Generative Retrieval with Few-shot Indexing

Arian Askari[1]*, Chuan Meng[2]*, Mohammad Aliannejadi[2], Zhaochun Ren[1], Evangelos Kanoulas[2], Suzan Verberne[1],

[1]Leiden University, [2]University of Amsterdam

{a.askari, z.ren, s.verberne}@liacs.leidenuniv.nl {c.meng, m.aliannejadi, e.kanoulas}@uva.nl

Abstract

Existing generative retrieval (GR) approaches rely on *training-based indexing*, i.e., fine-tuning a model to memorise the associations between a query and the document identifier (docid) of a relevant document. Training-based indexing has three limitations: high training overhead, under-utilisation of the pre-trained knowledge of large language models (LLMs), and challenges in adapting to a dynamic document corpus. To address the above issues, we propose a novel **few-shot** indexing-based **GR** framework (Few-Shot GR). It has a novel few-shot indexing process, where we prompt an LLM to generate docids for all documents in a corpus, ultimately creating a docid bank for the entire corpus. During retrieval, we feed a query to the same LLM and constrain it to generate a docid within the docid bank created during indexing, and then map the generated docid back to its corresponding document. Few-Shot GR relies solely on prompting an LLM without requiring any training, making it more efficient. Moreover, we devise few-shot indexing with *one-to-many mapping* to further enhance Few-Shot GR. Experiments show that Few-Shot GR achieves superior performance to state-of-the-art GR methods that require heavy training.

1 Introduction

Generative retrieval (GR) has emerged as a novel paradigm in information retrieval (IR) (Zeng et al., 2024b,a; Kuo et al., 2024; Li et al., 2024c,b). Unlike the traditional IR paradigm, decoupling indexing and retrieval processes, the paradigm of GR consolidates both processes into a single model (Tay et al., 2022). Studies in GR typically regard indexing and retrieval as training and inference processes, respectively. The indexing (training) process typically trains a seq2seq model (Raffel et al., 2020) to map queries to the docids corresponding to relevant documents, using extensive training data of query–docid pairs (Zhuang et al., 2022). In the retrieval (inference) process, the trained model takes a query text as input and directly generates potentially relevant docids.

Limitations. Existing studies typically rely on *training-based indexing* to memorise the associations between a query and its docid. The nature of training-based indexing results in three limitations: (i) The approach has a high training overhead (Li et al., 2024c). Existing studies typically use an LLM (or a pre-trained language model) (Lee et al., 2023; Li et al., 2024a) as the backbone and then fine-tune it with a new learning objective: mapping query text to docids. Fine-tuning an LLM with a new objective demands large-scale query–docid pairs, considerable time, and numerous GPUs. (ii) The approach does not make effective use of LLMs' pre-trained knowledge. Because there is a gap between the learning objectives of LLMs pre-training (natural language generation) and GR fine-tuning (query–docid mapping), fine–tuning an LLM with GR's objective may cause the LLM to forget its pre-trained knowledge (Li et al., 2024c). Little research has explored mainly using LLMs' pre-trained knowledge for GR indexing, without heavy training (Li et al., 2024c). (iii) It is challenging to handle a dynamic corpus. Training a model to memorise new documents inevitably leads to forgetting old ones (Li et al., 2024b). While existing studies propose solutions to mitigate this issue (Mehta et al., 2022; Kishore et al., 2023; Chen et al., 2023; Guo et al., 2024), the problem persists due to the inherent nature of training.

A new perspective on GR. To address the above limitations, we propose a **few-shot** indexing-based **GR** framework (Few-Shot GR). Unlike previous GR approaches based on training-based indexing, Few-Shot GR has a *few-shot indexing* process, where we index a document corpus without requiring any training. Specifically, in the few-shot index-

* These two authors contributed equally.

ing process, Few-Shot GR prompts an *open-source* LLM in a few-shot manner to generate a free-text docid for each document in a corpus. This process ultimately produces a *docid bank* for all documents in an entire corpus. Note that unlike the methods proposed by Li et al. (2023a,b), which first generate synthetic docids for documents (e.g., by prompting GPT-3.5) and then train another model to learn the mapping from query text to these docids, we do not need any training steps. During the retrieval process (inference), the same LLM used in few-shot indexing takes a query as input and uses constrained beam search (De Cao et al., 2020) to ensure the generated free-text docid matches a valid docid created during few-shot indexing.

