A Comparative Analysis of Generative and Dense Retrieval

Anonymous ACL submission

Abstract

Generative retrieval (GR) offers an alternative to dense retrieval (DR) by directly generating identifiers of documents relevant to a query. Relatively little is known about how the two approaches relate. We investigate, theoretically and empirically, how GR fundamentally differs from DR in both learning objective and representational capacity. GR performs globally normalized maximum-likelihood optimization and encodes corpus and relevance information directly in the model parameters, whereas DR adopts locally normalized objectives and represents the corpus with external embeddings before computing similarity via a bilinear interaction. Our analysis suggests that, under scaling, GR can overcome the inherent limitations of DR, yielding two major benefits. First, with larger corpora, GR avoids the sharp performance degradation caused by the optimization drift induced by DR's local normalization. Second, with larger models, GR's representational capacity scales with parameter size, unconstrained by the global low-rank structure that limits DR. We validate these theoretical insights through experiments on the Natural Questions and MS MARCO datasets, across varying negative sampling strategies, embedding dimensions, and model scales. However, despite its theoretical advantages, GR does not universally outperform DR in practice. We outline directions to bridge the gap between GR's theoretical potential and practical performance, providing guidance for future research in scalable and robust generative retrieval.

1 Introduction

Advances in deep learning and representation learning (Vaswani et al., 2017; Devlin et al., 2018) have established neural information retrieval (IR) as the dominant retrieval paradigm (Mitra et al., 2018; Fan et al., 2022). Within this paradigm, dense retrieval (DR) encodes queries and documents into vectors and measures their similarity through bilinear interactions, delivering state-of-the-art performance across retrieval tasks (Karpukhin et al., 2020; Khattab and Zaharia, 2020). Recently, driven by large language models (LLMs) (Radford et al., 2018; Yang et al., 2025b; Lewis et al., 2020), generative retrieval (GR) has emerged as a new branch of neural IR (Tay et al., 2022; Bevilacqua et al., 2022; Zhuang et al., 2022; Wang et al., 2022; Li et al., 2024; Zeng et al., 2024). GR directly generates identifiers of relevant documents (docids) for a given query, with corpus knowledge embedded in the model parameters. It typically adopts a sequence-to-sequence architecture trained with cross-entropy loss, while inference relies on constrained decoding to ensure valid docids.

Recent studies have examined the connection of GR to DR. Under strict assumptions, GR can be viewed as implicitly performing dot-product scoring within an LLM's parameters, giving rise to a unified framework for similarity computation across both paradigms (Nguyen and Yates, 2023; Wu et al., 2024). Beyond these assumptions, substantial architectural differences stand out: DR is encoder-only, and GR employs an autoregressive model with a decoder. This raises the question:

> *Do GR and DR fundamentally differ in their modeling mechanisms for retrieval?*

We proceed along two dimensions: (i) *learning objective*: DR trains with local normalization over a small candidate set in document space, whereas GR maps the problem to vocabulary space and optimizes a globally normalized likelihood; and (ii) *representational capacity*: DR encodes queries and documents as low-dimensional embeddings; GR uses the model parameters to memorize the corpus.

Our **theoretical analysis** examines these aspects and leads to the following conclusions: DR has intrinsic bottlenecks in both learning and representation that constrain its performance under scaling of corpus and model size, whereas GR does not.

First, local normalization in DR introduces calibration errors that grow with corpus size, whereas GR's global normalization avoids such optimization drift and benefits more from larger corpora. Second, the low-rank constraint imposed by DR's embedding dimension limits its ability to approximate the (often higher-rank) true query-document relevance matrix, whereas GR's parameterization allows higher-rank approximations, making it better suited to leverage large-scale models.

Unlike prior studies that analyze GR and DR only from a partial perspective, e.g., empirically examining the effect of DR embedding dimensionality (Karpukhin et al., 2020; Luan et al., 2021) or offering intuitions that GR has stronger representational capacity (Lee et al., 2022), we provide a theoretical characterization showing that GR and DR differ fundamentally in their learning objectives and representational capacity, leading to distinct scaling properties.

To **empirically** validate our theoretical analysis, we evaluate standard DR, multi-vector DR (MVDR, Khattab and Zaharia, 2020; Formal et al., 2021; Li et al., 2023), and two GR variants following the DSI (Tay et al., 2022) framework on the Natural Questions (NQ, Kwiatkowski et al., 2019) and MS MARCO (Bajaj et al., 2016) datasets. We conduct three studies: (i) By varying DR's negative sampling and embedding dimension, we evaluate their effects on calibration error and ranking metrics; experimental results show optimization limits due to local normalization and representation limits due to the embedding dimension. (ii) By scaling GR and DR with matched model sizes and training corpus sizes, we observe larger gains for GR, providing evidence that GR has the potential to overcome DR's bottlenecks when scaled. (iii) Using a larger model with 14B parameters, we conduct zero-shot and test-time scaling experiments for GR and observe promising performance, further supporting the potential scaling advantages of GR.

Our analyses reveal when and why GR's theoretical advantages arise at larger data and model scales. In practice, GR does not consistently outperform DR, as its performance depends on factors such as docid design, training data, and decoding strategies. We conclude by discussing these limitations and suggest directions to close the gap between its theoretical potential and practical results.

2 Preliminaries

Problem statement. Let $\mathcal{Q}$ be a set of queries and $\mathcal{D} = d_1, \ldots, d_N$ a document collection. Let $P^\star(d \mid q)$ denote the unknown ground-truth conditional distribution of documents given query q. Training pairs (q, d^+) are drawn from a data distribution $\mathcal{D}_{\text{train}}$, where d^+ is a relevant document under $P^\star(\cdot \mid q)$. The goal of IR is to approximate $P^\star(d \mid q)$ using a parametric model $P_\Theta(d \mid q)$, ensuring both probabilistic calibration and high ranking quality (Chowdhury, 2010).

Dense retrieval. Let $e_q \in \mathbb{R}^r$ and $e_d \in \mathbb{R}^r$ denote the query and document embeddings from encoders f_q and f_d, respectively (Karpukhin et al., 2020). DR computes relevance via inner-product scoring, $S(q, d) = e_q^\top e_d$, and is trained with a locally normalized (e.g., in-batch) softmax loss:

$$P_\Theta(d \mid q; \mathcal{N}) = \frac{\exp(S(q,d)/\tau)}{\sum_{d' \in \{d\} \cup \mathcal{N}(q)} \exp(S(q,d')/\tau)}, \quad (1)$$

where $\mathcal{N}(q)$ is the negative set and $\tau > 0$ is a temperature. The standard contrastive objective is:

$$\mathcal{L}_{\text{DR}}(\Theta) = \mathbb{E}_q\big[-\log P_\Theta(d^+ \mid q; \mathcal{N}(q))\big]. \quad (2)$$

Eq. 2 encourages $S(q, d^+)$ to exceed the scores of negatives within the current candidate pool. In practice, negatives may come from in-batch sampling (Karpukhin et al., 2020; Khattab and Zaharia, 2020) or hard-negative mining (Xiong et al., 2020; Zhan et al., 2021).

Generative retrieval. Each document has a tokenized docid $y_{1:L} \in \mathcal{V}^L$ from a finite vocabulary $\mathcal{V}$ (Tay et al., 2022). The GR training loss is defined by a sequence generation model $p_\Theta(y_t \mid y_{<t}, q)$:

$$\begin{aligned}\mathcal{L}_{\text{GR}}(\Theta) &= \mathbb{E}_q\big[-\log P_\Theta(d^+ \mid q)\big] \\ &= \mathbb{E}_q\Big[-\sum_{t=1}^{L} \log p_\Theta\big(y_t^+ \mid y_{<t}^+, q\big)\Big]. \end{aligned} \quad (3)$$

The mapping between sequences in $\mathcal{V}^L$ and $\mathcal{D}$ is constrained, so that decoding a sequence deterministically selects a document. At inference time, beam search is used with prefix constraints (e.g., trie) to guarantee valid docids.

3 Learning Objectives

3.1 DR: A locally normalized objective

The DR objective in Eq. 2 minimizes a surrogate defined on the set $\{d^+\} \cup \mathcal{N}(q)$, renormalizing scores via a softmax within K candidates per batch.

This makes the learning objective explicitly dependent on the sampled negatives, implying that the negative-sampling scheme (both the size of the candidate set and the quality of the negatives) has a substantial impact on the DR performance. One could use the entire set of non-relevant documents as negatives, but this is computationally infeasible under realistic resource constraints (Wang and Isola, 2020). This mismatch leads to a calibration gap between the global and local objectives.

