

GeoSID: Quantizer-Aware Geometric Shaping for Semantic IDs in Generative Recommendation

Xinrong He^{a*} Chenghao Wu^{*} Lianrong Chen Xuming Hu Menglin Yang^{b†}

The Hong Kong University of Science and Technology (Guangzhou)

^axhe543@connect.hkust-gz.edu.cn ^bmenglinyang@hkust-gz.edu.cn

Abstract

Semantic ID-based generative retrieval represents each item with a short sequence of discrete tokens derived from language-based item representations. Existing methods mainly instantiate this idea in two paradigms: autoregressive SID generation and parallel SID generation. In both paradigms, the quality of Semantic IDs depends critically on how continuous item embeddings are discretized before they become generation targets. While existing work has improved SID tokenization through better quantization objectives and supervision signals, it often treats the pre-tokenization embedding space as fixed, overlooking that pretrained item embeddings can be anisotropic and dominated by a few directions. This creates a pre-quantization distribution mismatch, where the global embedding geometry may not align with the structured spaces used for code assignment. We propose GeoSID, a tokenizer-side geometric shaping framework that keeps the downstream generative recommender unchanged. GeoSID uses a lightweight projected distribution matching regularizer to adjust item representations before SID assignment, and places the shaping objective in tokenizer-specific quantization spaces, including residual levels for residual quantization and OPQ/PQ blocks for product quantization. On the RPG backbone, GeoSID achieves up to 8.13% relative improvement in Recall@10 and 6.91% in NDCG@10. Further analyses show that GeoSID reduces projected distribution discrepancy, improves code usage balance, and lowers prefix-level SID collisions. Code is available at <https://github.com/hexinrong122/GeoSID>.

1 Introduction

Recommendation over large item catalogs has traditionally been framed as scoring or ranking candidate items (Kang and McAuley, 2018). Re-

cent Transformer-based generative recommendation models formulate next-item retrieval as conditional sequence generation, where a model predicts a token sequence that resolves to the next catalog item given user context, such as interaction history (Rajput et al., 2023; Hou et al., 2025b). This formulation brings recommendation closer to a central problem in NLP: the design of discrete target symbols for sequence models. In conventional recommendation systems, atomic item IDs are compact and easy to map back to the catalog, but they are semantically opaque and cannot share information across related items (Rajput et al., 2023; Wang et al., 2024). Directly generating item text provides richer semantics, but textual outputs are often long, ambiguous, and difficult to use as stable item identifiers. Semantic IDs (SIDs) address this trade-off by representing each item as a short sequence of discrete semantic tokens derived from item content representations (Rajput et al., 2023; Liu et al., 2024).

A typical SID pipeline first encodes item content into continuous semantic embeddings and then discretizes them into token sequences through quantization-based tokenizers, such as residual quantization or product quantization. The tokenizer therefore directly determines how continuous item semantics are compressed into discrete identifiers (Rajput et al., 2023; Hou et al., 2025a).

Most existing SID tokenizers focus on improving the discretization mechanism itself, while implicitly assuming that the input embedding space is already suitable for quantization. We argue that this assumption can fail in practice. Pretrained semantic embeddings are often anisotropic, with representations concentrated along a few dominant directions (Ethayarajh, 2019). Prior work suggests that non-uniform semantic embeddings can lead to imbalanced codeword assignments (Wei et al., 2026). We further hypothesize that such imbalanced geometry may reduce SID prefix separabil-

* Equal contribution.

† Corresponding author.

ity, making the representation distribution before code assignment an important but underexplored factor.

In residual quantization, later codebooks operate on residual spaces that may become progressively less informative. In product quantization, embeddings are split into independently quantized subspaces after rotation, where some blocks may still contain imbalanced or poorly shaped distributions (Hou et al., 2025a). A naive application of distribution regularization, like shaping the global input embedding, can therefore be insufficient or even harmful under structured quantizers, because the actual code assignment does not operate on the global embedding space. This motivates a quantizer-aware placement principle: the shaping granularity should match the decomposition structure of the tokenizer.

To address this issue, we propose GeoSID, a tokenizer-side geometric shaping framework for Semantic ID generation. GeoSID is a quantizer-aware placement framework rather than a new base regularizer. It adjusts item representations before SID assignment so that the inputs to quantization become more distributionally balanced, while keeping the original item encoder and downstream recommender unchanged. GeoSID aligns the shaping granularity with the structure of the target tokenizer. The shaping objective is implemented with a lightweight projected distribution matching regularizer.

GeoSID is plug-and-play because it improves the representation geometry directly before SID tokenization, making it compatible with existing generative recommendation pipelines. We evaluate GeoSID on multiple public recommendation datasets and SID backbones, including residual quantization-based and OPQ/PQ-based tokenizers. Our contributions are summarized as follows:

- We identify pre-quantization distribution imbalance as an overlooked factor in Semantic ID generation. We provide diagnostic evidence through PCA density visualization and SID quality metrics, showing that the geometry of tokenizer inputs is closely related to code usage balance and prefix-level SID separability.
- We propose GeoSID, a lightweight tokenizer-side geometric shaping framework that reshapes item representations before SID as-

signment without modifying the item encoder or downstream generative recommender. GeoSID reduces direction-dominated structures in the quantization input space through projected distribution matching.

- We instantiate GeoSID for two representative SID tokenizer paradigms: residual quantization and product quantization. Through residual-level shaping and OPQ-aligned block-wise shaping, GeoSID improves Recall@ K and NDCG@ K across multiple datasets and SID backbones. Further analysis shows reduced projected distribution discrepancy, more balanced code usage, and lower prefix-level SID collisions.

2 Related Work

2.1 Generative Recommendation with Semantic IDs

Generative recommendation formulates item retrieval as a token sequence generation problem, where a model generates item identifiers conditioned on user context (Rajput et al., 2023; Hou et al., 2025b).

In recommender systems, TIGER is a representative milestone of Semantic ID-based generative recommendation (Rajput et al., 2023). It derives Semantic IDs from item content embeddings and trains a Transformer model to autoregressively generate the next item’s identifier. Semantic IDs have also been studied for joint generative search and recommendation, where the choice of embedding sources and SID spaces affects downstream retrieval quality (Penha et al., 2025). GeoSID is complementary to this line of work: it keeps the downstream generative recommender unchanged and focuses on the representation space before SID tokenization.