We believe Few-Shot GR opens new avenues for GR by addressing the issues of high training overhead and under-utilisation of LLMs's pre-trained knowledge. Few-Shot GR can potentially alleviate the challenge of handling dynamic corpora posed by training-based indexing. This is because Few-Shot GR allows easy addition or removal of docids in the docid bank created during few-shot indexing, and so does not suffer from the forgetting issue.

However, the implementation of Few-Shot GR brings one new challenge: We found that generating only one docid per document during few-shot indexing results in limited retrieval quality. This occurs because a document can be relevant to multiple diverse queries; during retrieval, when the LLM is fed with different queries that share the same relevant document, it is hard for the LLM to always point to one docid.

We therefore further improve Few-Shot GR to address the challenge. Unlike most GR studies that generate a single docid per document, we devise few-shot indexing with *one-to-many mapping*, which enhances few-shot indexing by, for each document, generating multiple docids. This approach allows a relevant document to be mapped back by multiple various docids that are generated in response to different queries during retrieval.

Experiments. We equip Few-Shot GR with Llama-3-8B-Instruct (AI@Meta, 2024) for few-shot indexing and retrieval. Experiments on Natural Questions (NQ) (Kwiatkowski et al., 2019) demonstrate that Few-Shot GR outperforms or performs comparably to state-of-the-art GR approaches (Lee et al., 2023; Sun et al., 2024). Moreover, our analyses reveal that two critical factors contribute to the success of Few-Shot GR: conducting one-to-many mapping during few-shot indexing, and selecting an effective LLM. Finally, we demonstrate that few-shot indexing is significantly more efficient than training-based indexing.

Example1:
Query: Provide list of the olympic games?
Identifier: olympic-games-list

Example2:
Query: What is minority interest in accounting?
Identifier: subsidiary-corporation-parent

Example3:
Query: How does photosynthesis work in plants?
Identifier: photosynthesis-plant-process

Example4:
Query: {new query}
Identifier:

Figure 1: Prompt used for indexing and retrieval. The three queries in the demonstration examples are sampled from NQ's training set (Kwiatkowski et al., 2019), while their corresponding docids are annotated by the authors.

Our main contributions are as follows:

- We propose Few-Shot GR, a novel GR framework, which conducts GR indexing solely with prompting an LLM without requiring any training.
- We devise few-shot indexing with one-to-many mapping to further enhance Few-Shot GR's performance.
- We conduct experiments on the NQ dataset, showing that Few-Shot GR achieves superior performance to state-of-the-art GR methods that require heavy training.

2 Methodology

Few-Shot GR has two essential steps: (i) few-shot indexing with one-to-many mapping, and (ii) retrieval with constrained beam search.

Few-shot indexing with one-to-many mapping. Given a corpus $C = \{d_1, \cdots, d_i, \cdots, d_{|C|}\}$ with $|C|$ documents, this step uses an LLM to generate n distinct free-text docids $\{id_1, \cdots, id_j, \cdots, id_n\}$ for each document d in the corpus C. Ultimately, we create a *docid bank* B that contains docids for all documents (n docids for each document) in C.

Following the GR literature (Zhuang et al., 2022; Pradeep et al., 2023), which shows that replacing documents with their corresponding pseudo queries during indexing results in better retrieval quality, we use only pseudo queries for indexing. Specifically, we first generate n pseudo queries $\{\hat{q}_1, \cdots, \hat{q}_j, \cdots, \hat{q}_n\}$ for a document d_i and only feed the generated pseudo queries to the LLM to generate n corresponding docids $\{id_1, \cdots, id_j, \cdots, id_n\}$, formally:

$$\begin{aligned} \hat{q}_j =& \mathrm{QG}(d_i), \\ id_j =& \mathrm{LLM}(\hat{q}_j), \end{aligned} \tag{1}$$

where QG is a pseudo query generator, $i = 1, \cdots, |C|$ and $j = 1, \cdots, n$. As depicted in Figure 1, we prompt the LLM in a few-shot manner.

