

VARG: Value-Aware and Ranking-Aligned Generative Retrieval for Dynamic E-commerce Search

Xiaopeng Chu
University of Science and Technology
of China
Hefei, Anhui, China
cxphhh@mail.ustc.edu.cn

Jianbo Zhu
Taobao & Tmall Group of Alibaba
Hangzhou, China
zhu Jianbo.zjb@taobao.com

Mingmin Jin
Taobao & Tmall Group of Alibaba
Hangzhou, China
jimmy.jmm@taobao.com

Jing Wang
Taobao & Tmall Group of Alibaba
Hangzhou, China
jing.wangj1@taobao.com

Xing Fang^{*}
Nankai University
Tianjin, China
Taobao & Tmall Group of Alibaba
Hangzhou, China
fangxing.fx@taobao.com

Wenyi Zhang
University of Science and Technology
of China
Hefei, Anhui, China
wenyizha@ustc.edu.cn

Abstract

Integrating recall and pre-ranking in e-commerce search requires candidate generation to account for relevance, personalization, and business value before final ranking. To this end, we present VARG, a generative retrieval system for Tmall App search that directly admits generated item candidates to the existing final ranker. VARG-ID constructs semantic prefixes using RQ-VAE, enhances search relevance through bidirectional query-item contrastive learning, and combines these prefixes with a value-ordered third token to provide fine-grained item addresses and a business-value prior. Three-stage supervised fine-tuning progressively learns item-to-identifier mappings, query-semantic retrieval, and personalized retrieval. Personalized model training combines value-aware and hierarchy-aligned supervision with expanded user context, and uses local ordinal supervision (LO-SFT) to learn the local within-cluster ordering encoded by the third token. Prefix-GRPO combines gated rewards based on output legality, user behavior, ranker advantage, and search relevance with prefix-aware token weighting to align candidate generation with business value and ranking objectives. Coordinated daily product and model updates preserve existing item addresses while incorporating new products and behavioral feedback. Offline experiments on tens of millions of products validate identifier stability and demonstrate gains in retrieval quality and head-level value recall from SFT strategies and Prefix-GRPO over their respective baselines. In a 14-day online A/B test covering 20% of search traffic, VARG directly admits generated candidates to the final ranker and improves GMV by 1.45%,

per-user IPV by 0.22%, and PCTR by 0.31%. Online shopping-guide query evaluations further show that VARG maintains competitive relevance with a smaller candidate quota.

CCS Concepts

• **Information systems** → **Information retrieval; Retrieval models and ranking.**

Keywords

Generative Retrieval, E-commerce Search, Recall and Pre-ranking Integration, Semantic Identifiers, Reinforcement Learning

ACM Reference Format:

Xiaopeng Chu, Jianbo Zhu, Mingmin Jin, Jing Wang, Xing Fang, and Wenyi Zhang. 2018. VARG: Value-Aware and Ranking-Aligned Generative Retrieval for Dynamic E-commerce Search. In *Proceedings of [Conference Name]* (Conference acronym 'XX), June 03–05, 2018, Woodstock, NY. ACM, New York, NY, USA, 11 pages. <https://doi.org/XXXXXXX.XXXXXXX>

1 Introduction

E-commerce search matches users' shopping needs to a massive set of products updated daily through a recall-pre-rank-rank cascade [35, 2]. Lexical retrieval [20], dense encoders [9], efficient nearest-neighbor indexes [16], and late interaction [10] offer complementary relevance and efficiency trade-offs. Recall and pre-ranking jointly perform candidate selection, accounting for both item relevance and business value. Since downstream rankers cannot recover items omitted from the candidate set, candidate selection should retain high-quality items under strict latency and resource budgets.

Generative retrieval formulates candidate selection as identifier generation directly from context [12], with related work extending from document retrieval [25, 29] to hierarchical item IDs [19]. Recent industrial systems further explore unified retrieval and ranking [6, 33] and intent-enhanced search [2]. This formulation jointly models query semantics, user preferences, and behavioral context, expressing item selection and candidate priorities through generation probabilities.

^{*}Corresponding author.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
Conference acronym 'XX, Woodstock, NY
© 2018 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 978-1-4503-XXXX-X/2018/06
<https://doi.org/XXXXXXX.XXXXXXX>

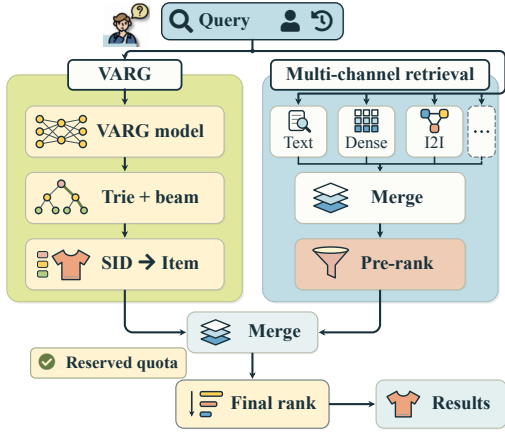


Figure 1: Online integration of VARG. Conventional multi-channel candidates undergo pre-ranking; VARG generates candidates using Trie-constrained beam search and SID-to-item mapping, bypassing pre-ranking via a reserved quota. Both streams are merged and deduplicated before final ranking.

Our setting is *recall-and-pre-ranking integration*: an additional generative channel forwards items directly to the existing final ranker rather than replacing the search cascade (Figure 1). Bypassing item-level pre-ranking requires generated candidates to provide relevance, personalization, and business value within the reserved quota. The system also needs to accommodate ongoing daily product and model updates.

This goal raises three connected challenges.

Item-level selection and addressing stability. Cluster-level identifiers improve item coverage during generation [35], but shared identifiers limit the model’s ability to distinguish individual items within a cluster. When item-level pre-ranking is bypassed, less relevant or lower-value items may occupy reserved candidate slots. Business-ranked identifiers provide finer addresses and value priors [15], but noisy conversion estimates and daily reordering can change existing item addresses and disrupt the model’s learned addressing relationships. Learning new products also risks forgetting existing knowledge [17].

Supervised learning for contextual retrieval. The model needs to learn item-to-identifier mappings and select items according to query intent and user preferences. The challenge is to extend item memorization into query-semantic and personalized retrieval while ensuring that supervision reflects the semantic hierarchy and within-cluster ordering encoded by the identifiers [28, 2].

Business-value and ranking alignment. An item’s conversion propensity varies across queries and users, while learning only from observed target items provides limited guidance for comparing the quality and value of other candidates. Policy optimization offers a way to further adjust candidate priorities [6, 35], but sparse click and purchase matches poorly distinguish unobserved candidates. Training feedback should therefore also cover candidate relevance and downstream ranking performance while preventing invalid outputs from receiving positive credit.

To address these challenges, we present VARG (Figure 2) with

the following main contributions:

- We introduce *VARG-ID*, a hierarchical item identifier with collision-free initial construction. Bidirectional Q2I contrastive learning injects search relevance into its semantic prefix, while stable empirical-Bayes conversion-rate ordering encodes business value in its third token. We further establish stable daily-refresh mechanisms for both items and the retrieval model.
- We build a three-stage SFT pipeline that progresses from item memorization to query-semantic retrieval and personalized retrieval. Personalized training combines Q2I value weighting, SID level-wise loss, and expanded user context, and uses local ordinal supervision (LO-SFT) to learn the within-cluster ordering encoded by the third token. We also explore lightweight parallel decoding to reduce inference cost.
- We develop Prefix-GRPO with gated multi-source rewards over SID legality, user behavior, ranker advantage, and search relevance, providing denser reward signals for policy optimization. Combined with prefix-aware token weighting, these rewards align candidate priorities with business value and downstream ranking objectives.

Offline experiments evaluate identifier stability and the effects of SFT and Prefix-GRPO on retrieval quality and value recall. A 14-day online A/B test validates the deployed recall-and-pre-ranking system, yielding a 1.45% GMV gain.

2 Related Work

2.1 Generative Retrieval and Training

Generative retrieval uses different forms of identifiers: GENRE generates entity names [5], DSI and NCI generate document identifiers [25, 29], and SEAL generates indexed document substrings [1]. Wu et al. [30] connect generative retrieval to multi-vector dense retrieval through a shared relevance-scoring framework. TIGER [19] extends this paradigm to recommendation with hierarchical semantic IDs.

