SSR-GRPO: Integrating Supervision and Semantic IDs into Reinforcement Learning for Dense Retrieval in E-commerce

Guangxin Song
1220635451@qq.com
Alibaba Group
Hangzhou, China

Xing Fang
fangxing.fx@taobao.com
Alibaba Group
Hangzhou, China

Mingmin Jin
jimmy.jmm@taobao.com
Alibaba Group
Hangzhou, China

Jing Wang
jing.wangj1@taobao.com
Alibaba Group
Hangzhou, China

Bokang Wang
wangbokang.wbk@alibaba-inc.com
Alibaba Group
Hangzhou, China

Zhentao Song
zhentao.szt@alibaba-inc.com
Alibaba Group
Hangzhou, China

Junjie Bai
baijunjie.bjj@taobao.com
Alibaba Group
Hangzhou, China

Jianbo Zhu
zhujianbo.zjb@taobao.com
Alibaba Group
Hangzhou, China

Abstract

Embedding-based retrieval (EBR) is a prevailing method in modern e-commerce search, retrieving relevant items from billions of candidates in response to users' queries. While recent methods often fine-tune large language models (LLMs) for representation learning, they typically lack robust mechanisms for handling complex and implicit semantics.

To address this, Retrieval-GRPO (R-GRPO) was recently proposed as a multi-objective Reinforcement Learning (RL) framework for dense retrieval. It computes rewards using top-K items dynamically retrieved per query. While this approach partially mitigates retrieval challenges, the limited candidate pool sampled per batch often introduces noisy samples into the top-K results. Moreover, the LLM used for computing relevance rewards in R-GRPO is trained with a RL framework similar to R-GRPO itself. This inherent similarity makes it difficult to provide unbiased and impartial assessment results. To tackle these issues, we propose Supervised Retrieval-GRPO with Semantic Identifiers (SSR-GRPO). Specifically, our method first proposes a dual-perspective framework for relevance assessment. It leverages both Semantic Identifiers (SIDs) produced by quantization learning and dense representation vectors to generate more unbiased relevance scores. Furthermore, leveraging the hierarchical similarity relationships of the generated SIDs, we mine a set of hard negative samples that serve two purposes: (1) to design a masking function integrated into R-GRPO, effectively filtering intra-group noisy samples; and (2) to construct a Retrieval-DPO task composed of positive and negative sample pairs, enabling the model to capture fine-grained semantic distinctions from a pair-wise perspective. By integrating these optimization strategies, we propose SSR-GRPO. Extensive offline and online experiments validate SSR-GRPO's effectiveness, and it has been deployed on a large-scale e-commerce platform.

CCS Concepts

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

Keywords

Dense Retrieval, Reinforcement Learning, Semantic Identifiers, Hard Negative Mining

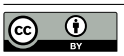
ACM Reference Format:

Guangxin Song, Xing Fang, Mingmin Jin, Jing Wang, Bokang Wang, Zhentao Song, Junjie Bai, and Jianbo Zhu. 2026. SSR-GRPO: Integrating Supervision and Semantic IDs into Reinforcement Learning for Dense Retrieval in E-commerce. In *Proceedings of the 35th ACM International Conference on Information and Knowledge Management (CIKM '26), November 07–11, 2026, Rome, Italy*. ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3799682.3840101

1 Introduction

Dense retrieval, also called embedding-based retrieval (EBR), is fundamental to modern e-commerce search [30]. It encodes queries and items into a shared embedding space [7, 10, 37] where semantic similarity is computed via distance metrics like cosine similarity. While early approaches used encoder-only architectures like BERT and RoBERTa [1, 20], recent LLM-based systems (e.g., Gemini [13], Qwen3 [38]) employ techniques like hard negative mining [27] and LLM-generated data synthesis [34]. However, most methods still follow BERT-era training paradigms.

Recent work explores integrating Reinforcement Learning (RL) with LLMs for retrieval tasks. Jiang et al. [8] proposed using RL to train LLMs for query generation/rewriting, achieving strong retrieval performance. This reflects a broader trend of applying policy optimization to information retrieval. In e-commerce dense retrieval, Retrieval-GRPO (R-GRPO) [19] leverages GRPO to train semantic retrieval models, capturing complex semantics through multi-objective rewards while eliminating manual hard negative mining. Specifically, at each training step, R-GRPO retrieves the



CIKM '26, Rome, Italy

ACM ISBN 979-8-4007-2539-5/2026/11
https://doi.org/10.1145/3799682.3840101

top-K most similar products from the current batch as hard negative samples. However, the products within each training batch represent only a limited subset of the overall product pool. Thus, there is no guarantee that the top-K items retrieved at each step are all valuable. Moreover, the retrieved top-K set may contain easy negative samples, which can introduce noise and misguidance into the model training process. Similar issues exist in the domain of LLMs. To address this, DeepSeekV3.2 proposed Off-policy Sequence Masking [17] strategy to mitigate the problem. It stabilizes training by masking sequences causing policy divergence. Beyond the noise of top-K items, R-GRPO's reward mechanism introduces another significant concern: the calculation of relevance reward relies on TaoSR1 [2], a 42B-MoE LLM. This dependency presents two main drawbacks: (1) The introduction of a high-parameter model for inference requires substantial extra computational resources during training; (2) Since TaoSR1 is trained using a reinforcement learning framework closely mirroring the one used in R-GRPO for query-item semantic matching, it tends to produce correlated relevance judgments. This correlation makes it difficult to provide unbiased and highly discriminative assessments.