At the end of the few-shot indexing process, we remove redundant docids from the *docid bank* B.

Retrieval with constrained beam search. Given a user query q and the *docid bank* B created in the previous stage, this step aims to use the same prompt (see Figure 1) and the LLM (see Equation 1) from the indexing phase to generate a docid id, formally:

$$id = \mathrm{LLM}(q), \tag{2}$$

Where we use constrained beam search (De Cao et al., 2020) to the LLM's decoding, ensuring the generated docid id matches a valid docid in the *docid bank* B. Finally, we map the matched valid docid back to its corresponding document. Note that the *docid bank* B undergoes de-duplication, ensuring that each docid uniquely corresponds to a single document.

3 Experimental setup

Datasets. We use NQ320K, a version of Natural Questions (NQ) (Kwiatkowski et al., 2019), has been widely used for GR evaluation (Lee et al., 2023; Sun et al., 2024; Tay et al., 2022). NQ320K consists of 320k relevant query–document pairs, 100k documents, and 7,830 test queries. Following recent studies (Lee et al., 2023; Sun et al., 2024), we fetch and process NQ320K using the script released by Wang et al. (2022),[1] to ensure our results are comparable with previous work.

Baselines. We use non-GR and GR baselines. Following Lee et al. (2023), we use the following non-GR baselines: BM25 (Robertson et al., 2009), DPR (Karpukhin et al., 2020), ANCE (Xiong et al., 2021) and SentenceT5 (Ni et al., 2022a), and GTR-base (Ni et al., 2022b). We use the following GR baselines (training-based indexing): (i) SEAL (Bevilacqua et al., 2022) learns to generate n-grams-based docids and applies FM-index (Ferragina and Manzini, 2000). (ii) DSI (Tay et al., 2022) learns to generate numeric identifiers. (iii) DSI-QG (Zhuang et al., 2022) augments DSI training by using pseudo queries; we replicate DSI-QG using the pseudo query generator provided by the original paper. (iv) DSI-QG (InPars) uses the pseudo query generator from InPars (Bonifacio et al., 2022). (v) GenRET (Sun et al., 2024) learns to assign numeric docids based on an auto-encoding scheme. (vi) TOME (Ren et al., 2023) learns to generate document URLs. (vii) GLEN (Lee et al., 2023) learns dynamic lexical docids.

Table 1: Retrieval quality of Few-Shot GR and baselines on NQ320K. DSI-QG (InPars) and Few-Shot GR use the query generator from InPars (Bonifacio et al., 2022) to generate pseudo queries. The best value in each column is marked in **bold**, and the second best is <u>underlined</u>.

Method	Recall@1	Recall@10	MRR@100
BM25	29.7	60.3	40.2
DocT5Query	38.0	69.3	48.9
DPR	50.2	77.7	59.9
ANCE	50.2	78.5	60.2
SentenceT5	53.6	83.0	64.1
GTR-base	56.0	84.4	66.2
SEAL	59.9	81.2	67.7
DSI	55.2	67.4	59.6
NCI	66.4	85.7	73.6
DSI-QG	63.1	80.7	69.5
DSI-QG (InPars)	63.9	82.0	71.4
GenRET	68.1	**88.8**	<u>75.9</u>
TOME	66.6	–	–
GLEN	<u>69.1</u>	86.0	75.4
Few-Shot GR	**70.1**	<u>87.6</u>	**77.4**

Evaluation metrics. In line with recent GR research (Lee et al., 2023; Sun et al., 2024), we adopt Recall@1, Recall@10 and MRR@100.

Implementation details. We equip Few-Shot GR with llama-3-8B-Instruct for indexing and retrieval. We generate 10 docids per document during few-shot indexing. We set the maximum and minimum lengths for docid generation to 15 and 3 tokens, respectively. We employ the query generator from InPars (Bonifacio et al., 2022) for generating pseudo queries in Equation 1. We conduct parameter tuning on the training set of NQ320K.

[1] https://github.com/solidsea98/Neural-Corpus-Indexer-NCI

4 Result & analysis

Comparison with baselines. Table 1 shows the retrieval quality of Few-Shot GR and all baselines on NQ320K. The leading observation is that Few-Shot GR outperforms all state-of-the-art baselines across all metrics, except for GenRET in terms of Recall@10. This indicates that our proposed few-shot indexing is a highly effective new GR paradigm compared to training-based indexing.