For generative retrieval training, P5 [8] unifies recommendation tasks through text-to-text learning and personalized prompts, while LC-Rec [32] aligns language and collaborative semantics around quantized item indices. Industrial systems further explore unified retrieval and ranking [6, 33]: CQ-SID [35] progressively learns item, query, and personalized mappings, OneSearch-V2 [2] models query intent through keyword-based reasoning and self-distillation, and TBGRecall [14] combines multi-session modeling with incremental training for e-commerce recommendation. These studies provide a foundation for item representation, staged training, and context modeling. For integrated recall and pre-ranking, a further consideration is whether the generative model can select items that balance relevance, personalization, and business value within a limited candidate quota.

2.2 Semantic IDs and Dynamic Indexing

Vector and residual quantization provide a discrete representation basis for hierarchical identifiers [26, 11, 19]. Subsequent studies further integrate identifier learning with retrieval objectives: GenRet [23] jointly learns document tokenization, reconstruction, and retrieval; IDGenRec [24] learns textual IDs with a recommender;

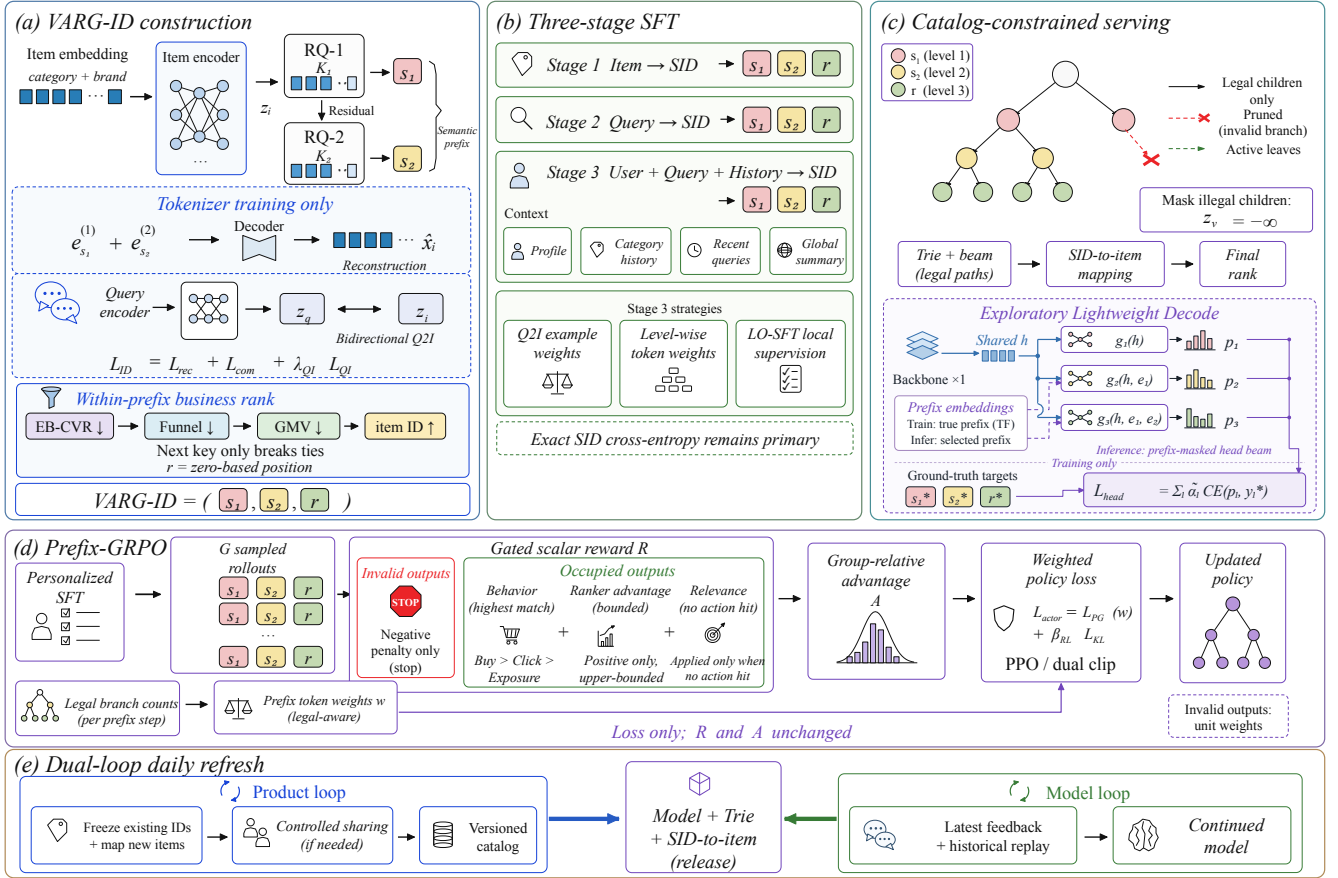


Figure 2: Overview of VARG. (a) VARG-ID with query-aligned semantic prefixes and value-ordered item addresses. (b) Three-stage SFT with context expansion and supervision strategies. (c) Trie-constrained serving and exploratory lightweight decoding. (d) Prefix-GRPO with gated multi-source rewards and prefix-aware token weighting. (e) Coordinated daily product and model updates.

and LETTER [28] combines semantic, collaborative, and diversity regularization with ranking-guided generation. D²-DocID [4] uses contrastive learning to build descriptive identifiers that distinguish similar documents, while FORGE [7] studies industrial SID design using intrinsic evaluation measures.

In search, CQ-SID [35] injects query relevance while retaining cluster-level aggregation, and CRID [15] uses intra-cluster business ranking for item-level addressing and incremental updates. For dynamic corpora, DSI++ [17] uses generative replay to mitigate forgetting, while CLEVER [3] combines incremental product quantization with memory-augmented learning. These studies involve two forms of stability: preserving model knowledge and maintaining stable item identifiers. Under daily product updates, identifier design should balance fine-grained addressing, business value, and decoding efficiency while reducing the disruption to existing addressing relationships caused by changes in item statistics.

2.3 Policy Optimization for Retrieval

Ranking-oriented training further considers the quality and ordering of generated candidates. LTRGR [13] adds a ranking-oriented

training phase without extra inference stages, while relevance-feedback RL [34] aligns token-level generation with document-level relevance. PPO, DPO, and GRPO provide general frameworks for policy or preference optimization [21, 18, 22].

In recommendation, OneRec [6] uses reward-model-derived preference pairs for iterative DPO, and OneRec-V2 [33] incorporates real interaction feedback. In search, EG-GRPO [35] injects expert SIDs to alleviate sparse positive feedback, while OneSearch-V2 [2] combines behavioral and relevance rewards with token-position marginal advantages. These methods demonstrate the value of additional feedback for improving generative retrieval. For hierarchical SID generation, we further study how to differentiate legal candidates without behavioral feedback, prevent invalid outputs from receiving positive credit, and distribute supervision across decisions at different levels to improve ranking quality within a limited candidate quota.

3 Method

Problem setting. Given a query q , user profile u , and history h , a Transformer [27] conditioned on context $x = (q, u, h)$ autoregress-

sively generates a three-token item identifier $y = (s_1, s_2, r)$. At inference, Trie-constrained beam search produces a ranked candidate list $\mathcal{Y}_K$ of K SIDs, which are expanded into item candidates through the SID-to-item map and forwarded directly to the existing final ranker as an additional retrieval channel. Under a fixed decoding budget and final-ranking candidate quota, the model should generate valid SIDs and select items that are relevant, personalized, and commercially valuable. The item map and model should also accommodate daily product updates and new user feedback.

3.1 VARG-ID: Semantics Followed by Stable Value

A VARG-ID consists of a semantic prefix and a value-ordered token: the query-aligned prefix supports semantic routing, while the third token provides fine-grained within-prefix item addresses and a business-value prior.

Semantic-prefix construction. The item embedding $x_i \in \mathbb{R}^{d_x}$ incorporates category and brand attributes and is mapped to $z_i \in \mathbb{R}^{d_z}$ by the item encoder E_I . The query encoder E_Q , used only during tokenizer training, maps the paired search query to $z_q \in \mathbb{R}^{d_z}$. Two codebooks of sizes K_1 and K_2 perform residual quantization sequentially:

$$\begin{aligned} s_{1,i} &= \arg \min_k \|z_i - e_k^{(1)}\|_2^2, & \delta_i^{(1)} &= z_i - \text{sg}(e_{s_{1,i}}^{(1)}), \\ s_{2,i} &= \arg \min_k \|\delta_i^{(1)} - e_k^{(2)}\|_2^2, & z_i^q &= e_{s_{1,i}}^{(1)} + e_{s_{2,i}}^{(2)}. \end{aligned} \quad (1)$$

Here sg denotes stop-gradient. The decoder uses the straight-through estimate $z_i^{\text{ST}} = z_i + \text{sg}(z_i^q - z_i)$, whose forward value equals z_i^q , while reconstruction gradients flow to the item encoder.