Assumptions. Negatives for each query q are drawn i.i.d. from a proposal sample policy $\pi(\cdot)$ over $\mathcal{D}$ (with $\mu(\cdot)$ the random sample policy) and scores are bounded as $|S(q,d)/\tau| \leq M$. We define the proposal-bias term

$$\delta(q) = \log \mathbb{E}_{d\sim\pi}\left[\mathrm{e}^{S(q,d)/\tau}\right] - \log \mathbb{E}_{d\sim\mu}\left[\mathrm{e}^{S(q,d)/\tau}\right]. \quad (4)$$

Theorem 3.1 (Lower bound under local normalization)**.** *Let $\widetilde{P}_\Theta(d \mid q)$ be the full-softmax distribution. Under the assumptions above, the expected gap satisfies the following condition:*

$$\mathbb{E}_q\Big[\log \widetilde{P}_\Theta(d^+ \mid q) - \log P_\Theta(d^+ \mid q; \mathcal{N}(q))\Big] \geq \log\frac{N}{K} - \mathbb{E}_q[\delta(q)], \quad (5)$$

where $N = |\mathcal{D}|$ and K is the batch candidate size.

Our proof in Appendix B exposes the mechanism: local normalization replaces the global partition function $Z(q)$ with a batch-level $Z_K(q)$ and, in expectation, $Z_K(q) \approx (K/N)\, Z(q)$ up to proposal bias, yielding a gap that shrinks only logarithmically in K, where $Z(q) = \sum_{d'} \exp(S(q,d')/\tau)$ and $Z_K(q) = \sum_{d'\in\{d^+\}\cup\mathcal{N}(q)} \exp(S(q,d')/\tau)$. A high-probability tail bound version of this theorem is provided in Appendix E.

Practical mitigations for the calibration gap. Increasing K and mining harder negatives can partially reduce the gap by better approximating the global normalization, and temperature scaling or post-hoc calibration further helps align scores (Xiong et al., 2020; Zhan et al., 2021). Nevertheless, as the corpus size N grows, the $\log(N/K)$ term dominates unless K scales proportionally with N, making it increasingly hard for DR to match the true posterior calibration.

3.2 GR: A globally normalized likelihood

The GR loss in Eq. 3 is the token-level negative log-likelihood of a fully normalized sequence model over docids. Averaging over tokens and queries, the cross-entropy decomposes as

$$\underbrace{\mathbb{E}_q\big[-\log P_\Theta(d^+ \mid q)\big]}_{\text{CE loss}} = \underbrace{\mathbb{E}_q\big[H(P^\star(\cdot \mid q))\big]}_{\text{entropy term}} + \underbrace{\mathbb{E}_q\Big[\mathrm{KL}\big(P^\star(\cdot \mid q) \,\|\, P_\Theta(\cdot \mid q)\big)\Big]}_{\text{KL divergence}}. \quad (6)$$

From the CE-KL decomposition in Eq. 6, the entropy term is constant with respect to the model parameters Θ. We therefore obtain the following proposition, for which a detailed proof is provided in Appendix A:

Proposition 3.2 (Global normalization and calibration of GR)**.** *Minimizing the GR loss in Eq. 3 is equivalent to minimizing the expected KL divergence in Eq. 6. Consequently, GR permits arbitrarily accurate in principle approximation of the true posterior $P^\star(d \mid q)$ and its objective is equivalent to likelihood-consistent optimization over the globally normalized candidate space.*

Note that teacher forcing makes gradients local to each conditional step, yet the objective itself remains globally normalized. Therefore, even under prefix constraints on the valid code space, improvements in likelihood translate directly into better probability calibration of $P_\Theta(d \mid q)$.

GR is expected to benefit under corpus scaling. Based on the above analysis, we conclude that under the assumptions in Section 3.1 for locally normalized DR (fixed negative-sample budget K and proposal bias $\delta(q)$), the gap between the ideal global partition $Z(q)$ and its sampled counterpart $Z_K(q)$ grows with $\log N$ when K and δ are not increased along with the corpus growth. In practice, this typically manifests as saturation or degradation in retrieval metrics unless K is increased or the sample quality is improved. In contrast, GR optimizes a globally normalized likelihood over the docid space. Assuming a fixed docid scheme with adequate coverage and in-distribution queries, GR does not incur the $\log N$ calibration drift and can keep benefiting from larger corpora without increasing K (albeit with higher computational costs).

4 Representational Capacity

4.1 DR: A low-rank bottleneck in relevance representation

DR learns a text-to-embedding mapping and computes relevance through a fixed post-interaction rule, typically a bilinear score such as the inner product $S(q,d) = e_q^\top e_d$. Consequently, all relevance information for a query or a document is

compressed into an r-dimensional vector (Weller et al., 2025). Formally, DR stacks m query embeddings into $Q \in \mathbb{R}^{m \times r}$ and N document embeddings into $D \in \mathbb{R}^{N \times r}$. The resulting relevance matrix is $S = QD^\top \in \mathbb{R}^{m \times N}$, which satisfies $\mathrm{rank}(S) \leq r$ regardless of the encoder architecture, as long as the final interaction is bilinear.

By the Eckart-Young-Mirsky theorem (Eckart and Young, 1936; Mirsky, 1960), among all matrices of rank at most r, the truncated SVD of any target logit matrix $S^\star$ achieves the best Frobenius-norm approximation, with minimal error equal to the sum of squared discarded singular values. We therefore state the following corollary:

Corollary 4.1 (Low-rank bottleneck of bilinear DR)**.** *Let r be the embedding dimension. Any bilinear DR with score $S(q, d) = e_q^\top e_d$ induces a relevance matrix $S = QD^\top$ with $\mathrm{rank}(S) \leq r$. Moreover, for a target $S^\star$, the optimal rank-r approximation error equals the squared singular-value tail $\sum_{i>r} \sigma_i(S^\star)^2$.*

Whenever $S^\star$ exhibits a heavy spectral tail, a fixed-r DR model inevitably suffers from an irreducible approximation error unless r is increased. This corollary is similar to the conclusion drawn in (Luan et al., 2021), but their analysis establishes the dimensionality limitation of DR from the perspective of document length and compression rate. The work most closely related to ours from the correlation-matrix viewpoint is contemporaneous work (Weller et al., 2025), which also identifies this limitation of DR, providing detailed proofs and experiments, and argues that late-interaction MVDR models (e.g., ColBERT (Khattab and Zaharia, 2020)) may mitigate the issue. However, we show that MVDR remains subject to a similar upper bound when tokens are grouped into channels (see Appendix D for details).

4.2 GR: Directly fitting the relevance mapping

Let $\mathcal{V}^L$ denote the docid space with a fixed bijection to documents. GR directly fits the query-document relevance mapping via the full model parameters.

Theorem 4.2 (Approximation of $P^\star$ by GR)**.** *Let $\epsilon > 0$. For any conditional distribution $P^\star(\cdot \mid q)$ supported on $\mathcal{D}$, there exist L and a decoder parameterization such that the induced GR model satisfies $\mathbb{E}_q\big[\mathrm{TV}(P^\star(\cdot \mid q), P_\Theta(\cdot \mid q))\big] < \epsilon$, where TV denotes the total variation distance.*

Theorem 4.2 states that under a fixed bijective docid coding and for in-distribution queries, a sufficiently expressive GR model can approximate the true query-document relevance mapping arbitrarily well (in expected total-variation distance). In other words, with adequate capacity, GR can represent documents, queries, and their relevance relations within the model itself. Note that Theorem 4.2 continues to hold when GR decodes under prefix-constrained decoding (see Appendix C for a detailed proof). Nevertheless, in practice the degree to which GR fits the query-document mapping is affected by several factors, including the quality of the docid tree design and the sufficiency and cleanliness of training data (Tay et al., 2022; Wang et al., 2022; Zhuang et al., 2022). Therefore, Theorem 4.2 is a capacity statement rather than a claim about sample or compute efficiency. It assumes an in-distribution query law and a fixed docid. A highly unbalanced or semantically incoherent docid trie can increase optimization difficulty even under universality, and no guarantee is made for out-of-distribution queries.

GR is expected to benefit under model scaling. Under the representation analysis in Section 4, GR can reduce the posterior approximation error by scaling its model capacity (given a fixed docid scheme), whereas DR with bilinear interactions is constrained by an effective rank bound $\mathrm{rank}(S) \leq r$ (or $\leq cr$ with c independent interaction channels). Hence, matching a heavy spectral tail requires proportionally increasing r or c. This predicts steeper gains for GR under equal-parameter scaling.

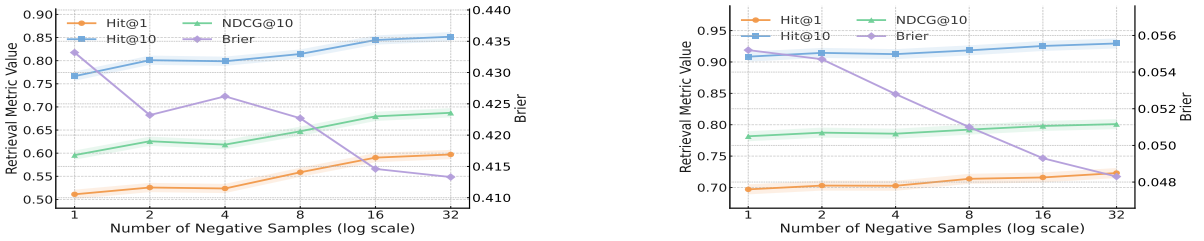
5 Experiments

We present (i) experiments that evaluate the theoretical limitations of DR, (ii) synchronized scaling experiments comparing GR and DR, and (iii) experiments that investigate the potential scaling advantages of GR.