2.2 Semantic ID Tokenization and Quantization

Semantic ID tokenization discretizes continuous item representations into token sequences through quantization. In this work, we focus on two representative structured paradigms widely used in SID-based recommendation: residual quantization and product quantization (Lee et al., 2022; Hou et al., 2025a). Residual quantization-based tokenizers encode an item through multiple residual codebooks, where later levels are expected to capture finer-grained residual information (Chen

et al., 2010). Product quantization provides another paradigm: PQ decomposes a representation into multiple subspaces and quantizes each subspace independently, while OPQ learns a rotation before PQ to reduce quantization distortion (Ge et al., 2013; Jegou et al., 2011). These two paradigms lead to different decomposition structures: residual spaces for RQ-based tokenizers and independently quantized blocks for OPQ/PQ-based tokenizers.

Recent SID tokenization methods mainly improve the quantization objective, supervision signal, or model-tokenizer coupling (Zhu et al., 2024; Liu et al., 2025; Wei et al., 2026). LETTER studies learnable item tokenization and improves code assignment with collaborative signals and code diversity regularization, aiming to alleviate codebook under-utilization or collapse (Wang et al., 2024). CARD studies non-uniform quantization for multi-modal Semantic ID construction and observes that non-uniform item semantic embeddings can lead to imbalanced codeword assignment and biased SID generation (Wei et al., 2026). Unlike these methods, GeoSID reshapes the representation space before SID assignment and matches the shaping granularity to the actual quantization units of the tokenizer.

2.3 Representation Geometry and Distribution Shaping

The geometry of pretrained representations has been widely studied in NLP. Prior work shows that contextual representations and sentence embeddings from pretrained language models can be anisotropic, with representations concentrated along a few dominant directions (Ethayarajh, 2019; Li et al., 2020). Several studies therefore post-process pretrained embeddings to make the representation space less dominated by a few common directions, for example by removing dominant components or balancing variance across dimensions (Su et al., 2021). In recommendation, however, the role of representation geometry before SID tokenization has received less systematic study, even though SID quality is directly affected by how continuous item embeddings are discretized.

Distribution regularization provides a possible way to control representation geometry. SIGReg introduces a projected distribution matching objective that encourages representations to align with an isotropic Gaussian-like reference distribution through random projections and characteristic-function matching (Balestriero and LeCun, 2025).

However, SIGReg is a general representation

regularizer, while SID tokenization imposes additional structural constraints. Residual quantization decomposes representations across residual levels, and product quantization decomposes an OPQ-rotated representation into independent subspaces. GeoSID therefore differs from standard distribution regularization.

3 Methodology

3.1 Overview of GeoSID

GeoSID is a tokenizer-side geometric shaping framework for Semantic ID generation. Given item representations to be discretized, GeoSID improves their geometric distribution before Semantic ID assignment while keeping the downstream generative recommender unchanged. Instead of designing a new tokenizer, GeoSID introduces a lightweight projected distribution matching loss, inspired by SIGReg, to reshape the representation spaces used for code assignment. Unlike PCA or whitening, which mainly adjust variance and covariance along principal components, projected CF matching regularizes the shape of projected densities over randomly sampled directions. This makes GeoReg better aligned with our goal of reducing pre-quantization density concentration before SID code assignment, rather than merely balancing second-order statistics.

The key principle of GeoSID is quantizer-aware distribution shaping. For RQ-based tokenizers, code assignment is performed over residual spaces, so GeoSID applies GeoReg to residual-level representations. For OPQ/PQ-based tokenizers, code assignment is performed in OPQ-aligned PQ blocks, so GeoSID applies GeoReg block-wise. Thus, GeoSID regularizes the spaces directly consumed by the tokenizer without modifying the main tokenization mechanism.

3.2 Projected Distribution Matching Loss

We use a projected distribution matching loss as the basic geometric shaping objective. Given a batch of representations, the loss samples multiple random one-dimensional projection directions and compares the empirical projected distributions with an isotropic Gaussian reference. Specifically, for each projection direction and frequency point, we compute the real-part and imaginary-part deviations between the empirical characteristic function and the Gaussian target. We write the projected

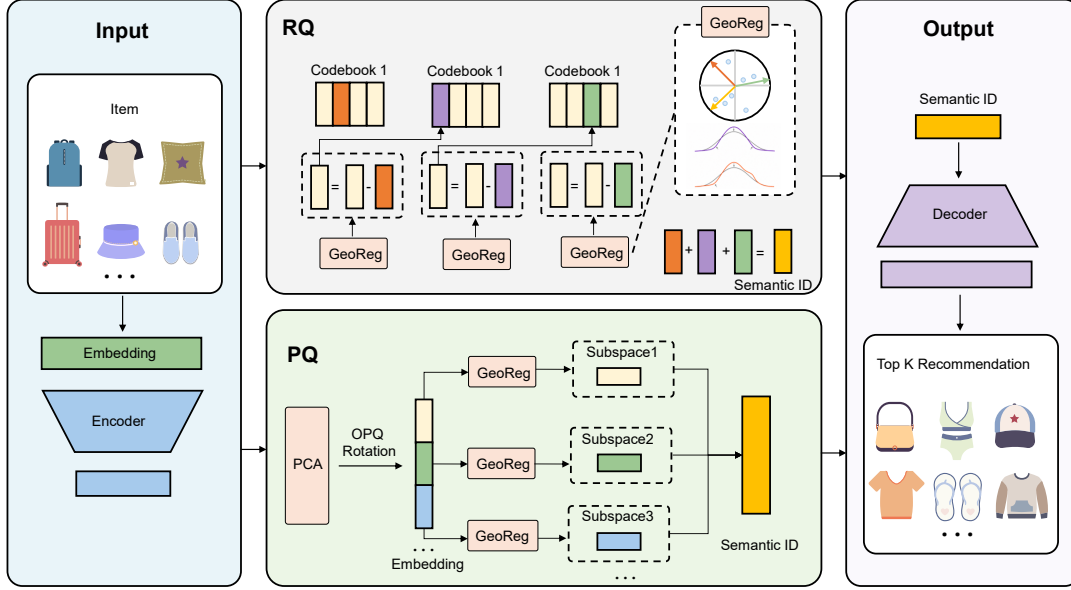


Figure 1: Overview of GeoSID.

distribution matching loss in a compact form:

$$\mathcal{L}_{\text{GeoReg}}(Z) = \sum_{m,k} \omega_k \left[(D_{m,k}^c)^2 + (D_{m,k}^s)^2 \right],$$

where $D_{m,k}^c$ and $D_{m,k}^s$ denote the real-part and imaginary-part deviations at the k -th frequency point along the m -th random projection direction, and ω_k is the corresponding frequency weight. This loss provides a distribution-level shaping signal through multiple projected views of the representation space. In GeoSID, it is applied by default to tokenizer-specific quantization spaces, such as residual spaces or OPQ/PQ blocks, rather than to the global embedding space. We use an isotropic Gaussian as the reference distribution because it discourages direction-dominated geometry in quantizer-facing spaces and has a closed-form characteristic function, making projected distribution matching efficient. We treat it as a practical surrogate for isotropic and quantization-friendly inputs. The distributional analysis in Section 5.1 provides empirical support for this choice.

