# Recent Advances in Generative Information Retrieval

Yubao Tang
Ruqing Zhang
CAS Key Lab of Network Data Science and Technology, ICT, CAS
University of Chinese Academy of Sciences
Beijing, China
{tangyubao,zhangruqing}@ict.ac.cn

Jiafeng Guo
CAS Key Lab of Network Data Science and Technology, ICT, CAS
University of Chinese Academy of Sciences
Beijing, China
guojiafeng@ict.ac.cn

Maarten de Rijke
University of Amsterdam
Amsterdam, The Netherlands
m.derijke@uva.nl

## ABSTRACT

Generative retrieval (GR) has become a highly active area of information retrieval (IR) that has witnessed significant growth recently. Compared to the traditional "index-retrieve-then-rank" pipeline, the GR paradigm aims to consolidate all information within a corpus into a single model. Typically, a sequence-to-sequence model is trained to directly map a query to its relevant document identifiers (i.e., docids). This tutorial offers an introduction to the core concepts of the GR paradigm and a comprehensive overview of recent advances in its foundations and applications. We start by providing preliminary information covering foundational aspects and problem formulations of GR. Then, our focus shifts towards recent progress in docid design, training approaches, inference strategies, and the applications of GR. We end by outlining remaining challenges and issuing a call for future GR research. This tutorial is intended to be beneficial to both researchers and industry practitioners interested in developing novel GR solutions or applying them in real-world scenarios.

## CCS CONCEPTS

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

**ACM Reference Format:**
Yubao Tang, Ruqing Zhang, Jiafeng Guo, and Maarten de Rijke. 2023. Recent Advances in Generative Information Retrieval. In *Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region (SIGIR-AP '23), November 26–28, 2023, Beijing, China.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3624918.3629547

## 1 TUTORIAL INFORMATION

**On-site tutorial.** All presenters will attend SIGIR-AP in person to deliver this tutorial and engage in Q&A with the audience.

**Intended audience.** The tutorial is open to those with a basic understanding of information retrieval (IR) and natural language processing (NLP). It will appeal to both academic researchers specializing in IR/NLP and industry practitioners.

**Length.** This tutorial is scheduled to last for three hours.


*SIGIR-AP '23, November 26–28, 2023, Beijing, China*

ACM ISBN 979-8-4007-0408-6/23/11.
https://doi.org/10.1145/3624918.3629547

## 2 PRESENTERS

**Yubao Tang** is a Ph.D. student at the Institute of Computing Technology, Chinese Academy of Sciences. She obtained her M.Sc. degree from the Institute of Information Engineering, Chinese Academy of Sciences, and her B.Eng. from Sichuan University. Her research focuses on information retrieval, and she is the first author of a full paper on generative retrieval at KDD'23 [38].

**Ruqing Zhang** is an Associate Researcher at the Institute of Computing Technology, Chinese Academy of Sciences. Her recent research focuses on information retrieval, with a particular emphasis on generative information retrieval, the robustness of neural ranking models, and trustworthy retrieval through the lens of causality. She has authored several papers in the field of generative retrieval [3–6, 23, 38]. And Ruqing co-organized the first workshop on generative information retrieval at SIGIR 2023 (Gen-IR@SIGIR23) [1], which aimed to foster discussions and innovations in GR. Ruqing is the main contact person.

**Jiafeng Guo** is a Researcher at the Institute of Computing Technology, Chinese Academy of Sciences (CAS) and a Professor at the University of Chinese Academy of Sciences. He is the director of the CAS key lab of network data science and technology. He has worked on a number of topics related to web search and data mining, with a current focus on neural models for information retrieval and natural language understanding. He has received multiple best paper (runner-up) awards at leading conferences (CIKM'11, SIGIR'12, CIKM'17, WSDM'22). He has been (co)chair for many conferences, e.g., reproducibility track co-chair of SIGIR'23, workshop co-chair of SIGIR'21 and short paper co-chair of SIGIR'20. He serves as an associate editor for ACM Transactions on Information Systems and Information Retrieval Journal. Jiafeng has previously taught tutorials at ACML, CCIR and CIPS ATT.

**Maarten de Rijke** is a Distinguished University Professor of Artificial Intelligence and Information Retrieval at the University of Amsterdam. His research is focused on designing and evaluating trustworthy technology to connect people to information, particularly search engines, recommender systems, and conversational assistants. He is the scientific director of the Innovation Center for Artificial Intelligence and a former editor-in-chief of ACM Transactions on Information Systems and of Foundations and Trends in Information Retrieval, and a current co-editor-in-chief of Springer's Information Retrieval book series, (associate) editor for various journals and book series. He has been general (co)chair or program (co)chair for CIKM, ECIR, ICTIR, SIGIR, WSDM, WWW, and has previously taught tutorials at these same venues and AAAI.