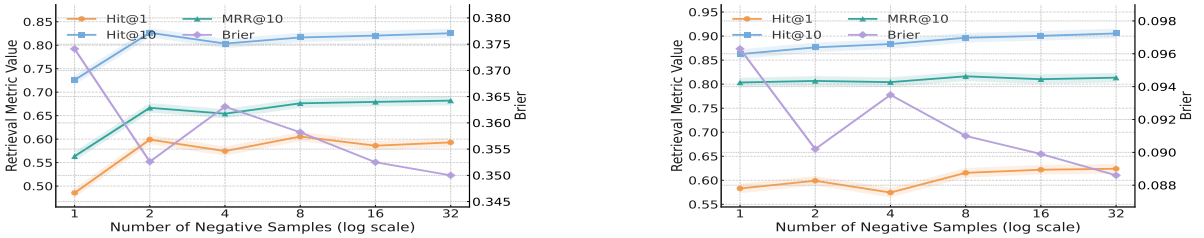
5.1 Experimental setup

We evaluate on two widely used retrieval benchmarks: (i) *Natural Questions* (NQ, Kwiatkowski et al., 2019): real user questions paired with supporting evidence from Wikipedia; and (ii) *MS MARCO Passage* (Bajaj et al., 2016): web search queries from Bing with associated relevant passages. We report the *Brier* score, defined as the mean squared error between the predicted relevance probability of the top-1 candidate and the ground truth per query. We also report standard retrieval metrics: Hits@k, NDCG@k, and MRR@k.

We implement representative DR and GR baselines, avoiding sophisticated variants to ensure fair



(a) Effect of the number of negative samples on Standard DR (Left) and MVDR (Right) on **NQ**.



(b) Effect of the number of negative samples on Standard DR (Left) and MVDR (Right) on **MS MARCO**.

Figure 1: DR's retrieval performance improves as the number of negative samples increases. The left y-axis shows retrieval metrics (higher is better), while the right y-axis shows the Brier score (lower is better). The plotted Brier values are raw and thus not comparable across different settings.

and transparent comparison. For DR, we use (i) a standard dual encoder with inner-product scoring (*Standard DR*) following DPR (Karpukhin et al., 2020), and (ii) a multi-vector late-interaction model in the style of ColBERT-v1 (*MVDR*) (Khattab and Zaharia, 2020). For GR, we follow a DSI-style training and inference pipeline (Tay et al., 2022) with two docid designs: (i) residual-quantization codebook docids, represented as length-6 sequences of 8-bit codes (*GR-codebook*), and (ii) text docids using document titles as identifiers (*GR-text*). All GR decoding is prefix-constrained by a trie over valid docids.

To control for capacity and pre-training, all DR models are built on `Qwen3-Embedding-0.6B`, and all GR models use `Qwen3-0.6B` (Yang et al., 2025a). *See Appendix F for full details of the experimental setup; Appendix G provides implementation details for each subsequent experiment.*

5.2 Limitations of DR

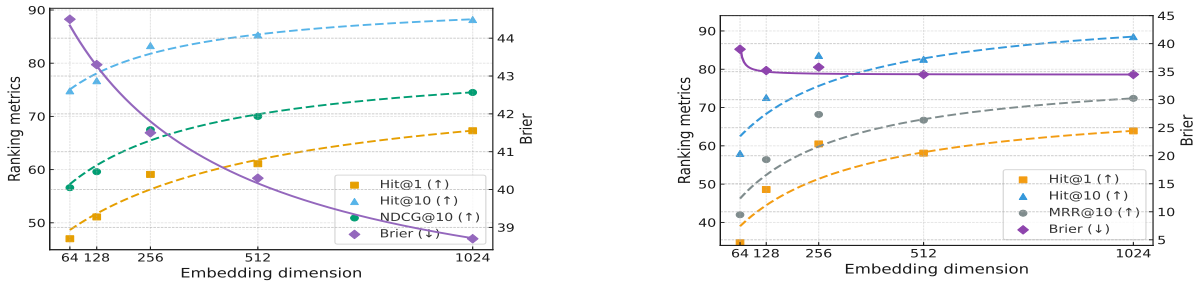
Optimization limitations introduced by local normalization. To assess the effect of local normalization in DR, we vary only the number of negative samples and the proportion of hard negatives while keeping all other settings fixed. Figure 1 illustrates how DR performance changes with *the number of negatives*. We observe that: (i) the calibration metric Brier and ranking metrics move in tandem, confirming that the predicted calibration drift affects retrieval performance; (ii) all metrics improve with more negatives and have not plateaued within our compute budget; and (iii) aside from a few outliers, Standard DR and MVDR show broadly consistent trends across both datasets.

Hard-negative ratio	Standard DR Hit @1	Standard DR Hit @10	Standard DR NDCG @10	MVDR Hit @1	MVDR Hit @10	MVDR NDCG @10
0	52.4	79.9	61.9	57.5	80.4	63.0
0.25	45.4	70.3	53.0	58.4	82.4	64.2
0.5	39.5	63.2	46.7	52.2	78.8	55.1
0.75	43.0	66.8	50.2	60.0	83.5	54.6
1.0	47.0	73.8	52.2	55.6	81.6	48.8

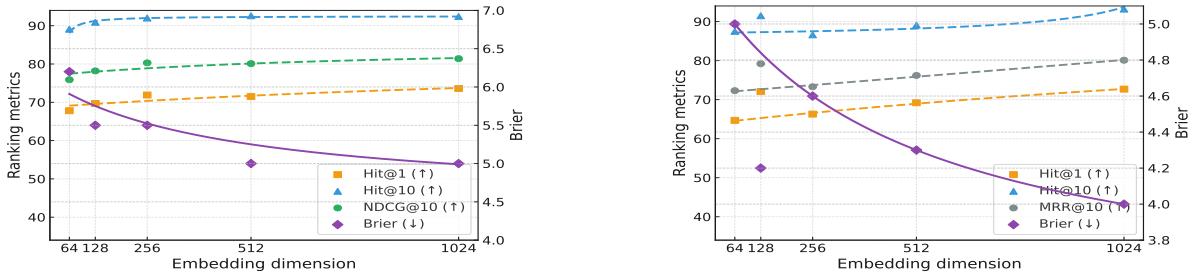
Table 1: Effect of the hard-negative ratio on DR and MVDR on the NQ dataset.

Table 1 illustrates the effect of *the number of hard negatives* on DR performance. For instance, when hard negatives make up half of the batch, Standard DR's Hit@1 drops by roughly 13% relative to using no hard negatives, whereas MVDR improves with a 25% mix of hard negatives. Without strict filtering, a large proportion of difficult negatives degrades DR performance, as overly hard or low-quality negatives introduce gradient noise. When hard negatives dominate (up to 100% in our experiments), the lack of easy negatives destabilizes optimization. These results corroborate the bias introduced by local normalization and highlight that DR's optimization quality depends strongly on negative sampling.

Representational limitations imposed by embedding dimensionality. To evaluate the constraints of bilinear interactions in DR, we vary the embedding size by adding a two-layer non-linear projection after the output layer and training it jointly with the backbone. Figure 2 shows the impact of em-



(a) Effect of the number of embedding dimension on Standard DR (Left) and MVDR (Right) on **NQ**.



(b) Effect of the number of embedding dimension on Standard DR (Left) and MVDR (Right) on **MS MARCO**.

Figure 2: DR's retrieval performance improves as the embedding dimension increases.

	NQ								MS MARCO							
	Standard DR				GR-codebook				Standard DR				GR-codebook			
Metric	Initial	Final	Abs. drop	Per-unit	Initial	Final	Abs. drop	Per-unit	Initial	Final	Abs. drop	Per-unit	Initial	Final	Abs. drop	Per-unit
Hit@1	52.4	45.5	6.9	1.0	64.2	60.9	3.3	0.5	57.5	48.4	9.1	1.3	42.3	39.6	2.7	0.4
Hit@10	79.9	73.6	6.3	0.9	82.5	79.2	3.3	0.5	80.4	73.3	7.1	1.0	70.8	64.5	6.3	0.9
NDCG@10	61.9	56.5	5.4	0.8	86.7	83.7	3.0	0.4	–	–	–	–	–	–	–	–
MRR@10	–	–	–	–	–	–	–	–	65.4	58.9	6.5	0.9	45.1	41.0	4.1	0.6

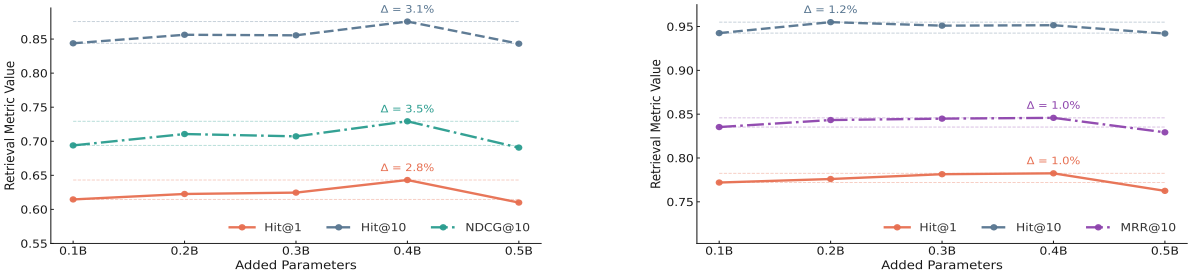
Table 2: DR degrades more sharply with corpus expansion on both NQ and MS MARCO.

bedding dimensionality on DR performance. We observe that: (i) calibration and ranking metrics vary consistently, confirming that theoretical limitations manifest in retrieval outcomes; (ii) increasing the embedding dimension substantially improves performance for both Standard DR and MVDR, with Standard DR gaining over 20% on NQ and MS MARCO; and (iii) even at 1024 dimensions, well above the common 768, performance continues to rise. These results suggest that embedding dimensionality can be a genuine bottleneck for DR, even on datasets smaller than real-world corpora.

