

Generative Retrieval with Few-shot Indexing

Arian Askari^{1†[0000-0003-4712-832X]}, Chuan Meng^{2†[0000-0002-1434-7596]},
 Mohammad Aliannejadi^{3[0000-0002-9447-4172]}, Zhaochun
 Ren^{1[0000-0002-9076-6565]}, Evangelos Kanoulas^{3[0000-0002-8312-0694]}, and Suzan
 Verberne^{1[0000-0002-9609-9505]}

¹ Leiden University, Leiden, The Netherlands

{a.askari, z.ren, s.verberne}@liacs.leidenuniv.nl

² The University of Edinburgh, Edinburgh, United Kingdom

chuan.meng@ed.ac.uk

³ University of Amsterdam, Amsterdam, The Netherlands

{m.aliannejadi, e.kanoulas}@uva.nl

Abstract. Existing generative retrieval (GR) methods rely on *training-based indexing*, which fine-tunes a model to memorise associations between queries and the document identifiers (docids) of relevant documents. Training-based indexing suffers from high training costs, under-utilisation of pre-trained knowledge in large language models (LLMs), and limited adaptability to dynamic document corpora. To address the issues, we propose a **few-shot** indexing-based **GR** framework (Few-Shot GR). It has a few-shot indexing process without any training, 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. 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 requiring heavy training.

Keywords: Generative retrieval · Neural ranking · Few-shot learning

1 Introduction

Generative retrieval (GR) [5,3,36,35,11,16,15] is a new paradigm in information retrieval (IR). Unlike traditional IR that decouples indexing and retrieval, GR unifies both processes into a single model [31]. Studies in GR typically regard indexing and retrieval as training and inference processes, respectively. The indexing (training) process typically trains a seq2seq model [28] to map queries to the docids corresponding to relevant documents, using extensive training data of query–docid pairs [37]. 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

[†]These two authors contributed equally.

indexing has two main limitations: (i) The approach has a high training overhead [16]. Existing studies typically use an LLM [13,14] 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 (text 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 [16]. Little research has explored mainly using LLMs’ pre-trained knowledge for GR indexing, without heavy training [16].

A new perspective on GR. To address the limitation, 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 indexing process, Few-Shot GR prompts an LLM in a few-shot way 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. During the retrieval process (inference), the same LLM used in few-shot indexing takes a query as input and uses constrained beam search [6] to ensure the generated docid matches a valid docid created during few-shot indexing.

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 LLMs for few-shot indexing and retrieval. Experiments on Natural Questions (NQ) [12] and MS MARCO show that Few-Shot GR outperforms or performs comparably to state-of-the-art GR methods [13,30]. 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.

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.
- Experiments show that Few-Shot GR achieves superior performance to state-of-the-art GR methods that require heavy training.

Example1	Example2
Query: Provide list of the olympic games?	Query: What is minority interest in ac-
Identifier: olympic-games-list	counting?
	Identifier: subsidiary-corporation-parent
Example3	Example4
Query: How does photosynthesis work in plants?	Query: {new query}
Identifier: photosynthesis-plant-process	Identifier:

Fig. 1: Prompt used for indexing and retrieval. The three queries in the demonstration examples are sampled from NQ’s training set [12], while their corresponding docids are annotated by the authors.

2 Methodology

Few-shot indexing with one-to-many mapping. Let $C = \{d_1, \dots, d_i, \dots, d_{|C|}\}$ be a corpus with $|C|$ documents; this step aims to use an LLM to generate n distinct free-text docids $\{id_1, \dots, id_j, \dots, id_n\}$ 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 [37,27], 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 $\{\hat{q}_1, \dots, \hat{q}_j, \dots, \hat{q}_n\}$ for a document d_i and only feed the generated pseudo queries to the LLM to generate n corresponding docids $\{id_1, \dots, id_j, \dots, id_n\}$, formally:

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

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

After few-shot indexing, we deduplicate docids in the *docid bank* B . The devised one-to-many mapping technique during few-shot indexing effectively captures diverse relevance signals, addressing limitations faced by prior methods relying on single identifier generation per document.