## 3 MOTIVATION

Information retrieval (IR) is a core task in a wide range of real-world applications, such as web search [9, 12, 15, 29, 34] and question answering [11, 14, 17, 35]. It aims to retrieve information from a large repository that is relevant to an information need. Most existing IR methods follow a common pipeline paradigm of "index-retrieve-then-rank," which includes (i) building an index for each document in the corpus [7, 22]; (ii) retrieving an initial set of candidate documents for a query [16, 27]; and (iii) determining the relevance degree of each candidate [22, 24]. Despite its wide usage, this paradigm has limitations: (i) during training, heterogeneous modules with different optimization objectives may lead to sub-optimal performance, and capturing fine-grained relationships between queries and documents is challenging; and (ii) during inference, a large document index is needed to search over the corpus, which may come with substantial memory and computational requirements.

Recently, a fundamentally different paradigm, known as *generative retrieval* (GR) [26], has garnered attention to replace the long-standing "index-retrieve-then-rank" paradigm. The key idea of the GR paradigm is to parameterize the indexing, retrieval, and ranking components of traditional IR systems into a single consolidated model. A sequence-to-sequence (Seq2Seq) model is trained to directly map queries to their relevant document identifiers (docids). Such a single-step generative model dramatically simplifies the search process, can be optimized in an end-to-end manner, and can better leverage the capabilities of large language models (LLMs).

**Importance and timeliness.** In 2021, Metzler et al. [26] envisioned a model-based IR approach that replaces the long-standing "index-retrieve-then-rank" paradigm with a single consolidated model. With the expectation that similar successes achieved with generative language models in other areas like natural language processing could be replicated in IR, we have witnessed substantial growth in GR research, both in academia and industry, in recent years. A plethora of publications have emerged in reputable conferences, e.g., SIGIR [4, 5], CIKM [3, 6, 42], KDD [38], NeurIPS [2, 39, 41], ICLR [10], and ACL [8, 18, 21, 33], in Gen-IR@SIGIR2023 [23, 30, 31, 48], in journals [47], and on arXiv [20, 28, 37, 45, 46].

The first workshop on generative information retrieval at SIGIR 2023 (Gen-IR@SIGIR2023) [1] welcomed many submissions and attendees, underscoring the research community's current keen interest in this field. To the best of our knowledge, there is currently no tutorial that provides a comprehensive overview of the advances in GR. This is an opportune moment to provide such a tutorial that can arouse the interest of more researchers and help them gain a better understanding of this novel field.

## 4 OBJECTIVES

**1. Introduction.** We start by reminding our audience of the required background and examining the motivation behind GR.

**2. Preliminaries.** With GR, the document retrieval task is formulated as a Seq2Seq problem, i.e., directly generating identifiers of relevant documents with respect to the given query. To achieve this functionality, GR encompasses two fundamental training tasks [39], based on an encoder-decoder architecture: (i) *indexing* – this task aims to establish associations between each document and its corresponding docid; the GR model takes each original document as input and generates its docid as output in a straightforward Seq2Seq fashion; and (ii) *retrieval* – this task focuses on mapping each query to its relevant docids; given a query, the GR model learns to generate its relevant docid string.

It is crucial to store document information as comprehensively as possible during the indexing process, thus ensuring that the subsequent retrieval process is not hindered by information loss [8]. Using these two operations, a GR model can be trained to index a corpus of documents and optionally fine-tune with an available set of labeled query-document pairs. Thereafter, during inference, the optimized generative retriever can be used to efficiently retrieve relevant documents within a single neural model.

Building on these preliminaries, we will cover docid design, training approaches, inference strategies, and applications of GR in downstream scenarios.

**3. Docid designs.** With GR, employing identifiers, rather than generating original documents directly, could reduce irrelevant information in documents and make it easier for the model to memorize the corpus. Therefore, one of the key challenges in GR is how to assign a high-quality identifier to represent a document. An effective docid should be unique to enable effective distinction among different documents and concise for ease of generation. Therefore, we proceed to discuss the work related to docid designs.

Most existing GR approaches utilize pre-defined static docids, i.e., these docids are fixed and are not learnable during training the indexing and retrieval tasks. To be specific, these works usually leverage a single docid to represent the document, and several types of identifiers have been explored, including number-based and word-based docids. The number-based docids encompass atomic unique integers [25, 28, 39, 47], structured integer strings [39], semantically structured strings [30, 39, 41], product quantization code [3, 46], while the word-based docids primarily involve document titles [5, 6, 10, 18, 40], n-grams [2, 4, 20], important word sets [45], pseudo-queries [38], and URLs [33, 46]. Given that a document has the potential to answer multiple queries from different views, some research advocates the use of multiple types of identifiers to comprehensively represent a document [20, 21].