Query-aligned tokenizer objective. The tokenizer jointly optimizes reconstruction, commitment, and bidirectional query-item contrastive losses:

$$\begin{aligned} \mathcal{L}_{\text{rec}} &= \|D(z_i^{\text{ST}}) - x_i\|_2^2, \\ \mathcal{L}_{\text{com}} &= \frac{1}{2} (\|z_i - \text{sg}(e_{s_{1,i}}^{(1)})\|_2^2 + \|\delta_i^{(1)} - \text{sg}(e_{s_{2,i}}^{(2)})\|_2^2), \\ \mathcal{L}_{\text{QI}} &= \frac{1}{2} [\text{CE}(S, I) + \text{CE}(S^T, I)], \quad S_{ab} = \frac{\tilde{z}_{i,a}^T \tilde{z}_{q,b}}{\tau_{\text{QI}}}, \\ \mathcal{L}_{\text{ID}} &= \mathcal{L}_{\text{rec}} + \mathcal{L}_{\text{com}} + \lambda_{\text{QI}} \mathcal{L}_{\text{QI}}. \end{aligned} \quad (2)$$

Reconstruction preserves item embedding information, commitment encourages encoder outputs to approach discrete codes, and the bidirectional InfoNCE term brings paired query and item representations closer in continuous space, injecting search relevance before quantization. Here, $\tau_{\text{QI}} > 0$ is the contrastive temperature and $\lambda_{\text{QI}} > 0$ is the contrastive-loss weight. The diagonal of the contrastive matrix S contains positive pairs, with labels $I = (0, \dots, N-1)$; the remaining valid query-item combinations in the local minibatch serve as negatives. Samples without queries optimize only the reconstruction and commitment losses. The codebooks are initialized by deterministic K-means and updated by exponential moving averages.

The first two codes form the semantic prefix $c(i) = (s_{1,i}, s_{2,i})$, and the third token r_i represents the item's value-ranked position within that prefix, together yielding a unique item address at initial

construction:

$$y_i = (s_{1,i}, s_{2,i}, r_i). \quad (3)$$

Here r_i is zero-based and satisfies $0 \leq r_i < K_3$, where K_3 is the rank-token vocabulary size.

Value-ordered item address. Let b_i , a_i , and c_i denote an item's purchase, add-to-cart, and click counts in a fixed ranking window. We set $v_i = b_i + \eta_a a_i$ and estimate conversion value with empirical-Bayes smoothing:

$$p_0 = \frac{\sum_i v_i}{\sum_i c_i}, \quad \hat{p}_i = \frac{v_i + \alpha_{\text{ID}} p_0}{c_i + \alpha_{\text{ID}}}. \quad (4)$$

We define the behavioral score $V_i^{\text{funnel}} = \gamma_b b_i + \gamma_a a_i + \gamma_c c_i$ and rank items within each prefix using the following funnel rule, comparing its keys lexicographically:

$$\hat{p}_i \downarrow \longrightarrow V_i^{\text{funnel}} \downarrow \longrightarrow \text{GMV}_i \downarrow \longrightarrow \text{ID}_i \uparrow. \quad (5)$$

Here $0 < \eta_a < 1$, $\alpha_{\text{ID}} > 0$, and $\gamma_b > \gamma_a > \gamma_c > 0$, reflecting the behavioral priority of purchase > add-to-cart > click. Empirical-Bayes smoothing reduces low-frequency noise while preserving conversion-value information; Section 4.2 shows that its ordering is more stable than raw click and volume ordering.

3.2 Dual-Loop Daily Refresh

Product loop. Within the same primary SID version, we freeze the RQ-VAE and SID vocabulary and preserve existing item SIDs. New items in existing semantic prefixes reuse historical slots according to their within-cluster value ranks; newly observed prefixes receive addresses starting from zero.

For example, existing items A, B, and C under one semantic prefix have third tokens 0, 1, and 2. If a new item D ranks between A and B by value, D reuses slot 1 and shares B's SID, while the addresses of A, B, and C remain unchanged.

For prefix $c = (a, b)$, $C_c^{(t)} = C_{ab}^{(t)}$ denotes the number of distinct occupied rank slots at snapshot t . An occupied prefix has maximum rank $M_t(c) = C_c^{(t)} - 1$; unseen prefixes have capacity zero. Let $\rho_t(i)$ be the zero-based position of new item i after jointly ordering all existing and new items in the same prefix at snapshot t according to Equation (5). Its assigned rank is

$$r_t(i) = \begin{cases} \min\{\rho_t(i), M_{t-1}(c(i))\}, & C_{c(i)}^{(t-1)} > 0, \\ \min\{\rho_t(i), K_3 - 1\}, & C_{c(i)}^{(t-1)} = 0. \end{cases} \quad (6)$$

Equation (6) maps new items beyond the available slot range to the last slot; for a new prefix, the maximum slot index is $K_3 - 1$. Assigned addresses remain fixed within the primary version, and items sharing a slot are expanded into candidates for downstream ranking at serving time. Below, C_{ab} denotes the slot capacity of the relevant fixed snapshot, with the time index omitted.

Model loop. Each day, the latest click and purchase data are labeled using the SID mapping in effect when the behavior occurred. Within the same primary SID version, we continue training the last accepted personalized generation model for $E_{\text{day}} \in \mathbb{N}^+$ epochs, mixing historical replay with proportion $0 \leq \lambda_{\text{replay}} \leq 1$ to incorporate fresh feedback while retaining learned addressing relationships. After validation, the model, Trie, and item map are atomically published as a compatible version combination.

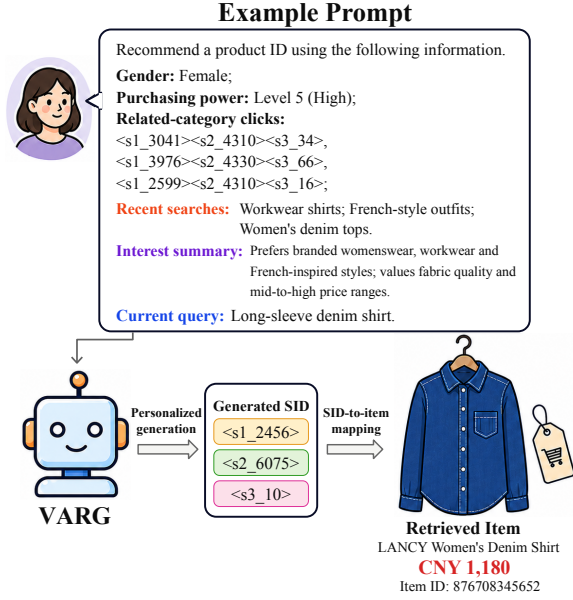


Figure 3: Schematic personalized prompt-to-item example (English translation).

Monthly SID rebuilding introduces a new primary version, under which training labels, SID-encoded histories, and replay data are re-encoded. The retrieval model is retrained before joint deployment, and the accepted model initializes the next daily-update cycle.

3.3 Three-Stage SFT

Three-stage supervised fine-tuning. We add K_ℓ dedicated tokens for SID level ℓ to the vocabulary of Qwen2.5-0.5B-Instruct [31]. Given context x , the model autoregressively generates the target identifier $y = (s_1, s_2, r)$:

$$p_\theta(y | x) = \prod_{\ell=1}^3 p_\theta(y_\ell | x, y_{<\ell}). \quad (7)$$

Training proceeds sequentially across stages. Stage 1 learns Item→SID from product content, establishing item-address memory. Stage 2 continues from Stage 1 using Query→SID examples to learn query-semantic retrieval. Stage 3 continues from Stage 2 using User+Query+History→SID examples to learn personalized retrieval.

For personalized candidate generation, context expansion adds short-term intent and long-term preference signals, Q2I weighting emphasizes query-specific conversion value, level weighting adjusts supervision strength across SID levels, and LO-SFT introduces local ordinal supervision encoded by the third token. The final Stage 3 checkpoint initializes the trainable policy π_θ and fixed reference policy π_{ref} for Prefix-GRPO.