To tackle these challenges, we propose a dense-sparse dual-perspective relevance reward evaluation method to replace the single LLM inference assessment. In the dense component, we utilize representation embeddings from a multimodal relevance LLM to compute inner product scores. In the sparse perspective, we employ Semantic Identifiers (SIDs) generated via Residual Quantized Variational Autoencoder (RQ-VAE) for items and through a next-token-prediction task for queries. The relevance score is then calculated based on the matching depth between the query and item SIDs. Furthermore, we design a joint training framework that enables the model to learn fine-grained semantic information and implicit patterns from multiple perspectives. Specifically, we introduce the Retrieval-DPO (R-DPO) approach to enable the model to learn representation differences from a pair-wise perspective. This is achieved by mining hard negative samples through the hierarchical similarity relationships of SIDs, combined with traditional hard negative mining strategies to construct a novel R-DPO loss and a masking function that provides explicit supervision signals to R-GRPO.

To the best of our knowledge, we are the first to leverage SIDs generated by RQ-VAE for mining hard negative samples in dense retrieval, and the first to integrate SID-based matching into reinforcement learning reward calculation. In summary, our proposed SSR-GRPO introduces the following innovations:

- We propose a dual-perspective relevance reward calculation method to enhance the reliability of relevance assessment.
- We introduce a hard negative sample mining strategy based on SID similarity. The newly mined samples are utilized to establish a joint training paradigm integrating R-DPO and R-GRPO, thereby enhancing the generalization capability of the reinforcement learning framework.
- We design a hard negative-based masking function for R-GRPO to reduce noise from simple negatives.
- By integrating the aforementioned components, we develop SSR-GRPO, a novel semantic retrieval paradigm rigorously validated through extensive offline and online A/B testing.

2 Related Work

2.1 Dense Retrieval

Embedding-based retrieval (EBR) based on dual-encoder architectures has been fundamental to search systems, enabling efficient similarity computation between queries and items [6, 21]. Early work focused on personalized representations [15] and BERT-based models [5, 23], while SimCSE [4] introduced contrastive learning for dense retrieval. Recently, LLMs have emerged as powerful backbones for generating universal semantic representations, with models like RepLLaMA [22] and newer state-of-the-art embeddings [11, 38] demonstrating strong retrieval capabilities.

2.2 Reinforcement Learning in Information Retrieval

Reinforcement learning has gained traction for improving LLM performance and controllability. While PPO [28] dominated early RLHF approaches, recent methods like DPO [25] and GRPO [29] offer more efficient alternatives. In retrieval, RL has been applied to query generation [8], relevance reasoning [36], and dense retrieval optimization [19]. DeepRetrieval [8] combines GRPO with LLM to enable unsupervised query generation, offering a more efficient training paradigm for information retrieval. Zeng et al. [36] formulate relevance modeling as a reasoning task and introduce a RL-based training framework to enhance the grounded reasoning capabilities of generative relevance models. Retrieval-GRPO [19] applies GRPO strategy to dense retrieval, which eliminates the dependency on labor-intensive hard negative sample mining while mitigating the seesaw effect inherent in multi-objective optimization.

2.3 Semantic IDs in LLMs and Information Retrieval

Generative Retrieval [18, 26] represents products using Semantic Identifiers (SIDs)—discrete token sequences that encode product semantics into generatable identifiers [16, 35]. SIDs are typically generated through an encoder-quantization framework [3, 12] that maps multimodal features into discrete spaces. This approach enables semantically similar products to receive similar SIDs, facilitating more accurate matching in LLM-based e-commerce retrieval systems.

3 Methods

3.1 Preliminary

Dense retrieval aims to map queries and items into a shared embedding space for efficient relevance matching via geometric proximity. Given query q_i and item collection $\mathcal{D}$, we employ a dual-encoder model $f_\theta(\cdot)$ to encode them as:

$$\mathbf{q_i} = f_\theta(q_i), \quad \mathbf{d_j} = f_\theta(d_j) \in \mathbb{R}^D. \tag{1}$$

Both query and item embeddings are L2-normalized to unit vectors, i.e., $\|\mathbf{q_i}\|_2 = 1$ and $\|\mathbf{d_j}\|_2 = 1$. Under this normalization, retrieval computes similarity scores using inner product:

$$s(q_i, d_j) = \langle \mathbf{q_i}, \mathbf{d_j} \rangle, \tag{2}$$

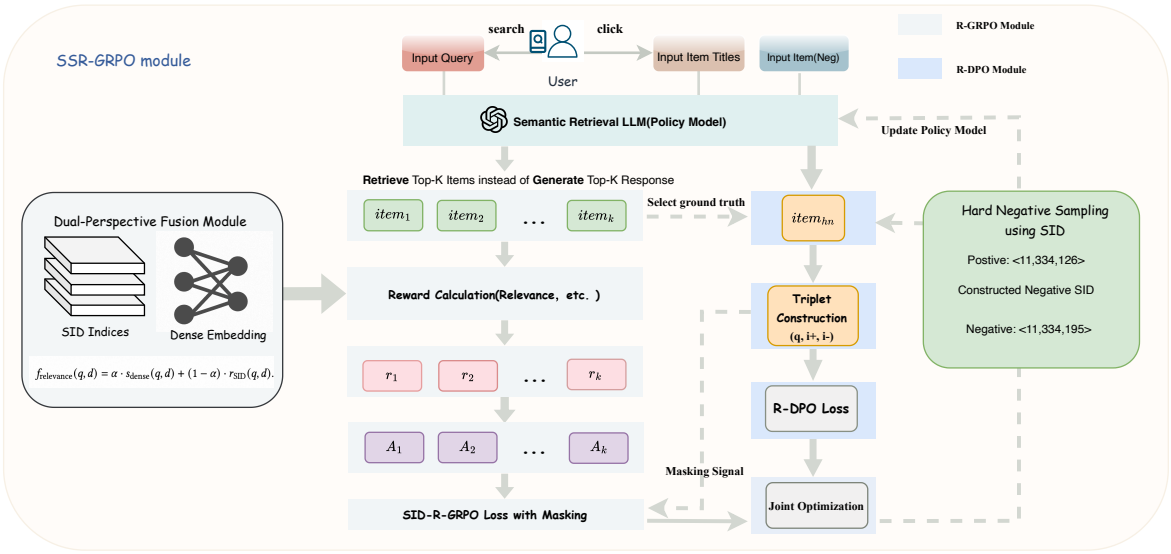


Figure 1: The SSR-GRPO overall training framework consists of four components: (1) a semantic retrieval LLM trained via SFT; (2) R-GRPO module containing the dual-perspective reward calculation based on SIDs and masking mechanism; (3) R-DPO Optimization using hard negatives derived from hierarchical SIDs; (4) SID-R-GRPO and R-DPO joint training based on dynamic weighting.

