Multi-level Relevance Document Identifier Learning for Generative Retrieval

Fuwei Zhang[1], Xiaoyu Liu[1], Xinyu Jia[2], Yingfei Zhang[2], Shuai Zhang[2], Xiang Li[2], Fuzhen Zhuang[1,3]†, Wei Lin[2]†, Zhao Zhang[4]†

[1]Institute of Artificial Intelligence, Beihang University, Beijing, China
[2]Meituan, Beijing, China [3]Zhongguancun Laboratory, Beijing, China
[4]SKLCCSE, School of Computer Science and Engineering, Beihang University, Beijing, China
{zhangfuwei, liuxiaoyv, zhuangfuzhen}@buaa.edu.cn
{jiaxinyu04, zhangyingfei03, zhangshuai51, lixiang245, linwei31}@meituan.com
zhangzhao.cs.ai@gmail.com

Abstract

Generative Retrieval (GR) introduces a new information retrieval paradigm that directly generates unique document identifiers (DocIDs). The key challenge of GR lies in creating effective yet discrete DocIDs that preserve semantic relevance for similar documents while differentiating dissimilar ones. However, existing methods generate DocIDs solely based on the textual content of documents, which may result in DocIDs with weak semantic connections for similar documents due to variations in expression. Therefore, we propose using queries as a bridge to connect documents with varying relevance levels for learning improved DocIDs. In this paper, we propose **M**ulti-l**E**vel **R**elevance document identifier learning for **G**enerative r**E**trieval (MERGE), a novel approach that utilizes multi-level document relevance to learn high-quality DocIDs. MERGE incorporates three modules: a multi-relevance query-document alignment module to effectively align document representations with related queries, an outer-level contrastive learning module to capture binary-level relevance, and an inner-level multi-level relevance learning module to distinguish documents with different relevance levels. Our approach encodes rich hierarchical semantic information and maintains uniqueness across documents. Experimental results on real-world multilingual e-commerce search datasets demonstrate that MERGE significantly outperforms existing methods, underscoring its effectiveness. The source code is available at `https://github.com/zhangfw123/MERGE`.

1 Introduction

Information Retrieval (IR) plays a vital role in helping users find relevant information from large datasets, with applications spanning web search (Liu et al., 2017; Zhang et al., 2020, 2022a) and question answering (Karpukhin et al., 2020a; Nie et al., 2020; Liu et al., 2024a,b; Zhang et al., 2022b, 2024a,b). Traditionally, IR methods are categorized into sparse retrieval and dense retrieval. Sparse retrieval, such as BM25 (Robertson et al., 2009), relies on probabilistic ranking using term frequency, inverse document frequency, and document length normalization to rank documents effectively. As IR evolved, dense retrieval emerged, leveraging deep learning to transform queries and documents into dense vectors, thereby capturing more nuanced semantic relationships and enhancing retrieval accuracy (Karpukhin et al., 2020b; Ni et al., 2022b; Xiong et al., 2020). Recently, the development of pre-trained language models (PLMs) has given rise to Generative Retrieval (GR) (Kuo et al., 2024; Li et al., 2025; Wang et al., 2024; Pan et al., 2024). A notable method is the Differentiable Search Index (DSI) (Tay et al., 2022a), which leverages the contextual understanding capabilities of PLMs to generate document identifiers (DocIDs) directly from queries, thereby obviating the need for additional indexing or ranking mechanisms. GR demonstrates strong query understanding capabilities, enabling it to handle complex long-tail queries in e-commerce scenarios (Yuan et al., 2024; Wu et al., 2024b).

One critical aspect of learning in GR is the generation of DocIDs. In GR, documents are typically represented as sequences of tokens to generate their corresponding DocIDs. A well-constructed DocID not only maintains semantic relevance for each document but also ensures unique identification, allowing similar documents to share similar IDs. While several studies have explored DocID learning (Tang et al., 2024; Li et al., 2023; Wang et al., 2023), most remain preliminary in terms of incorporating relevance, often introducing relevance only at a binary level or not at all. Many GR-related works generally employ standard methods to obtain DocIDs and subsequently focus on improvements during

†Corresponding authors: Fuzhen Zhuang, Wei Lin, and Zhao Zhang

Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 10066–10080
July 27 - August 1, 2025 ©2025 Association for Computational Linguistics

the model training phase. In this paper, we argue that DocID is crucial for effective GR, and relying solely on a document's representation is insufficient for generating high-quality DocIDs. Instead, it is essential to utilize multi-level relevance information between queries and documents. Specifically, queries can act as a bridge to establish connections among documents. Typically, the relevance between a query and a document set captures explicit similarities among documents, with those exhibiting higher relevance levels aligning more closely with the given query. Incorporating this information into DocID learning can effectively capture hierarchical semantics. This approach ensures that semantically related documents produce similar DocIDs while distinguishing between documents with varying levels of relevance under the same query. Consequently, this ensures the uniqueness and discriminability of the generated IDs.

To this end, we propose a novel approach named **M**ulti-l**E**vel **R**elevance document identifier learning for **G**enerative r**E**trieval (MERGE), designed to generate high-quality DocIDs. Our method captures multi-level relevance through a learnable DocID generation method named Residual Quantization Variational Autoencoder (RQ-VAE) (Rajput et al., 2023), which employs multi-layer codebooks to generate hierarchical indices for documents. First, we design a multi-relevance query-document alignment mechanism that aligns document representations with their related queries. To further ensure the relevance of similar documents, we introduce an outer-level contrastive learning module that captures binary-level relevance by drawing relevant documents closer together while pushing irrelevant ones apart. Additionally, to enhance differentiation among relevant documents, we propose an inner-level multi-level relevance learning strategy that distinguishes documents with different levels of relevance. By integrating these components, we achieve semantically rich and well-differentiated DocIDs, effectively training the GR model for improved retrieval performance. Here, we summarize our contributions:

- We incorporate multi-level relevance into the learning process for DocID generation, aiming to generate semantically rich and well-differentiated DocIDs.

- To effectively capture multi-level relevance, we design a query-document alignment mechanism alongside outer-level contrastive learning and inner-level multi-level relevance learning, aiming to capture semantic relevance across different levels.

- We perform experiments across three languages using a challenging e-commerce product search dataset, and the results clearly demonstrate the superiority of our method.

2 Related Work

2.1 Sparse and Dense Retrieval

Sparse and dense retrieval are fundamental in information retrieval. Sparse retrieval, like BM25 (Robertson et al., 2009), focuses on term-document matching. Dense retrieval, exemplified by Dense Passage Retrieval (DPR) (Karpukhin et al., 2020b), uses neural embeddings for semantic matching. ANCE (Xiong et al., 2020) enhances this by improving training efficiency. Hybrid models, such as COIL (Ma et al., 2021), combine sparse and dense approaches to leverage both term interactions and dense representations. Sentence-transformers (Reimers, 2019) further advance dense retrieval by generating high-quality sentence embeddings for better semantic understanding. Recent works, like learned sparse representations by Zhou et al., bridge the gap between traditional and neural methods.