Table 4 in the appendix gives an example of 10 distinct docids generated by Few-Shot GR for a specific document in NQ320K [13,30,31].

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:

$$id = \text{LLM}(q), \quad (2)$$

Where we use constrained beam search [6] 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.

Table 1: Retrieval quality of Few-Shot GR and baselines on NQ320K and MS300K. DSI-QG (InPars) and Few-Shot GR use the query generator from InPars [2] to generate pseudo queries. Methods marked \dagger are our reimplementations; all other results are from the corresponding papers [34,30,29,34]. The best value in each column is marked in **bold**, and the second best is underlined.

Method	NQ320K			MS300K		
	Recall@1	Recall@10	MRR@100	Recall@1	Recall@10	MRR@10
BM25	29.7	60.3	40.2	39.1	69.1	48.6
DocT5Query	38.0	69.3	48.9	46.7	76.5	56.2
ANCE	50.2	78.5	60.2	45.6	75.7	55.6
SentenceT5	53.6	83.0	64.1	41.8	75.4	52.8
GTR-base	56.0	84.4	66.2	—	—	—
SEAL	59.9	81.2	67.7	25.9	68.6	40.2
DSI	55.2	67.4	59.6	32.4	69.9	44.3
NCI	66.4	85.7	73.6	30.1	64.3	41.7
DSI-QG \dagger	63.1	80.7	69.5	41.0	71.2	50.7
DSI-QG (InPars) \dagger	63.9	82.0	71.4	41.3	71.5	50.0
TOME	66.6	—	—	—	—	—
GLEN	69.1	86.0	75.4	—	—	—
GenRET	68.1	<u>88.8</u>	75.9	47.9	79.8	58.1
NOVO	<u>69.3</u>	89.7	<u>76.7</u>	<u>49.1</u>	<u>80.8</u>	59.2
Few-Shot GR	70.1	87.6	77.4	49.6	81.2	<u>59.1</u>

3 Experimental setup

Datasets. We evaluate on NQ320K [13,30,31] and MS300K [34,18]; both have widely been used for GR evaluation. NQ320K is a version of Natural Questions (NQ) [12]; NQ320K consists of 320k relevant query–document pairs, 100k documents, and 7,830 test queries. MS300K is a version of MS MARCO; MS300K contains 300k query–document pairs, 320k documents, and 5,187 test queries.

Baselines. We use non-GR and GR baselines. Following [13], we use the following non-GR baselines: BM25, DPR [9], SentenceT5 [25], and GTR-base [26]. We use the following GR baselines (training-based indexing): (i) SEAL [1] learns to generate n-grams-based docids and applies FM-index [7]. (ii) DSI [31] learns to generate numeric identifiers. (iii) DSI-QG [37] 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 [2]. (v) TOME [29] learns to generate document URLs. (vi) GLEN [13] learns dynamic lexical docids. (vii) GenRET [30] learns to assign numeric docids based on an auto-encoding scheme. (viii) NOVO [34] learns interpretable docids.

Evaluation metrics. In line with recent GR work [34,13,30], we report Recall@1,10 on both datasets, plus MRR@100 (NQ320K) and MRR@10 (MS300K).

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

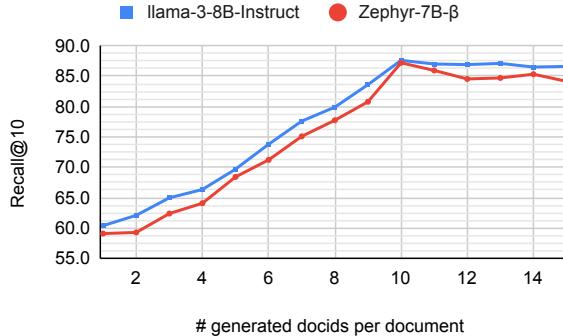


Fig. 2: Few-Shot GR’s retrieval quality w.r.t. # generated docids per document in few-shot indexing on NQ320K.