User context expansion. The personalized input is $x = (q, u, h^{\text{cat}}, h^{\text{recent}}, g)$, where q is the current query, u is the user profile including gender and purchasing power, h^{cat} is related-category click history represented by historical SIDs,

h^{recent} contains the L_{recent} most recent search queries, and g is a global-interest summary distilled from cross-domain behaviors. Recent queries provide additional signals of short-term user intent, while the global summary represents long-term user interests by summarizing category, brand, and price preferences. Figure 3 shows an example of item candidate generation from the full input context.

Q2I example weighting. To account for conversion differences across query-item pairs, we construct example weights using purchase and click statistics from a fixed historical window. Let b_j and c_j denote the purchase and click counts for query-item pair j . The global prior, smoothed conversion rate, and example weights are

$$p_{0q} = \frac{\sum_j b_j}{\sum_j c_j}, \quad \hat{p}_j = \frac{b_j + \alpha_q p_{0q}}{c_j + \alpha_q},$$

$$\tilde{w}_j = \text{clip} \left(\frac{\hat{p}_j}{p_{0q}}, w_{\min}, w_{\max} \right), \quad w_j = \frac{\tilde{w}_j}{\mathbb{E}[w]}. \quad (8)$$

Here, $\alpha_q > 0$ is the prior strength, and the clipping bounds satisfy $0 < w_{\min} < 1 < w_{\max}$. Empirical-Bayes smoothing reduces noise in sparse statistics; the ratio of the smoothed conversion rate to the global prior is clipped and normalized by its mean, assigning greater weights to query-item pairs with higher conversion propensity. We activate this signal only in Stage 3; Stages 1–2 remain equally weighted to learn unbiased item memory and query semantics.

Level-aware token weighting. The three target positions have distinct decision roles. s_1 selects a coarse semantic region, s_2 chooses the fine semantic prefix, and r only identifies an item after that prefix is correct. In Trie-constrained Top-1000 evaluation, the HR gap is 16.94 percentage points between L1 and L1+L2, versus only 3.80 points between L1+L2 and the complete SID, indicating that the second level is the main prediction bottleneck. We therefore assign supervision weights based on prediction performance at each level. Let $\ell_{j,t} = -\log p_\theta(y_{j,t} | x_j, y_{j,<t})$ denote per-token cross-entropy, $m_{j,t}$ the response mask, and $\lambda(y_{j,t})$ the token level. The level-weighted loss is

$$\overline{\mathcal{L}}_j^{\text{level}} = \frac{\sum_t m_{j,t} \alpha_{\lambda(y_{j,t})} \ell_{j,t}}{\sum_t m_{j,t} \alpha_{\lambda(y_{j,t})}}. \quad (9)$$

The level coefficients $(\alpha_1, \alpha_2, \alpha_3)$ satisfy $\alpha_2 > \alpha_1 > \alpha_3 > 0$ to strengthen supervision at the first two levels; non-SID response tokens retain unit weight. Combining this loss with Q2I example weights gives the Stage 3 training objective

$$\mathcal{L}_{\text{SFT3}} = \frac{\sum_j w_j \overline{\mathcal{L}}_j^{\text{level}}}{\sum_j w_j}. \quad (10)$$

Local ordinal supervision (LO-SFT). Standard cross-entropy treats rank tokens as discrete categories without explicitly using their ordinal distances. LO-SFT continues fine-tuning from the trained Stage-3 SFT Base checkpoint, adding local supervision around the target position based on the value ranks encoded by the third token.

For prefixes with slot capacity $C = C_{ab} \geq C_{\min}$, we construct a local target distribution centered on the true rank r^* that decays

with distance, while retaining the exact-token loss:

$$q(r | r^*) = \frac{\exp(-|r - r^*|/\tau_{LO})}{Z(r^*)} \mathbb{I}[|r - r^*| \leq R_{LO}] \times \mathbb{I}[0 \leq r < C],$$

$$\mathcal{L}_{LO} = \mathcal{L}_{\text{exact}} - \lambda_{LO} \sum_{r=0}^{C-1} q(r | r^*) \log p_{\theta}(r | x, s_1, s_2), \quad (11)$$

where $C_{\min}$ is the minimum prefix-capacity threshold, R_{LO} is the neighborhood radius, $\tau_{LO} > 0$ controls distance decay, $\lambda_{LO} > 0$ is the auxiliary-loss weight, and $Z(r^*)$ is the normalization constant over the legal neighborhood. The exact loss supervises the target SID, while the auxiliary term guides the model to learn the local ordinal relationships encoded by EB-CVR ordering.

3.4 Prefix-GRPO

Building on supervised learning, Prefix-GRPO adjusts candidate generation probabilities according to rewards to further align candidate priorities with business value and ranking objectives. We use group-relative policy optimization [22], combining gated multi-source rewards with prefix-aware token weighting. For each prompt, the policy samples G responses, with $(T_{\text{roll}}, p_{\text{roll}}, k_{\text{roll}})$ controlling sampling temperature, nucleus sampling, and top- k truncation, respectively.

Response contract and gated reward. After removing control tokens, a response must be exactly $y = \langle s1_a \rangle \langle s2_b \rangle \langle s3_r \rangle$, where $0 \leq a < K_1$, $0 \leq b < K_2$, and $0 \leq r < K_3$. In the RL training snapshot, a SID with $C_{ab} > 0$ and $0 \leq r < C_{ab}$ corresponds to an occupied slot. Only well-formed, occupied SIDs participate in subsequent reward computation; other outputs receive only their corresponding negative rewards.

Let $\mathcal{P}_x$, $\mathcal{C}_x$, and $\mathcal{E}_x$ denote the sets of SIDs matched to purchase, click, and exposure behaviors for request x , respectively. The behavior reward uses the highest matched action under the priority purchase > click > exposure:

$$R_{UA}(x, y) = \begin{cases} \beta_{\text{buy}}, & y \in \mathcal{P}_x, \\ \beta_{\text{clk}}, & y \notin \mathcal{P}_x, y \in \mathcal{C}_x, \\ \beta_{\text{exp}}, & y \notin \mathcal{P}_x \cup \mathcal{C}_x, y \in \mathcal{E}_x, \\ 0, & \text{otherwise.} \end{cases} \quad (12)$$

The ranker-advantage reward favors items with scores above the candidate mean for the same request. Let $f_x(y)$ be the production ranker's final score and μ_x^f, σ_x^f its mean and standard deviation over mapped recall candidates for that request. The reward is

$$a_x(y) = \frac{f_x(y) - \mu_x^f}{\sigma_x^f},$$

$$\tilde{a}_x(y) = \begin{cases} a_x(y), & 0 < a_x(y) < \infty, \\ 0, & \text{otherwise,} \end{cases}$$

$$R_{\text{adv}}(x, y) = \beta_{\text{adv}} \tanh\left(\frac{\tilde{a}_x(y)}{\tau_{\text{adv}}}\right). \quad (13)$$

For occupied SIDs without matched user behavior, the relevance

grade $g_x(y)$ provides supplementary feedback:

$$R_{\text{rel}}(x, y) = \begin{cases} \beta_{\text{rel}}^{(3)}, & g_x(y) = 3, \\ \beta_{\text{rel}}^{(2)}, & g_x(y) = 2, \\ -\beta_{\text{rel}}^{(1)}, & g_x(y) = 1, \\ 0, & \text{otherwise.} \end{cases} \quad (14)$$

Let $\mathcal{F}$ denote malformed or out-of-range outputs, $\mathcal{U}$ structurally valid but unoccupied outputs, and $\mathcal{H}_x = \mathcal{P}_x \cup \mathcal{C}_x \cup \mathcal{E}_x$. The complete reward is

$$R(x, y) = \begin{cases} -\beta_F, & y \in \mathcal{F}, \\ -\beta_U, & y \in \mathcal{U}, \\ R_{UA} + R_{\text{adv}}, & y \text{ occupied, } y \in \mathcal{H}_x, \\ R_{\text{adv}} + R_{\text{rel}}, & y \text{ occupied, } y \notin \mathcal{H}_x. \end{cases} \quad (15)$$

Here, $\beta_F > \beta_U > 0$, $\beta_{\text{buy}} > \beta_{\text{clk}} > \beta_{\text{exp}} > 0$, $\beta_{\text{adv}}, \tau_{\text{adv}} > 0$, $\beta_{\text{rel}}^{(3)} > \beta_{\text{rel}}^{(2)} > 0$, and $\beta_{\text{rel}}^{(1)} > 0$. Relevance rewards apply only to SIDs without matched behavior, avoiding double counting of correlated labels.