2.2 Generative Retrieval

Recent advances in generative retrieval have introduced various innovative approaches. DSI (Tay et al., 2022a) transforms documents into DocIDs and utilizes transformers for end-to-end retrieval, while SE-DSI (Tang et al., 2023) enhances this with semantic learning strategies. SEAL (Bevilacqua et al., 2022) introduces autoregressive search engines generating substrings as DocIDs. GenRRL (Zhou et al., 2023) incorporates reinforcement learning to obtain the relevance feedback. LTRGR (Li et al., 2024) utilizes the ranking task to optimize GR models. NOVO (Wang et al., 2023) generates learnable document identifiers, and RIPOR (Zeng et al., 2024) focuses on scalable generative retrieval frameworks. Wu et al. proposed multi-vector dense retrieval, and GDR (Yuan et al., 2024) addresses memory efficiency in generative dense retrieval. SEATER (Si et al., 2023) employs Constrained K-means to construct a balanced K-ary tree, adding an alignment loss to improve token relationship understanding. GenRet (Sun et al., 2024) uses an Encoder-Decoder structure to generate ID tokens step-by-step. Hi-gen (Wu et al., 2024b) utilizes prior category information for clustering with k-means. D2-DocID (Cheng et al., 2025) introduces learnable document identifiers that are both descriptive and discriminative. GR^2 (Tang et al., 2024) incorporates multi-graded relevance during the retrieval process and implements multi-graded constrained contrastive training. However, during the ID encoding phase, it only uses the simplest relevant and non-relevant information for contrastive learning, which is significantly different from our approach. Our method aims to incorporate multi-level relevance information during the encoding phase to achieve two objectives: **(1) ensuring that DocIDs of irrelevant documents are more distant, and (2) ensuring that DocIDs of relevant documents are closer.**

3 Methodology

In the following, we present MERGE, including the DocID generation method and the model training process. Given a query q, the set of documents relevant to q is represented as: $\mathcal{D}_q = \{\mathcal{D}_q^1, \ldots, \mathcal{D}_q^L\}, \mathcal{D}_q^n = \{d_1^{q,n}, d_2^{q,n}, \ldots\} (q \in \mathcal{Q})$ where $\mathcal{Q}$ is the set of all queries, $\mathcal{D}_q^n$ represents the n-th level relevant document set of q, and L denotes the highest relevance level, representing the documents most relevant to the query q. We employ a pre-trained language model, such as BERT (Kenton and Toutanova, 2019) or T5 (Raffel et al., 2020), to encode the document text into dense vector embeddings. These embeddings are subsequently utilized to train the RQ-VAE (Rajput et al., 2023) method. During the learning process, we implement multi-level relevance learning by crafting loss functions with distinct objectives, thereby producing high-quality DocIDs. For GR model training, we utilize queries as input and DocIDs as output to train the sequence-to-sequence model.

3.1 DocID Learning via Multi-level Relevance

In retrieval tasks, the number of documents is large. Previous GR methods of statically constructing IDs, such as hierarchical clustering, cannot effectively capture the relationships between documents. To achieve relevance learning, we adopt a learnable DocID generation method named RQ-VAE (Rajput et al., 2023), which is a multi-level vector quantization method that generates semantic IDs by recursively quantizing residuals at each level using separate codebooks. Additionally, its multi-level construction of ID facilitates the introduction of multi-level relevance in our tasks.

3.1.1 Basic DocID Learning

The RQ-VAE process consists of three key phases: **Document Encoding.** Given a document d with text information (e.g., title or content), we first extract its semantic embedding $\mathbf{d}$ using PLMs. **Residual Quantization (RQ).** RQ encodes the input embedding $\mathbf{d}$ to learn a latent representation $\mathbf{z} = E(\mathbf{d})$ (E is a deep neural network (DNN) encoder that maps the input embedding to a low-dimensional vector), which serves as the initial residual at the 0-th level, denoted as $\mathbf{r}_0 = \mathbf{z}$. At each quantization level l, there is a codebook $\mathcal{C}^l = \{\mathbf{e}_k^l\}_{k=1}^K$, where $\mathbf{e}_k^l$ is the k-th vector in codebook at level l and K is the codebook size. To prevent codebook collapse, each codebook is initialized using k-means clustering (the cluster number is the codebook size k) on the latent representations $\{\mathbf{z}_i\}$ from the first training batch.

The initial residual $\mathbf{r}_0$ is then used to find the index of the nearest embedding in $\mathcal{C}^0$, given by $c_0 = \arg\min_k \|\mathbf{r}_0 - \mathbf{e}_k^0\|_2$. More generally, at each level l, the residual update follows the rule $c_l = \arg\min_k \|\mathbf{r}_l - \mathbf{e}_k^l\|_2$, and the residual is iteratively updated as $\mathbf{r}_{l+1} = \mathbf{r}_l - \mathbf{e}_{c_l}^l$. This recursive residual approximation ultimately generates an ID tuple $(c_0, \ldots, c_{m-1})$ with a coarse-to-fine granularity, where m is the maximum level of RQ. **Reconstruction & Training.** After generating DocIDs, the quantized representation $\hat{\mathbf{z}} = \sum_{l=0}^{m-1} \mathbf{e}_{c_l}^l$ is fed into a DNN decoder D to reconstruct the input $\mathbf{d}$ via a reconstruction loss:

$$\mathcal{L}_{\text{recon}} = \|\mathbf{d} - D(\hat{\mathbf{z}})\|_2^2. \tag{1}$$

The training objective combines reconstruction loss $\mathcal{L}_{\text{recon}}$ with codebook commitment $\mathcal{L}_{\text{rq}}$:

$$\mathcal{L}_{\text{RQ-VAE}} = \mathcal{L}_{\text{recon}} + \mathcal{L}_{\text{rq}},$$
$$\mathcal{L}_{\text{rq}} = \sum_{l=0}^{m-1} \left(\|\text{sg}[\mathbf{r}_l] - \mathbf{e}_{c_l}^l\|_2^2 + \alpha \|\mathbf{r}_l - \text{sg}[\mathbf{e}_{c_l}^l]\|_2^2 \right), \tag{2}$$

where $\text{sg}[\cdot]$ denotes the stop-gradient operation, which prevents gradient updates for the quantized embeddings during backpropagation. The first term in $\mathcal{L}_{\text{rq}}$ ensures that the codebook vectors $\mathbf{e}_{c_l}^l$ are close to the corresponding residuals $\mathbf{r}_l$. The second term, weighted by the hyperparameter α, constrains the residuals to remain close to the selected codebook entries.