3.3 GeoSID with RQ-based Tokenizers

RQ-based tokenizers encode an item representation through a sequence of residual codebooks. Each quantization level approximates the remaining residual from previous levels, and the final Semantic ID is formed by the code indices selected across all levels. Therefore, the quality of later

residual spaces is important: if residual representations become highly concentrated or geometrically degenerate, later codebooks may fail to capture fine-grained item differences.

To address this issue, GeoSID applies GeoReg at each residual-quantization layer rather than only on the original item embedding. For the ℓ -th RQ layer, we first perform level-wise normalization to reduce scale discrepancies across different residual spaces. We then apply GeoReg to the normalized residual batch, so that the distributional constraint is imposed on the representations directly consumed by each residual codebook.

The layer-wise GeoReg loss is defined as:

$$\mathcal{L}_{\text{GeoReg}}^{\text{layer}} = \sum_{\ell} \lambda_{\ell} \mathcal{L}_{\text{GeoReg}} \left(\tilde{\mathbf{R}}^{(\ell)} \right), \quad (1)$$

where $\tilde{\mathbf{R}}^{(\ell)}$ denotes the normalized residual representations at the ℓ -th RQ layer, and λ_{ℓ} controls the regularization strength for this layer.

The final objective for an RQ-based tokenizer is:

$$\mathcal{L}_{\text{GeoSID}}^{\text{RQ}} = \mathcal{L}_{\text{base}} + \mathcal{L}_{\text{GeoReg}}^{\text{layer}}. \quad (2)$$

where $\mathcal{L}_{\text{base}}$ denotes the original training objective of the tokenizer, such as reconstruction loss, commitment loss, or clustering objective. By aligning GeoReg with the residual decomposition structure, GeoSID encourages each RQ layer to produce a more quantization-friendly residual space while preserving the original tokenizer design.

3.4 GeoSID with OPQ-based Tokenizers

OPQ-based tokenizers follow a block-wise quantization structure. Taking RPG as an example, item representations are first transformed into an OPQ-rotated space and then partitioned into multiple PQ subspaces, where each subspace is independently quantized to produce one Semantic ID token. In the original RPG pipeline, the tokenizer mainly performs OPQ/PQ-based code assignment, while the downstream recommendation model is trained with the recommendation objective. GeoSID introduces an additional tokenizer-side shaping stage before code assignment, aiming to improve the geometry of the representations used for quantization without changing the downstream RPG model. In this OPQ-based variant, the shaped representation $\mathbf{Z}$ is directly optimized as an item-level free variable initialized from the OPQ-rotated representation, rather than generated by an adapter or neural network.

Let $\mathbf{X} \in \mathbb{R}^{B \times D}$ denote the OPQ-rotated item representations, and let $\mathbf{Z} \in \mathbb{R}^{B \times D}$ denote the shaped representations initialized from $\mathbf{X}$. To preserve the original semantic structure, we first define two preservation terms:

$$\mathcal{L}_{\text{mse}} = \text{MSE}(\mathbf{Z}, \mathbf{X}). \quad (3)$$

$$\mathcal{L}_{\text{cos}} = 1 - \cos(\mathbf{X}, \mathbf{Z}). \quad (4)$$

The overall preservation loss is:

$$\mathcal{L}_{\text{keep}} = r_w \mathcal{L}_{\text{mse}} + c_{kw} \mathcal{L}_{\text{cos}}. \quad (5)$$

The keep loss prevents distribution shaping from moving item representations to statistically regular but semantically distorted regions. The MSE term constrains the shaped representation to stay close to the original OPQ-rotated representation, and the cosine term preserves directional similarity.

For distribution shaping, GeoSID aligns the regularization granularity with the PQ block partition. Specifically, $\mathbf{Z}^{(g)}$ denotes the g -th OPQ-aligned PQ block of $\mathbf{Z}$. The block-wise GeoReg loss is defined as:

$$\mathcal{L}_{\text{GeoReg}}^{\text{block}} = \frac{1}{G} \sum_{g=1}^G \mathcal{L}_{\text{GeoReg}}(\mathbf{Z}^{(g)}). \quad (6)$$

This loss encourages each PQ subspace to form a more balanced projected distribution before independent code assignment, making the local representations more suitable for product quantization.

Dataset	#Users	#Items	#Interactions	Avg.Len
Sports	35,598	18,357	260,739	8.32
Beauty	22,363	12,101	176,139	8.87
Toys	19,412	11,924	148,185	8.63

Table 1: Statistics of the processed datasets.

Finally, the OPQ-based GeoSID objective is:

$$\mathcal{L}_{\text{GeoSID}}^{\text{OPQ}} = \mathcal{L}_{\text{keep}} + \lambda \mathcal{L}_{\text{GeoReg}}^{\text{block}}. \quad (7)$$

After optimization, the shaped representations $\mathbf{Z}$ are used for OPQ/PQ code assignment and Semantic ID generation. Therefore, GeoSID modifies only the tokenizer-side representation geometry, while keeping the OPQ/PQ tokenizer structure and the downstream RPG training framework unchanged.

4 Experiments

In this section, we conduct extensive experiments to evaluate the effectiveness and generalizability of GeoSID. Specifically, we aim to answer the following research questions:

RQ1: Is GeoSID a plug-and-play method that improves multiple Semantic ID tokenization backbones across datasets?

RQ2: Where should distribution regularization be applied for Semantic ID generation?

RQ3: Does GeoSID reshape pre-quantization item embeddings into a more isotropic and quantization-friendly distribution?

RQ4: How effective is GeoReg compared with alternative regularization objectives for tokenizer-side distribution shaping?

4.1 Experimental Setup

Datasets We conduct experiments on three categories from the Amazon Reviews dataset (McAuley et al., 2015): *Beauty*, *Sports and Outdoors*, and *Toys and Games*. These datasets are commonly used for benchmarking semantic ID-based generative recommendation methods. Following prior studies, we treat users’ historical reviews as implicit interactions and order them chronologically to construct user behavior sequences. Table 1 reports the dataset statistics. The interaction count excludes the last item held out for testing, whereas the average sequence length is computed from the complete user sequence, including the held-out item.

Evaluation Metrics. We evaluate recommendation performance using $\text{Recall}@K$ and $\text{NDCG}@K$, where $K \in \{5, 10\}$, following prior generative recommendation studies (Rajput et al., 2023). For each method, we select the checkpoint with the best validation performance and report the results on the test set.