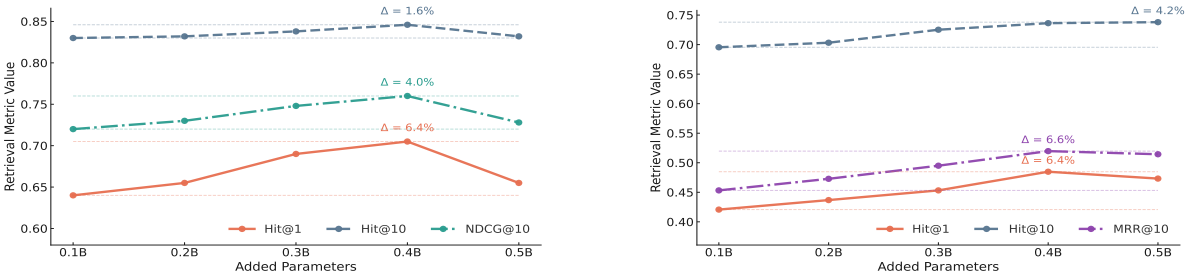
5.3 Scaling trends of GR and DR

Corpus scaling. To assess the impact of normalization on corpus-level scaling, we compare GR and DR on progressively larger subsets of documents and queries sampled from the official training and evaluation sets, keeping all hyperparameters and the training budget fixed. Candidate set sizes are increased logarithmically from the base number of training documents (300K for NQ, 1M for MS MARCO). As shown in Table 2, both datasets exhibit similar trends: (i) performance of both GR and DR declines as the candidate set grows, reflecting increased task difficulty; (ii) GR degrades more slowly than DR, both in magnitude and rate. E.g., on NQ, DR's Hit@1 drops 6.9% and Hit@10 6.3%, while GR's Hit@1 and Hit@10 decrease only 3.3% each. This aligns with our theoretical analysis: corpus expansion amplifies DR's optimization drift due to local normalization, whereas GR's globally normalized objective renders it less sensitive to additional non-relevant documents. Results for MVDR and GR-text are provided in Appendix J.

Model scaling. We study model scaling by comparing GR and DR under matched increases in parameter budget. We attach randomly initialized adapters of the same size to both models and train the adapters jointly with the backbone, then track ranking metrics. The adapters range from 0.1B to 0.8B parameters; at the largest setting, the adapter exceeds the backbone in size, making this a meaningful scaling regime. We do not vary backbone size due to the fixed and sparse availability of pre-trained models (e.g., Qwen3 offers only 0.6B, 4B, and 8B variants below 10B), which prevents smooth scaling analysis. Sparse backbone-scaling results are reported in Appendix I and are consis-



(a) Standard DR shows no clear trend of improved retrieval performance with increasing parameter scale on NQ (Left) and MS MARCO (Right).



(b) GR shows a clear upward scaling trend in retrieval performance on NQ (Left) and MS MARCO (Right).

Figure 3: Comparison of DR and GR under synchronized model scaling. Only the increasing range is shown here. All models drop after 0.4B due to adding too many new parameters. See Appendix K for the full curve.

tent with the adapter-scaling conclusions. Figure 3 shows that GR consistently improves with model scale, with all metrics increasing by about 5% on both NQ and MS MARCO. In contrast, DR exhibits flat or only marginal gains (around 1%), indicating limited benefit from parameter scaling. This trend is consistent across datasets, suggesting that GR is better positioned to leverage increasing model capacity in the LLM era, whereas DR may require larger embeddings or richer contrastive pretraining to benefit from scaling. Additional results for MVDR and GR-text are provided in Appendix K.

Here, GR underperforms DR on MS MARCO, seemingly contradicting its theoretical advantages. We attribute this to two factors: (i) GR is much more sensitive than DR to data design and task priors, whereas DR is relatively robust to such choices; and (ii) GR benefits from repeated document exposure, while MS MARCO exhibits low query-document coverage and limited train-validation overlap. For fairness, we do not tune these factors for MS MARCO. We focus on the per-parameter improvement rate, which still indicates that GR is better suited to scaling.

5.4 Potential advantages of GR

Next, we examine GR at larger scale using a 14B-parameter model. We focus on GR-text on NQ, where document titles serve as natural text docids drawn from Wikipedia. Since both documents and titles are observed during pre-training, this setting directly uses the LLM's pre-trained world knowledge and reasoning capabilities.

	Hit@1	Hit@10	NDCG@10
Zero-shot GR	18.1	23.8	33.3
Standard GR	45.7	63.5	88.6
TTS GR	47.3	65.8	89.1

Table 3: Retrieval performance on the NQ dataset for standard GR-text and its zero-shot and TTS variants.

Zero-shot GR. GR generates docids token by token; when docids are textual, this process naturally aligns with the LLM's next-token prediction pre-training objective. This motivates testing whether retrieval can be performed without any task-specific training. Accordingly, we evaluate zero-shot GR using a well-pre-trained LLM with no post-hoc fine-tuning, applying constrained decoding to restrict outputs to valid docids. Unlike prior ZeroGR methods that still use lightweight instruction tuning (Sun et al., 2025), this setting isolates retrieval ability derived purely from pre-training.

Test-time scaling (TTS) GR. We further study TTS via a "think-then-retrieve" procedure. At inference, the model first generates a brief free-form reasoning step, which is concatenated with the original query and then fed into constrained decoding for retrieval. This reasoning augmentation is applied only at test time; training follows the standard GR setup.

Table 3 reports results for standard, zero-shot, and TTS GR. We observe that (i) zero-shot GR achieves non-trivial, though modest, performance,

suggesting that with larger models, better prompts, and suitable docids, practical training-free GR may be feasible; and (ii) even without fine-tuning, a pre-retrieval reasoning step consistently improves performance over the no-reasoning baseline, indicating that GR's internalized document knowledge aids retrieval via query reformulation. Together, these results reinforce GR's advantages at larger model scales.

6 Discussion

Practical challenges of GR. Despite its theoretical appeal and scaling advantages, in practice, GR often falls short of its optimum for three main reasons. First, *noisy or biased supervision and limited training* can induce an irreducible mismatch between the learned model and the target posterior (Zhuang et al., 2022). Second, *prefix-constrained autoregressive decoding* suffers from error propagation, where early mistakes compound downstream, especially under poor docid designs (e.g., unbalanced hierarchies or weakly informative text docids) (Bevilacqua et al., 2022; Zhang et al., 2024). Third, retrieval performance is highly sensitive to docid design, yet *constructing identifiers that are both compact and semantically expressive* remains an open problem. These factors limit GR's empirical performance (Section 5). Beyond optimality, practical constraints further hinder deployment. *Token-by-token decoding incurs high latency* compared to ANN-based DR, which supports near-instant lookup after indexing (Appendix H). And under continual corpus drift, GR typically requires re-training or local fine-tuning to update docids or hierarchies (Chen et al., 2023; Kishore et al., 2023), whereas DR often supports index-only updates.

Potential solutions for GR. We outline possible directions to mitigate GR's practical challenges.

Data noise and undertraining. Two complementary approaches are promising. (i) Treat relevance as the pre-training objective by training a decoder-only model from scratch on large-scale, noise-controlled (q, d) pairs to directly optimize $-\log P(d \mid q)$, as in recent generative recommendation work (e.g., OneRec (Deng et al., 2025)); this is appropriate when relevance is explicitly defined by human rules (e.g., e-commerce query-item (Rajput et al., 2023), ads matching (Fan et al., 2019), FAQ-KB pairs (Sakata et al., 2019)). (ii) Better exploit LLM world knowledge and reasoning by teaching the semantics and interface of retrieval via explicit, reasoning-time instructions, rather than memorizing full-corpus relevance. This is most effective when relevance is already encoded in pre-training data (e.g., Wikipedia or encyclopedic retrieval (Petroni et al., 2020)). ZeroGR (Sun et al., 2025) explores a similar path, but it still depends on some lightweight post-hoc instruction fine-tuning.