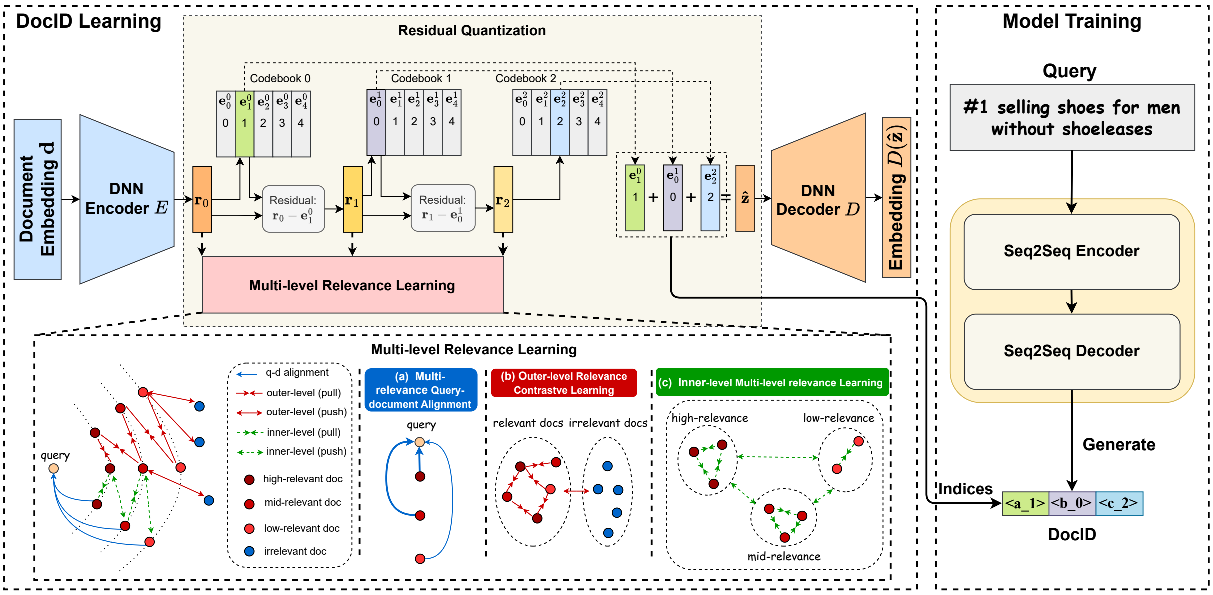

DocID Learning
Residual Quantization
Codebook 0
Codebook 1
Codebook 2
$\mathbf{e}_0^0$ $\mathbf{e}_1^0$ $\mathbf{e}_2^0$ $\mathbf{e}_3^0$ $\mathbf{e}_4^0$
0 1 2 3 4
$\mathbf{e}_0^1$ $\mathbf{e}_1^1$ $\mathbf{e}_2^1$ $\mathbf{e}_3^1$ $\mathbf{e}_4^1$
0 1 2 3 4
$\mathbf{e}_0^2$ $\mathbf{e}_1^2$ $\mathbf{e}_2^2$ $\mathbf{e}_3^2$ $\mathbf{e}_4^2$
0 1 2 3 4
Document Embedding d
DNN Encoder E
$\mathbf{r}_0$
Residual: $\mathbf{r}_0 - \mathbf{e}_1^0$
$\mathbf{r}_1$
Residual: $\mathbf{r}_1 - \mathbf{e}_0^1$
$\mathbf{r}_2$
$\mathbf{e}_1^0$ 1 + $\mathbf{e}_0^1$ 0 + $\mathbf{e}_2^2$ 2 = $\hat{\mathbf{z}}$
DNN Decoder D
Embedding $D(\hat{\mathbf{z}})$
Multi-level Relevance Learning
Model Training
Query
#1 selling shoes for men without shoeleases
Seq2Seq Encoder
Seq2Seq Decoder
Generate
Indices
<a_1> <b_0> <c_2>
DocID
Multi-level Relevance Learning
query
q-d alignment
outer-level (pull)
outer-level (push)
inner-level (pull)
inner-level (push)
high-relevant doc
mid-relevant doc
low-relevant doc
irrelevant doc
(a) Multi-relevance Query-document Alignment
query
(b) Outer-level Relevance Contrastve Learning
relevant docs
irrelevant docs
(c) Inner-level Multi-level relevance Learning
high-relevance
low-relevance
mid-relevance

Figure 1: Overall architecture of MERGE. (a) Multi-relevance Query-document Alignment, which aligns the document representations to the related queries with different levels. (b) Outer-level Relevance Contrastive Learning, which is employed to differentiate between relevant and irrelevant documents. (c) Inner-level Multi-level relevance Learning, which is utilized to distinguish between varying levels of relevance among documents.

3.1.2 Multi-level Relevance DocID Learning

To incorporate multi-level relevance into DocID, we design Multi-relevance Query-document alignment, Outer-level Relevance Contrastive Learning, and Inner-level Multi-level Relevance Learning to optimize the learning of RQ-VAE.

Multi-relevance Query-document alignment. In the context of document retrieval, similar documents might have different expressions, leading to embeddings generated by PLMs that may not position relevant documents in close proximity. To address this, we introduce a query-document alignment mechanism during text encoding. Specifically, in real-world datasets, queries are typically short, whereas documents are often much longer, resulting in inconsistencies in the distribution of the encoding space. To avoid this, we represent a query by averaging the embeddings of the document set $\mathcal{D}_q^L$ with the highest relevance level under the given query, formulated as follows:

$$\mathbf{q} = \frac{1}{|\mathcal{D}_q^L|} \sum_{i=1}^{|\mathcal{D}_q^L|} \mathbf{z}_i^{q,L}, \tag{3}$$

where $\mathbf{z}_i^{q,L}$ denotes the embedding of the i-th document in the set $\mathcal{D}_q^L$ obtained by RQ encoder E.

Then, we introduce a hierarchical alignment loss function to align the representations of all relevant documents of query q, as follows:

$$\mathcal{L}_{align} = \frac{1}{|Q|} \sum_{q \in Q} w_j \sum_{j=1}^{L} \frac{1}{|\mathcal{D}_q^j|} \sum_{k=1}^{|\mathcal{D}_q^j|} \text{Dist}(\mathbf{d}_k^{q,j} - \mathbf{q}). \tag{4}$$

We adopt the cosine similarity as distance for alignment. $w_j = 1/(L - j + 1)$ represents the alignment weight for the j-th level of relevance.

Outer-level Relevance Contrastive Learning. The core of this module lies in **binary-level relevance** learning, which seeks to discriminate irrelevant documents and simultaneously enhance the connections among relevant documents. Specifically, let $\mathcal{D}_q^{\text{rel}} = \{d_1^q, d_2^q, \ldots\}$ denote the document set relevant to query q in a training batch D^{batch}. We employ InfoNCE (Oord et al., 2018) loss at each quantization layer l to pull the residual vector of relevant documents closer while pushing irrelevant ones apart, as follows:

$$\mathcal{L}_{\text{outer}}^{l} = \sum_{d_i^q, d_j^q \in \mathcal{D}_q^{\text{rel}}} \log \frac{\exp(\text{sim}(\mathbf{r}_l^{d_i^q}, \mathbf{r}_l^{d_j^q})/\tau)}{\sum_{d \in \mathcal{D}^{\text{batch}}} \exp(\text{sim}(\mathbf{r}_l^{d_i^q}, \mathbf{r}_l^{d})/\tau)}, \tag{5}$$

where τ controls similarity distribution sharpness, $\text{sim}(\cdot,\cdot)$ is cosine similarity, and $\mathbf{r}_l^d$ denotes the residual vector at the l-th quantization layer for document d in RQ.

Inner-level Multi-level Relevance Learning. After learning binary-level relevance, the model still fails to perceive the different relevance levels among documents, leading to similar DocIDs being assigned to all relevant documents. To address this, we further introduce **inner-level multi-level relevance learning**, which optimizes document representations across different relevance levels, thereby enabling the DocID assignments to distinguish documents with different levels of relevance. Specifically, for a query q, we employ a triplet loss (Schroff et al., 2015) to enforce differentiation among documents of different relevance levels, formulated as follows:

$$\begin{aligned}\mathcal{L}^{l}_{\text{inner}} = \sum_{(d,d^+,d^-)\in\mathcal{T}} & \max(0,\gamma+ \\ & \text{sim}(\mathbf{r}^{d}_{l},\mathbf{r}^{d^-}_{l}) - \text{sim}(\mathbf{r}^{d}_{l},\mathbf{r}^{d^+}_{l})),\end{aligned} \tag{6}$$

where $\mathcal{T}$ is the set of learning triplets. $\mathbf{r}^{d}_{l}, \mathbf{r}^{d^+}_{l}$, and $\mathbf{r}^{d^-}_{l}$ represent the residuals at l-th level of document d, d^+, and d^-, respectively. $d, d^+ \in \mathcal{D}^{a}_{q}$ and $d^- \in \mathcal{D}^{b}_{q}$ $(a > b)$, ensuring that the positive document d^+ and anchor document d have higher relevance levels than the negative document d^-. This encourages the model to learn a representation space where more relevant documents are closer to the query than less relevant ones, thereby distinguishing different levels of relevance effectively.

3.1.3 Full DocID Learning

Finally, we incorporate the aforementioned multi-level relevance learning method in the learning progress of RQ-VAE. For outer-level and inner-level learning in section 3.1.2 and 3.1.2, we introduce a trade-off weight β_l $(0 \leq \beta_l \leq 1)$ for the l-th layer in RQ to ensure distinct optimization objectives across different codebook layers. Specifically, in the early layers of RQ, we prioritize optimizing $\mathcal{L}_{\text{outer}}$ to ensure that irrelevant documents are assigned to different ID tokens. In contrast, in the later layers, we strengthen optimizing $\mathcal{L}_{\text{inner}}$ to further differentiate documents with different levels of relevance. This hierarchical optimization strategy enables the generated DocIDs to encode both hierarchical semantic information and discriminative capacity. The combined relevance learning loss is formulated as follows:

$$\mathcal{L}_{\text{rel}} = \frac{1}{m}\sum_{l=0}^{m-1}\left(\beta_l\mathcal{L}^{l}_{\text{outer}} + (1-\beta_l)\mathcal{L}^{l}_{\text{inner}}\right). \tag{7}$$

Here β_l decreases as the layer l increases. The newly formulated training objective for RQ-VAE is presented as follows:

$$\mathcal{L}_{\text{IDgen}} = \mathcal{L}_{\text{RQ-VAE}} + \lambda_1\mathcal{L}_{\text{align}} + \lambda_2\mathcal{L}_{\text{rel}}, \tag{8}$$

where λ_1, λ_2 are trade-off parameters to balance the weights of losses.