which is equivalent to computing the cosine similarity between the two vectors. The system then retrieves top-K relevant items:

$$\mathcal{D}_{q_i} = \text{TopK}_{d_j \in \mathcal{D}} s(q_i, d_j). \tag{3}$$

3.2 SFT

Our framework employs a two-stage training process: supervised fine-tuning (SFT) followed by SSR-GRPO. In SFT, (q, d) positive pairs and in-batch negatives [15, 21] are used to pull query representations close to relevant items while pushing them away from negative items via InfoNCE loss [24]:

$$\mathcal{L}_{\text{InfoNCE}} = -\log \frac{\exp(s(q_i, d_j^+)/\tau)}{\sum_{d_j \in \mathcal{B}} \exp(s(q_i, d_j)/\tau)}, \tag{4}$$

where $s(\cdot)$ is the similarity score, $\mathcal{B}$ the in-batch negative samples, and τ a temperature parameter [14, 32] smoothing the fitted data distribution.

3.3 SSR-GRPO

3.3.1 Reward Function in R-GRPO. The reward function in SSR-GRPO follows R-GRPO's multi-objective formulation, combining relevance, quality, and exclusivity components as in Eq. (5):

$$r = f_{\text{relevance}}(q, d) + g_{\text{quality}}(d) + h_{\text{exclusivity}}(q, d). \tag{5}$$

While g_{quality} and $h_{\text{exclusivity}}$ remain identical to R-GRPO, we propose a novel dual-perspective relevance assessment for $f_{\text{relevance}}$. This integrates: (1) a dense signal via inner product between multi-modal embeddings $s(\mathbf{e}_q, \mathbf{e}_d)$, and (2) a sparse signal using SIDs, where relevance is scored based on hierarchical matching depth. The SID construction and scoring strategy are detailed next.

3.3.2 Item SID Learning via Contrastive RQ-VAE.. We adopt a query-bridged contrastive quantization framework to learn hierarchical Semantic Identifiers (SIDs) for items. The process begins with pre-trained relevance embeddings $\mathbf{e}_q = \text{RelevanceEmb}_q(q)$ and $\mathbf{e}_d = \text{RelevanceEmb}_d(d)$, obtained from a dual-tower model trained on large-scale interaction data. Lightweight MLP encoders then project these embeddings into a shared latent space: $\mathbf{z}_q = \text{Enc}_q(\mathbf{e}_q)$, $\mathbf{z}_d = \text{Enc}_d(\mathbf{e}_d)$. A shared Residual Quantized Variational Autoencoder (RQ-VAE) with L(e.g., L=3) levels of codebooks $\{C_1, C_2, \ldots, C_L\}$ performs residual quantization on the item representations:

$$\mathbf{r}^0 = \mathbf{z}_d, \quad c^\ell = \arg\min_{e \in C_\ell} \|\mathbf{r}^{\ell-1} - e\|_2^2, \quad \mathbf{r}^\ell = \mathbf{r}^{\ell-1} - c^\ell, \tag{6}$$

producing the item SID as $\text{SID}_i = [k_1, k_2, \ldots, k_L]$. The RQ-VAE is optimized through a joint training objective that combines a commitment loss, a reconstruction loss and an InfoNCE contrastive loss :

$$\mathcal{L} = \lambda_{\text{infoNCE}} \mathcal{L}_{\text{InfoNCE}} + \lambda_{\text{commit}} \mathcal{L}_{\text{commit}} + \lambda_{\text{recon}} \mathcal{L}_{\text{recon}}. \tag{7}$$

Crucially, the InfoNCE loss encourages items co-retrieved by the same query to share consistent SIDs, turning geometric clustering into relevance-aware hierarchical partitioning.

3.3.3 Query SID Generation via Generative LLM.. We formulate the generation of Query SIDs as a next-token-prediction (NTP) task, fine-tuned on an auto-regressive Large Language Model (LLM). The training data is constructed from historical search logs, where each query is paired with the Semantic Identifiers (SIDs) of items that received positive user interactions—such as clicks, add-to-cart actions, and purchases. These interacted item SIDs serve as the ground-truth targets for the model. The model is trained to minimize the negative

log-likelihood loss:

$$\mathcal{L}_{\text{SFT}} = -\sum_{t=1}^{L} \log P(k_t \mid q, k_{<t}; \theta), \tag{8}$$

where k_t denotes the codebook index at level t and θ represents the LLM parameters. During inference, beam search is employed to generate the top-K most probable SID sequences $\{\text{SID}_q^1, \ldots, \text{SID}_q^K\}$. This approach naturally captures the one-to-many mapping from queries to relevant item clusters, effectively addressing query ambiguity.