Early error propagation. Relaxing constraints or enabling parallel decoding may reduce cascading errors. Specifically, for relaxing constraints, allowing each document to belong to multiple clusters (especially for boundary cases) may reduce early errors, and incorporating backoff mechanisms or, when necessary, allowing tokens outside the constraint set can help the model recover from early mistakes. For parallel decoding, some recent work has made initial attempts in this direction by modeling GR with diffusion models (Liu et al., 2025; Zhao et al., 2025).

Engineering efficiency. A unified GR-DR system is a practical direction. For instance, GR can decode a shallow prefix for coarse-grained category recall, followed by DR for fine-grained retrieval within that category. This coarse-to-fine design leverages GR's capacity to model relevance, while reducing error accumulation and latency of deep prefix-constrained decoding all the way down to full docids. Some industrial systems deployed in practice may already have adopted a similar design (Deng et al., 2025; Zhang et al., 2025).

7 Conclusion

We have systematically compared DR and GR in terms of learning objectives and representational capacity. Theoretically, GR performs globally normalized maximum likelihood over the docid space, avoiding the calibration gap of DR's locally normalized contrastive learning. DR is limited by a low-rank bottleneck under fixed bilinear interactions, whereas GR supports higher-rank approximations. Empirically, results on the NQ and MS MARCO datasets show that calibration and ranking metrics corroborate these theoretical differences. Under comparable corpus and parameter scaling, GR achieves larger gains and shows benefits in zero-shot and test-time scaling. Overall, GR promises to overcome DR's bottlenecks, though several practical challenges remain. For future work, we will extend our analysis to practical factors such as docid design, noisy supervision, and decoding, and leverage these insights to develop more effective and efficient GR methods.

8 Limitations

This work has several limitations. (i) We aim to systematically analyze the fundamental differences between GR and DR as two retrieval paradigms. As a result, our comparison is conducted at a highly abstract level, and we do not propose design-specific improvement techniques. (ii) Our theoretical analysis relies on idealized formulations of GR and DR, and does not fully model factors such as training data noise, docid design, and decoding/search strategies. Incorporating these practical factors into a unified analytical framework may introduce new interactions and additional assumptions, making the conclusions depend on tasks and implementation details in a more complex way. (iii) Due to resource constraints, we are unable to systematically compare GR and DR at larger model and corpus scales. In future work, we plan to complete the scaling curves over a broader range of model sizes and larger candidate sets. (iv) For fairness, our comparison does not include the latest system variants of GR and DR. These variants may narrow or widen the gap discussed in this paper under specific conditions, and a more careful alignment and controlled comparisons are needed in future. And (v) although we discuss several promising directions to improve the practical usability of GR, we do not conduct preliminary experiments to validate their effectiveness. Some contemporaneous work partially supports a subset of our discussion, but deeper system designs and large-scale empirical studies remain for future work.

9 Ethical Considerations

This work analyzes generative and dense retrieval methods using existing benchmark datasets and reports aggregate results only. It does not collect new user data or involve human subjects.

References

Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, and 1 others. 2016. MS MARCO: A human generated machine reading comprehension dataset. *arXiv preprint arXiv:1611.09268*.

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.

Jiangui Chen, Ruqing Zhang, Jiafeng Guo, Maarten de Rijke, Wei Chen, Yixing Fan, and Xueqi Cheng. 2023. Continual learning for generative retrieval over dynamic corpora. In *Proceedings of the 32nd ACM international conference on information and knowledge management*, pages 306–315.

Gobinda G Chowdhury. 2010. *Introduction to modern information retrieval*. Facet publishing.

Jiaxin Deng, Shiyao Wang, Kuo Cai, Lejian Ren, Qigen Hu, Weifeng Ding, Qiang Luo, and Guorui Zhou. 2025. Onerec: Unifying retrieve and rank with generative recommender and iterative preference alignment. *arXiv preprint arXiv:2502.18965*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 3(8):4171–4186.

Carl Eckart and Gale Young. 1936. The approximation of one matrix by another of lower rank. *Psychometrika*, 1(3):211–218.

Miao Fan, Jiacheng Guo, Shuai Zhu, Shuo Miao, Mingming Sun, and Ping Li. 2019. Mobius: towards the next generation of query-ad matching in baidu's sponsored search. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2509–2517.

Yixing Fan, Xiaohui Xie, Yinqiong Cai, Jia Chen, Xinyu Ma, Xiangsheng Li, Ruqing Zhang, Jiafeng Guo, and 1 others. 2022. Pre-training methods in information retrieval. *Foundations and Trends in Information Retrieval*, 16(3):178–317.

Thibault Formal, Benjamin Piwowarski, and Stéphane Clinchant. 2021. Splade: Sparse lexical and expansion model for first stage ranking. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2288–2292.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick SH Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *EMNLP (1)*, pages 6769–6781.

Omar Khattab and Matei Zaharia. 2020. Colbert: Efficient and effective passage search via contextualized late interaction over bert. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pages 39–48.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Varsha Kishore, Chao Wan, Justin Lovelace, Yoav Artzi, and Kilian Q Weinberger. 2023. IncDSI: Incrementally updatable document retrieval. In *International*

conference on machine learning, pages 17122–17134. PMLR.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, and 1 others. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466.

Hyunji Lee, Sohee Yang, Hanseok Oh, and Minjoon Seo. 2022. Generative multi-hop retrieval. *arXiv preprint arXiv:2204.13596*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880.

Minghan Li, Sheng-Chieh Lin, Xueguang Ma, and Jimmy Lin. 2023. SLIM: Sparsified late interaction for multi-vector retrieval with inverted indexes. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1954–1959.

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.

Zhao Liu, Yichen Zhu, Yiqing Yang, Guoping Tang, Rui Huang, Qiang Luo, Xiao Lv, Ruiming Tang, Kun Gai, and Guorui Zhou. 2025. DiffGRM: Diffusion-based generative recommendation model. *arXiv preprint arXiv:2510.21805*.

Yi Luan, Jacob Eisenstein, Kristina Toutanova, and Michael Collins. 2021. Sparse, dense, and attentional representations for text retrieval. *Transactions of the Association for Computational Linguistics*, 9:329–345.

Leon Mirsky. 1960. Symmetric gauge functions and unitarily invariant norms. *The quarterly journal of mathematics*, 11(1):50–59.

Bhaskar Mitra, Nick Craswell, and 1 others. 2018. An introduction to neural information retrieval. *Foundations and Trends® in Information Retrieval*, 13(1):1–126.

Thong Nguyen and Andrew Yates. 2023. Generative retrieval as dense retrieval. *arXiv preprint arXiv:2306.11397*.

Fabio Petroni, Aleksandra Piktus, Angela Fan, Patrick Lewis, Majid Yazdani, Nicola De Cao, James Thorne, Yacine Jernite, Vladimir Karpukhin, Jean Maillard, and 1 others. 2020. KILT: A benchmark for knowledge intensive language tasks. *arXiv preprint arXiv:2009.02252*.

Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, and 1 others. 2018. Improving language understanding by generative pre-training. https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf.

Shashank Rajput, Nikhil Mehta, Anima Singh, Raghunandan Hulikal Keshavan, Trung Vu, Lukasz Heldt, Lichan Hong, Yi Tay, Vinh Tran, Jonah Samost, and 1 others. 2023. Recommender systems with generative retrieval. *Advances in Neural Information Processing Systems*, 36:10299–10315.

Wataru Sakata, Tomohide Shibata, Ribeka Tanaka, and Sadao Kurohashi. 2019. FAQ retrieval using query-question similarity and BERT-based query-answer relevance. In *Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval*, pages 1113–1116.

Weiwei Sun, Keyi Kong, Xinyu Ma, Shuaiqiang Wang, Dawei Yin, Maarten de Rijke, Zhaochun Ren, and Yiming Yang. 2025. ZeroGR: A generalizable and scalable framework for zero-shot generative retrieval. *arXiv preprint arXiv:2510.10419*.

Yi Tay, Vinh Tran, Mostafa Dehghani, Jianmo Ni, Dara Bahri, Harsh Mehta, Zhen Qin, Kai Hui, Zhe Zhao, Jai Gupta, and 1 others. 2022. Transformer memory as a differentiable search index. *Advances in Neural Information Processing Systems*, 35:21831–21843.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Tongzhou Wang and Phillip Isola. 2020. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International conference on machine learning*, pages 9929–9939. PMLR.

Yujing Wang, Yingyan Hou, Haonan Wang, Ziming Miao, Shibin Wu, Qi Chen, Yuqing Xia, Chengmin Chi, Guoshuai Zhao, Zheng Liu, and 1 others. 2022. A neural corpus indexer for document retrieval. *Advances in Neural Information Processing Systems*, 35:25600–25614.

Orion Weller, Michael Boratko, Iftekhar Naim, and Jinhyuk Lee. 2025. On the theoretical limitations of embedding-based retrieval. *arXiv preprint arXiv:2508.21038*.