For DocIDs that remain colliding generated by RQ-VAE, we adopted the Sinkhorn algorithm(Cuturi, 2013) to re-assign unique DocIDs.

3.2 Model Training

Using the trained RQ-VAE model, we generate a unique ID for each document. If a document is assigned the ID tuple (1, 0, 2), we organize it into a DocID in the form of: `<a_1><b_0><c_2>`. Each element in this sequence, such as `<a_1>`, represents a distinct token added to the vocabulary for training.

We map the documents in the query-document pairs of the training data to the constructed DocIDs, forming the training data for GR. Specifically, we place a special token `<retrieval>` at the beginning of the query text to indicate that the model needs to perform a retrieval task. The retrieval task aims to generate a unique DocID as the optimization target. To make the model aware that the earlier generated tokens are more important, we introduce a position weight to improve the original loss function. Finally, we train the sequence-to-sequence transformer model (Raffel et al., 2020; Xue et al., 2021)) with the following loss function:

$$\mathcal{L} = -\sum_{t=1}^{T} w_t \cdot \log P(y_t|y_{<t}, q), \tag{9}$$

where q is the input query, T denotes the length of DocID, y_t is the t-th token of the target DocID, $y_{<t}$ represents the previously generated tokens, and $w_t = 1/\sqrt{t}$ is the position weight at position t, weighting scheme ensures that the model pays more attention to the accuracy of the earlier tokens in the DocID.

4 Experiment

In this section, we analyze our model's effectiveness and address: 1) **RQ1:** How does MERGE compare to state-of-the-art baselines? 2) **RQ2:** How do different modules impact MERGE's performance? 3) **RQ3:** How do different loss weights affect MERGE? 4) **RQ4:** What is the quality of MERGE-generated DocIDs?

4.1 Experimental Setup

Datasets. We utilize the publicly available ESCI dataset (Reddy et al., 2022), a large-scale, multilingual dataset for query-product semantic matching. ESCI includes challenging search queries with up to 40 potentially relevant products per query, annotated with relevance labels (Exact, Substitute, Complement, Irrelevant). Each query-product pair is enriched with additional metadata, spanning English, Japanese, and Spanish. Notably, the dataset features many queries with negation conditions (e.g., "not" and "without") and numerous product attribute constraints (e.g., size, price, functionality). These factors result in low term overlap between queries and documents, posing significant challenges for retrieval models. Appendix A presents more information of datasets.

Baselines. To evaluate our model, we use three baseline categories: sparse, dense, and generative retrieval models. For sparse retrieval, we use BM25 (Robertson et al., 2009). For dense retrieval, we consider DPR (Karpukhin et al., 2020c), Sentence-T5 (Ni et al., 2022a) for English, and multilingual MPNet (Song et al., 2020) for other languages. For generative retrieval, we include DSI (Tay et al., 2022a) (DSI$_{\text{naive}}$ and DSI$_{\text{semantic}}$), NCI (Wang et al., 2022), LTRGR (Li et al., 2024), and RIPOR (Zeng et al., 2024). Appendix B provides a detailed description of the baselines.

Evaluation Metrics. We evaluate our model using **Recall@k** (R@k) and **NDCG@k**, two standard metrics for ranking tasks. **Recall@k** measures the fraction of relevant items retrieved in the top-k results, reflecting the model's ability to identify relevant candidates. **NDCG@k** evaluates the ranking quality by considering the position of relevant items, with higher weights given to top-ranked results. We choose Recall@10, 100 (R@10, 100), and NDCG@100 as the evaluation metrics, which are widely used in information retrieval.

Implementation Details. We reproduce the results of all baseline models on the ESCI dataset using official open-source implementations to ensure consistency and comparability. For English-language datasets, we employ a pre-trained T5-base model as the backbone for generative retrieval, while for datasets in other languages, we use the mT5-base model. During the DocID learning stage, these models are utilized to extract semantic representations of documents. For the Codebook configuration, we define the DocID length as 4, which requires the use of four codebooks, each with a size of 256. The RQ-VAE is trained for 300 epochs using a batch size of 2048 and the AdamW optimizer. The λ_1 for $\mathcal{L}_{\text{align}}$ is set to 0.01. The λ_2 for $\mathcal{L}_{\text{rel}}$ is set to 0.001. The β_l balancing the inner- and outer-level losses are set as $\beta_0 = 1.0, \beta_1 = 0.75, \beta_2 = 0.5, \beta_3 = 0.25$. All experiments are conducted on a computing platform equipped with eight A100 80G GPUs. For detailed hyperparameter settings and additional implementation specifics, please refer to the Appendix C.

4.2 Performance of MERGE (RQ1)

1) MERGE outperforms state-of-the-art GR models on three ESCI datasets, demonstrating its effectiveness. Additionally, it surpasses the dense retrieval model Sentence-T5 in NDCG@100 and RECALL@10, highlighting its ability to generate highly relevant documents. 2) While RIPOR demonstrates competitive performance (achieving second-best results in GR models), its training methodology necessitates a complex multi-phase optimization procedure requiring multiple warm-up stages, which introduces significant implementation complexity. In contrast, our approach establishes superior performance by generating high-quality DocIDs and employing a simple generative model training, which is simple and effective. 3) MERGE significantly outperforms the vanilla RQ-VAE, indicating the importance of learning high-quality DocIDs.

4.3 Ablation Studies (RQ2)

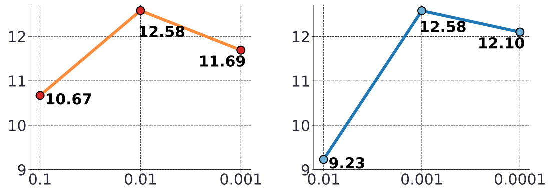
To explore the roles of different modules in MERGE, we present experimental results of several variants of MERGE on ESCI-English, as shown in Table 2. Here, *w/o* $\mathcal{L}_{\text{align}}$ indicates the removal of the loss $\mathcal{L}_{\text{align}}$; *w/o* $\mathcal{L}_{\text{outer}}$ refers to using only the loss $\mathcal{L}_{\text{inner}}$, which means setting β_l to 0.0; *w/o* $\mathcal{L}_{\text{inner}}$ sets β_l to 1.0; and *w/o* w_t means not using positional weights during model training. We draw the following conclusions: 1) Performance drops after removing each component, confirming the effectiveness of the three losses and weighted training strategy. 2) Removing the outer-level relevance contrastive loss $\mathcal{L}_{\text{outer}}$ leads to the largest decline, highlighting the critical role of relevance in DocID learning and its ability to mitigate poor encoding quality from low-quality text. 3) All MERGE variants outperform vanilla RQ-VAE, underscoring the importance of each module.

Table 1: Performance of different models on ESCI in terms of R@10, 100 (%) and NDCG@100 (%). The best results are highlighted in **bold**, while the second-best results are underlined in all groups.