3.3.4 Dual-Perspective Relevance Reward with SID Fusion. Given query SIDs $\{\text{SID}_q^j\}_{j=1}^K$ and item SID $\text{SID}_i = [k_1^i, k_2^i, k_3^i]$, we compute the maximum matching depth:

$$\ell^*(q,d) = \max_{j=1}^{K} \max \left\{ \ell \in \{1,\ldots,L\} : k_t^{q,j} = k_t^i,\ \forall t \le \ell \right\}, \tag{9}$$

where a deeper shared prefix signifies finer-grained semantic coherence. This discrete hierarchical score is then mapped to a scalar value:

$$r_{\text{SID}}(q,d) = \begin{cases} 0.0 & \text{if } \ell^* = 0 \quad \text{(No match)}, \\ 0.25 & \text{if } \ell^* = 1 \quad \text{(Coarse-level match)}, \\ 0.5 & \text{if } \ell^* = 2 \quad \text{(Mid-level match)}, \\ 1.0 & \text{if } \ell^* = 3 \quad \text{(Fine-level match)}. \end{cases} \tag{10}$$

The final relevance reward is obtained by fusing the dense embedding similarity with the sparse SID matching score, forming the core of our dual-perspective assessment:

$$f_{\text{relevance}}(q,d) = \alpha \cdot r_{\text{SID}}(q,d) + (1-\alpha) \cdot s_{\text{dense}}(q,d). \tag{11}$$

Here, $\alpha \in [0,1]$ is a balancing hyperparameter. This dual-perspective mechanism leverages the complementary strengths of both representations: the dense embeddings provide a smooth, continuous measure of overall semantic similarity, while the discrete SIDs offer sharp, hierarchical discrimination based on key attribute alignments. Together, they form a more resilient and accurate relevance reward, crucial for guiding the policy optimization in R-GRPO.

3.3.5 Hard Negative Mining and Masking Mechanism. We introduce a novel method which leverages SIDs introduced in Section 3.3.2 for hard negative mining, combined with a masking mechanism. This approach provides explicit supervision signals to augment and stabilize the R-GRPO process. As for the hard negative mining, we select hard negatives from two complementary perspectives to address different aspects of retrieval challenges:

Type I: Commercial Performance-based Hard Negatives. We target items that consistently reach the pre-ranking stage but fail final exposure due to low scores from subsequent ranking stages. These ranking models employ sophisticated cross-features and sequential modeling, providing accurate conversion efficiency assessments. Thus, low-scoring items typically exhibit genuinely poor conversion performance (e.g., low CTR/CVR). Although semantically relevant to queries, they demonstrate weak commercial performance. Using them as hard negatives helps the model better discriminate between semantically relevant but commercially weak items and those with strong conversion potential.

Type II: Semantic Fine-grained Hard Negatives. To capture subtle semantic distinctions, we leverage hierarchical SIDs. For a positive item with $\text{SID}_{i^+} = [k_1, k_2, k_3]$, we select negatives sharing the first two SID levels (k_1, k_2) but differing at the third (k_3). This creates challenging negatives that are semantically proximate at coarse granularity yet distinct in fine-grained attributes, forcing the model to learn nuanced semantic boundaries essential for e-commerce relevance.

To provide explicit supervision signals for filtering out trivial negative samples within the top-K items in R-GRPO, we design a novel masking mechanism using the mined hard negatives as quality benchmarks. In the GRPO framework, these hard negatives (denoted as d_{hn}) serve as quality benchmarks. We apply a masking function designed to filter out easy negative samples by removing any top-K item whose relevance reward score falls below the benchmark. The masking function is defined as:

$$M_i = \begin{cases} 0, & f_{\text{relevance}}(q,d_i) < f_{\text{relevance}}(q,d_{\text{hn}}) \\ 1, & \text{otherwise}, \end{cases} \tag{12}$$

where M_i is a binary function, if the current item's relevance reward is lower than the relevance reward of the hard negative, then $M_i = 0$ (masked out), otherwise $M_i = 1$.

3.3.6 SID-R-GRPO. Combining the dual-perspective relevance reward (Section 3.3.4) with the masking mechanism (Section 3.3.5), we obtain SID-R-GRPO. In SID-R-GRPO, we treat the top-K items retrieved by the semantic retrieval LLM illustrated in Figure 1 as candidates. Using the reward models integrating SIDs from Section 3.3.1, we compute real-time rewards and penalize low-scoring items to dynamically adjust predictions. Candidates are gathered across devices into a large batch $\hat{B}$ ($k \ll \hat{B}$), with top-k selected for reward computation. The inter-group advantage A_i among top-K items under the same query reflects multi-perspective user preferences, determining reward/penalty assignments. The SID-R-GRPO loss is defined as:

$$\mathcal{L}_{\text{SID-R-GRPO}} = -\frac{1}{K}\sum_{i=1}^{K} \Bigg\{ \min \Bigg(\pi_\theta(s(q,d_i)|q,d_i)\hat{A}_i, \\ \text{clip}\left(\pi_\theta(s(q,d_i)|q,d_i), 1-\varepsilon, 1+\varepsilon\right)\hat{A}_i \Bigg) M_i \Bigg\} + \beta \mathcal{L}_{\text{InfoNCE}}, \tag{13}$$

where

$$M_i = \begin{cases} 0, & f_{\text{relevance}}(q,d_i) < f_{\text{relevance}}(q,d_{\text{hn}}) \\ 1, & \text{otherwise}, \end{cases} \tag{14}$$

$$A_i = \frac{r_i - mean(\mathbf{r})}{std(\mathbf{r})}, \tag{15}$$

and K is the size of the sample set composed of top-k for the current query, π_θ is the distribution of the policy model, A_i represents the advantage value of the i-th item in top-k computed by multi-objective rewards. The collection $\mathbf{r} = \{r_1, r_2, \ldots, r_K\}$ represents the set of reward scores for the K candidate items in the group. The function $\text{mean}(\mathbf{r})$ computes the average reward of the group, and $\text{std}(\mathbf{r})$ calculates the corresponding standard deviation. Different from conventional GRPO, we use InfoNCE rather than KL divergence as a semantic regularizer. Token-level KL divergence is ill-suited for continuous embedding spaces, whereas InfoNCE

directly aligns the learned semantic space with the retrieval objective, leading to more stable and effective optimization for semantic retrieval.