Implementation Details. For RVQ, we follow the implementation and hyperparameter settings in GRID (Ju et al., 2025). For RPG, we follow its original OPQ-based semantic ID construction and parallel generative recommendation protocol (Hou et al., 2025a). To ensure a fair comparison, we keep the original data preprocessing, item tokenization configuration, model architecture, training strategy, and evaluation protocol unchanged for each backbone. GeoSID is only applied before semantic ID assignment, introducing no additional downstream parameters or inference-time cost.

Baselines. We evaluate GeoSID on two representative SID tokenization backbones: RVQ and RPG. RVQ represents the residual quantization family, where Semantic IDs are constructed through multiple residual codebooks. RPG represents the OPQ/PQ family, where long Semantic IDs are constructed with optimized product quantization and predicted in parallel.

Following the evaluation protocol of RPG, we compare against both conventional item ID-based sequential recommenders and Semantic ID-based generative recommenders. The item ID-based baselines include Caser (Tang and Wang, 2018), BERT4Rec (Sun et al., 2019), SASRec (Kang and McAuley, 2018), FDSA (Zhang et al., 2019), S3-Rec (Zhou et al., 2020) and HSTU (Zhai et al., 2024), covering convolutional, self-attentive and self-supervised sequential recommendation architectures. The Semantic ID-based baselines include LETTER-TIGER (Wang et al., 2024), VQ-Rec (Hou et al., 2023), TIGER (Rajput et al., 2023), and ActionPiece (Hou et al., 2025b). These methods evaluate different ways of incorporating discrete or semantic item representations into recommendation.

For each GeoSID backbone, we compare three variants: the original tokenizer, a global shaping variant that applies distribution shaping to the original embedding space, and our quantizer-aware local shaping variant. For RVQ, the local variant applies residual-level shaping; for RPG, it applies OPQ-aligned block-wise shaping. All GeoSID vari-

ants and their baselines are rerun with three random seeds; we report mean performance in the table.

4.2 Main Results

We compare GeoSID with item ID-based sequential recommenders, Semantic ID-based generative recommenders, and GeoSID variants built on RVQ and RPG. The results are shown in Table 2. Overall, Semantic ID-based methods generally outperform conventional item ID-based baselines, showing the benefit of using semantic token sequences as generation targets.

Effectiveness. GeoSID improves both representative SID backbones. On RVQ, GeoSID(Residual) consistently outperforms the original RVQ across datasets and metrics. On RPG, GeoSID(OPQ-Block) improves over the original RPG in all cases and achieves the strongest performance among GeoSID variants. These results show that GeoSID can be applied to both residual-quantization and OPQ/PQ-based Semantic ID pipelines.

Placement matters. The improvement depends strongly on where the shaping objective is applied. Under RVQ, GeoSID(Global) is unstable and often slightly worse than the backbone, whereas residual-level shaping consistently improves RVQ. Under RPG, OPQ-block shaping performs better than global shaping. Since the global and local variants use the same type of distribution loss and mainly differ in the regularized space, this comparison supports the quantizer-aware placement principle.

Plug-and-play property. GeoSID acts as a lightweight tokenizer-side enhancement and keeps the downstream generative recommender unchanged. Since it only reshapes the representations before Semantic ID assignment, it can be integrated into different SID tokenization backbones with minimal changes. The consistent gains on RVQ and RPG show that GeoSID provides a general pre-quantization improvement rather than a backbone-specific modification.

5 Further Analysis

5.1 Distributional Analysis

OPQ/PQ decomposes item representations into independently quantized subspaces. We therefore examine whether GeoSID improves the geometry of item embeddings in the OPQ-rotated chunk space used by RPG.

Figure 3 presents both local and global evidence. For the local view, we use the Beauty dataset and

Method	Beauty				Toys				Sports			
	R@5	R@10	N@5	N@10	R@5	R@10	N@5	N@10	R@5	R@10	N@5	N@10
<i>Item ID-based baselines</i>												
Caser	0.0205	0.0347	0.0131	0.0176	0.0166	0.0270	0.0107	0.0141	0.0116	0.0194	0.0072	0.0097
BERT4Rec	0.0203	0.0347	0.0124	0.0170	0.0116	0.0203	0.0071	0.0099	0.0115	0.0191	0.0075	0.0099
SASRec	0.0387	0.0605	0.0249	0.0318	0.0463	0.0675	0.0306	0.0374	0.0233	0.0350	0.0154	0.0192
FDSA	0.0267	0.0407	0.0163	0.0208	0.0228	0.0381	0.0140	0.0189	0.0182	0.0288	0.0122	0.0156
S ³ -Rec	0.0387	0.0647	0.0244	0.0327	0.0443	0.0700	0.0294	0.0376	0.0251	0.0385	0.0161	0.0204
HSTU	0.0469	0.0704	0.0314	0.0389	0.0433	0.0669	0.0281	0.0357	0.0258	0.0414	0.0165	0.0215
<i>Semantic ID-based baselines</i>												
LETTER-TIGER	0.0431	0.0672	0.0286	0.0364	0.0418	0.0625	0.0288	0.0373	<u>0.0309</u>	<u>0.0456</u>	<u>0.0205</u>	0.0292
VQ-Rec	0.0457	0.0664	0.0317	0.0383	0.0497	0.0737	0.0346	0.0423	0.0208	0.0300	0.0144	0.0173
TIGER	0.0454	0.0648	0.0321	0.0384	0.0521	0.0712	0.0371	0.0432	0.0264	0.0400	0.0181	0.0225
ActionPiece	0.0484	0.0762	0.0322	0.0412	0.0473	0.0725	0.0310	0.0391	0.0285	0.0441	0.0187	0.0238
<i>GeoSID variants and backbones</i>												
RVQ	0.0427	0.0641	0.0285	0.0351	0.0413	0.0612	0.0279	0.0335	0.0239	0.0368	0.0155	0.0198
RVQ + GeoSID(Global)	0.0425*	0.0637*	0.0282*	0.0350*	0.0410*	0.0603*	0.0272*	0.0334*	0.0236*	0.0364*	0.0152*	0.0193*
RVQ + GeoSID(Residual)	0.0456*	0.0665*	0.0305*	0.0370*	0.0431*	0.0629*	0.0287*	0.0350*	0.0256*	0.0385*	0.0166*	0.0208*
RPG	0.0522	0.0763	0.0361	0.0444	0.0556	0.0845	0.0372	0.0458	0.0293	0.0438	0.0199	0.0246
RPG + GeoSID(Global)	<u>0.0552*</u>	<u>0.0809*</u>	<u>0.0381*</u>	<u>0.0461*</u>	<u>0.0573*</u>	<u>0.0875*</u>	<u>0.0379*</u>	<u>0.0469*</u>	0.0282*	0.0430*	0.0194*	0.0242*
RPG + GeoSID(OPQ-Block)	0.0554*	0.0825*	0.0383*	0.0471*	0.0579*	0.0879*	0.0388*	0.0484*	0.0310*	0.0467*	0.0211*	<u>0.0263*</u>

Table 2: Performance comparison between GeoSID and other baselines across three datasets. The best score for each metric is shown in **bold**, and the second-best score is underlined. Statistical significance is assessed using a paired t-test across three matched random seeds against the corresponding backbone. An asterisk indicates statistical significance at $p < 0.05$.