Type	Model	ESCI-English			ESCI-Spanish			ESCI-Japanese		
		R@10	R@100	NDCG@100	R@10	R@100	NDCG@100	R@10	R@100	NDCG@100
Sparse	BM25(2009)	0.28	1.41	1.07	0.71	2.21	1.91	0.23	1.29	1.11
Dense	DPR(2020c)	5.43	23.79	10.95	4.61	21.86	10.42	6.02	21.36	11.12
	sentence-T5(2022a)	6.35	**27.85**	11.64	-	-	-	-	-	-
	multilingual MPNet(2019)	2.99	12.24	5.32	2.72	11.62	5.71	1.21	4.22	2.30
Generative	DSI$_{\text{naive}}$(2022a)	0.49	1.74	1.36	0.97	4.63	3.08	0.42	2.36	1.59
	DSI$_{\text{semantic}}$(2022a)	1.88	7.21	3.94	7.26	26.40	15.64	5.99	22.30	13.33
	NCI(2022)	4.22	15.39	8.40	7.66	29.02	16.64	6.20	22.84	14.00
	vanilla RQ-VAE(2023)	5.19	19.52	9.94	8.03	29.50	17.21	6.24	23.01	14.09
	LTRGR(2024)	2.68	10.55	8.67	7.85	28.56	17.88	6.38	23.31	14.45
	RIPOR(2024)	5.49	22.17	10.82	8.73	31.28	18.20	6.43	23.80	14.47
	MERGE	**6.73**	25.02	**12.58**	**9.86**	**32.39**	**18.79**	**6.67**	**25.92**	**15.42**

Table 2: Ablation Study on ESCI-English dataset.

Model	R@10	R@100	NDCG@100
MERGE	**6.73**	**25.02**	**12.58**
w/o $\mathcal{L}_{\text{align}}$	6.22	22.91	11.51
w/o $\mathcal{L}_{\text{outer}}$	6.14	20.94	10.81
w/o $\mathcal{L}_{\text{inner}}$	6.23	23.52	11.69
w/o w_t	6.34	23.62	11.86
vanilla RQ-VAE	5.19	19.52	9.94



(a) λ_1

(b) λ_2

Figure 2: Parameter analysis on λ_1 and λ_2.

4.4 Parameter Analysis (RQ3)

To investigate the impact of different weights of losses, we conducted experiments with varying values of λ_1 and λ_2. The value of λ_1 determines the weight of the multi-relevance query-document alignment, while λ_2 controls the multi-level relevance learning among documents. Figure 2 presents the NDCG@100 results on the ESCI-English dataset for different combinations of λ_1 and λ_2. As shown in Figure 2(a), as the weight of the multi-relevance query-document alignment decreases, the NDCG@100 score initially increases and then decreases. This is because an excessively large λ_1 forces the representations of query-related documents to cluster too closely, reducing the discriminability among documents. Conversely, an overly small λ_1 leads to insufficient alignment, resulting in performance degradation. Figure 2(b) illustrates the results under different levels of relevance learning, which also exhibit an initial increase followed by a decrease. We hypothesize that this trend arises because the magnitude of λ_2 determines the degree to which document representations are pulled closer or pushed apart, thus leading to the observed pattern.

4.5 Analysis of Generated DocIDs (RQ4)

To further validate the quality of the DocIDs generated by MERGE, we conduct a statistical analysis to count the number of unique ID tokens assigned by each layer of RQ for the same query. Table 3 shows the average Unique ID Count per layer across all queries. A smaller value indicates a higher overlap of relevant DocIDs for the same query. As can be seen from the table, the DocIDs generated by MERGE allocate fewer and more concentrated ID tokens in the first three layers, while the vanilla RQ-VAE exhibits a more dispersed distribution. In the final layer, the number of ID tokens allocated by MERGE is similar to that of vanilla RQ-VAE, which further demonstrates the goal of our proposed DocID: an ID that incorporates better hierarchical semantic information while maintaining independence.

To further explore the distribution across different ID layers, we illustrates the connections among

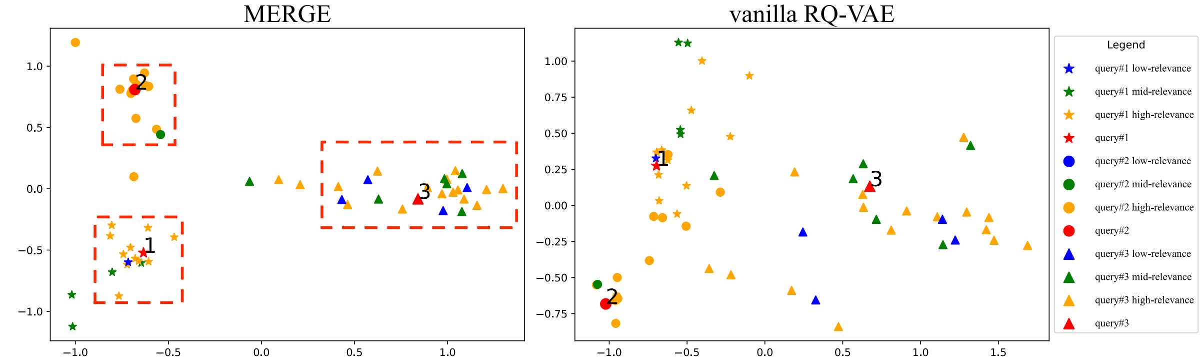


Figure 3: Visualization of different documents under three queries: 1) query #1: “car wash cannon”. 2) query #2: “sasquatch cookie cutter”. 3) query #3: “logitech mx master 3”.

Table 3: Average unique ID count per layer for all queries.

Model	Layer_1	Layer_2	Layer_3	Layer_4
MERGE	4.65	12.00	13.85	15.39
vanilla RQ-VAE	6.55	12.90	14.54	15.56

different layers of DocIDs in Figure 4. Red nodes denote different queries, while other colors represent the first through fourth layers of tokens in DocIDs, with each layer comprising 256 tokens. The node size signifies the degree of connectivity, and the edge thickness indicates the extent of overlapping edges. It is evident that MERGE more effectively achieves the objective that relevant documents have overlap distributions at the lower levels of DocIDs (e.g., layer 1 and 2), demonstrating a proclivity to assign identical IDs. Conversely, the distributions at the higher layers of DocIDs is similar to the vanilla RQ-VAE, exhibiting a more dispersed allocation. This suggests that MERGE can utilize queries as bridges to effectively capture document hierarchies, generating semantically meaningful and distinct DocIDs.

Besides, we analyze three distinct queries, visualizing their document representations encoded by DNN encoder of RQ-VAE via PCA (Figure 3). The visualization indicates that the vanilla RQ-VAE results exhibit some overlap in the document representations associated with these three queries. Notably, some results for query #2 (depicted by circles) in the vanilla RQ-VAE are positioned near query #1, despite the two queries being largely unrelated. In contrast, our model, MERGE, demonstrates a superior ability to distinguish these documents, as emphasized by the square dashed box. As indicated by the rectangular dashed box, our model effectively draws numerous highly relevant documents (shown in orange) closer to the query (depicted in red), thus demonstrating the effectiveness of our carefully designed multi-level relevance learning approach. For additional analysis of DocIDs, see Appendix D.

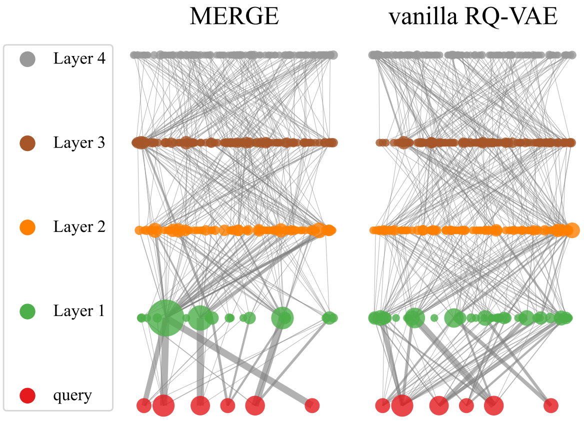


Figure 4: Layer distribution of DocIDs to six queries.

5 Conclusion

This paper explores opportunities to enhance GR by more comprehensively aligning DocID learning with the multi-level relevance observed between queries and documents. To address this, we propose **M**ulti-l**E**vel **R**elevance document identifier learning for **G**enerative r**E**trieval (MERGE), a novel approach that leverages multi-level relevance learning to generate high-quality DocIDs. MERGE learns effective DocIDs by aligning document representations with queries and incorporating binary-level and multi-level relevance. We design relevance learning modules, including query-document alignment, outer-level contrastive learning, and inner-level multi-level relevance learning, and introduce positional weighting to emphasize earlier tokens. We conduct extensive experiments to demonstrate the effectiveness of MERGE. Finally, we performed extensive analysis to verify the quality of the DocIDs generated by MERGE.