Prefix-aware token weighting. The number of legal token alternatives varies across prefixes. We use these counts to assign token-level policy-loss weights, giving higher weights to decisions with more branches. In the RL training snapshot, the numbers of legal child tokens at the second and third levels are

$$n_2(a) = \sum_{b=0}^{K_2-1} \mathbb{I}[C_{ab} > 0], \quad n_3(a, b) = C_{ab}, \quad (16)$$

Let $N_2 = \max_a n_2(a)$ and $N_3 = \max_{a,b} n_3(a, b)$. We map the legal child-token count to $[d, 1]$ using

$$\phi(n; N) = \begin{cases} d + (1-d) \frac{\log n}{\log N}, & N > 1, \\ d, & N = 1, \end{cases} \quad 0 < d \leq 1. \quad (17)$$

For an occupied path, token weights are

$$w_{s_1} = 1, \quad w_{s_2} = \phi(n_2(a); N_2),$$

$$w_{s_3} = \phi(n_3(a, b); N_3), \quad w_{\text{ctrl}} = d. \quad (18)$$

The first-level weight is fixed at 1, while later levels are weighted according to the number of legal alternatives under the current prefix. Branches with only one legal option, as well as EOS and control tokens, use the floor d . For malformed or unoccupied responses, all valid tokens use unit weight. These weights enter the policy loss in Equation (20).

Weighted GRPO optimization. Let $R_{x,g}$ be the reward of response g to prompt x , and $\bar{R}_x, s_x$ the mean and sample standard deviation of the G response rewards for that prompt. Based on GRPO's within-group reward comparison [22], we construct advantages through standardization, whitening, and clipping:

$$\hat{R}_{x,g} = \text{clip}\left(\frac{R_{x,g} - \bar{R}_x}{s_x + \epsilon_{\text{num}}}, -c_R, c_R\right),$$

$$A_{x,g} = \text{clip}\left(\text{Whiten}(\hat{R}_{x,g}), -c_A, c_A\right). \quad (19)$$

Here, $\varepsilon_{\text{num}} > 0$ provides numerical stability, and $c_R, c_A > 0$ are clipping thresholds. All valid tokens of a response share the same advantage $A_{x,g}$.

Using a PPO-style clipped objective [21], let $\rho = \pi_\theta(y_t | x, y_{<t}) / \pi_{\text{old}}(y_t | x, y_{<t})$ be the token probability ratio and $\bar{\rho} = \text{clip}(\rho, 1 - \varepsilon_{\text{ppo}}, 1 + \varepsilon_{\text{ppo}})$. The surrogate objective is $J = \min(\rho A, \bar{\rho} A)$. When $A < 0$, dual clipping further applies $\max(J, c_{\text{dual}} A)$, where $c_{\text{dual}} > 1$; otherwise, J is retained. Denote the result by $J_{x,g,t}^{\text{dual}}$.

For a minibatch of B sampled responses, let $m_{x,g,t}$ be the response mask. The prefix-weighted policy loss is

$$\mathcal{L}_{\text{PG}} = -\frac{1}{B} \sum_{x,g} \frac{\sum_t m_{x,g,t} w_{x,g,t} J_{x,g,t}^{\text{dual}}}{\sum_t m_{x,g,t} w_{x,g,t}}. \quad (20)$$

Adding KL regularization relative to the fixed SFT reference policy gives the final training objective:

$$\mathcal{L}_{\text{actor}} = \mathcal{L}_{\text{PG}} + \beta_{\text{KL}} \mathcal{L}_{\text{KL}}. \quad (21)$$

Here, $\beta_{\text{KL}} > 0$ is the regularization coefficient. $\mathcal{L}_{\text{KL}}$ averages clipped KL-divergence estimates over valid tokens within each response and then across responses.

3.5 Serving

At inference, the model generates Top- K SID paths using Trie-constrained beam search conditioned on the expanded context. The Trie is built from the published SID set. Let $\mathcal{T}(p)$ denote the set of legal child tokens after partial path p . Before each beam expansion, logits are masked as

$$\tilde{z}_v(p) = \begin{cases} z_v(p), & v \in \mathcal{T}(p), \\ -\infty, & v \notin \mathcal{T}(p). \end{cases} \quad (22)$$

At the third level, only occupied rank tokens satisfying $0 \leq r < C_{ab}$ are allowed. Masking prunes invalid branches before beam expansion, so the generated complete SIDs can be mapped to valid items.

The Trie is rebuilt by the product-refresh loop and kept version-consistent with the SID-to-item map. Generated SIDs are expanded into item candidates through the version-matched map and directly admitted to the existing final ranker as an additional retrieval channel, augmenting the production multi-channel retrieval stack.

3.6 Exploratory Lightweight Decoder

Standard autoregressive decoding requires a Transformer forward pass at each SID generation step. To reduce inference cost, we compute the prompt-end hidden state h once and then predict SID tokens at each level using lightweight, prefix-conditioned classification heads:

$$\begin{aligned} p_1(s_1 | h) &= \text{softmax}(g_1(h)), \\ p_2(s_2 | h, s_1) &= \text{softmax}(g_2([h; e_1(s_1)])), \\ p_3(s_3 | h, s_1, s_2) &= \text{softmax}(g_3([h; e_1(s_1); e_2(s_2)])), \end{aligned} \quad (23)$$

where e_l is the embedding of the SID token at level l and g_l is the corresponding prediction head. Training uses teacher forcing, with the second- and third-level heads conditioned on ground-truth

upper-level tokens. For $L = 3$ levels, we normalize the loss coefficients $\alpha_l > 0$ to obtain the training objective

$$\tilde{\alpha}_l = \frac{L\alpha_l}{\sum_{j=1}^L \alpha_j}, \quad \mathcal{L}_{\text{head}} = \sum_{l=1}^L \tilde{\alpha}_l \text{CE}(p_l, s_l). \quad (24)$$

At inference, hierarchical beam expansion proceeds across the prediction heads, with prefix masks retaining occupied VARG-ID child tokens. The backbone is evaluated only once, while subsequent decisions use small prediction heads. This eliminates repeated Transformer forward passes. Section 4.6 evaluates the decoder's efficiency–retrieval-quality trade-off; it remains an exploratory approach and is not used in the main VARG serving pipeline.

4 Experiments

4.1 Experimental Setup

Dataset. All experiments were conducted on real search logs from Tmall App, a major mobile e-commerce platform. VARG-ID covers 51.43M products, with its tokenizer trained on query–item pairs and item-only examples. The three-stage generative model was trained on 51.43M samples for Item2SID, 48.00M for Query2SID, and 105.69M for personalized User–Query2SID. For Prefix-GRPO, we construct a training pool of approximately 1.03M records sampled from request-level data. Personalized evaluation and SID re-ordering audits used 3.85M samples; lightweight decoding was evaluated on a separate query-only set of 180,324 samples.

Metrics. HR@ K measures whether the target SID appears among the top- K deduplicated predictions; NDCG@ K assesses ranking quality. GMV Recall and Q2I-GMV Recall measure the fraction of evaluation-sample value covered by these hits, weighted by the item's 30-day GMV and the query–item pair's 14-day GMV, respectively. The online A/B test evaluates a fixed candidate quota for direct admission over 14 days with 20% of search traffic, reporting relative changes in business and engagement metrics against the production control.

Implementation Details. We extend Qwen2.5-0.5B-Instruct with SID tokens and perform full-parameter SFT on 128 GPUs. Learning rates are 10^{-4} for Stage 1, 4×10^{-5} for Stages 2–3, and 5×10^{-6} for LO-SFT. We evaluate SFT strategies with matched epoch budgets. Prefix-GRPO runs on 64 GPUs at a learning rate of 10^{-6} , sampling eight responses per prompt with 1,024 prompts per rollout batch.

Hyperparameter Settings. For VARG-ID, both semantic codebooks and the third-level rank vocabulary have 8,192 entries ($K_1 = K_2 = K_3 = 8192$). Reconstruction and commitment losses each have unit weight; the bidirectional Q2I contrastive loss uses $\lambda_{\text{QI}} = 0.001$ and temperature $\tau_{\text{QI}} = 0.1$, with at most 128 valid query–item pairs sampled per local minibatch.

For SFT strategies, the Q2I prior strength is $\alpha_q = 5$, and intent expansion includes the three most recent search queries ($L_{\text{recent}} = 3$). LO-SFT uses neighborhood radius $R_{\text{LO}} = 3$, distance-decay temperature $\tau_{\text{LO}} = 1.0$, and auxiliary loss coefficient $\lambda_{\text{LO}} = 0.02$.