Table 2: Retrieval quality of Few-Shot GR 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

and 3 tokens, respectively. We employ the query generator from InPars [2] for generating pseudo queries in Equation 1. We conduct parameter tuning on the training set of NQ320K or MS300K.

4 Result and analysis

Comparison with baselines. Table 1 shows the retrieval quality of Few-Shot GR and all baselines on NQ320K and MS300K. The leading observation is that Few-Shot GR outperforms all baselines across all metrics, except GenRET/NOVO on Recall@10 (NQ320K)/MRR@10 (MS300K). This shows that our proposed few-shot indexing is highly effective versus training-based indexing. Notably, while GenRET/NOVO is slightly better on those metrics, it requires large training corpora and heavy model-specific training, which may not be feasible in low-resource settings. In contrast, Few-Shot GR achieves strong results using only a small set of examples, making it more practical.

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 on NQ320K; we equip Few-Shot GR with llama-3-8B-Instruct or Zephyr-7B- β [33]. We found that Few-Shot GR’s performance improves as it generates more docids per document during indexing, reaching saturation when generating 10 docids. E.g., with Llama-3, increasing the number of generated docids from 1 to 10 yields a 27.2% improvement in Recall@10. It suggests that our devised “one-to-many mapping” is key to the success of few-shot indexing. The trend is similar on MS300K; we report only NQ320K hereafter due to space.

The impact of LLMs choices. Table 2 shows Few-Shot GR’s performance using different LLMs on NQ320K; here we compare T5-base, Zephyr-7B- β , and

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	$\approx 16,800$	72
Few-Shot GR	37	98

llama-3-8B-Instruct. We found that Llama-3-8B-Instruct performs the best across most 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.

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 [37] and GenRET [30]. 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 [30] 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 & Future Work

We have proposed a new, efficient, and effective GR paradigm, Few-Shot GR, featuring a 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. We have designed *few-shot indexing with one-to-many mapping* to further enhance Few-Shot GR’s indexing. Experimental results show that GR achieves superior performance to training-intensive state-of-the-art GR methods.

Suitability for dynamic corpora. Training-based indexing struggles with dynamic corpora, as training on new documents often causes forgetting of old ones [15]. Although several studies attempt to mitigate this issue [17,10,4,8], it remains inherent to training-based methods. Few-Shot GR alleviates this challenge by enabling easy addition or removal of docids in the few-shot indexing docid bank, thus avoiding catastrophic forgetting. Future work can further explore this direction.

The datasets used in this paper, NQ320K and MS300K, contain corpora of 100K and 320K documents, respectively. So it is worthwhile to test whether Few-Shot GR’s effectiveness would generalise to a million-document corpus. Also, it is worth testing Few-Shot GR on other datasets (e.g., BEIR [32] and conversational search domains [24,22,23,19]). Finally, exploring automatic retrieval quality prediction for generative retrieval methods is another promising direction [21,20].