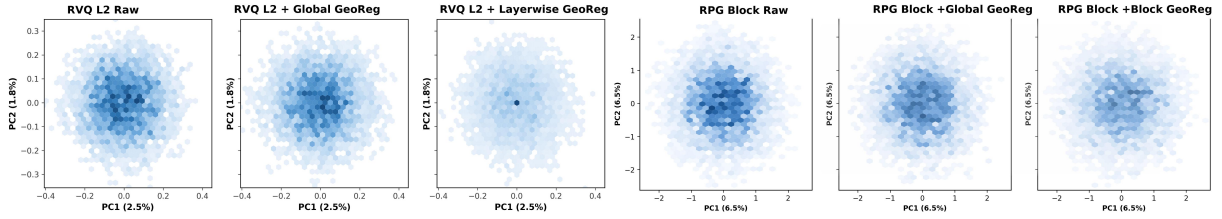


Figure 2: PCA density visualization of tokenizer-side quantization spaces on the Beauty dataset. The left three panels show layer-2 residual representations in RVQ, while the right three panels show OPQ-block representations in RPG.

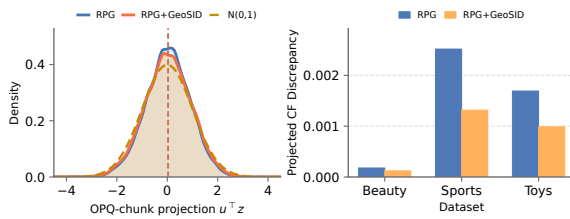


Figure 3: Distributional comparison of OPQ-chunk representations before and after GeoSID. The left panel illustrates the projected density on the Beauty dataset, and the right panel reports the projected CF discrepancy averaged over OPQ chunks across three datasets.

project a representative OPQ chunk onto a fixed random direction. The projected densities of RPG and RPG+GeoSID are normalized using RPG projection statistics, with $\mathcal{N}(0, 1)$ as the reference. For the global view, we compute the projected characteristic-function (CF) discrepancy over all

OPQ chunks. For each dataset and chunk, the score is averaged over 256 random unit projections, where a lower value indicates a closer match to the isotropic Gaussian target.

The local density plot shows that RPG produces a sharper central peak and a larger deviation from the Gaussian reference, whereas RPG+GeoSID yields a smoother density with better overlap and no obvious median shift. The global statistics show the same trend: GeoSID reduces the projected CF discrepancy by 32.1%, 47.7%, and 41.7% on Beauty, Sports, and Toys, respectively. These results suggest that GeoSID improves the OPQ-facing representation geometry by making chunk-level projected distributions more Gaussian-like under the projected CF criterion.

5.2 SID Quality Analysis

We further examine whether GeoSID improves the quality of the generated semantic IDs. For hierarchical SID representations, excessive concentration on a small subset of codes or severe prefix-level collisions may weaken item separability and affect the stability of downstream generative retrieval. We therefore analyze SID quality from two perspectives: code usage balance and prefix-level separability.

We compare RPG and RPG+GeoSID on the Beauty dataset with 12,101 items and 32 SID layers. We report code-level metrics, including Code Gini, Max/Mean, Top-k Share, and Effective Ratio, together with prefix-level metrics, including Prefix Collision@k and Prefix Entropy@k. Prefix Collision@k measures how often different items share the same length- k SID prefix, where lower values indicate better separability; Prefix Entropy@k measures the dispersion of prefix buckets, where higher values indicate a more balanced hierarchical partition.

As shown in Table 3, GeoSID improves the structural quality of Semantic IDs from both code usage and prefix separability perspectives. For code usage balance, Code Gini and Max/Mean decrease by 2.87% and 2.65%, respectively, indicating reduced code concentration. Other code-usage indicators show the same direction: the Top-10% and Top-20% usage shares decrease, while the Effective Ratio slightly increases, suggesting broader utilization of the code space.

The improvement is more evident at the prefix level. GeoSID reduces Prefix Collision@2 and Prefix Collision@3 by 2.89% and 7.87%, respectively, indicating that fewer items collapse into the same short SID prefixes. Prefix Entropy@2 and Prefix Entropy@3 also increase slightly, suggesting a more dispersed prefix distribution. Overall, these results show that GeoSID improves both continuous representation geometry and the structural quality of the resulting discrete Semantic IDs.

5.3 Comparison with Alternative Balancing Methods

We compare GeoSID with two direct code-balancing objectives under the same RVQ and RPG backbones. As shown in Table 4, GeoSID achieves the best Recall@10 and NDCG@10 under both tokenizer families. Under RVQ, GeoSID improves Recall@10 and NDCG@10 by 3.74%

Metric	RPG	+GeoSID	Change
Code Gini ↓	0.1500	0.1457	-2.87%
Max/Mean ↓	1.9443	1.8927	-2.65%
Top-10% Share ↓	0.1490	0.1478	-0.81%
Top-20% Share ↓	0.2771	0.2755	-0.56%
Eff. Ratio ↑	0.9600	0.9626	+0.27%
Prefix Col.@2 ↓	0.3374	0.3277	-2.89%
Prefix Col.@3 ↓	0.1258	0.1159	-7.87%
Prefix Ent.@2 ↑	13.0507	13.0737	+0.18%
Prefix Ent.@3 ↑	13.3708	13.3898	+0.14%

Table 3: SID quality comparison between RPG and GeoSID(OPQ-Block) on the Beauty dataset.

Backbone	Method	R@10	N@10
RVQ	Original	0.0641	0.0351
RVQ	+ Code-usage entropy	0.0651	0.0355
RVQ	+ Balanced assignment	0.0646	0.0352
RVQ	+ GeoSID (Residual)	0.0665	0.0370
RPG	Original	0.0763	0.0444
RPG	+ Code-usage entropy	0.0808	0.0459
RPG	+ Balanced assignment	0.0796	0.0457
RPG	+ GeoSID (OPQ-Block)	0.0825	0.0471

Table 4: Direct code-balancing comparisons on the Beauty dataset under the RVQ and RPG backbones.

and 5.41%, respectively. Under RPG, the corresponding improvements are 8.13% and 6.08%. In comparison, directly regularizing code usage or assignment balance yields smaller gains. These controlled results indicate that shaping the continuous quantizer-facing geometry provides benefits that are not reproduced by directly balancing discrete assignments alone.