For Prefix-GRPO, malformed and unoccupied outputs receive rewards of -2.0 and -1.0 , respectively. Purchase, click, and exposure rewards are 3.0, 1.0, and 0.1. The ranker-advantage reward uses $\beta_{\text{adv}} = 0.2$ and $\tau_{\text{adv}} = 1.5$; relevance grades 3, 2, and 1 contribute

Table 1: Evaluation of SIDs and constrained decoding on production data. (a) Initial identifier capacity and collision. (b) Full-reordering stability. (c) Trie-constrained decoding effectiveness and overhead.

(a) Initial identifier capacity and collision.							(b) Full-reordering stability.					
Identifier	Tokens	Unique	Exclusive	Collision	Avg.	P99 / Max	Policy	Change	Avg. shift	P90	P99	Top20 keep
RQ-VAE-3	3	16.73%	8.75%	91.25%	5.98	76 / 7,580	EB-CVR	69.54%	22.00	51	291	93.31%
RQ-VAE-4	4	51.84%	38.42%	61.58%	1.93	15 / 2,364	Click	77.61%	26.58	61	395	91.29%
RQ+OPQ	5	81.21%	71.62%	28.38%	1.23	5 / 968	Volume	76.93%	26.99	59	389	91.97%
VARG prefix	2	10.74%	4.32%	95.68%	9.31	129 / 4,221	Frozen update	0.00%	0.00	0	0	100.00%
VARG-ID	3	100%	100%	0%	1.00	1 / 1						

(c) Trie-constrained decoding.						
Method	Legal rate (%)	Δ HR@20	Δ HR@100	Δ HR@1000	Inference time Δ	Memory Δ
Trie	93.73 $\rightarrow$ 100	+0.0915	+0.2343	+0.3857	+3.16%	+0.13 GiB

Notes. (a) For N items and U occupied SIDs, $\text{Unique}=U/N$ and $\text{Avg.}=N/U$. Exclusive is the fraction of items with an unshared SID, and $\text{Collision}=1 - \text{Exclusive}$; P99/Max are the 99th percentile and maximum item counts per occupied SID. (b) Change is the fraction of items whose third token changes; Avg. shift is mean absolute rank displacement, and P90/P99 are displacement quantiles. Top20 keep is the retention rate of old Top-20 items in the same prefix's new Top-20. (c) HR gains are in percentage points. Time is cumulative forward-pass and beam-search time, and memory is peak GPU memory usage; both are measured on rank 0.

+0.08, +0.03, and -0.10 for occupied SIDs without matched user actions. The prefix-weight floor is $d = 0.1$, also used for EOS and control tokens, and the KL regularization coefficient is $\beta_{\text{KL}} = 0.8$.

4.2 SID Observations: What Identifier Fits Recall-and-Pre-Ranking?

Table 1(a) shows that adding semantic quantization levels reduces collisions, but some items still share identifiers and decoding becomes longer. VARG-ID retains a two-level semantic prefix and distinguishes items within each prefix using a value-ordered third token, achieving collision-free addressing at initial construction. Both semantic codebooks are fully utilized, and the rank vocabulary has sufficient capacity for the largest observed item cluster.

Table 3 shows that the VARG-ID SFT base improves Q2I-GMV Recall@1/10/20 by 15.50/13.65/0.93 percentage points over the SFT base using three-level RQ-VAE identifiers. These results indicate that VARG-ID achieves better value recall before RL, helping prioritize high-value items under limited candidate budgets.

EB-CVR is the most stable of the value-based reordering policies in Table 1(b): compared with Click, it yields fewer third-token changes, smaller rank displacements, and higher Top-20 retention. We further evaluate how identifier changes affect the trained model's retrieval performance. We keep the SFT model parameters fixed and reorder all items within their semantic prefixes. After reordering, L1 HR@100 remains nearly constant (92.08% $\rightarrow$ 92.00%) and L1+L2 HR remains stable (71.80% $\rightarrow$ 71.56%), but complete-SID HR@100 drops by **14.45 points**, and third-token teacher-forcing accuracy falls from 36.75% to 21.24%. This degradation reflects temporal drift in SID labels: as item statistics change, within-prefix reordering changes the third token assigned to the same item, creating a mismatch with the addressing relationships learned by the model. We therefore adopt frozen incremental updates to preserve existing item addresses.

During daily product updates, existing item addresses remain unchanged, while incremental insertion permits controlled SID sharing. This update policy incorporates new items while preserv-

ing the addressing relationships learned by the model.

Table 1(c) shows that Trie constraints improve SID legality and retrieval hit rates with only small increases in inference time and memory usage.

4.3 SFT Strategies

Value and hierarchy supervision. Joint Q2I and level weighting yields small gains in most reported HR metrics (Table 2(a)). Value-recall gains are concentrated near the beam head: GMV Recall@1 and Q2I-GMV Recall@1 increase by 2.18 and 6.23 percentage points, respectively, while changes at Top-50 are small (Table 2(b)). Level weighting also yields small HR gains at larger candidate budgets.

Intent expansion. Adding recent queries and the global-interest summary improves all reported HR metrics, with gains of 0.30 and 0.31 percentage points at HR@20 and HR@50, respectively. These results suggest that incorporating short-term intent and long-term preferences helps improve personalized retrieval.

Local ordinal supervision. LO-SFT yields small gains in all reported HR metrics, with HR@50 increasing from 46.19% to 46.33%. Further analysis shows that NDCG@100 increases from 20.91% to 21.00%, while third-token top-1 accuracy, conditioned on the gold two-level semantic prefix, increases from 36.74% to 37.02%.

4.4 Overall Prefix-GRPO Results

Relative to the SFT model using VARG-ID, Prefix-GRPO improves head-level hit rate, ranking quality, and value recall (Table 4). GMV Recall@1 and Q2I-GMV Recall@1 increase by 5.47 and 2.48 percentage points, respectively; HR@1 improves by 0.36 points for purchase-labeled requests and 0.03 points for click-labeled requests.

Reward and prefix ablation. Table 5 compares SFT Base with three RL configurations. All three retain the legality gates. Behavior GRPO uses user-behavior rewards for occupied SIDs; Multi-source GRPO adds ranker advantage and relevance rewards; Full Prefix-

Table 2: Evaluation of Stage-3 SFT strategies. (a) Absolute plain HR of the base and gains from the evaluated methods. (b) GMV and Q2I-GMV recall through Top-50 for the Q2I-example- and level-weighted checkpoint. All changes are in percentage points.

(a) Plain HR: absolute baseline (%) and gains from the evaluated strategies (pp).							
Direction	Method	@1	@10	@20	@50	@100	@1000
Baseline	SFT Base (absolute, %)	6.39	26.33	34.68	46.19	54.59	72.84
Value	Q2I weight + level loss	-0.01	+0.05	+0.10	+0.13	+0.18	+0.16
Hierarchy	Level-weighted loss	-0.02	0.00	+0.02	+0.05	+0.08	+0.15
Intent	Recent queries + global summary	+0.10	+0.26	+0.30	+0.31	+0.28	+0.23
Ordering	LO-SFT	+0.06	+0.13	+0.12	+0.14	+0.12	+0.07

(b) Value-weighted recall (%).							
K	GMV Recall			Q2I-GMV Recall			Δ
	Base	Q2I + level	Δ	Base	Q2I + level	Δ	
1	30.88	33.06	+2.18	36.12	42.35	+6.23	
10	92.96	94.42	+1.46	95.18	97.01	+1.83	
20	97.57	97.73	+0.16	98.06	99.06	+1.00	
50	99.35	99.36	+0.01	99.81	99.83	+0.02	