Shiguang Wu, Wenda Wei, Mengqi Zhang, Zhumin Chen, Jun Ma, Zhaochun Ren, Maarten de Rijke, and Pengjie Ren. 2024. 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.

Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul Bennett, Junaid Ahmed, and Arnold Overwijk. 2020. Approximate nearest neighbor negative contrastive learning for dense text retrieval. *arXiv preprint arXiv:2007.00808*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025a. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

An Yang, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoyan Huang, Jiandong Jiang, Jianhong Tu, Jianwei Zhang, Jingren Zhou, and 1 others. 2025b. Qwen2. 5-1m technical report. *arXiv preprint arXiv:2501.15383*.

Hansi Zeng, Chen Luo, and Hamed Zamani. 2024. Planning ahead in generative retrieval: Guiding autoregressive generation through simultaneous decoding. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 469–480.

Jingtao Zhan, Jiaxin Mao, Yiqun Liu, Jiafeng Guo, Min Zhang, and Shaoping Ma. 2021. Optimizing dense retrieval model training with hard negatives. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*, pages 1503–1512.

Peitian Zhang, Zheng Liu, Yujia Zhou, Zhicheng Dou, Fangchao Liu, and Zhao Cao. 2024. Generative retrieval via term set generation. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 458–468.

Yingchen Zhang, Ruqing Zhang, Jiafeng Guo, Wenjun Peng, Sen Li, and Fuyu Lv. 2025. Retrieval-in-the-chain: Bootstrapping large language models for generative retrieval. *arXiv preprint arXiv:2510.13095*.

Xinpeng Zhao, Yukun Zhao, Zhenyang Li, Mengqi Zhang, Jun Feng, Ran Chen, Ying Zhou, Zhumin Chen, Shuaiqiang Wang, Zhaochun Ren, and 1 others. 2025. DiffuGR: Generative document retrieval with diffusion language models. *arXiv preprint arXiv:2511.08150*.

Shengyao Zhuang, Houxing Ren, Linjun Shou, Jian Pei, Ming Gong, Guido Zuccon, and Daxin Jiang. 2022. Bridging the gap between indexing and retrieval for differentiable search index with query generation. *arXiv preprint arXiv:2206.10128*.

A Cross-entropy and KL Decomposition

For completeness, we give a concise derivation of Eq. 6. Let P be the data distribution and Q_Θ the model on the same finite support. By definition,

$$\begin{aligned}\mathrm{CE}(P, Q_\Theta) &= \mathbb{E}_{x\sim P}\big[-\log Q_\Theta(x)\big] \\ &= \mathbb{E}_{x\sim P}\Big[\log \frac{P(x)}{Q_\Theta(x)}\Big] + \mathbb{E}_{x\sim P}\big[-\log P(x)\big].\end{aligned} \tag{7}$$

The first term equals $\mathrm{KL}(P\|Q_\Theta)$ and the second equals $H(P)$, hence $\mathrm{CE}(P, Q_\Theta) = H(P) + \mathrm{KL}(P\|Q_\Theta)$. For conditional sequence models (GR), summing token-wise cross-entropies yields the same identity after taking expectations over queries.

B Proof of Theorem 3.1

For a query q, define the global and in-batch partition functions

$$\begin{aligned}Z(q) &= \sum_{d'\in\mathcal{D}} \exp\big(S(q,d')/\tau\big), \\ Z_K(q) &= \sum_{d'\in\{d^+\}\cup\mathcal{N}(q)} \exp\big(S(q,d')/\tau\big).\end{aligned} \tag{8}$$

Then

$$\begin{aligned}&\log \widetilde{P}_\Theta(d^+ \mid q) - \log P_\Theta(d^+ \mid q; \mathcal{N}) \\ &= \log Z_K(q) - \log Z(q).\end{aligned} \tag{9}$$

Let μ be the corpus marginal (uniform over $\mathcal{D}$) and π the negative-sampling proposal,

$$\delta(q) = \log \mathbb{E}_{d\sim\pi}\big[\mathrm{e}^{S(q,d)/\tau}\big] - \log \mathbb{E}_{d\sim\mu}\big[\mathrm{e}^{S(q,d)/\tau}\big]. \tag{10}$$

Taking expectation over the sampling of $\mathcal{N}(q)$ and using Jensen's inequality,

$$\begin{aligned}\mathbb{E}\big[\log Z_K(q)\big] &\geq \log \mathbb{E}\big[Z_K(q)\big] \\ &\geq \log K \;+\; \log \mathbb{E}_{d\sim\pi}\big[\mathrm{e}^{S(q,d)/\tau}\big].\end{aligned} \tag{11}$$

where we use the fact that $\mathbb{E}[Z_K(q)] \geq K\,\mathbb{E}_{d\sim\pi}[\mathrm{e}^{S(q,d)/\tau}]$. Since $Z(q) = N\,\mathbb{E}_{d\sim\mu}[\mathrm{e}^{S(q,d)/\tau}]$, we obtain

$$\mathbb{E}\big[\log Z_K(q) - \log Z(q)\big] \geq \log \frac{K}{N} \;-\; \delta(q). \tag{12}$$

Averaging over queries gives Theorem 3.1.

C Constructive Universality for GR

Fix a bijection between $\mathcal{D}$ and the leaves of a $|\mathcal{V}|$-ary trie of depth L. Given a target posterior $P^\star(\cdot \mid q)$, assign at each internal node the conditional distribution over its children to match the subtree mass under $P^\star$: for node u with children $\{v\}$, set

$$p^\star(v \mid u, q) = \frac{\sum_{\text{leaves } \ell \in \text{subtree}(v)} P^\star(\ell \mid q)}{\sum_{\text{leaves } \ell \in \text{subtree}(u)} P^\star(\ell \mid q)}. \quad (13)$$

A decoder with sufficient capacity can approximate each local conditional $p^\star(\cdot \mid u, q)$ arbitrarily well. By the chain rule along any root-to-leaf path, the product of these conditionals approximates the target leaf mass, hence the induced leaf distribution approaches $P^\star(\cdot \mid q)$ in total variation. Under prefix-constrained decoding, the same construction applies because valid leaves are exactly the trie leaves corresponding to $\mathcal{D}$.

D Low-rank Limitation for DR

Let $S^* \in \mathbb{R}^{m \times N}$ be a ground-truth logit matrix whose (i, j)-entry is a monotone transform of $\log P^*(d_j \mid q_i)$. Any bilinear DR model with embedding dimension r factorizes as $S = QD^\top$ and thus $\operatorname{rank}(S) \le r$ (or $\le cr$ with c independent interaction channels). By the Eckart-Young-Mirsky theorem,

$$\min_{\operatorname{rank}(S) \le r} \|S - S^*\|_F^2 = \sum_{i > r} \sigma_i(S^*)^2, \quad (14)$$

the squared Frobenius norm of the spectral tail beyond rank r.

Consequently, if S^* has a heavy spectral tail, any fixed-r DR model incurs an irreducible posterior approximation error unless r (or the number of interaction channels) is increased.

E A High-probability Bound for $\log Z_K - \log Z$

Fix a query q and define $X = \mathrm{e}^{S(q,d)/\tau}$ for $d \sim \pi(\cdot \mid q)$ with mean $\mu_\pi = \mathbb{E}_\pi[X]$ and variance $\sigma_\pi^2 = \operatorname{Var}_\pi[X]$. Let $X_1, \ldots, X_K$ be i.i.d. copies and $\bar{X}_K = \frac{1}{K}\sum_{i=1}^K X_i$. Assuming X is sub-exponential (e.g., bounded or with a finite moment generating function in a neighborhood of 0), a Bernstein-type inequality gives, for any $\epsilon \in (0, 1)$,

$$\Pr\left[\log \bar{X}_K \le \log \mu_\pi - \epsilon\right] \le \exp\Big(-\frac{K\,\epsilon^2}{2\big(\sigma_\pi^2/\mu_\pi^2 + \epsilon/3\big)}\Big). \quad (15)$$

Since $Z_K(q) = \sum_{d \in \mathcal{N}(q)} \mathrm{e}^{S(q,d)/\tau} = K\,\bar{X}_K$ and $Z(q) = N\,\mu_\mu$ with $\mu_\mu = \mathbb{E}_{d \sim \mu}[\mathrm{e}^{S(q,d)/\tau}]$, we have with probability at least $1 - \exp(-cK\epsilon^2)$ (for a constant c depending on moments of X):

$$\begin{aligned} &\log Z_K(q) - \log Z(q) \\ &\ge \log \frac{K}{N} - \big(\log \mu_\mu - \log \mu_\pi\big) - \epsilon \\ &= \log \frac{K}{N} - \delta(q) - \epsilon. \end{aligned} \quad (16)$$

Averaging over q yields a high-probability version of Theorem 3.1. We emphasize that this bound holds under i.i.d. negatives from π; for adaptive or "hard-negative" proposals $\pi_t(\cdot \mid q, \Theta_t)$, the same form holds with an additional bias term in $\delta_t(q)$ that captures proposal/model dependence.