Although pre-defined static docids have demonstrated some effectiveness, they are not tailored to the retrieval objectives, limiting their capacity to adapt to semantic relationships within documents during the training process. Consequently, recent research [37, 42] has introduced document tokenization learning methods to acquire learnable docids for GR.

**4. Training approaches.** Here, we consider two main scenarios for training the GR model. The first, a more straightforward one, assumes a stationary learning scenario where the document collection is fixed and no longer updates. The second, a more practical scenario, is a dynamic corpora setting where information changes and new documents emerge incrementally over time.

The majority of GR research [2, 10, 39, 41, 48] primarily focuses on implementing GR in a stationary learning scenario. These works can be further categorized into supervised learning methods and pre-training methods, depending on the availability of labeled query-docid pairs. (i) For supervised learning methods, Tay et al. [39] introduced fundamental training strategies, jointly optimizing indexing and retrieval tasks using the standard Seq2Seq objective,

i.e., maximum likelihood estimation [43] with teacher forcing. Building upon this foundation, a series of improvements [31, 38, 41, 48] have been proposed, significantly enhancing performance. These solutions involve direct fine-tuning of off-the-shelf pre-trained generative models on downstream labeled datasets. (ii) In IR research, limited labeled data is often a challenge. Some researchers explore the design of self-supervised pre-training objectives to generate a large number of pseudo pairs of queries and docids [6]. The pre–trained model can then be further fine-tuned to improve retrieval performance for various downstream tasks.

In many scenarios, document collections are dynamic, with new documents continuously being added to the corpus, old documents being removed, or updated. A significant challenge in GR is how to enable the model to capture and remember information from new documents while minimizing the forgetting of information from previously learned documents. Mehta et al. [25] demonstrate that continually memorizing new documents leads to considerable forgetting of old documents. They achieved this by assigning each new document an arbitrary unique integer identifier and sampling some old documents using experience replay for incremental updates. Several follow-up approaches have been proposed to address this issue, such as updating a partial quantization codebook [3] and modifying training dynamics to reduce forgetting [44].

**5. Inference strategies.** During inference, when given a new query, we can easily employ the learned GR model to provide relevant documents through autoregressive generation. In cases where a single docid represents a document, the trained GR model autoregressively generates a ranked list of candidate docids in descending order of output likelihood conditioned on each query. To ensure the validity of the generated docids, three classical approaches are commonly used: constrained beam search [5, 6, 10, 19, 37, 38], constrained greedy search [45] and FM-index [2, 4, 42]. In cases where multiple docids represent a single document, some research [20, 21] combines the aforementioned approaches and designs heuristic scoring functions to determine the ranking order of relevant docids.

**6. Applications.** After discussing the basic building blocks of GR, we will demonstrate how GR models are adapted to downstream applications. First, we will discuss methods designed to enhance GR models for specific offline tasks, such as entity retrieval [10], fact checking [5, 6], recommender systems [32], multi-hop retrieval [19] and code generation [28]. Then, we will explore methods tailored for building more powerful GR models in industrial applications, such as the Baidu search system [38]. These examples underscore the tremendous promise and value of the GR paradigm in IR.

**7. Conclusions and future directions.** We conclude our tutorial by discussing several important questions and future directions, including (i) Most existing studies only demonstrate the effectiveness of their approaches over relatively small corpora or task-specific datasets. Evaluating at a larger scale remains a significant challenge for GR [31]. Therefore, one potential future direction is to explore how we can enhance the scalability of GR models to support complex, diverse, and dynamically changing retrieval tasks. (ii) Previous retrieval models are primarily discriminative models, where the core focus is on measuring the matching degree based on query–document pairs [13]. In contrast, GR takes the query as input and directly generates docids. It is evident that the relevance modeling mechanism centered around matching is no longer applicable. Consequently, there is an urgent need to understand the differences and connections between generative models and discriminative models in terms of fundamental indexing and retrieval mechanisms. (iii) When it comes to the practical deployment of retrieval systems, transparency, trustworthiness, and user-friendly interaction are pivotal for ensuring secure applications [36]. Additionally, it's worth considering how this technical evolution of search engines may effect the current content ecosystem.