6 Limitations

Although MERGE demonstrates superior capability in capturing multi-level relevance between queries and documents, it still faces limitations inherent to generative retrieval (GR) approaches. GR, which relies on generative language models, remains substantially less efficient compared to traditional dense retrieval (DR). Furthermore, while it shows advantages in generating more relevant documents, the framework exhibits inferior document coverage performance relative to DR. This is evident in the RECALL@100 metric on the ESCI-English dataset, where MERGE underperforms compared to DR baselines.

Acknowledgements

This work was supported by the National Key Research and Development Program of China under Grant No. 2024YFF0729003, the National Natural Science Foundation of China under Grant Nos. 62176014, 62276015, 62206266, the Fundamental Research Funds for the Central Universities.

References

Michele Bevilacqua, Giuseppe Ottaviano, Patrick Lewis, Scott Yih, Sebastian Riedel, and Fabio Petroni. 2022. Autoregressive search engines: Generating substrings as document identifiers. *Advances in Neural Information Processing Systems*, 35:31668–31683.

Jiehan Cheng, Zhicheng Dou, Yutao Zhu, and Xiaoxi Li. 2025. Descriptive and discriminative document identifiers for generative retrieval. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 11518–11526.

Marco Cuturi. 2013. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26.

Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen Tau Yih. 2020a. Dense passage retrieval for open-domain question answering. In *2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020*, pages 6769–6781. Association for Computational Linguistics (ACL).

Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen Tau Yih. 2020b. Dense passage retrieval for open-domain question answering. In *2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020*, pages 6769–6781. Association for Computational Linguistics (ACL).

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020c. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781.

Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of naacL-HLT*, volume 1. Minneapolis, Minnesota.

Tzu-Lin Kuo, Tzu-Wei Chiu, Tzung-Sheng Lin, Sheng-Yang Wu, Chao-Wei Huang, and Yun-Nung Chen. 2024. A survey of generative information retrieval. *arXiv preprint arXiv:2406.01197*.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466.

Xiaoxi Li, Jiajie Jin, Yujia Zhou, Yuyao Zhang, Peitian Zhang, Yutao Zhu, and Zhicheng Dou. 2025. From matching to generation: A survey on generative information retrieval. *ACM Transactions on Information Systems*, 43(3):1–62.

Yongqi Li, Nan Yang, Liang Wang, Furu Wei, and Wenjie Li. 2023. Multiview identifiers enhanced generative retrieval. In *61st Annual Meeting of the Association for Computational Linguistics, ACL 2023*, pages 6636–6648. Association for Computational Linguistics (ACL).

Yongqi Li, Nan Yang, Liang Wang, Furu Wei, and Wenjie Li. 2024. Learning to rank in generative retrieval. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 8716–8723.

Jiaqi Liu, Zhiwen Yu, Bin Guo, Cheng Deng, Luoyi Fu, Xinbing Wang, and Chenghu Zhou. 2024a. Evolvekg: a general framework to learn evolving knowledge graphs. *Frontiers of Computer Science*, 18(3):183309.

Qi Liu, Qinghua Zhang, Fan Zhao, and Guoyin Wang. 2024b. Uncertain knowledge graph embedding: an effective method combining multi-relation and multi-path. *Frontiers of Computer Science*, 18(3).

Shichen Liu, Fei Xiao, Wenwu Ou, and Luo Si. 2017. Cascade ranking for operational e-commerce search. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1557–1565.

I Loshchilov. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Xinyu Ma, Jiafeng Guo, Ruqing Zhang, Yixing Fan, and Xueqi Cheng. 2021. Contextualized late interaction over dense and sparse representations for information retrieval. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2401–2405.

Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. Ms marco: A human-generated machine reading comprehension dataset.

Jianmo Ni, Gustavo Hernandez Abrego, Noah Constant, Ji Ma, Keith Hall, Daniel Cer, and Yinfei Yang. 2022a. Sentence-t5: Scalable sentence encoders from pre-trained text-to-text models. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1864–1874.

Jianmo Ni, Chen Qu, Jing Lu, Zhuyun Dai, Gustavo Hernandez Abrego, Ji Ma, Vincent Zhao, Yi Luan, Keith Hall, Ming-Wei Chang, et al. 2022b. Large dual encoders are generalizable retrievers. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9844–9855.

Ping Nie, Yuyu Zhang, Xiubo Geng, Arun Ramamurthy, Le Song, and Daxin Jiang. 2020. Dc-bert: Decoupling question and document for efficient contextual encoding. In *Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval*, pages 1829–1832.

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

Shirui Pan, Linhao Luo, Yufei Wang, Chen Chen, Jiapu Wang, and Xindong Wu. 2024. Unifying large language models and knowledge graphs: A roadmap. *IEEE Transactions on Knowledge and Data Engineering*, 36(7):3580–3599.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67.

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*, 36:10299–10315.

Chandan K Reddy, Lluís Màrquez, Fran Valero, Nikhil Rao, Hugo Zaragoza, Sambaran Bandyopadhyay, Arnab Biswas, Anlu Xing, and Karthik Subbian. 2022. Shopping queries dataset: A large-scale esci benchmark for improving product search. *arXiv preprint arXiv:2206.06588*.

N Reimers. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Stephen Robertson, Hugo Zaragoza, et al. 2009. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4):333–389.

Florian Schroff, Dmitry Kalenichenko, and James Philbin. 2015. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823.

Zihua Si, Zhongxiang Sun, Jiale Chen, Guozhang Chen, Xiaoxue Zang, Kai Zheng, Yang Song, Xiao Zhang, Jun Xu, and Kun Gai. 2023. Generative retrieval with semantic tree-structured item identifiers via contrastive learning. *arXiv preprint arXiv:2309.13375*.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. Mpnet: Masked and permuted pretraining for language understanding. *Advances in neural information processing systems*, 33:16857–16867.

Weiwei Sun, Lingyong Yan, Zheng Chen, Shuaiqiang Wang, Haichao Zhu, Pengjie Ren, Zhumin Chen, Dawei Yin, Maarten Rijke, and Zhaochun Ren. 2024. Learning to tokenize for generative retrieval. *Advances in Neural Information Processing Systems*, 36.

Yubao Tang, Ruqing Zhang, Jiafeng Guo, Jiangui Chen, Zuowei Zhu, Shuaiqiang Wang, Dawei Yin, and Xueqi Cheng. 2023. Semantic-enhanced differentiable search index inspired by learning strategies. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 4904–4913.

Yubao Tang, Ruqing Zhang, Jiafeng Guo, Maarten de Rijke, Wei Chen, and Xueqi Cheng. 2024. Generative retrieval meets multi-graded relevance. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Yi Tay, Vinh Tran, Mostafa Dehghani, Jianmo Ni, Dara Bahri, Harsh Mehta, Zhen Qin, Kai Hui, Zhe Zhao, Jai Gupta, et al. 2022a. Transformer memory as a differentiable search index. *Advances in Neural Information Processing Systems*, 35:21831–21843.

Yi Tay, Vinh Tran, Mostafa Dehghani, Jianmo Ni, Dara Bahri, Harsh Mehta, Zhen Qin, Kai Hui, Zhe Zhao, Jai Gupta, et al. 2022b. Transformer memory as a differentiable search index. *Advances in Neural Information Processing Systems*, 35:21831–21843.

Wenjie Wang, Honghui Bao, Xinyu Lin, Jizhi Zhang, Yongqi Li, Fuli Feng, See-Kiong Ng, and Tat-Seng Chua. 2024. Learnable item tokenization for generative recommendation. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pages 2400–2409.

Yujing Wang, Yingyan Hou, Haonan Wang, Ziming Miao, Shibin Wu, Qi Chen, Yuqing Xia, Chengmin Chi, Guoshuai Zhao, Zheng Liu, et al. 2022. A neural corpus indexer for document retrieval. *Advances in Neural Information Processing Systems*, 35:25600–25614.