3.3.7 R-DPO Loss. We also introduce a retrieval-specific direct preference optimization objective, termed R-DPO, which adapts the DPO framework for vector retrieval tasks by removing the reference policy constraint. It provides explicit supervision for fine-grained semantic distinctions. Specifically, positive pairs (q_i, d_i) are extracted from real user click logs, while negatives d_{hn} are hard negatives mined by the SID-based approach in Section 3.3.5. Given these pairs, the R-DPO loss is formulated as:

$$\mathcal{L}_{\text{R-DPO}} = -\log \sigma \left(\beta \left(s(q_i, d_i) - s(q_i, d_{\text{hn}}) \right) \right), \tag{16}$$

where $s(q, d)$ is the relevance scoring function, $\sigma(\cdot)$ is the sigmoid function, and β is a temperature parameter controlling the margin strength. Unlike conventional DPO that regularizes against a reference policy, our R-DPO focuses purely on maximizing the scoring margin between positive and hard negative examples. This design is particularly suitable for retrieval tasks where we aim to explicitly widen the relevance gap between semantically similar items.

3.3.8 SSR-GRPO: Joint Optimization Framework. By combining the loss formulas of SID-R-GRPO and R-DPO, we obtain the final loss formulation of SSR-GRPO. To mitigate gradient conflicts and eliminate manual tuning of loss weights, we use uncertainty-based dynamic weighting [9] with trainable scalars $\sigma_1, \sigma_2 > 0$:

$$\mathcal{L}_{\text{SSR-GRPO}} = \frac{1}{2\sigma_1^2}\mathcal{L}_{\text{SID-R-GRPO}} + \frac{1}{2\sigma_2^2}\mathcal{L}_{\text{R-DPO}} + \log(\sigma_1\sigma_2), \tag{17}$$

Here, $\frac{1}{2\sigma_i^2}$ acts as an adaptive weight: tasks with higher noise or difficulty (larger σ) are down-weighted to prevent gradient dominance, while $\log(\sigma)$ serves as a regularizer. This mechanism ensures stable convergence by dynamically balancing the high-variance RL signals from SID-R-GRPO with the deterministic supervised gradients from R-DPO, effectively aligning their optimization trajectories without hyperparameter sensitivity.

Based on the above descriptions and optimization steps, the overall pipeline of SSR-GRPO can be summarized and organized in Figure 1 and the specific implementation steps are as follows:

(1) **Candidate Sample Selection:** Retrieve top-k candidates via nearest-neighbor search over query and item representations.
(2) **Dense-Sparse Perspective Reward Computation:** The multi-objective rewards are computed by Equation (5). The relevance reward is comprehensively derived from the dual-perspective dense-sparse relevance reward scores introduced in 3.3.4.
(3) **R-DPO data construction:** Construct R-DPO pairs using hard negative samples obtained from the hierarchical SID-based approach and a related-but-unexposed-item strategy introduced in 3.3.7.
(4) **Masking Mechanism Calculation** Relevance-based hard negatives are employed as supervision to calculate a masking function by Equation (12) for filtering out easy negatives in the group.
(5) **SSR-GRPO Loss Optimization** Minimize Eq. (17), jointly updating model parameters θ and uncertainty scalars $\sigma_{1,2}$.

4 Experiments

In this section, we introduce our industrial-scale dataset from a real-world scenario, evaluation metrics, experimental configurations, and model hyperparameters. We present the results of our proposed model through both offline evaluations and online A/B tests.

4.1 Experimental Setup

4.1.1 Dataset and Metrics. We utilized over 0.3 billion Tmall click-and-purchase logs as positive samples. The model undergoes two-stage training: SFT with InfoNCE loss on the full three-month dataset, followed by SSR-GRPO optimization on recent 15-day interactions. Performance is evaluated via Hitrate@4K (HR@4k) and Goodrate@100 (GR@100) [15, 37].

- **Hitrate@4k**: This metric measures the recall rate of the ground-truth items within the top-4000 candidates retrieved by the dense retrieval model. It evaluates the system's ability to successfully recover all relevant items.
- **Goodrate@100**: This metric is defined as the precision of items rated as "Good" (i.e., score 1 for "Related") among the top-100 retrieved results. It quantifies the relevance quality of the top results presented to users.

Next, we provide a concise description of the two offline test sets:

- **General Retrieval Test Set**: To avoid overfitting, this set is built from a distinct e-commerce scenario's user interactions. It contains clicked and purchased items, filtered by relevance model and human annotation, with over 450,000 queries and 1,330,000 query-item pairs.
- **Long-Tail Query Test Set**: Constructed by collecting long-tail queries from end-to-end search logs across other e-commerce scenarios. These queries, which can be viewed as out-of-distribution data, were annotated using both relevance models and manual labeling, followed by rigorous filtering. The final test set contains over 30,000 queries and 90,000 query-item pairs.

4.1.2 Implementation and Deployment Details. For the SFT stage, we use Tbstars-3B (an internal 3B-parameter model) as our backbone encoder. Training employs 32 GPUs with 96GB memory each, batch size 140, AdamW optimizer (lr=1e-5), max input length 100 tokens, and 128-dimensional L2-normalized embeddings for 1 epoch. In the SSR-GRPO phase, we use 15 days of data with reduced learning rate 5e-6, maintaining other hyperparameters from SFT. The InfoNCE loss weight β in Equation (13) is set to 0.5 for best performance, and training runs for 3 epochs.

Offline Training Process: During offline training, we input item titles and basic attribute texts to the LLM for the item side, and search queries to the LLM for the query side, following the SSR-GRPO training procedure.

Online Deployment: The trained SSR-GRPO LLM is used to infer embeddings for the entire product pool, which are cached online. The LLM is then deployed to production, where it performs real-time inference on user search queries to generate query vectors.