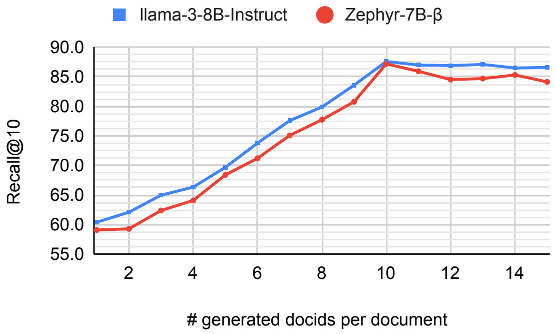

Figure 2: Few-Shot GR's retrieval quality w.r.t. # generated docids per document in few-shot indexing.

The impact of # docids generated per document. Figure 2 shows Few-Shot GR's performance w.r.t. # generated docids per document during few-shot indexing; we equip Few-Shot GR with llama-3-8B-Instruct or Zephyr-7B-β (Tunstall et al., 2023). We found that Few-Shot GR's performance improves as it generates more docids per document during indexing, reaching saturation when generating 10 docids per document. E.g., when using Llama-3, increasing the number of generated docids from 1 to 10 leads to an improvement of 27.2% in Recall@10. It suggests that our devised "one-to-many mapping" is a key factor in the success of few-shot indexing. Table 4 in the appendix gives an example of 10 distinct docids generated by Few-Shot GR for a specific document.

Table 2: Retrieval quality of Few-Shot GR equipped with different LLMs on NQ320K.

Method	Recall@1	Recall@10	MRR@100
T5-base	52.4	66.4	55.8
Zephyr-7B-β	69.9	87.2	77.8
llama-3-8B-Instruct	**70.1**	**87.6**	**77.4**

The impact of LLMs choices. Table 2 shows Few-Shot GR's performance using different LLMs; here we compare T5-base, Zephyr-7B-β, and llama-3-8B-Instruct. We found that Llama-3-8B-Instruct performs the best across all metrics, followed by Zephyr-7B-β. However, both markedly outperform T5-base in terms of performance. It suggests that selecting an effective LLM is another critical factor contributing to the success of Few-Shot GR.

Table 3: Efficiency of indexing and retrieval for Few-Shot GR and training-based GR baselines on NQ320K. Few-Shot GR uses llama-3-8B-Instruct and generates 10 docids per document during few-shot indexing.

Method	Indexing (hr)	Retrieval (ms)
DSI-QG	240	72
GenRET	≈16,800	72
Few-Shot GR	37	98

Efficiency of indexing and retrieval. Table 3 presents the indexing time and retrieval latency for Few-Shot GR compared to two training-based GR methods, DSI-QG (Zhuang et al., 2022) and GenRET (Sun et al., 2024). The time cost of indexing is measured in hours (hr) on the training set of NQ320K, while the retrieval query latency is measured in milliseconds (ms) on the test set of NQ320K. We perform all measurements on a single A100 GPU (80GB) with a batch size of 16, except for the indexing (training) of GenRET. We inquired with the authors of GenRET (Sun et al., 2024) about GenRET's indexing (training) time, and they indicated it took 7 days on 100 A100 GPUs. This implies it may take approximately 16,800 hours on a single A100 GPU. We found that Few-Shot GR is significantly more efficient in indexing than existing GR methods. Also, Few-Shot GR achieves similar retrieval query latency compared to existing GR methods.

5 Conclusions

We have proposed a new, efficient, and effective GR paradigm, Few-Shot GR, featuring a novel few-shot indexing process that solely relies on prompting an LLM to record associations between queries and their docids, eliminating the need for any training steps. Given a query, Few-Shot GR conducts retrieval by promoting the LLM used for indexing and constraining it to generate a docid within the recorded docid created in indexing. We have designed *few-shot indexing with one-to-many mapping* to further enhance Few-Shot GR's indexing. Experimental results demonstrate that GR achieves superior performance to state-of-the-art GR methods that require heavy training.