6 Conclusion

In this paper, we revisited Semantic ID generation from the perspective of pre-quantization representation geometry. We showed that distribution mismatch in quantizer-facing spaces is closely related to code usage, prefix-level separability, and downstream recommendation performance. To address this issue, we proposed GeoSID, a tokenizer-side geometric shaping framework that applies GeoReg to residual spaces in RVQ and OPQ-aligned blocks in RPG without changing the downstream generative recommender. Experiments and analyses demonstrate that GeoSID improves both recommendation performance and Semantic ID quality, suggesting that representation geometry before quantization is an important factor for effective

Semantic ID generation.

Limitations

This work has several limitations. First, GeoSID is evaluated on three public Amazon Reviews categories and two representative SID tokenizer backbones, so its effectiveness in larger industrial catalogs, streaming recommendation scenarios, and cross-domain or cold-start settings remains to be further examined. Second, although we keep the downstream recommender, tokenizer configuration, and evaluation protocol unchanged within each backbone, the results may still be affected by hyperparameter selection and shaping optimization settings. Finally, our analysis mainly relies on recommendation metrics, projected CF discrepancy, code usage balance, and prefix-level SID quality, which may not fully capture semantic preservation or user-level utility.

Acknowledgments

We thank the anonymous reviewers and area chairs for their valuable feedback and suggestions. This work is partially supported by the Research Travel Grant of the Base of Red Bird MPhil (RBM) at The Hong Kong University of Science and Technology (Guangzhou). We would like to thank our RBM Project Supervisor Ning Li for the academic support.

References

- Randall Balestriero and Yann LeCun. 2025. LeJEPa: Provable and scalable self-supervised learning without the heuristics. *arXiv preprint arXiv:2511.08544*.
- Yongjian Chen, Tao Guan, and Cheng Wang. 2010. Approximate nearest neighbor search by residual vector quantization. *Sensors*, 10(12):11259–11273.
- Kawin Ethayarajh. 2019. How contextual are contextualized word representations? comparing the geometry of BERT, ELMo, and GPT-2 embeddings. 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)*, pages 55–65.
- Tiezhen Ge, Kaiming He, Qifa Ke, and Jian Sun. 2013. Optimized product quantization for approximate nearest neighbor search. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2946–2953.
- 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*, pages 1162–1171.
- Yupeng Hou, Jiacheng Li, Ashley Shin, Jinsung Jeon, Abhishek Santhanam, Wei Shao, Kaveh Hassani, Ning Yao, and Julian McAuley. 2025a. Generating long Semantic IDs in parallel for recommendation. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pages 956–966.
- Yupeng Hou, Jianmo Ni, Zhankui He, Naveen Sachdeva, Wang-Cheng Kang, Ed H. Chi, Julian McAuley, and Derek Zhiyuan Cheng. 2025b. ActionPiece: Contextually tokenizing action sequences for generative recommendation. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 24004–24024. PMLR.
- Herve Jegou, Matthijs Douze, and Cordelia Schmid. 2011. Product quantization for nearest neighbor search. *IEEE transactions on pattern analysis and machine intelligence*, 33(1):117–128.
- Clark Mingxuan Ju, Liam Collins, Leonardo Neves, Bhuvish Kumar, Louis Yufeng Wang, Tong Zhao, and Neil Shah. 2025. Generative recommendation with Semantic IDs: A practitioner’s handbook. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management*, pages 6420–6425.
- Wang-Cheng Kang and Julian McAuley. 2018. Self-attentive sequential recommendation. In *2018 IEEE international conference on data mining (ICDM)*, pages 197–206. IEEE.
- 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*, pages 11523–11532.
- Bohan Li, Hao Zhou, Junxian He, Mingxuan Wang, Yiming Yang, and Lei Li. 2020. On the sentence embeddings from pre-trained language models. In *Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP)*, pages 9119–9130.
- Enze Liu, Bowen Zheng, Cheng Ling, Lantao Hu, Han Li, and Wayne Xin Zhao. 2025. Generative recommender with end-to-end learnable item tokenization. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 729–739.
- Han Liu, Yinwei Wei, Xuemeng Song, Weili Guan, Yuan-Fang Li, and Liqiang Nie. 2024. MMGRec: Multimodal generative recommendation with transformer model. *arXiv preprint arXiv:2404.16555*.

- 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*, pages 43–52.
- Gustavo Penha, Edoardo D’Amico, Marco De Nadai, Enrico Palumbo, Alexandre Tamborrino, Ali Vardasbi, Max Lefarov, Shawn Lin, Timothy Heath, Francesco Fabbri, and Hugues Bouchard. 2025. Semantic IDs for joint generative search and recommendation. In *Proceedings of the Nineteenth ACM Conference on Recommender Systems*, pages 1296–1301.
- Shashank Rajput, Nikhil Mehta, Anima Singh, Raghunandan Hulikal Keshavan, Trung Vu, Lukasz Heldt, Lichan Hong, Yi Tay, Vinh Tran, Jonah Samost, Maciej Kula, Ed Chi, and Maheswaran Sathiamoorthy. 2023. Recommender systems with generative retrieval. *Advances in Neural Information Processing Systems*, 36:10299–10315.
- Jianlin Su, Jiarun Cao, Weijie Liu, and Yangyiwen Ou. 2021. Whitening sentence representations for better semantics and faster retrieval. *arXiv preprint arXiv:2103.15316*.
- 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*, pages 1441–1450.
- 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*, pages 565–573.
- Wenjie Wang, Honghui Bao, Xinyu Lin, Jizhi Zhang, Yongqi Li, Fuli Feng, See-Kiong Ng, and Tat-Seng Chua. 2024. Learnable item tokenization for generative recommendation. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pages 2400–2409.
- Yibiao Wei, Jie Zou, Pengfei Zhang, Xiao Ao, Weikang Guo, Zeyu Ma, and Yang Yang. 2026. CARD: Non-Uniform Quantization of Visual Semantic Unit for Generative Recommendation. *arXiv preprint arXiv:2604.26427*.
- Jiaqi Zhai, Lucy Liao, Xing Liu, Yueming Wang, Rui Li, Xuan Cao, Leon Gao, Zhaojie Gong, Fangda Gu, Jiayuan He, Yinghai Lu, and Yu Shi. 2024. Actions speak louder than words: Trillion-parameter sequential transducers for generative recommendations. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 58484–58509. PMLR.
- Tingting Zhang, Pengpeng Zhao, Yanchi Liu, Victor S. Sheng, Jiajie Xu, Deqing Wang, Guanfeng Liu, and Xiaofang Zhou. 2019. Feature-level deeper self-attention network for sequential recommendation. In *IJCAI*, pages 4320–4326.
- 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*, pages 1893–1902.
- Jieming Zhu, Mengqun Jin, Qijiong Liu, Zexuan Qiu, Zhenhua Dong, and Xiu Li. 2024. CoST: Contrastive quantization based semantic tokenization for generative recommendation. In *Proceedings of the 18th ACM Conference on Recommender Systems*, pages 969–974.