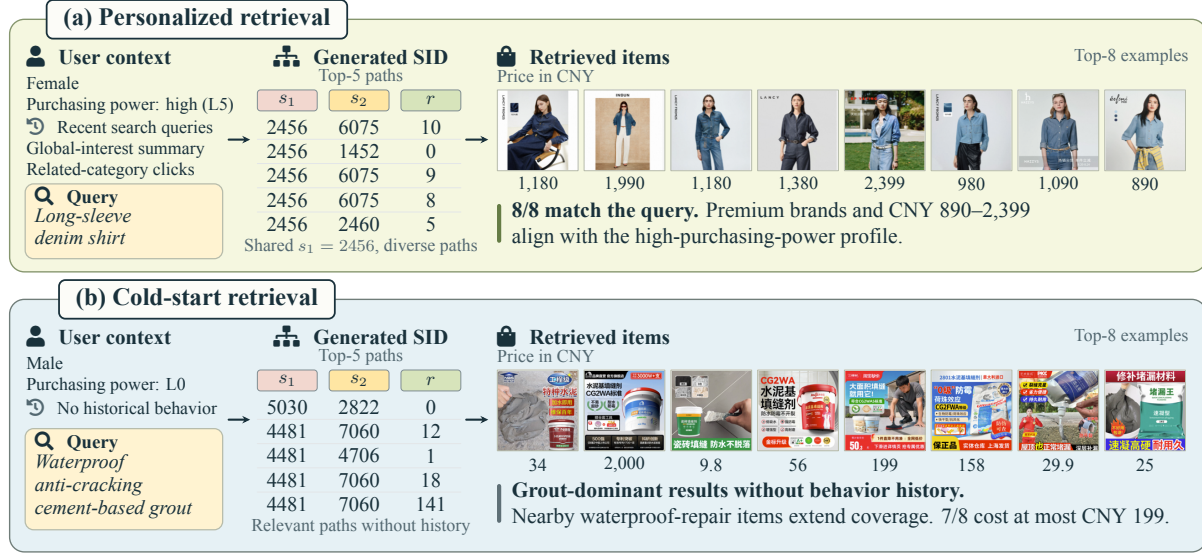


Figure 4: Personalized and cold-start retrieval examples. Case A shows personalized retrieval with user context; Case B shows cold-start retrieval without behavioral history. Prices are in CNY.

Table 3: Q2I-GMV recall of SFT Base models. Absolute scores are percentages; gains are percentage points.

Method	@1	@10	@20
RQ-VAE-3	20.62	81.53	97.13
VARG-ID SFT Base	36.12	95.18	98.06
Δ (pp)	+15.50	+13.65	+0.93

GRPO further enables prefix-aware token weighting. The first two configurations use unit weights for valid response tokens in the policy loss.

Relative to SFT Base, Behavior GRPO improves Q2I-GMV Recall@1/10 by 2.19/1.64 percentage points. Adding ranker and relevance feedback further improves these metrics by 0.21/0.30 points and NDCG@10 by 0.11 points. With the same reward, prefix weighting adds 0.08/0.09 points to Q2I-GMV Recall@1/10 and 0.04 points to HR@20. Behavioral feedback accounts for most of the observed head-value gain, ranker and relevance feedback further improve value recall and ranking, and prefix weighting provides small but consistent additional gains.

Table 4: Overall Prefix-GRPO results using the SFT base in Table 2: head-level retrieval (left), value-weighted recall (right), and behavior-stratified HR@1 (bottom). Absolute scores are percentages; changes are percentage points.

Plain metric	SFT	Prefix-GRPO	Δ	Value metric	SFT	Prefix-GRPO	Δ
HR@1	6.39	6.44	+0.05	GMV Recall@1	30.88	36.35	+5.47
HR@10	26.33	26.46	+0.13	GMV Recall@10	92.96	95.20	+2.24
HR@20	34.68	34.85	+0.17	GMV Recall@20	97.57	98.06	+0.49
HR@50	46.19	46.30	+0.11	Q2I-GMV Recall@1	36.12	38.60	+2.48
NDCG@10	15.15	15.30	+0.15	Q2I-GMV Recall@10	95.18	97.21	+2.03
NDCG@20	17.26	17.44	+0.18	Q2I-GMV Recall@20	98.06	99.39	+1.33
<i>Behavior-stratified HR@1</i>							
Buy HR@1	13.51	13.87	+0.36	Click HR@1	5.97	6.00	+0.03

Table 5: Progressive ablation of reward signals and prefix-aware token weighting. All scores are percentages; best scores are bold.

Method	HR			Q2I-GMV Recall		NDCG
	@1	@10	@20	@1	@10	@10
SFT Base	6.39	26.33	34.68	36.12	95.18	15.15
Behavior GRPO	6.43	26.39	34.76	38.31	96.82	15.18
Multi-source GRPO	6.43	26.45	34.81	38.52	97.12	15.29
Full Prefix-GRPO	6.44	26.46	34.85	38.60	97.21	15.30

Table 6: Online A/B results over 14 days and 20% of search traffic. All values are relative changes against the production control.

Setting	GMV	IPV/user	Exp. PV/user	UCTR	UCTCVR	PCTR
VARG	+1.45%	+0.22%	-0.09%	+0.13%	+0.13%	+0.31%

Table 7: Relevance Good rate (%) in two independent evaluations of AI shopping-guide queries under different candidate-quota configurations.

System	Eval. 1	Eval. 2
Orig. pipeline (largest quota)	76.71	76.81
Orig. pipeline (reduced quota)	75.77	75.73
Gen. baseline (DA)	75.57	-
Gen. baseline (DA, larger quota)	75.90	76.03
VARG (DA, smaller quota)	76.77	76.58

Eval. 1/2: independent evaluation rounds. DA: direct admission to the final ranker, bypassing pre-ranking.

4.5 Online A/B Test

Online system validation. We conducted a 14-day online A/B test in Tmall App covering 20% of search traffic (Table 6). VARG uses Trie-constrained decoding with layer-wise beam widths of (20, 50, 500); generated item candidates bypass pre-ranking and enter the existing final ranker directly through a reserved quota. Compared with the production control, GMV increases by 1.45%, per-user IPV by 0.22%, UCTR and UCTCVR by 0.13% each, and PCTR by 0.31%. These results support the effectiveness of VARG for integrated recall and pre-ranking.

AI shopping-guide query evaluation. We conducted two independent relevance evaluations on 6K+ AI shopping-guide queries using the online search system. Good rate is the fraction of evaluated query-item pairs judged relevant. Table 7 shows that the original pipeline’s Good rate decreases when its candidate quota is reduced. With a smaller candidate quota, VARG outperforms the generative direct-admission baselines and achieves a Good rate comparable to or slightly higher than that of the original pipeline at its largest quota. These results indicate that VARG maintains competitive relevance under a smaller candidate budget.

4.6 Efficiency-Effectiveness Trade-off

We benchmarked the standard autoregressive decoder and the lightweight decoder in Section 3.6 using PyTorch and Transformers, without an additional inference engine. Our pilot study shows that, relative to standard autoregressive decoding, the lightweight decoder reduces mean latency from 10.41 to 0.94 ms and achieves approximately 11.1× the single-card QPS, but HR@1/20/100 drops by 1.13/8.38/10.77 percentage points. The lightweight decoder is not yet used in the main online VARG pipeline and requires further improvement in retrieval quality before deployment.

4.7 Case Study

Figure 4 presents personalized retrieval (Case A) and cold-start retrieval (Case B). In Case A, recent queries reflect short-term intent, the global-interest summary captures long-term preferences, and related-category clicks provide behavioral evidence. All eight displayed items match the query “long-sleeve denim shirt”; the premium brands and price range are consistent with the user’s high purchasing power, illustrating personalized retrieval aligned with user preferences. Case B contains no behavioral history. For the query “waterproof anti-cracking cement-based grout,” the displayed results mainly comprise grout products and related waterproof-repair items, showing that retrieval remains semantically aligned with the query in this cold-start example.

5 Conclusion

We presented VARG, a generative retrieval system for large-scale industrial e-commerce search that integrates recall and pre-ranking by directly admitting generated item candidates to the existing final ranker. VARG-ID uses bidirectional Q2I contrastive learning to align semantic prefixes with queries and stable EB-CVR ordering to assign the third token, providing

collision-free, fine-grained item addresses and a business-value prior at initial construction. Three-stage SFT advances from item memorization to query-semantic and personalized retrieval, combining context expansion, value-aware and hierarchical supervision, and LO-SFT to improve candidate generation. Prefix-GRPO uses gated multi-source rewards and prefix-aware weighting to further align candidate priorities with business value and ranking objectives. Coordinated daily product and model updates preserve existing item addresses while incorporating new products and the latest behavioral feedback. Offline experiments evaluate identifier stability and demonstrate gains in retrieval quality and value recall. A 14-day online A/B test improves GMV by 1.45% and PCTR by 0.31%, supporting the effectiveness of the complete recall-and-pre-ranking system. Future work will explore quality–efficiency trade-offs under tighter candidate budgets and improve retrieval quality in lightweight decoding while retaining its efficiency gains.