Table 1: Overall performance comparison on the different methods in terms of Hitrate@4k and Goodrate@100

Model	HR@4k		GR@100	
	General	Long-Tail	General	Long-Tail
StructBERT	0.7166	0.4019	0.8837	0.5732
Qwen2.5-1.5B	0.7466	0.4189	0.8894	0.6415
Qwen3-0.6B	0.7495	0.4330	0.8958	0.6524
Tbstars-3B	0.7577	0.4506	0.9088	0.6583
Tbstars-3B SFT (base)	0.7872	0.4754	0.9174	0.6941
Tbstars-3B SFT + R-DPO	0.7916	0.5857	0.9261	0.7014
Tbstars-3B SFT + R-GRPO	0.8158	0.5857	0.9305	0.7126
Tbstars-3B SFT + SSR-GRPO	**0.8218**	**0.5943**	**0.9372**	**0.7195**
Tbstars-3B R-GRPO only	0.7616	0.4492	0.9213	0.6820
Tbstars-3B SSR-GRPO only	0.7628	0.4562	0.9230	0.6844

These vectors retrieve Top-K items from the cached product embeddings, which are subsequently passed to downstream modules.

4.1.3 Baseline Models. We compare SSR-GRPO against three categories of baselines: the production Tmall baseline (Tbstars-3B SFT), the primary competitor R-GRPO, and several state-of-the-art text embedding models. The specific baselines are as follows:

- StructBERT [33]: A structure-aware encoder that integrates linguistic structural priors (e.g., word order and sentence coherence) into pre-training, with 0.1 billion parameters.
- Qwen3-0.6B: The Qwen3-Embedding0.6B model [38] is designed for text embedding and ranking tasks.
- Qwen2.5-1.5B [31]: One of the Qwen2.5 series with 1.5 billion parameters, designed for generating long texts and understanding structured data.
- Tbstars-3B: Dense retrieval model trained based on an internal closed-source Tbstars with 3B parameters, modifying to bi-directional attention.
- Tbstars-3B SFT (base): This model is built upon Tbstars-3B and fine-tuned on Tmall search logs using the InfoNCE loss, which can be viewed as the first phase of SSR-GRPO in Section 3.2. It had been deployed in the online system, serving hundreds of millions of users.
- Tbstars-3B SFT + R-GRPO: The R-GRPO training framework which includes SFT and R-GRPO, representing a direct ablation of our SSR-GRPO framework.

4.2 Offline Evaluation

The experimental results are shown in Table 1. The proposed Tbstars-3B SFT + SSR-GRPO achieves the best performance across all evaluation settings, obtaining the highest scores on both General Retrieval Test and Long-Tail Query test sets. This demonstrates the effectiveness of our SSR-GRPO framework in enhancing both retrieval coverage and relevance quality. Among the baseline models, Tbstars-3B outperforms both Qwen2.5 and Qwen3-0.6B across all metrics, validating its superior architecture for e-commerce retrieval tasks. The Tbstars-3B SFT (base) model shows significant improvements over the pre-trained Tbstars-3B, confirming the value of domain-specific fine-tuning on Tmall search logs. The integration of reinforcement learning methods consistently improves upon the SFT baseline. Tbstars-3B SFT + R-GRPO shows substantial gains, while our proposed SSR-GRPO demonstrates clear advantages over R-GRPO in all metrics (especially showing a 0.6% gain in HR@4k and 0.86% in Long-Tail HR@4k), validating the effectiveness of incorporating supervised signals and SIDs for enhancing relevance reward. Notably, the standalone GRPO variants without SFT initialization show limited performance, indicating the SFT stage remains essential for effective training.

Table 2: Ablation study on the contributions of SSR-GRPO components.

Model	Hitrate@4k	GR@100
SFT+SSR-GRPO (Full)	+0.00%	+0.00%
(w/o Dual-Perspective Reward)	-0.17%	-0.25%
(w/o R-DPO Loss)	-0.49%	-0.36%
(w/o Masking Function)	-0.12%	-0.08%
(w/o Dynamic Weights)	-0.15%	-0.17%

4.3 Ablation Study

In this section, we only present HR@4k on the general evaluation set and GR@100 on the long-tail dataset to provide a more intuitive experimental comparison.

4.3.1 Analysis of SSR-GRPO Components. To validate the effectiveness of each component in SSR-GRPO compared to R-GRPO, we conducted ablation experiments on SSR-GRPO. To perform the ablation studies, we removed each of the three components one at a time. Dual-Perspective Reward represents the component described in Section 3.3.4, R-DPO loss means Equation (16), Masking Function is Equation (12) and Dynamic Weights means the uncertainty-based dynamic weighting in (17). As Table 2 shows, each ablated configuration resulted in a performance degradation across evaluation metrics. Most notably, the removal of the R-DPO Loss caused the most significant drop in HR@4k by 0.49%, underscoring the critical and irreplaceable importance of hard negative mining. This finding also demonstrates the complementary effect of R-DPO and R-GRPO in the task. In contrast to methods solely dependent on reward computation, which are plagued by high variance, the introduction of

Figure 2: Comparison of retrieval results: Top-exposed items from online A/B tests for our model and R-GRPO.

R-DPO furnishes the training process with stable and well-defined gradients, thereby effectively constraining the overall optimization trajectory.

Additionally, SID integration contributes consistently to both metrics, particularly benefiting relevance judgment as evidenced by the 0.25% drop in GR@100 when removed. This highlights the value of combining dense and sparse representations for a comprehensive semantic understanding. Although the masking mechanism contributes the least among the four components, the 0.12% decrease in Hitrate@4K nonetheless validates its effectiveness. Moreover, this masking approach efficiently filters out unnecessary simple negative samples, thereby improving computational efficiency during training.