A Additional Experimental Setup

We provide additional details on the experimental protocol used in our GeoSID experiments. A key principle of our evaluation is *backbone-faithful comparison*: GeoSID does not impose a unified preprocessing pipeline across different SID backbones. For each backbone, we follow its original data preprocessing, tokenization configuration, generation model, training strategy, and evaluation protocol. GeoSID is inserted only as a tokenizer-side representation shaping step before Semantic ID assignment.

A.1 Datasets

We conduct experiments on three categories from the Amazon Reviews dataset: *Beauty*, *Sports and Outdoors*, and *Toys and Games*. We use all public datasets and backbone implementations only for academic benchmarking and follow their intended research-use settings. The Amazon Reviews dataset records user feedback on products from the Amazon e-commerce platform and provides both user-item interaction histories and item metadata. These properties make it a standard benchmark for sequential recommendation and Semantic ID-based generative recommendation, where item identifiers are constructed from item-side semantic representations.

The three categories cover different product domains. *Beauty* contains beauty and personal care products with fine-grained semantic distinctions such as product type, brand, and usage. *Sports and Outdoors* includes sports, outdoor, and fitness products, forming a larger and more heterogeneous item catalog. *Toys and Games* contains toys, games,

and entertainment products, where item semantics vary across themes, age groups, and usage scenarios. The diversity across these categories allows us to examine whether GeoSID is effective under different item distributions and catalog structures.

A.2 Dataset Processing

We conduct experiments on three categories from the Amazon Reviews dataset: *Beauty*, *Sports and Outdoors*, and *Toys and Games*. Following prior generative recommendation studies, user reviews are treated as implicit feedback and each user’s historical interactions are ordered chronologically to construct behavior sequences. Since GeoSID is evaluated on two different SID backbone families, we follow the original data processing protocol of each backbone rather than forcing a single preprocessing pipeline.

A.3 Implementation Details

We provide additional implementation details for reproducing GeoSID under two representative Semantic ID quantization paradigms. For residual quantization (RQ), we use the RVQ tokenizer in GRID as the representative backbone; for product quantization (PQ), we use the OPQ tokenizer in RPG as the representative backbone. Across both backbones, we keep the original data processing, tokenizer structure, downstream generative recommender, decoding strategy, and evaluation protocol unchanged. GeoSID only changes where GeoReg is applied during Semantic ID construction.

GeoReg settings. GeoReg is implemented with projected characteristic-function matching. For each loss forward pass, projection directions are sampled from a Gaussian distribution and normalized to unit norm. The frequency grid uses 17 linearly spaced knots with $t_{\max} = 3.0$. Frequency weights follow a trapezoidal integration approximation multiplied by the Gaussian characteristic-function term, rather than uniform weighting. We use $n_{\text{proj}} = 256$ by default and optionally select $n_{\text{proj}} = 512$ according to validation performance. The GeoReg strength λ is selected on the validation split.

GRID/RVQ backbone. For the RQ paradigm, we follow the original GRID pipeline and use its RVQ tokenizer as the backbone. The RVQ baseline is obtained by disabling GeoReg, i.e., setting the GeoReg loss weight to zero. Both RVQ +

GeoSID(Global) and RVQ + GeoSID(Residual) inject GeoReg into RVQ quantizer training. The former regularizes the global item embedding scope, while the latter regularizes the residual-level representations consumed by each residual codebook.

RPG/OPQ backbone. For the PQ paradigm, we follow the original RPG pipeline and use its OPQ tokenizer as the backbone. RPG encodes item textual metadata into semantic embeddings, applies PCA and OPQ rotation, and then performs PQ code assignment to construct long unordered Semantic IDs. Both RPG + GeoSID(Global) and RPG + GeoSID(OPQ-Block) are implemented in the tokenizer-side pre-quantization shaping stage before OPQ code assignment. The global variant applies GeoReg to the whole pre-quantization representation space, while the OPQ-block variant applies GeoReg separately within each OPQ-aligned PQ block. After shaping, both variants use the original RPG PQ encoding, parallel training objective, and graph-constrained decoding.

In RPG + GeoSID, shaped representations are initialized from the OPQ-rotated item representations and optimized as item-level free variables. We use Adam for shaping optimization with 100 shaping epochs. The preservation loss uses an MSE preservation weight of 2.0 and a cosine preservation weight of 1.0 in the main configuration. The GeoReg strength λ and shaping learning rate are selected according to validation performance.

Fairness and Controlled Comparison. Within each backbone, the global and quantizer-aware variants use the same GeoReg configuration and validation-based selection protocol; they differ only in the space where GeoReg is computed. All main results are evaluated with Recall@5, Recall@10, NDCG@5, and NDCG@10 under the original evaluation protocol of the corresponding backbone.

B Method Pseudocode

We provide implementation-level pseudocode for GeoSID under two representative Semantic ID tokenization backbones. For RVQ/RQ-VAE, GeoSID is applied to residual-level representations used by the residual quantizer. For RPG, GeoSID is applied after OPQ rotation and before PQ code assignment. The downstream generative recommender and decoding protocol remain unchanged.

Algorithm 1 RVQ + GeoSID (Residual)

Require: Item embeddings X ; an L -level residual quantizer Q ; GeoSID hyperparameters λ , n_{proj} , $\mathcal{T}$, and t_{max}
Ensure: Semantic IDs and recommendation predictions
function RESIDUALGEOSID(X, Q)
 Initialize tokenizer Q with L residual levels
 $\theta \leftarrow (n_{\text{proj}}, \mathcal{T}, t_{\text{max}})$
 for each training step **do**
 Sample mini-batch X_{batch}
 $(\mathbf{c}, \{R_\ell\}_{\ell=1}^L, \hat{X}) \leftarrow Q(X_{\text{batch}})$
 $\mathcal{L}_{\text{quant}} \leftarrow \text{QuantizationLoss}(\hat{X}, X_{\text{batch}}, \mathbf{c})$
 $\mathcal{L}_{\text{GeoReg}} \leftarrow \sum_{\ell=1}^L \mathcal{L}_{\text{CF}}(R_\ell; \theta)$
 $\mathcal{L}_{\text{total}} \leftarrow \mathcal{L}_{\text{quant}} + \lambda \mathcal{L}_{\text{GeoReg}}$
 Update Q by minimizing $\mathcal{L}_{\text{total}}$
 end for
 SID $\leftarrow \text{ExportSID}(Q, X)$
 Train the downstream recommender with SID
 Evaluate using the original backbone protocol
 return Semantic IDs and predictions
end function