F Detailed Experimental Setup

Datasets. We evaluate on two standard retrieval benchmark datasets: (i) **Natural Questions** (NQ, Kwiatkowski et al., 2019). This is a collection of real-user questions paired with supporting Wikipedia evidence. We use the official train (313K) and test (7K) splits. To make generative retrieval feasible, we ensure that each test query's gold document appears in the docid inventory constructed from the training corpus (i.e., the gold docid is seen during training); and (ii) **MS MARCO Passage** (Bajaj et al., 2016). This is a set of web search queries from Bing with associated passages. We use the passage-ranking subset and sample 1M training pairs and 2K evaluation queries from the official train/test splits. Unlike NQ, we do not enforce the "seen-document" constraint on MS MARCO (because enforcing it would shrink the evaluation set to only few hundred queries).

Models used for comparison. We implement two representative systems for both DR and GR and intentionally avoid complex variants to keep comparisons fair and transparent. For DR, we implement (i) a *standard bi-encoder* in the spirit of DPR (Karpukhin et al., 2020) with inner-product scoring; and (ii) a *multi-vector late-interaction* variant like ColBERT v1 (Khattab and Zaharia, 2020). For GR, we implement two varying about the docid design and train/inference follow the DSI-style (Tay et al., 2022): (i) *codebook docids* built via residual quantization, each docid is a length-6 sequence of 8-bit code indices; and (ii) *textual docids* that directly use the title as the document identifier. All GR

decoding is prefix-constrained by a trie constructed from the set of valid docids.

Metrics. We report the calibration metric *Brier*, which is the mean squared error between the predicted relevance probability and the ground truth over the query's rank-1 candidate. We report unnormalized (raw) Brier scores, consequently, they are comparable only within the same dataset and experimental series, and the values are not comparable across experiments. We also report four retrieval metrics: (i) *Hits@*k indicates whether at least one relevant document appears in the top-k results for a query; (ii) *NDCG@*k is the normalized discounted cumulative gain at cutoff k, using binary gains with logarithmic discounting by rank; and (iii) *MRR@*k is the mean reciprocal rank of the first relevant document within the top-k.

Training and inference. To control for capacity and pre-training, all DR models are built on Qwen3-Embedding-0.6B, and all GR models use Qwen3-0.6B (Yang et al., 2025a). Unless otherwise noted, we train with the Adam optimizer (Kingma and Ba, 2014) using its default settings. At inference time, DR retrieves top-k candidates using FAISS-based ANN search (Xiong et al., 2020), while GR performs top-k constrained decoding over the docid trie.

G Detailed Experimental Implementation

DR negative sampling. The goal is to assess how negative sampling affects DR performance along two dimensions: size and quality. For size, we use random negatives and vary the number of negatives during training. For quality, we experiment only on NQ, which provides both standard and hard negatives: we vary the proportion of hard negatives in the sampled batch. If the official hard negatives are insufficient, we first fill with the provided standard negatives, and if still insufficient we complete the batch with random negatives. In this experiment, both query and document embedding dimensionality is fixed at 128, and MVDR and DR share identical settings.

DR embedding size. The goal is to examine the constraint imposed by the embedding dimension on DR. We append a two-layer non-linear projection (ReLU activations) after the model's output layer to map embeddings to the target dimension and this projection is trained jointly with the backbone. Random negative sampling is used, and MVDR shares the same settings as DR.

Corpus scaling. The goal is to observe how GR and DR behave when the training corpus size is increased by the same amount. We control the number of documents in the corpus and require that each document appears at least once as a positive in the training set; the test set is a subset of this corpus. In this experiment, DR uses random negative sampling and 128-dimensional embeddings. GR-codebook and GR-text follow the configurations described in the main text. GR-text is evaluated only on NQ, where the official titles can serve as textual docids.

Model scaling. The goal is to compare GR and DR when model capacity is scaled by the same budget. We equip each layer with randomly initialized adapters of matched size and control the scaling by the total number of newly introduced parameters and adapters are trained jointly with the backbone. Note that the largest adapter budget can exceed the original backbone size. All other settings mirror those in the Corpus Scaling experiment.

GR zero-shot. The goal is to evaluate GR's retrieval ability without fine-tuning, relying solely on pre-trained knowledge. This experiment is conducted only on NQ with the GR-text, because NQ's documents and their titles (used as docids) come from Wikipedia which is thoroughly covered during LLM pre-training making zero-shot GR feasible. We employ a larger model (Qwen3-14B) for this study. Specifically, we do not fine-tune Qwen3-14B, instead, we prepend a prompt to each query: `Given the question, predict the document title that most likely contains the answer. The title is:` and then enforce trie-constrained decoding to produce the docid.

GR TTS. The goal is to assess whether GR can leverage an LLM's reasoning capability and its internalized document knowledge to improve performance via a "think-then-retrieve" procedure. This experiment is conducted only on NQ with the GR-text, using Qwen3-14B as the backbone. During training, we prepend a retrieval instruction I_r to each query: `Given the question, predict the document title that most likely contains the answer. The title is:` and fine-tune GR with LoRA. During inference, the model first performs unconstrained "thinking" given the prompt: `Briefly think about the document title that may contain the answer to this question.` The generated reasoning is then concatenated with the original query and the retrieval

Corpus size	DR-Flat	DR-ANN	GR-codebook
0.1M	0.062×	0.008×	1.000×
200M	0.393×	0.080×	1.124×

Table 4: Inference latency on different size of NQ. We report ratios relative to GR on the 0.1M corpus (set to 1.00×).

Size	0.6B	4B	8B
DR	61.4%	62.8%	62.6%
GR-codebook	63.8%	64.5%	64.9%

Table 5: Retrieval performance on NQ when varying the Qwen3 backbone size.

instruction I_r, and constrained decoding is applied to produce the docid.

H Computation and Storage Cost

In terms of training computation, GR usually requires longer training time and higher compute cost than DR. This mainly stems from (i) the autoregressive objective of GR, which requires teacher forcing; and (ii) the fact that more training steps are often needed for convergence. However, in practical retrieval deployments, training cost is rarely the bottleneck, since the system is typically trained once and then deployed for a long period.

To evaluate decoding latency, we measure end-to-end latency on NQ with two corpus sizes (0.1M documents and 2M documents), and compare three methods: DR-Flat (brute-force search), DR-ANN (HNSW indexing), and GR (constrained beam search). The results are shown in Table 4. We observe that although GR has higher absolute latency, the latency of DR-Flat increases sharply as the corpus grows and thus offers no order-of-magnitude advantage at large scale. DR-ANN substantially reduces latency, but it typically incurs a noticeable drop in accuracy. In contrast, a notable advantage of GR is that its latency remains stable and is largely insensitive to corpus size, since constrained decoding does not rely on external index lookup.

Regarding storage requirements, GR is more lightweight than DR. DR must store a dense embedding vector for each document as well as the associated ANN index structures. By contrast, GR only needs to store docids and a prefix trie (or other constraint structures) built over these docids, whose size is often several orders of magnitude smaller than the dense-vector index for the same corpus.

I Extended Experiments with Larger Backbone

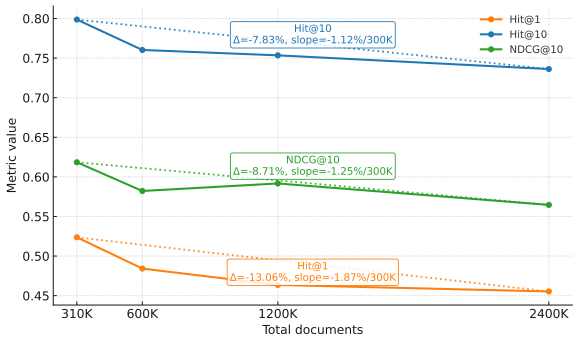
In Section 5.3 of the main text, to plot smooth scaling curves, we vary the trainable parameter budget by adjusting the adapter size. In this section, we provide retrieval performance on NQ when varying the backbone model size. We only consider the 4B and 8B variants in the Qwen3 family, because (i) latency-constrained retrieval systems rarely deploy extremely large models; and (ii) relevance supervision is limited, and training larger models often leads to overfitting and related issues. The results are shown in Table 5. Although the data points are limited, the trend is clear: GR benefits more from scaling than DR, consistent with our main conclusion.

J Extended Results on Corpus Scaling

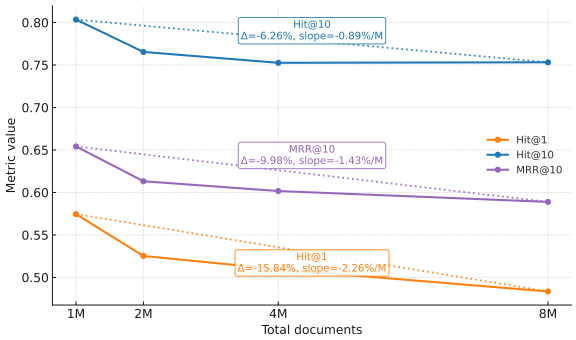
This section supplements the corpus scaling experiments in Section 5.3. Figure 4 presents the full performance trends under corpus scaling for all models (including MVDR and GR-text, which are not covered in the main text Table 2). The conclusions mirror those in the main text: overall, DR exhibits a larger performance drop than GR as the corpus size increases.

