NDCG @5	Recall @10	NDCG @10	Recall @5	NDCG @5	Recall @10	NDCG @10	Recall @5	NDCG @5	Recall @10	NDCG @10
Random ID	0.007	0.005	0.0116	0.0063	0.0296	0.0205	0.0434	0.0250	0.0362	0.0270	0.0448	0.0298
LSH SID	0.0215	0.0146	0.0321	0.0180	0.0379	0.0259	0.0533	0.0309	0.0412	0.0299	0.0566	0.0349
RQ-VAE SID	**0.0264**	**0.0181**	**0.0400**	**0.0225**	**0.0454**	**0.0321**	**0.0648**	**0.0384**	**0.0521**	**0.0371**	**0.0712**	**0.0432**

- RQ-VAE基本优于局部敏感哈希（LSH）。
- 给定相同的基于内容的语义嵌入，通过非线性深度神经网络（DNN）架构学习语义ID比使用随机投影产生更好的量化。
- 语义ID始终优于随机ID基线，突出了利用基于内容的语义信息的重要性。

冷启动推荐



(a) Recall@K vs. K, ($\epsilon = 0.1$). (b) Recall@10 vs. ϵ.

Figure 5: Performance in the cold-start retrieval setting.

- 基于“Beauty”数据集上的对比
- 将测试物品中的 5 %从训练集中移除，作为“未见过”物品； ε表示TIGER框架中能够选择到的“未见过”物品比例
- 与KNN相比，在ε=0.1，TIGER在所有K值上的Recall性能更优；针对不同ε，也显示出性能更优。
- 相比之下，TIGER框架可以很容易地执行冷启动推荐，因为它在预测下一个物品时利用了物品语义。

推荐多样性

Table 3: The entropy of the category distribution predicted by the model for the Beauty dataset. A higher entropy corresponds more diverse items predicted by the model.

Temperature	Entropy@10	Entropy@20	Entropy@50
T = 1.0	0.76	1.14	1.70
T = 1.5	1.14	1.52	2.06
T = 2.0	1.38	1.76	2.28

Table 4: Recommendation diversity with temperature-based decoding.

Target Category	Most-common Categories for top-10 predicted items	
	T = 1.0	T = 2.0
Hair Styling Products	Hair Styling Products	Hair Styling Products, Hair Styling Tools, Skin Face
Tools Nail	Tools Nail	Tools Nail, Makeup Nails
Makeup Nails	Makeup Nails	Makeup Nails, Skin Hands & Nails, Tools Nail
Skin Eyes	Skin Eyes	Hair Relaxers, Skin Face, Hair Styling Products, Skin Eyes
Makeup Face	Tools Makeup Brushes,Makeup Face	Tools Makeup Brushes, Makeup Face,Skin Face, Makeup Sets, Hair Styling Tools
Hair Loss Products	Hair Loss Products,Skin Face, Skin Body	Skin Face, Hair Loss Products, Hair Shampoos,Hair & Scalp Treatments, Hair Conditioners

- Entropy@K，计算top-K物品的ground-truth的类别分布；
- 解码器过程中基于温度系数的采样 temperature-based sampling可以应用于任何现有的推荐模型来控制结果多样性，但由于RQ-VAE语义id的特性，TIGER允许跨不同层次的采样。

消融实验：模型层数及用户信息的作用

Table 5: Recall and NDCG metrics for different number layers.

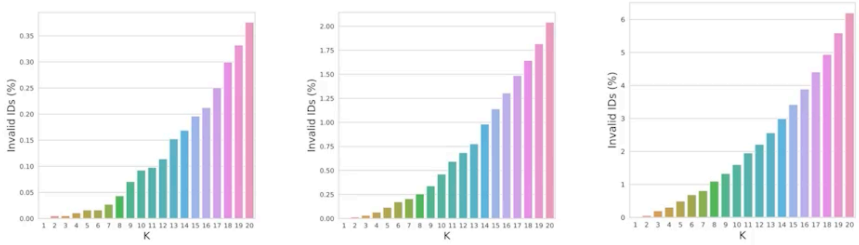
Number of Layers	Recall@5	NDCG@5	Recall@10	NDCG@10
3	0.04499	0.03062	0.06699	0.03768
4	0.0454	0.0321	0.0648	0.0384
5	0.04633	0.03206	0.06596	0.03834

Table 8: The effect of providing user information to the recommender system

Recall@5	NDCG@5	Recall@10	NDCG@10	
No user information	0.04458	0.0302	0.06479	0.0367
With user id (reported in the paper)	0.0454	0.0321	0.0648	0.0384

- 当网络变大（层数变多），模型性能优轻微的提升。
- 为语言模型提供用户信息对性能有一定的帮助

不正确编码/Invalid ID



(a) Sports and Outdoors (b) Beauty (c) Toys and Games

Figure 6: Percentage of invalid IDs when generating Semantic IDs using Beam search for various values of K. As shown, $\sim 0.3\% - 6\%$ of the IDs are invalid when retrieving the top-20 items.