Table 3: Ablation results on Relevance Reward

Model	HR@4k	GR@100
R-GRPO (w TaoSR1)	0.8152	0.7136
R-GRPO (w Dense)	0.8093	0.7108
R-GRPO (w Sparse SID)	0.8154	0.7143
R-GRPO (w Dual-Perspective SID)	**0.8161**	**0.7165**

4.3.2 Effect of Relevance Reward. SSR-GRPO replaces the computationally heavy TaoSR1 LLM in R-GRPO with a lightweight dual-perspective reward mechanism. As shown in Table 3, the Sparse SID-only variant achieves performance comparable to, and even slightly better than TaoSR1 (HR@4k: 0.8154 vs. 0.8152). This demonstrates that representing items as discrete semantic clusters via SIDs provides a robust and unbiased signal for GRPO reward estimation, effectively capturing coarse-grained relevance without the bias inherent in LLM-based judges. While the Dense-only variant shows limited capability, the Dual-Perspective fusion further boosts performance by integrating fine-grained dense nuances. In our implementation, we set the balancing coefficient $\alpha = 0.8$ in Eq. (11), which yields the optimal trade-off between dense semantic smoothness and sparse hierarchical discrimination.

4.3.3 Case Study. To qualitatively assess the semantic alignment capability of SSR-GRPO, we analyze a complex multi-attribute query: "2026 new stylish UV protection face mask for women running". As illustrated in Figure 2, the baseline R-GRPO retrieves items that are visually appealing but fail to capture the specific "running" scenario. This indicates that R-GRPO is prone to Reward Hacking when the Top-K candidate pool lacks perfect matches. In contrast, SSR-GRPO comprehensively covers the full semantic requirements, retrieving masks that balance high aesthetics with running-specific functionalities (e.g., breathability and secure fit).

The superior performance on this case is attributed to the SID-guided optimization mechanism. By leveraging hierarchical SIDs to construct hard negatives, the R-DPO component forces the model to learn fine-grained semantic distinctions, preventing collapse into coarse-grained local optima. Meanwhile, the masking function filters out trivial negatives. A compelling evidence of this improved coherence is the distribution of SIDs in the retrieved results. As shown in Figure 2, the baseline SIDs are highly dispersed across different semantic clusters (e.g., the second index level varies significantly: 335, 208, 163, . . .). Conversely, SSR-GRPO results exhibit strong clustering consistency, with the second index level concentrating on only two values (208, 108). This concentration confirms our method effectively reduces semantic noise and aligns the retrieved items with the user's precise multi-faceted intent.

4.4 Online Evaluation

We deployed the semantic retrieval model in production search on TmallAPP, a major mobile e-commerce application. Our online evaluation focused on key business metrics: UCTCVR, GMV, Ex-Imp(the proportion of items seen by users that are solely retrieved by the semantic deep retrieval channel in our multi-channel framework) and Goodrate (the mean relevance ratio derived by sampling search queries and corresponding items from search logs).

Table 4: Online A/B testing results (relative improvement over baseline).

Model	UCTCVR	GMV	Ex. Imp.	Goodrate
Base	-	-	-	-
R-GRPO	+0.38%	+1.01%	+0.37%	+0.20%
SSR-GRPO	**+0.99%**	**+1.40%**	**+0.48%**	**+0.61%**

The online A/B test was conducted over a period of two weeks to ensure statistical reliability. Compared to the strong baseline R-GRPO, SSR-GRPO achieved statistically significant improvements, with a 0.61% lift in UCTCVR and a 0.39% increase in GMV. These results further validate the business value of our proposed method in a real-world production environment.

5 Conclusion

This paper presents SSR-GRPO, a novel RL framework for dense retrieval that leverages hierarchical SIDs to provide unbiased relevance rewards, replacing costly LLMs. By integrating SID-based hard negative mining for R-DPO supervision and a masking mechanism to filter noise in R-GRPO, our method ensures stable and precise optimization. Experiments show SSR-GRPO significantly outperforms state-of-the-art baselines, offering superior semantic alignment and robust business value.

References

[1] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*. 4171–4186.

[2] Chenhe Dong, Shaowei Yao, Pengkun Jiao, Jianhui Yang, Yiming Jin, Zerui Huang, Xiaojiang Zhou, Dan Ou, Haihong Tang, and Bo Zheng. 2025. TaoSR1: The thinking model for e-commerce relevance search. *arXiv preprint arXiv:2508.12365* (2025).

[3] Patrick Esser, Robin Rombach, and Bjorn Ommer. 2021. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 12873–12883.

[4] Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. Simcse: Simple contrastive learning of sentence embeddings. *arXiv preprint arXiv:2104.08821* (2021).

[5] Weiwei Guo, Xiaowei Liu, Sida Wang, Huiji Gao, Ananth Sankar, Zimeng Yang, Qi Guo, Liang Zhang, Bo Long, Bee-Chung Chen, et al. 2020. Detext: A deep text ranking framework with bert. In *Proceedings of the 29th ACM international conference on information & knowledge management*. 2509–2516.

[6] Jui-Ting Huang, Ashish Sharma, Shuying Sun, Li Xia, David Zhang, Philip Pronin, Janani Padmanabhan, Giuseppe Ottaviano, and Linjun Yang. 2020. Embedding-based retrieval in facebook search. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2553–2561.

[7] Po-Sen Huang, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry Heck. 2013. Learning deep structured semantic models for web search using clickthrough data. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*. 2333–2338.

[8] Pengcheng Jiang, Jiacheng Lin, Lang Cao, Runchu Tian, SeongKu Kang, Zifeng Wang, Jimeng Sun, and Jiawei Han. 2025. Deepretrieval: Hacking real search engines and retrievers with large language models via reinforcement learning. *arXiv preprint arXiv:2503.00223* (2025).

[9] Alex Kendall, Yarin Gal, and Roberto Cipolla. 2018. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 7482–7491.

[10] Weize Kong, Swaraj Khadanga, Cheng Li, Shaleen Kumar Gupta, Mingyang Zhang, Wensong Xu, and Michael Bendersky. 2022. Multi-aspect dense retrieval. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 3178–3186.