K Extended Results on Model Scaling

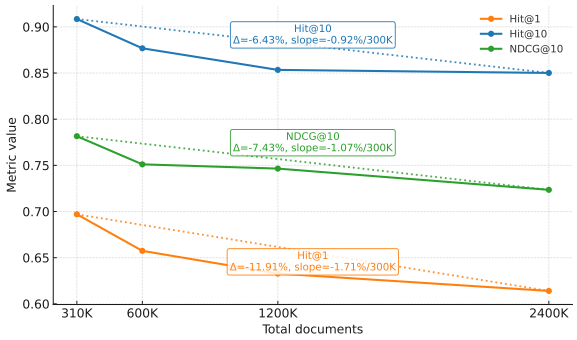
This section supplements the model scaling experiments in Section 5.3. Figure 5 presents the full performance trends under model scaling for all models (including MVDR and GR-text, which are not covered in the main text, Figure 3). The end-of-curve downturn observed in all traces is likely due to the addition of excessive parameters. Ignoring this effect and focusing on the initial stage where model scaling yields gains, the conclusion aligns with the main text: GR derives greater benefits from increases in parameter scale.



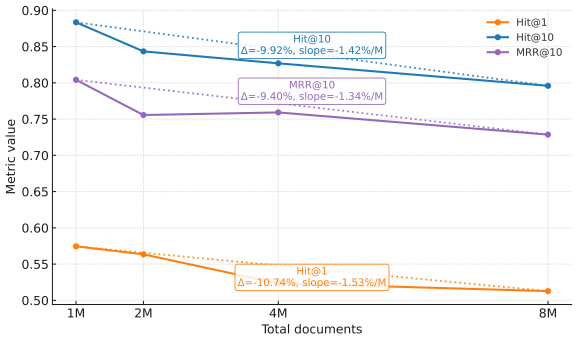
(a) Standard DR on NQ



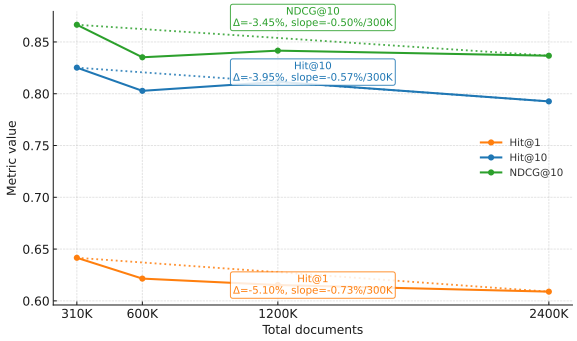
(b) Standard DR on MS MARCO



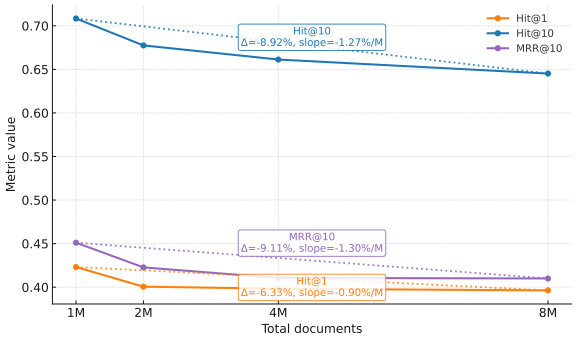
(c) MVDR on NQ



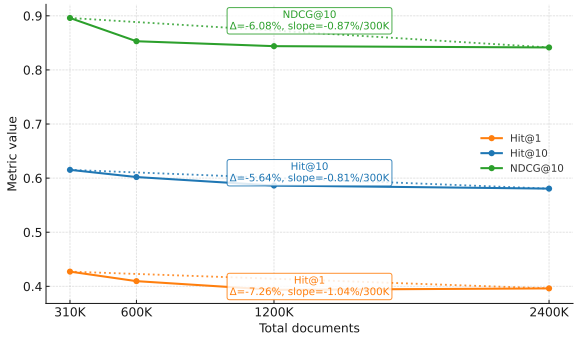
(d) MVDR on MS MARCO



(e) GR-codebook on NQ

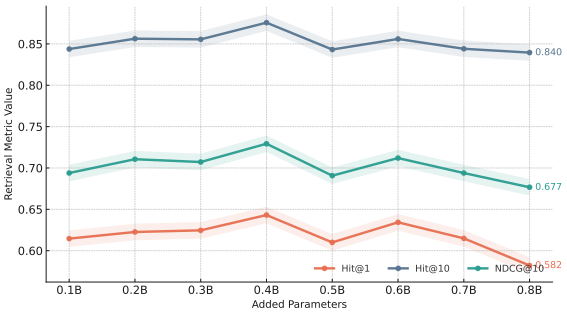


(f) GR-codebook on MS MARCO

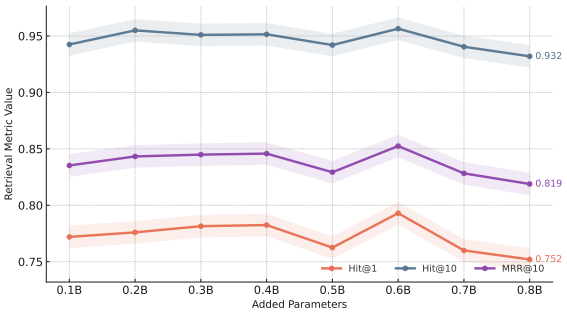


(g) GR-text on NQ

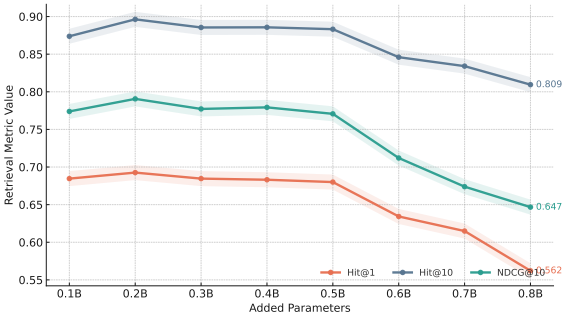
Figure 4: Extended results of corpus scaling.



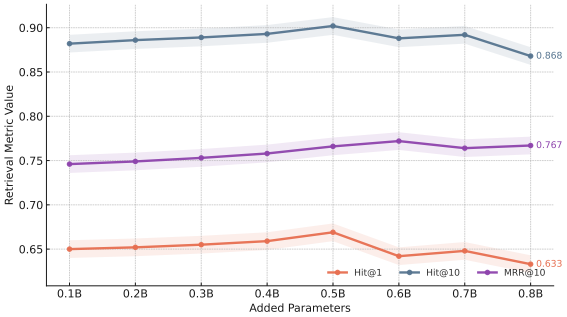
(a) Standard DR on NQ



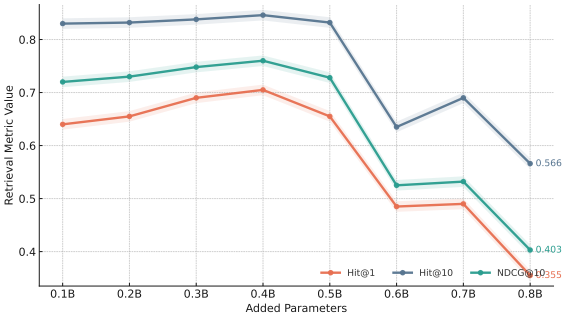
(b) Standard DR on MS MARCO



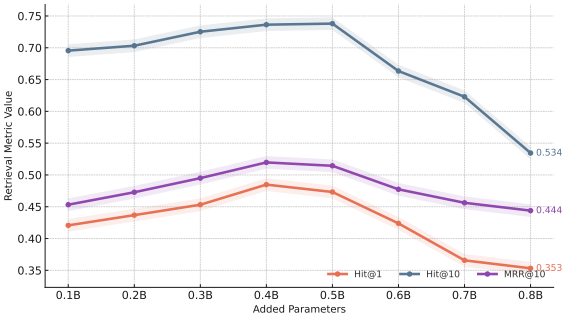
(c) MVDR on NQ



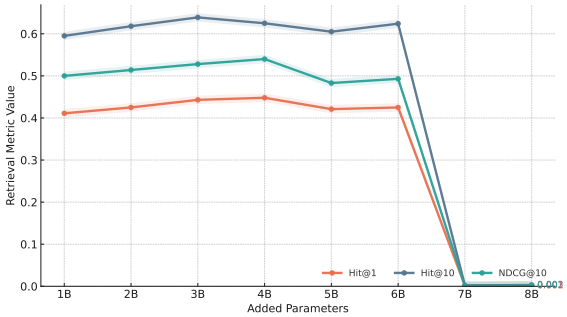
(d) MVDR on MS MARCO



(e) GR-codebook on NQ



(f) GR-codebook on MS MARCO



(g) GR-text on NQ

Figure 5: Extended results of model scaling.