Algorithm 2 RPG + GeoSID (OPQ-Block)

Require: Item embeddings X ; training mask $\mathcal{M}$; OPQ/PQ configuration; GeoSID hyperparameters
Ensure: Semantic IDs and recommendation predictions
function OPQBLOCKGEOSID($X, \mathcal{M}$)
 Encode item information into representations X
 Optionally apply PCA following RPG
 Learn the OPQ rotation R using $X_{\mathcal{M}}$
 $X^{\text{rot}} \leftarrow X R$
 $Z \leftarrow X_{\mathcal{M}}^{\text{rot}}$
 $\theta \leftarrow (n_{\text{proj}}, \mathcal{T}, t_{\text{max}})$
 for $e = 1, \dots, E_{\text{shape}}$ **do**
 Split Z into blocks $\{Z_b\}_{b=1}^{B_q}$
 $\mathcal{L}_{\text{GeoReg}} \leftarrow \sum_{b=1}^{B_q} \mathcal{L}_{\text{CF}}(Z_b; \theta)$
 $\mathcal{L}_{\text{keep}} \leftarrow \text{KeepLoss}(Z, X_{\mathcal{M}}^{\text{rot}})$
 $\mathcal{L}_{\text{total}} \leftarrow \mathcal{L}_{\text{keep}} + \lambda \mathcal{L}_{\text{GeoReg}}$
 Update Z using η_{shape}
 end for
 Replace $X_{\mathcal{M}}^{\text{rot}}$ with Z
 SID $\leftarrow \text{PQEncode}(X^{\text{rot}})$
 Train and evaluate RPG using its original protocol
 return Semantic IDs and predictions
end function

C Hyperparameter Sensitivity

We analyze the sensitivity of RPG + GeoSID(OPQ-Block) on the Beauty dataset by varying one GeoSID-specific hyperparameter at a time while keeping all other settings fixed. Figure 4 reports Recall@10 and NDCG@10 under different values of the GeoReg strength λ , the cosine preservation weight c_{kw} .

GeoReg strength λ . The best performance is obtained around $\lambda = 0.0015$. Smaller values provide weaker distribution shaping, while larger values lead to a slight performance drop, suggesting that

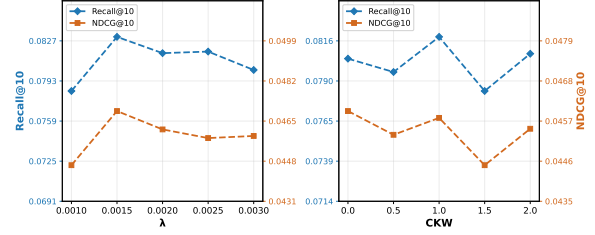


Figure 4: Parameter sensitivity analysis of RPG + GeoSID(OPQ-Block) on the Beauty dataset.

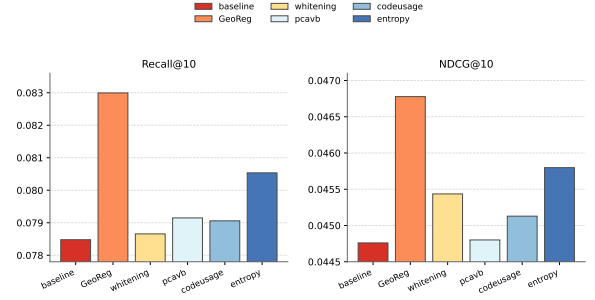


Figure 5: Comparison of different tokenizer-side shaping objectives on the Beauty dataset. GeoReg achieves better SID structural quality than alternative objectives under the same RPG setting.

overly strong regularization may disturb the original representation structure.

Cosine preservation weight c_{kw} . The model is relatively stable across moderate values of c_{kw} . Performance peaks around $c_{kw} = 1.0$ and decreases when the preservation constraint becomes too strong, indicating a trade-off between semantic preservation and geometric reshaping.

D Additional Evidence for GeoReg

We provide additional evidence to examine whether the advantage of GeoReg comes from its specific distribution-shaping objective rather than from simply adding an auxiliary regularizer. All compared methods are evaluated under the same RPG tokenizer-side setting on the Beauty dataset. The downstream recommender, OPQ/PQ tokenizer configuration, and evaluation protocol are kept unchanged, and only the shaping objective before Semantic ID assignment is replaced.

Figure 5 and Table 5 report representative SID quality metrics for different tokenizer-side shaping objectives. For code usage balance, GeoReg achieves the lowest Code Gini, the lowest Max/Mean, and the highest Effective Ratio among all compared methods. Compared with the RPG baseline, GeoReg reduces Code Gini from 0.1500

to 0.1457 and Max/Mean from 1.9443 to 1.8927, while increasing the Effective Ratio from 0.9600 to 0.9626. These results indicate that GeoReg reduces code concentration more consistently than whitening, PCA variance balancing, code-usage regularization, and entropy regularization.

For prefix-level separability, GeoReg obtains the lowest Prefix Collision@2 among all compared methods. This indicates that fewer items share the same short SID prefix after GeoReg shaping, which is important for generative recommendation because early SID tokens directly affect item discrimination during sequence generation and decoding. In contrast, whitening, PCA variance balancing, code-usage regularization, and entropy regularization reduce prefix collision only partially, while showing weaker or even negative effects on code usage balance.

Overall, GeoReg provides a more balanced improvement across code usage and prefix-level separability. It is the only objective that simultaneously improves Code Gini, Max/Mean, Effective Ratio, and Prefix Collision@2 over the RPG baseline. This supports our design choice of using GeoReg as the tokenizer-side shaping objective for Semantic ID generation.

Method	Gini↓	M/M↓	Eff.↑	Col.@2↓
RPG	0.1500	1.9443	0.9600	0.3374
+White.	0.1518	1.9324	0.9596	0.3354
+PCA-VB	0.1520	1.9377	0.9595	0.3359
+CodeUse	0.1510	1.8954	0.9598	0.3353
+Entropy	0.1513	1.8954	0.9596	0.3364
+GeoReg	0.1457	1.8927	0.9626	0.3277

Table 5: SID quality comparison of different tokenizer-side shaping objectives on the Beauty dataset. Gini, M/M, Eff., and Col.@2 denote Code Gini, Max/Mean, Effective Ratio, and Prefix Collision@2.