References

1. Bevilacqua, M., Ottaviano, G., Lewis, P., Yih, S., Riedel, S., Petroni, F.: Autoregressive search engines: Generating substrings as document identifiers. In: NeurIPS. vol. 35, pp. 31668–31683 (2022)
2. Bonifacio, L., Abonizio, H., Fadaee, M., Nogueira, R.: InPars: Unsupervised dataset generation for information retrieval. In: SIGIR. p. 2387–2392 (2022)
3. Cai, H., Li, Y., Yuan, R., Wang, W., Zhang, Z., Li, W., Chua, T.S.: Exploring training and inference scaling laws in generative retrieval. In: SIGIR. pp. 1339–1349 (2025)
4. Chen, J., Zhang, R., Guo, J., de Rijke, M., Chen, W., Fan, Y., Cheng, X.: Continual learning for generative retrieval over dynamic corpora. In: CIKM. pp. 306–315 (2023)
5. Cheng, J., Dou, Z., Zhu, Y., Li, X.: Descriptive and discriminative document identifiers for generative retrieval. In: AAAI. vol. 39, pp. 11518–11526 (2025)
6. De Cao, N., Izacard, G., Riedel, S., Petroni, F.: Autoregressive entity retrieval. In: ICLR (2020)
7. Ferragina, P., Manzini, G.: Opportunistic data structures with applications. In: Proceedings 41st annual symposium on foundations of computer science. pp. 390–398. IEEE (2000)
8. Guo, J., Zhou, C., Zhang, R., Chen, J., de Rijke, M., Fan, Y., Cheng, X.: Corpus-brain++: A continual generative pre-training framework for knowledge-intensive language tasks. arXiv preprint arXiv:2402.16767 (2024)
9. Karpukhin, V., Oguz, B., Min, S., Lewis, P., Wu, L., Edunov, S., Chen, D., Yih, W.t.: Dense passage retrieval for open-domain question answering. In: EMNLP. pp. 6769–6781 (2020)
10. Kishore, V., Wan, C., Lovelace, J., Artzi, Y., Weinberger, K.Q.: Incdsi: Incrementally updatable document retrieval. In: International Conference on Machine Learning. pp. 17122–17134. PMLR (2023)
11. Kuo, T.L., Chiu, T.W., Lin, T.S., Wu, S.Y., Huang, C.W., Chen, Y.N.: A survey of generative information retrieval. arXiv preprint arXiv:2406.01197 (2024)
12. Kwiatkowski, T., Palomaki, J., Redfield, O., Collins, M., Parikh, A., Alberti, C., Epstein, D., Polosukhin, I., Devlin, J., Lee, K., et al.: Natural questions: A benchmark for question answering research. TACL **7**, 453–466 (2019)
13. Lee, S., Choi, M., Lee, J.: Glen: Generative retrieval via lexical index learning. In: EMNLP. pp. 7693–7704 (2023)
14. Li, X., Dou, Z., Zhou, Y., Liu, F.: Corpuslm: Towards a unified language model on corpus for knowledge-intensive tasks (2024)
15. Li, X., Jin, J., Zhou, Y., Zhang, Y., Zhang, P., Zhu, Y., Dou, Z.: From matching to generation: A survey on generative information retrieval. arXiv preprint arXiv:2404.14851 (2024)
16. Li, Y., Lin, X., Wang, W., Feng, F., Pang, L., Li, W., Nie, L., He, X., Chua, T.S.: A survey of generative search and recommendation in the era of large language models. arXiv preprint arXiv:2404.16924 (2024)
17. Mehta, S.V., Gupta, J., Tay, Y., Dehghani, M., Tran, V.Q., Rao, J., Najork, M., Strubell, E., Metzler, D.: Dsi++: Updating transformer memory with new documents. arXiv preprint arXiv:2212.09744 (2022)
18. Mekonnen, K.A., Tang, Y., de Rijke, M.: Lightweight and direct document relevance optimization for generative information retrieval. In: SIGIR. pp. 1327–1338 (2025)