Zihan Wang, Yujia Zhou, Yiteng Tu, and Zhicheng Dou. 2023. Novo: learnable and interpretable document identifiers for model-based ir. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pages 2656–2665.

Shiguang Wu, Wenda Wei, Mengqi Zhang, Zhumin Chen, Jun Ma, Zhaochun Ren, Maarten de Rijke, and Pengjie Ren. 2024a. Generative retrieval as multi-vector dense retrieval. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1828–1838.

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

Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul N Bennett, Junaid Ahmed, and Arnold Overwijk. 2020. Approximate nearest neighbor negative contrastive learning for dense text retrieval. In *International Conference on Learning Representations*.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mt5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498.

Peiwen Yuan, Xinglin Wang, Shaoxiong Feng, Boyuan Pan, Yiwei Li, Heda Wang, Xupeng Miao, and Kan Li. 2024. Generative dense retrieval: Memory can be a burden. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2835–2845.

Hansi Zeng, Chen Luo, Bowen Jin, Sheikh Muhammad Sarwar, Tianxin Wei, and Hamed Zamani. 2024. Scalable and effective generative information retrieval. In *Proceedings of the ACM on Web Conference 2024*, pages 1441–1452.

Fuwei Zhang, Zhao Zhang, Xiang Ao, Dehong Gao, Fuzhen Zhuang, Yi Wei, and Qing He. 2022a. Mind the gap: Cross-lingual information retrieval with hierarchical knowledge enhancement. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 4345–4353.

Fuwei Zhang, Zhao Zhang, Xiang Ao, Fuzhen Zhuang, Yongjun Xu, and Qing He. 2022b. Along the time: Timeline-traced embedding for temporal knowledge graph completion. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, pages 2529–2538.

Fuwei Zhang, Zhao Zhang, Fuzhen Zhuang, Zhiqiang Zhang, Jun Zhou, and Deqing Wang. 2024a. Multi-view temporal knowledge graph reasoning. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pages 4263–4267.

Fuwei Zhang, Zhao Zhang, Fuzhen Zhuang, Yu Zhao, Deqing Wang, and Hongwei Zheng. 2024b. Temporal knowledge graph reasoning with dynamic memory enhancement. *IEEE Transactions on Knowledge and Data Engineering*, 36(11):7115–7128.

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*, pages 2407–2416.

Yujia Zhou, Zhicheng Dou, and Ji-Rong Wen. 2022. Learning sparse representations for end-to-end dense retrieval. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2359–2364.

Yujia Zhou, Zhicheng Dou, and Ji-Rong Wen. 2023. 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.

A Details of Datasets

In the process of selecting a dataset, we require a publicly available dataset with multi-level relevance annotations. The commonly used evaluation datasets for existing GR models, such as NQ (Kwiatkowski et al., 2019) and MS-MARCO (Nguyen et al., 2016) (NQ is an open-domain question-answering dataset released by Google, and MSMARCO is a dataset released by Microsoft), are binary-level relevance datasets. Moreover, the number of documents relevant to each query is almost always one, making them unsuitable for our multi-level relevance learning. Therefore, we did not choose these two datasets for evaluation. To validate the effectiveness of multi-level relevance learning, we considered search datasets from the e-commerce domain, where retrieval often needs to account for different levels of relevance. Consequently, we employed the ESCI dataset[0] (Reddy et al., 2022), an Amazon e-commerce product search dataset.

[0] https://github.com/amazon-science/esci-data

The ESCI dataset is a comprehensive dataset designed to advance research in the semantic matching of queries and products. It includes challenging search queries and provides up to 40 potentially relevant results for each query, along with ESCI relevance judgments (Exact, Substitute, Complement, Irrelevant) that indicate the relevance of the product to the query. Each query-product pair is enriched with additional information. The dataset is multilingual, featuring queries in English, Japanese, and Spanish. Each example within the dataset contains the following fields: *example_id, query, query_id, product_id, product_locale, esci_label, small_version, large_version, split, product_title, product_description, product_bullet_point, product_brand, product_color, and source*. The primary aim of releasing this dataset is to establish a benchmark for developing new ranking strategies and simultaneously identifying interesting result categories, such as substitutes, to enhance the customer experience during product searches. One of the key tasks explored in the literature using this dataset is Query-Product Ranking, where the objective is to rank the products such that relevant products are positioned above non-relevant ones. Table 4 gives an example from ESCI dataset. From the given example, it is evident that the query encompasses both positive product conditions and negative constraints. In this query, the text in red font represents the size, the text in blue font indicates the color, the text in orange font denotes the name of items, the text in green font signifies certain negation conditions, and the final text in black font describes some functional attributes. Items rated as E (Exact) optimally fulfill the query's requirements. Conversely, items rated as S (Substitute) and C (Complement), while partially matching the query, fail to meet the criteria set by the negative conditions.

Due to the absence of some documents from the training set in the test set, the DocIDs in the test set might not have been learned. Therefore, we filtered out the untrained documents from the test set. We choose the small version of ESCI for evaluation. Table 5 presents the statistics of ESCI dataset.

B Details of Baselines

Here, we will provide detailed information about baselines:

- **BM25** (Robertson et al., 2009): BM25 is a classic sparse retrieval model that emphasizes term-document matching by leveraging term frequency and inverse document frequency for effective information retrieval.
- **DPR** (Karpukhin et al., 2020c): DPR employs neural embeddings to facilitate semantic matching, enhancing retrieval effectiveness by capturing deeper contextual relationships between queries and documents.
- **Sentence-T5** (Ni et al., 2022a): Sentence-T5 advances dense retrieval by generating high-quality sentence embeddings that improve semantic understanding.
- **Multilingual MPNet** (Song et al., 2020): Multilingual MPNet extends dense retrieval capabilities to multiple languages, utilizing transformer-based embeddings to capture semantic nuances across diverse linguistic contexts.
- **DSI$_{\text{naive}}$** (Tay et al., 2022a): DSI$_{\text{naive}}$ employs numerical IDs as DocIDs and performs end-to-end retrieval using transformers.
- **DSI$_{\text{semantic}}$** (Tay et al., 2022a): DSI$_{\text{semantic}}$ utilizes a hierarchical k-means approach to cluster document representations, combining category indices from each layer to form the DocID.
- **vanilla RQ-VAE** (Rajput et al., 2023): The vanilla RQ-VAE generates DocIDs using the RQ-VAE method without incorporating any additional modules. Subsequently, it trains either the T5 or mT5 model to perform retrieval tasks.
- **NCI** (Wang et al., 2022): NCI leverages neural architectures to improve document retrieval effectiveness.
- **LTRGR** (Li et al., 2024): LTRGR utilizes an auxiliary ranking task to optimize already trained GR models.
- **RIPOR** (Zeng et al., 2024): RIPOR emphasizes scalable generative retrieval frameworks, providing robust solutions for handling large-scale retrieval tasks efficiently.

C Implementation Details

For all baseline models, since they have not been evaluated on the ESCI dataset (Reddy et al., 2022),

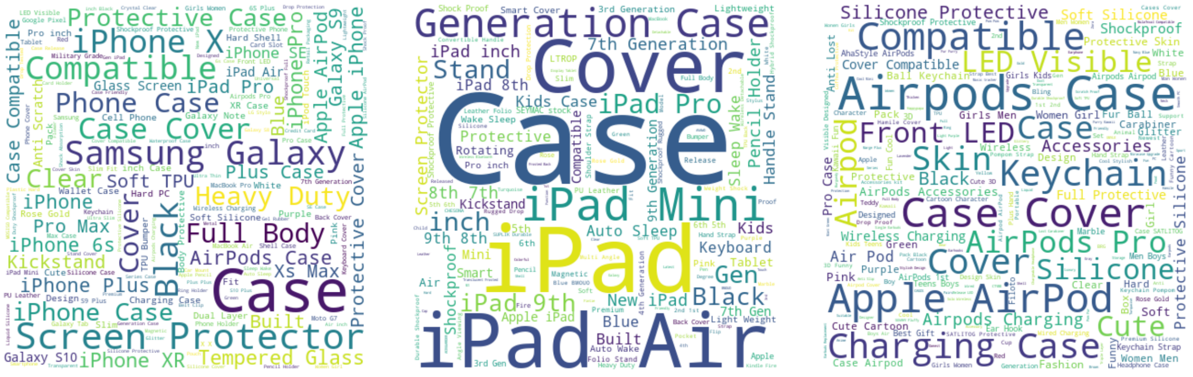
Table 4: Example of ESCI-English dataset.