Limitations

We acknowledge the limitations of our work and outline avenues for future research. First, we only verify Few-Shot GR's effectiveness on one dataset, NQ320k. It is valuable to investigate Few-Shot GR 's performance on other ranking datasets, such as MS MARCO (Bajaj et al., 2016) and BEIR (Thakur et al., 2021). Second, existing work has shown that GR methods using training-based indexing perform worse as the corpus size increases (Pradeep et al., 2023). Since the dataset we used in our paper (NQ320K) has a document corpus with only 100k documents, we have yet to validate the effectiveness of Few-Shot GR on a document corpus with millions of documents. It is worthwhile to investigate whether Few-Shot GR's effectiveness would generalise to a large-scale document corpus. Third, we claim that Few-Shot GR can potentially deal with a dynamic document corpus better than training-based indexing, because of Few-Shot GR's non-training nature. This is because Few-Shot GR can easily add or remove docids in the docid bank created during few-shot indexing, and it does not suffer from the issue of forgetting. Several existing GR methods that use training-based indexing attempt to address this issue (Mehta et al., 2022; Kishore et al., 2023; Chen et al., 2023; Guo et al., 2024). It would be valuable to design experiments in the future to compare Few-Shot GR with these methods.

Acknowledgments

This research was partially supported by EU Horizon 2020 ITN/ETN on Domain Specific Systems for Information Extraction and Retrieval (H2020-EU.1.3.1., ID: 860721), and the China Scholarship Council (CSC) under grant number 202106220041.

References

AI@Meta. 2024. Llama 3 model card.

Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Xiaodong Liu Jianfeng Gao, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, Mir Rosenberg, Xia Song, Alina Stoica, Saurabh Tiwary, and Tong Wang. 2016. Ms marco: A human generated machine reading comprehension dataset. In *NIPS*.

Michele Bevilacqua, Giuseppe Ottaviano, Patrick Lewis, Scott Yih, Sebastian Riedel, and Fabio Petroni. 2022. Autoregressive search engines: Generating substrings as document identifiers. In *NeurIPS*, volume 35, pages 31668–31683.

Luiz Bonifacio, Hugo Abonizio, Marzieh Fadaee, and Rodrigo Nogueira. 2022. InPars: Unsupervised dataset generation for information retrieval. In *SIGIR*, page 2387–2392, New York, NY, USA. Association for Computing Machinery.

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 *CIKM*, pages 306–315.

Nicola De Cao, Gautier Izacard, Sebastian Riedel, and Fabio Petroni. 2020. Autoregressive entity retrieval. In *ICLR*.

Paolo Ferragina and Giovanni Manzini. 2000. Opportunistic data structures with applications. In *Proceedings 41st annual symposium on foundations of computer science*, pages 390–398. IEEE.

Jiafeng Guo, Changjiang Zhou, Ruqing Zhang, Jiangui Chen, Maarten de Rijke, Yixing Fan, and Xueqi Cheng. 2024. Corpusbrain++: A continual generative pre-training framework for knowledge-intensive language tasks. *arXiv preprint arXiv:2402.16767*.

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

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.

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. *TACL*, 7:453–466.

Sunkyung Lee, Minjin Choi, and Jongwuk Lee. 2023. Glen: Generative retrieval via lexical index learning. In *EMNLP*, pages 7693–7704.

Haoxin Li, Phillip Keung, Daniel Cheng, Jungo Kasai, and Noah A Smith. 2023a. Acid: Abstractive, content-based ids for document retrieval with language models. *arXiv preprint arXiv:2311.08593*.

Xiaoxi Li, Zhicheng Dou, Yujia Zhou, and Fangchao Liu. 2024a. Corpuslm: Towards a unified language model on corpus for knowledge-intensive tasks.

Xiaoxi Li, Jiajie Jin, Yujia Zhou, Yuyao Zhang, Peitian Zhang, Yutao Zhu, and Zhicheng Dou. 2024b. From matching to generation: A survey on generative information retrieval. *arXiv preprint arXiv:2404.14851*.