[11] Chankyu Lee, Rajarshi Roy, Mengyao Xu, Jonathan Raiman, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. 2024. Nv-embed: Improved techniques for training llms as generalist embedding models. *arXiv preprint arXiv:2405.17428* (2024).

[12] 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*. 11523–11532.

[13] Jinhyuk Lee, Feiyang Chen, Sahil Dua, Daniel Cer, Madhuri Shanbhogue, Iftekhar Naim, Gustavo Hernández Ábrego, Zhe Li, Kaifeng Chen, Henrique Schechter Vera, et al. 2025. Gemini embedding: Generalizable embeddings from gemini. *arXiv preprint arXiv:2503.07891* (2025).

[14] Mingming Li, Chunyuan Yuan, Binbin Wang, Jingwei Zhuo, Songlin Wang, Lin Liu, and Sulong Xu. 2023. Learning query-aware embedding index for improving e-commerce dense retrieval. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 3265–3269.

[15] Sen Li, Fuyu Lv, Taiwei Jin, Guli Lin, Keping Yang, Xiaoyi Zeng, Xiao-Ming Wu, and Qianli Ma. 2021. Embedding-based product retrieval in taobao search. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 3181–3189.

[16] Xinyu Lin, Haihan Shi, Wenjie Wang, Fuli Feng, Qifan Wang, See-Kiong Ng, and Tat-Seng Chua. 2025. Order-agnostic identifier for large language model-based generative recommendation. In *Proceedings of the 48th international ACM SIGIR conference on research and development in information retrieval*. 1923–1933.

[17] Aixin Liu, Aoxue Mei, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, et al. 2025. Deepseek-v3. 2: Pushing the frontier of open large language models. *arXiv preprint arXiv:2512.02556* (2025).

[18] 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*. 729–739.

[19] Xingxian Liu, Dongshuai Li, Tao Wen, Jiahui Wan, Gui Ling, Fuyu Lv, Dan Ou, and Haihong Tang. 2025. TaoSearchEmb: A Multi-Objective Reinforcement Learning Framework for Dense Retrieval in Taobao Search. *arXiv preprint arXiv:2511.13885* (2025).

[20] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692* (2019).

[21] Yiqun Liu, Kaushik Rangadurai, Yunzhong He, Siddarth Malreddy, Xunlong Gui, Xiaoyi Liu, and Fedor Borisyuk. 2021. Que2search: fast and accurate query and document understanding for search at facebook. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 3376–3384.

[22] Xueguang Ma, Liang Wang, Nan Yang, Furu Wei, and Jimmy Lin. 2024. Fine-tuning llama for multi-stage text retrieval. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2421–2425.

[23] Xichuan Niu, Bofang Li, Chenliang Li, Rong Xiao, Haochuan Sun, Hongbo Deng, and Zhenzhong Chen. 2020. A dual heterogeneous graph attention network to improve long-tail performance for shop search in e-commerce. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 3405–3415.

[24] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748* (2018).

[25] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems* 36 (2023), 53728–53741.

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

[27] Joshua Robinson, Ching-Yao Chuang, Suvrit Sra, and Stefanie Jegelka. 2020. Contrastive learning with hard negative samples. *arXiv preprint arXiv:2010.04592* (2020).

[28] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347* (2017).

[29] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300* (2024).

[30] Parikshit Sondhi, Mohit Sharma, Pranam Kolari, and ChengXiang Zhai. 2018. A taxonomy of queries for e-commerce search. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*. 1245–1248.

[31] Qwen Team et al. 2024. Qwen2 technical report. *arXiv preprint arXiv:2407.10671* 2, 3 (2024).

[32] Binbin Wang, Mingming Li, Zhixiong Zeng, Jingwei Zhuo, Songlin Wang, Sulong Xu, Bo Long, and Weipeng Yan. 2023. Learning multi-stage multi-grained semantic embeddings for e-commerce search. In *Companion Proceedings of the ACM Web Conference 2023*. 411–415.

[33] W Wang, B Bi, M Yan, C Wu, Z Bao, J Xia, L Peng, and L Si. 2019. Structbert: Incorporating language structures into pre-training for deep language understanding. arXiv. *arXiv preprint arXiv:1908.04577* (2019).

[34] Mingrui Wu and Sheng Cao. 2024. Llm-augmented retrieval: Enhancing retrieval models through language models and doc-level embedding. *arXiv preprint arXiv:2404.05825* (2024).

[35] Yanjing Wu, Yinfu Feng, Jian Wang, Wenji Zhou, Yunan Ye, Rong Xiao, and Jun Xiao. 2024. Hi-gen: Generative retrieval for large-scale personalized e-commerce search. *arXiv preprint arXiv:2404.15675* (2024).

[36] Ziyang Zeng, Heming Jing, Jindong Chen, Xiangli Li, Hongyu Liu, Yixuan He, Zhengyu Li, Yige Sun, Zheyong Xie, Yuqing Yang, et al. 2025. Optimizing Generative Ranking Relevance via Reinforcement Learning in Xiaohongshu Search. *arXiv preprint arXiv:2512.00968* (2025).

[37] Han Zhang, Songlin Wang, Kang Zhang, Zhiling Tang, Yunjiang Jiang, Yun Xiao, Weipeng Yan, and Wen-Yun Yang. 2020. Towards personalized and semantic retrieval: An end-to-end solution for e-commerce search via embedding learning. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2407–2416.

[38] Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, et al. 2025. Qwen3 Embedding: Advancing Text Embedding and Reranking Through Foundation Models. *arXiv preprint arXiv:2506.05176* (2025).

GenAI Usage Disclosure

During the preparation of this work, GenAI tools were used solely to improve the spelling and grammar of the author-written text. All ideas, theoretical frameworks, experimental designs, results, and conclusions are the original work of the authors.