- 由于模型是自回归地解码生成目标语义ID，因此模型可能会预测出无效的ID（即在推荐数据集中没有对应物品的ID）。**我们观察到模型几乎总是预测出有效的ID**。在**图6**中，**对于前10个预测结果，三个数据集中无效ID的比例在约0.1%至1.6%之间变化。可以增加beam search波束搜索的大小并过滤掉无效ID**。
- 尽管会生成无效ID，但与用于顺序推荐的其他流行方法相比，TIGER仍然实现了最优的性能。一种处理无效token的方法是，当模型生成无效token时进行**前缀匹配**。RQ-VAE token具有**层次结构**，**前缀匹配可以看作是模型预测物品类别而不是物品索引**，这样的扩展可以进一步提高召回率/NDCG指标。

其它讨论

Table 10: Testing scalability by generating the Semantic IDs on the combined dataset vs generating the Semantic IDs on only the Beauty dataset.

	Recall@5	NDCG@5	Recall@10	NDCG@10
Semantic ID [Combined datasets]	0.04355	0.3047	0.06314	0.03676
Semantic ID [Amazon Beauty]	0.0454	0.0321	0.0648	0.0384

- **语义ID长度和码本大小的影响：**通过试验（尝试6-元组的语义ID，每层64个codewords），TIGER的推荐指标对语义ID的长度和codebook大小的变化具有鲁棒性。然而着ID变长，输入序列的长度也会增加，这会使基于Transformer的序列到序列模型的计算成本更高。
- **可扩展性**：将所有三个数据集合并，与仅从Beauty数据集生成语义ID的原始实验结果进行比较的结果如**表10**所示。我们发现，性能仅略有下降。
- **推理成本：**由于使用了自回归解码的beam search，模型在推理期间的计算成本可能比基于ANN的模型更高。论文开辟了一个新的研究领域：基于生成检索的推荐系统。我们将考虑使模型更小的方法或探索其他提高推理效率的方法。
- **查找表的内存成本：**我们为TIGER维护了两个查找哈希表：一个是从物品ID到语义ID的表，另一个是从语义ID到物品ID的表。它们在基于RQ-VAE的语义ID生成模型训练后生成，每个语义ID由一个4个整数的元组组成，，每个查找表的大小将为64N位左右，其中N是数据集中的物品数量。
- **嵌入表的内存成本：**与传统推荐系统相比，TIGER使用的嵌入表要小得多。这是因为传统推荐系统为每个物品存储一个嵌入，而TIGER只为每个语义码字存储一个嵌入。

总结

- 本文提出了一种新的推荐范式，称为TIGER，使用生成检索模型在序列推荐中“生成”下一个可能的交互对象。
- 支撑这种方法的是一种新的物品语义ID表示，它在内容嵌入上使用层次量化器（RQ-VAE）来生成形成语义ID。
- 基于Transformer的序列生成模型：将用户的历史行为表示为语义ID序列，使用Encoder-Decoder结构的Transformer模型学习用户偏好，并生成下一个物品的语义ID。
- 嵌入表的基数不会随着物品空间的基数线性增长，这与需要在训练期间创建大型嵌入表或为每个单独物品生成索引的系统相比，更为有利。
- 通过在三个数据集上的实验，证明了我们的模型可以达到由于SOTA检索的性能，同时可以泛化到新的和未见过的物品。

不足

论文中的语义ID建模是静态的，没有与user-item交互相关，也就是没有融合协作信息，后续的生成式模型有不少改进空间。

基于生成式检索的推荐：发展前沿

Recommender Systems with Generative Retrieval	2023 NeurIPS	谷歌提出TIGER框架，首次将生成式检索用于推荐系统，通过语义ID序列，利用Transformer自回归生成候选集，
Actions Speak Louder than Words: Trillion-Parameter Sequential Transducers for Generative Recommendations	2024 ICML	Meta提出生成式推荐新范式，设计HSTU架构处理高基数动态数据，将推荐任务转化为序列变换问题，
MTGR: Industrial-Scale Generative Recommendation Framework in Meituan	2025 CIKM	美团设计工业级生成框架，提升多语义建模，
Sparse Meets Dense: Unified Generative Recommendations with Cascaded Sparse-Dense Representations	2025 arXiv	百度提出COBRA框架，在广告场景显著优化业务指标
OneRec: Unifying Retrieve and Rank with Generative Recommender and Preference Alignment	2025 arXiv	快手提出端到端单阶段生成框架，采用MoE扩展模型容量 在线实验提升总观看时长1.68%

基于生成式检索的推荐：发展前沿

Learnable Item Tokenization for Generative Recommendation	对齐推荐
DAS: Dual-Aligned Semantic IDs Empowered Industrial Recommender System	对齐推荐
OneRec Technical Report - RQ-KMeans *Towards Scalable Semantic Representation for Recommendation*	多套semantic ID
Generating Long Semantic IDs in Parallel for Recommendation	多套semantic ID
MMQ: Multimodal Mixture-of-Quantization Tokenization for Semantic ID Generation and User Behavioral Adaptation	多套semantic ID