Yongqi Li, Xinyu Lin, Wenjie Wang, Fuli Feng, Liang Pang, Wenjie Li, Liqiang Nie, Xiangnan He, and Tat-Seng Chua. 2024c. A survey of generative search and recommendation in the era of large language models. *arXiv preprint arXiv:2404.16924*.

Yongqi Li, Nan Yang, Liang Wang, Furu Wei, and Wenjie Li. 2023b. Multiview identifiers enhanced generative retrieval. In *ACL*, pages 6636–6648.

Sanket Vaibhav Mehta, Jai Gupta, Yi Tay, Mostafa Dehghani, Vinh Q Tran, Jinfeng Rao, Marc Najork, Emma Strubell, and Donald Metzler. 2022. Dsi++: Updating transformer memory with new documents. *arXiv preprint arXiv:2212.09744*.

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 ACL*, 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 *EMNLP*, pages 9844–9855.

Ronak Pradeep, Kai Hui, Jai Gupta, Adam D Lelkes, Honglei Zhuang, Jimmy Lin, Donald Metzler, and Vinh Q Tran. 2023. How does generative retrieval scale to millions of passages? *arXiv preprint arXiv:2305.11841*.

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. *JMLR*, 21(140):1–67.

Ruiyang Ren, Wayne Xin Zhao, Jing Liu, Hua Wu, Ji-Rong Wen, and Haifeng Wang. 2023. Tome: A two-stage approach for model-based retrieval. In *ACL*, pages 6102–6114.

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

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. *NeurIPS*, 36.

Yi Tay, Vinh Tran, Mostafa Dehghani, Jianmo Ni, Dara Bahri, Harsh Mehta, Zhen Qin, Kai Hui, Zhe Zhao, Jai Gupta, et al. 2022. Transformer memory as a differentiable search index. In *NeurIPS*, volume 35, pages 21831–21843.

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2021. Beir: A heterogeneous benchmark for zero-shot evaluation of information retrieval models. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro von Werra, Clémentine Fourrier, Nathan Habib, et al. 2023. Zephyr: Direct distillation of lm alignment. *arXiv preprint arXiv:2310.16944*.

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. In *NeurIPS*, volume 35, pages 25600–25614.

Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul N. Bennett, Junaid Ahmed, and Arnold Overwijk. 2021. Approximate nearest neighbor negative contrastive learning for dense text retrieval. In *ICLR*.

Hansi Zeng, Chen Luo, Bowen Jin, Sheikh Muhammad Sarwar, Tianxin Wei, and Hamed Zamani. 2024a. Scalable and effective generative information retrieval. In *WWW*, pages 1441–1452.

Hansi Zeng, Chen Luo, and Hamed Zamani. 2024b. Planning ahead in generative retrieval: Guiding autoregressive generation through simultaneous decoding. *arXiv preprint arXiv:2404.14600*.

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 Appendix

Case study of docids generated by Few-Shot GR. Table 4 gives an example of 10 distinct docids generated by Few-Shot GR for a specific document in NQ320K. It shows that docids generated by Few-Shot GR are various.

Table 4: Case study of the 10 docids generated by Few-Shot GR for a document on NQ320K. Few-Shot GR uses Llama-3-8B-Instruct.

Document text	Pseudo queries	docids
In accounting, minority interest (or non-controlling interest) is the portion of a subsidiary corporation's stock that is not owned by the parent corporation. The magnitude of the minority interest in the subsidiary company is generally less than 50% of outstanding shares, or the corporation would generally cease to be a subsidiary of the parent.	What is minority interest in accounting?	minority-interest-accounting
	What is non-controlling interest in accounting?	non-controlling-interest-accounting
	How is minority interest defined in accounting?	minority-interest-definition
	How is minority interest calculated in accounting?	minority-interest-calculation
	What is the significance of minority interest in accounting?	minority-interest-significance
	How does minority interest affect financial statements in accounting?	minority-interest-financial-statements
	How is minority interest treated in consolidated financial statements in accounting?	minority-interest-consolidated-financial-statements
	What is the impact of minority interest on the parent company's earnings per share in accounting?	minority-interest-impact-eps
	How is minority interest reported in financial statements of a subsidiary company in accounting?	minority-interest-reporting-subsidiary
	What is the role of minority interest in a parent corporation's financial statements?	minority-interest-parent-corporation