19. Meng, C., Arabzadeh, N., Aliannejadi, M., De Rijke, M.: Query performance prediction: From ad-hoc to conversational search. In: SIGIR. pp. 2583–2593 (2023)
20. Meng, C., Arabzadeh, N., Askari, A., Aliannejadi, M., de Rijke, M.: Query performance prediction using relevance judgments generated by large language models. TOIS **43**(4), 1–35 (2025)
21. Meng, C., Fagioli, G., Aliannejadi, M., Ferro, N., Mothe, J.: Report on the 2nd workshop on query performance prediction and its applications in the era of large language models (QPP++ 2025) at ECIR 2025 **59**(1), 1–8 (2025)
22. Meng, C., Mo, F., Aliannejadi, M., Dalton, J., Nie, J.Y.: Conversational search: From fundamentals to frontiers in the age of agents. In: WWW (2026)
23. Meng, C., Tonolini, F., Mo, F., Aletras, N., Yilmaz, E., Kazai, G.: Bridging the gap: From ad-hoc to proactive search in conversations. In: SIGIR. pp. 64–74 (2025)
24. Mo, F., Meng, C., Aliannejadi, M., Nie, J.Y.: Conversational search: From fundamentals to frontiers in the LLM era. In: SIGIR. pp. 4094–4097 (2025)
25. Ni, J., Abrego, G.H., Constant, N., Ma, J., Hall, K., Cer, D., Yang, Y.: Sentence-t5: Scalable sentence encoders from pre-trained text-to-text models. In: Findings of ACL. pp. 1864–1874 (2022)
26. Ni, J., Qu, C., Lu, J., Dai, Z., Abrego, G.H., Ma, J., Zhao, V., Luan, Y., Hall, K., Chang, M.W., et al.: Large dual encoders are generalizable retrievers. In: EMNLP. pp. 9844–9855 (2022)
27. Pradeep, R., Hui, K., Gupta, J., Lelkes, A.D., Zhuang, H., Lin, J., Metzler, D., Tran, V.Q.: How does generative retrieval scale to millions of passages? arXiv preprint arXiv:2305.11841 (2023)
28. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P.J.: Exploring the limits of transfer learning with a unified text-to-text transformer. JMLR **21**(140), 1–67 (2020)
29. Ren, R., Zhao, W.X., Liu, J., Wu, H., Wen, J.R., Wang, H.: Tome: A two-stage approach for model-based retrieval. In: ACL. pp. 6102–6114 (2023)
30. Sun, W., Yan, L., Chen, Z., Wang, S., Zhu, H., Ren, P., Chen, Z., Yin, D., Rijke, M., Ren, Z.: Learning to tokenize for generative retrieval. NeurIPS **36** (2024)
31. Tay, Y., Tran, V., Dehghani, M., Ni, J., Bahri, D., Mehta, H., Qin, Z., Hui, K., Zhao, Z., Gupta, J., et al.: Transformer memory as a differentiable search index. In: NeurIPS. vol. 35, pp. 21831–21843 (2022)
32. Thakur, N., Reimers, N., Rücklé, A., Srivastava, A., Gurevych, I.: 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) (2021)
33. Tunstall, L., Beeching, E., Lambert, N., Rajani, N., Rasul, K., Belkada, Y., Huang, S., von Werra, L., Fourrier, C., Habib, N., et al.: Zephyr: Direct distillation of lm alignment. arXiv preprint arXiv:2310.16944 (2023)
34. Wang, Z., Zhou, Y., Tu, Y., Dou, Z.: Novo: Learnable and interpretable document identifiers for model-based ir. In: CIKM. pp. 2656–2665 (2023)
35. Zeng, H., Luo, C., Jin, B., Sarwar, S.M., Wei, T., Zamani, H.: Scalable and effective generative information retrieval. In: WWW. pp. 1441–1452 (2024)
36. Zeng, H., Luo, C., Zamani, H.: Planning ahead in generative retrieval: Guiding autoregressive generation through simultaneous decoding. arXiv preprint arXiv:2404.14600 (2024)
37. Zhuang, S., Ren, H., Shou, L., Pei, J., Gong, M., Zuccon, G., Jiang, D.: Bridging the gap between indexing and retrieval for differentiable search index with query generation. arXiv preprint arXiv:2206.10128 (2022)

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: Pseudo queries and corresponding docids generated by Few-Shot GR using Llama-3-8B-Instruct for a document from NQ320K.

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