Query	Relevance Label	Product Title
1-1/2 inch black sink drain without overflow -pop	E(Exact)	Pop Up Sink Drain Stopper Without Overflow, Bathroom Faucet Lavatory Vessel Pop Up, Black
1-1/2 inch black sink drain without overflow -pop	S(Substitute)	KES Bathroom Sink Drain with Strainer Basket Hair Catcher Anti Clog Pop Up Drain Stopper Vanity Vessel Sink with Overflow, Matte Black S2013A-BK
1-1/2 inch black sink drain without overflow -pop	C(Complement)	Universal Bathtub Stopper for Shower and Jacuzzi Drain Stopper, Kitchen Silicone Sink Stopper (Black1)
1-1/2 inch black sink drain without overflow -pop	I(Irrelevant)	Royal Imports 5lb Small Decorative Ornamental River Pebbles Rocks for Fresh Water Fish Animal Plant Aquariums, Landscaping, Home Decor etc. with Netted Bag, Small - Natural

Table 5: Statistics of the Dataset

		Train		Test	
Dataset	#Documents	# Queries	# Q-D Pairs	# Queries	# Q-D Pairs
ESCI-English (US)	482,105	20,888	348,537	6,803	37,220
ESCI-Spanish (ES)	167,761	5,632	126,419	1,743	14,763
ESCI-Japanese (JP)	233,850	7,284	174,640	2,210	18,482

which contains multi-level relevance labels, we reproduce their results on this dataset using the official open-source implementations. While RIPOR, NCI, and LTRGR all employ doc2query-generated pseudo queries (PQs, a technique proven to enhance performance) (Tay et al., 2022b), our method operates without relying on this component. To ensure fairness: For LTRGR — which explicitly states model-agnostic design — we implement their framework on our DSI baseline excluding PQs. For RIPOR and NCI retain their original PQ implementations due to deep architectural dependencies. All these models are trained based on T5-base (Raffel et al., 2020) or mT5-base (Xue et al., 2021). When reproducing the Dense retrieval model, the candidate set for each query consists of all the documents, which is closer to practical application scenarios as it provides a comprehensive assessment of the model's performance in real-world conditions. For English-language datasets, we employ a pre-trained T5-base (Raffel et al., 2020) model as the backbone for generative retrieval. For datasets in other languages, we use mT5-base (Xue et al., 2021) as the backbone. During the DocID learning stage, we leverage T5 or mT5 to extract semantic representations of documents. Regarding the Codebook configuration, we define the DocID length as 4, requiring four codebooks, each with a size of 256. In the training of RQ-VAE, we random sample one relevant document for each document in a batch for relevance learning. we set the hyperparameters as follows: the weight α of the commitment loss $\mathcal{L}rq$ is set to 1.0; in $\mathcal{L}rel$, the weights β_l balancing the inner- and outer-level losses are



(a) First Layer ID:<a_253> (b) Second Layer ID:<a_253><b_36> (c) Second Layer ID:<a_253><b_240>

Figure 5: Case studies of Word Clouds on the generative DocIDs.

Table 6: Comparison of generated DocIDs for the same queries with different levels of relevance using MERGE and vanilla RQ-VAE.

Query	Model	Relevance Label	Product ID	Generated IDs
size 16 white jeans for girls	MERGE	E	B009NGM9PG B00502KEGS	<a_220><b_102><c_227><d_33> <a_220><b_102><c_227><d_245>
		S	B07LFTL35F B076MNHVVT B07F1BTMQD	<a_220><b_102><c_191><d_31> <a_220><b_102><c_145><d_215> <a_220><b_102><c_106><d_42>
	vanilla RQ-VAE	E	B009NGM9PG B00502KEGS	<a_22><b_71><c_97><d_125> <a_22><b_71><c_186><d_191>
		S	B07LFTL35F B076MNHVVT B07F1BTMQD	<a_87><b_13><c_179><d_110> <a_22><b_15><c_180><d_70> <a_87><b_97><c_92><d_7>

set as $\beta_0 = 1.0, \beta_1 = 0.75, \beta_2 = 0.5, \beta_3 = 0.25$; the weight λ_1 of the alignment loss $\mathcal{L}_{\text{align}}$ is set to 0.01; the weight of $\mathcal{L}_{\text{rel}}$ is set to 0.001; the temperature τ in $\mathcal{L}_{\text{outer}}$ is set to 0.7; and the margin γ in $\mathcal{L}_{\text{inner}}$ is set to 0.2. RQ-VAE is trained for 300 epochs with a batch size of 2048 using the AdamW optimizer (Loshchilov, 2017). During the model training phase, we use a learning rate of 5e-4 and train for 100 epochs. All experiments are conducted on a computing platform equipped with eight A100 80G GPUs. We employed AI exclusively for grammatical refinement and sentence polishing.

D Other Analysis of DocIDs

D.1 Word clouds of product titles at different ID levels

We provide several case studies to further illustrate the DocIDs. Figure 5 presents word clouds of product titles at different ID levels. Figure 5(a) shows the word cloud generated from product titles with the first-layer ID <a_253>, which primarily represents protective cases for electronic products of various brands, including Apple and Samsung. Figures 5(b) and 5(c) display the word clouds of product titles corresponding to two sub-IDs under <a_253>. It can be observed that <a_253><b_36> mostly represents cases for tablets, while <a_253><b_240> focuses on cases for earphones, such as airpods. This demonstrates that our model can effectively learn fine-grained category information of products and assign appropriate DocIDs.

D.2 DocIDs comparison between MERGE and vanilla RQ-VAE

We present several cases to demonstrate the DocIDs generated by the MERGE model and the vanilla RQ-VAE. Table 6 shows the DocIDs with different relevance levels encoded by the two models under the same query. The red font indicates the common prefix of DocIDs generated by the MERGE model,

Table 7: Comparison of generated DocIDs for two similar queries with different levels of relevance using MERGE and vanilla RQ-VAE.

Model	Query	Relevance Label	Product ID	Generated IDs
MERGE	black pride blackout curtains	E S	B07TB5DQYH B08DKNMJ7Z	<a_41><b_250><c_169><d_76> <a_41><b_116><c_9><d_246>
	bathroom curtains window pink	E S	B07XDFYSJG B078PRT27T	<a_41><b_146><c_185><d_128> <a_41><b_160><c_148><d_135>
vanilla RQ-VAE	black pride blackout curtains	E S	B07TB5DQYH B08DKNMJ7Z	<a_215><b_237><c_64><d_231> <a_168><b_255><c_170><d_6>
	bathroom curtains window pink	E S	B07XDFYSJG B078PRT27T	<a_144><b_58><c_140><d_155> <a_104><b_220><c_40><d_96>

while the blue font indicates the common prefix for vanilla RQ-VAE. It can be observed that the DocIDs of relevant documents generated by the MERGE model have a longer common prefix and can distinguish documents of different relevance levels. In contrast, the results from vanilla RQ-VAE are less effective, as relevant documents may not be assigned the same first-digit ID.

Table 7 presents the results of documents under two similar queries. Both queries are related to "curtain." From these, we can also observe that the query acts as an effective bridge, as the results generated by MERGE have a common first-digit ID prefix, distinguishing different documents only at the second digit. In contrast, vanilla RQ-VAE assigns different and non-overlapping DocIDs to each item.