References

- [1] Michele Bevilacqua, Giuseppe Ottaviano, Patrick Lewis, Wen-tau Yih, Sebastian Riedel, and Fabio Petroni. Autoregressive search engines: Generating substrings as document identifiers. *arXiv preprint arXiv:2204.10628*, 2022.
- [2] Ben Chen, Siyuan Wang, Yufei Ma, Zihan Liang, Xuxin Zhang, Yue Lv, Ying Yang, Huangyu Dai, Lingtao Mao, Tong Zhao, et al. OneSearch-V2: The latent reasoning enhanced self-distillation generative search framework. *arXiv preprint arXiv:2603.24422*, 2026.
- [3] Jiangui Chen, Ruqing Zhang, Jiafeng Guo, Maarten de Rijke, Wei Chen, Yixing Fan, and Xueqi Cheng. Continual learning for generative retrieval over dynamic corpora. In *Proceedings of the ACM International Conference on Information and Knowledge Management*, 2023.
- [4] Jiehan Cheng, Zhicheng Dou, Yutao Zhu, and Xiaoxi Li. Descriptive and discriminative document identifiers for generative retrieval. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 11518–11526, 2025.
- [5] Nicola De Cao, Gautier Izacard, Sebastian Riedel, and Fabio Petroni. Autoregressive entity retrieval. In *International Conference on Learning Representations*, 2021.
- [6] Jiaxin Deng, Shiyao Wang, Kuo Cai, Lejian Ren, Qigen Hu, Weifeng Ding, Qiang Luo, and Guorui Zhou. OneRec: Unifying retrieve and rank with generative recommender and iterative preference alignment. *arXiv preprint arXiv:2502.18965*, 2025.
- [7] Kairui Fu, Tao Zhang, Shuwen Xiao, Ziyang Wang, Xinming Zhang, Chenchi Zhang, Yuliang Yan, Junjun Zheng, Xiangheng Kong, Shengyu Zhang, Kun Kuang, and Yuning Jiang. FORGE: Forming semantic identifiers for generative retrieval in industrial datasets. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2026.
- [8] Shijie Geng, Shuchang Liu, Zuohui Fu, Yingqiang Ge, and Yongfeng Zhang. Recommendation as language processing (RLP): A unified pretrain, personalized prompt & predict paradigm (P5). In *Proceedings of the ACM Conference on Recommender Systems*, 2022.
- [9] Vladimir Karpukhin, Barlas Ögüz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. *arXiv preprint arXiv:2004.04906*, 2020.
- [10] Omar Khattab and Matei Zaharia. ColBERT: Efficient and effective passage search via contextualized late interaction over BERT. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2020.
- [11] Doyup Lee, Chihyeon Kim, Saehoon Kim, Minsu Cho, and Wook-Shin Han. Autoregressive image generation using residual quantization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.
- [12] Yongqi Li, Xinyu Lin, Wenjie Wang, Fuli Feng, Liang Pang, Wenjie Li, Liqiang Nie, Xiangnan He, and Tat-Seng Chua. A survey of generative search and recommendation in the era of large language models. *arXiv preprint arXiv:2404.16924*, 2024.
- [13] Yongqi Li, Nan Yang, Liang Wang, Furu Wei, and Wenjie Li. Learning to rank in generative retrieval. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2024.
- [14] Zida Liang, Changfa Wu, Dunxian Huang, Weiqiang Sun, Ziyang Wang, Yuliang Yan, Jian Wu, Yuning Jiang, Bo Zheng, Ke Chen, Silu Zhou, and Yu Zhang. TBGRRecall: A generative retrieval model for e-commerce recommendation scenarios. *arXiv preprint arXiv:2508.11977*, 2025.
- [15] Gui Ling, Zhihong Chen, Yu Li, Tong Xiong, Kunhai Lin, Kaixuan Zhang, Yuliang Yan, Dan Ou, Haihong Tang, and Bo Zheng. Beyond semantic ids: Encoding business-value ranking into document identifiers for generative retrieval. *arXiv preprint arXiv:2607.11392*, 2026.
- [16] Yu. A. Malkov and D. A. Yashunin. Efficient and robust approximate nearest neighbor search using Hierarchical Navigable Small World graphs. *arXiv preprint arXiv:1603.09320*, 2016.
- [17] Sanket Vaibhav Mehta, Jai Gupta, Yi Tay, Mostafa Dehghani, Vinh Q. Tran, Jin-feng Rao, Marc Najork, Emma Strubell, and Donald Metzler. DSI++: Updating transformer memory with new documents. *arXiv preprint arXiv:2212.09744*, 2022.
- [18] Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *arXiv preprint arXiv:2305.18290*, 2023.
- [19] Shashank Rajput, Nikhil Mehta, Anima Singh, Raghunandan H. Keshavan, Trung Vu, Lukasz Heldt, Lichan Hong, Yi Tay, Vinh Q. Tran, Jonah Samost, Maciej Kula, Ed H. Chi, and Maheswaran Sathiamoorthy. Recommender systems with generative retrieval. In *Advances in Neural Information Processing Systems*, 2023.
- [20] Stephen Robertson and Hugo Zaragoza. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends in Information Retrieval*, 3(4):333–389, 2009.
- [21] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.
- [22] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeek-Math: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.
- [23] Weiwei Sun, Lingyong Yan, Zheng Chen, Shuaiqiang Wang, Haichao Zhu, Pengjie Ren, Zhumin Chen, Dawei Yin, Maarten de Rijke, and Zhaochun Ren. Learning to tokenize for generative retrieval. In *Advances in Neural Information Processing Systems*, 2023.
- [24] Juntao Tan, Shuyuan Xu, Wenyue Hua, Yingqiang Ge, Zelong Li, and Yongfeng Zhang. IDGenRec: LLM-RecSys alignment with textual ID learning. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2024.
- [25] Yi Tay, Vinh Q. Tran, Mostafa Dehghani, Jianmo Ni, Dara Bahri, Harsh Mehta, Zhen Qin, Kai Hui, Zhe Zhao, Jai Gupta, Tal Schuster, William W. Cohen, and Donald Metzler. Transformer memory as a differentiable search index. *arXiv preprint arXiv:2202.06991*, 2022.
- [26] Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning. In *Advances in Neural Information Processing Systems*, 2017.
- [27] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, 2017.
- [28] Wenjie Wang, Honghui Bao, Xinyu Lin, Jizhi Zhang, Yongqi Li, Fuli Feng, See-Kiong Ng, and Tat-Seng Chua. Learnable item tokenization for generative recommendation. *arXiv preprint arXiv:2405.07314*, 2024.
- [29] Yujing Wang, Yingyan Hou, Haonan Wang, Ziming Miao, Shibin Wu, Hao Sun, Qi Chen, Yuqing Xia, Chengmin Chi, Guoshuai Zhao, Zheng Liu, Xing Xie, Hao Allen Sun, Weiwei Deng, Qi Zhang, and Mao Yang. A neural corpus indexer for document retrieval. In *Advances in Neural Information Processing Systems*, 2022.
- [30] Shiguang Wu, Wenda Wei, Mengqi Zhang, Zhumin Chen, Jun Ma, Zhaochun Ren, Maarten de Rijke, and Pengjie Ren. Generative retrieval as multi-vector dense retrieval. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2024.
- [31] An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.
- [32] Bowen Zheng, Yupeng Hou, Hongyu Lu, Yu Chen, Wayne Xin Zhao, Ming Chen, and Ji-Rong Wen. Adapting large language models by integrating collaborative semantics for recommendation. In *Proceedings of the IEEE International Conference on Data Engineering*, 2024.
- [33] Guorui Zhou, Hengrui Hu, Hongtao Cheng, Huanjie Wang, Jiaxin Deng, Jinghao Zhang, Kuo Cai, Lejian Ren, Lu Ren, Liao Yu, et al. OneRec-V2 technical report. *arXiv preprint arXiv:2508.20900*, 2025.
- [34] Yujia Zhou, Zhicheng Dou, and Ji-Rong Wen. Enhancing generative retrieval with reinforcement learning from relevance feedback. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12481–12490. Association for Computational Linguistics, 2023.
- [35] Jianbo Zhu, Xing Fang, Jing Wang, Mingmin Jin, Bokang Wang, Guangxin Song, Zhenyu Xie, and Junjie Bai. Efficient generative retrieval for e-commerce search with semantic cluster ids and expert-guided rl. *arXiv preprint arXiv:2605.14434*, 2026.