## 5 RELEVANCE TO THE IR COMMUNITY

GR has garnered significant attention within the IR community on the back of the emergence of generative language models. At SIGIR 2023, Marc Najork, serving as the keynote speaker, provided a comprehensive summary of existing generative information retrieval systems and discussed many open challenges in this emerging field. And the first workshop on generative information retrieval also took place at SIGIR 2023. One of the research tracks at SIGIR-AP is dedicated to search and ranking, with a particular focus on topics such as web search, and retrieval models and ranking. GR aligns well with the theme of SIGIR-AP.

## 6 TUTORIAL OUTLINE

1. **Introduction** (15 minutes)
   - An overview of the tutorial
   - Why generative retrieval?
2. **Preliminaries** (15 minutes)
   - Retrieval task formulation: generative models vs. discriminative models
   - Basic concepts in generative retrieval
3. **Generative retrieval: Docid design** (30 minutes)
   - Pre-defined static docids
     - Single docids: number-based and word-based docids
     - Multiple docids
   - Learnable docids: jointly with retrieval tasks
4. **Generative retrieval: Training approaches** (40 minutes)
   - Static corpora: supervised learning with labeled data, and pre-training with unlabeled data
   - Dynamic corpora: continual learning
5. **Generative retrieval: Inference strategies** (25 minutes)
   - For a single docid: constrained beam search, constrained greedy search and FM-index
   - For multiple docids: heuristic scoring functions
6. **Generative retrieval: Applications** (35 minutes)
   - Offline application: e.g., entity retrieval, fact checking, recommender systems, multi-hop retrieval and code generation
   - Industry applications
7. **Conclusions and future directions** (20 minutes)

## 7 TUTORIAL MATERIALS

We plan to make all teaching materials available online for attendees, including: (i) Slides: The slides will be made publicly available. (ii) Annotated bibliography: This compilation will contain references listing all works discussed in the tutorial, serving as a valuable resource for further study. (iii) Code: We will provide an annotated list of pointers to open-source code bases and datasets related to

works discussed in the tutorial. Besides, we are open to the publication of the slides and video recordings in the ACM anthology.

## 8 ACKNOWLEDGEMENTS

This work was funded by the National Natural Science Foundation of China (NSFC) under Grants No. 62006218, the Lenovo-CAS Joint Lab Youth Scientist Project, and the project under Grants No. JCKY2022130C039. This work was also (partially) funded by the Hybrid Intelligence Center, a 10-year program funded by the Dutch Ministry of Education, Culture and Science through the Netherlands Organisation for Scientific Research, https://hybrid-intelligence-centre.nl, and project LESSEN with project number NWA.1389.20.183 of the research program NWA ORC 2020/21, which is (partly) financed by the Dutch Research Council (NWO). All content represents the opinion of the authors, which is not necessarily shared or endorsed by their respective employers and/or sponsors.

## REFERENCES

[1] Gabriel Bénédict, Ruqing Zhang, and Donald Metzler. 2023. Gen-IR@ SIGIR 2023: The First Workshop on Generative Information Retrieval. In *SIGIR*. 3460–3463.
[2] Michele Bevilacqua, Giuseppe Ottaviano, Patrick Lewis, Wen-tau Yih, Sebastian Riedel, and Fabio Petroni. 2022. Autoregressive Search Engines: Generating Substrings as Document Identifiers. In *NeurIPS*. 31668–31683.
[3] 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*.
[4] Jiangui Chen, Ruqing Zhang, Jiafeng Guo, Maarten de Rijke, Yiqun Liu, Yixing Fan, and Xueqi Cheng. 2023. A Unified Generative Retriever for Knowledge-Intensive Language Tasks via Prompt Learning. In *SIGIR*. 1448–1457.
[5] Jiangui Chen, Ruqing Zhang, Jiafeng Guo, Yixing Fan, and Xueqi Cheng. 2022. GERE: Generative Evidence Retrieval for Fact Verification. In *SIGIR*. 2184–2189.
[6] Jiangui Chen, Ruqing Zhang, Jiafeng Guo, Yiqun Liu, Yixing Fan, and Xueqi Cheng. 2022. CorpusBrain: Pre-train a Generative Retrieval Model for Knowledge-Intensive Language Tasks. In *CIKM*. 191–200.
[7] Ruey-Cheng Chen, Luke Gallagher, Roi Blanco, and J. Shane Culpepper. 2017. Efficient Cost-aware Cascade Ranking in Multi-stage Retrieval. In *SIGIR*.
[8] Xiaoyang Chen, Yanjiang Liu, Ben He, Le Sun, and Yingfei Sun. 2023. Understanding Differential Search Index for Text Retrieval. In *Findings of ACL*. 10701–10717.
[9] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, and Ellen M. Voorhees. 2020. Overview of the TREC 2019 Deep Learning Track. *arXiv preprint arXiv:2003.07820* (2020).
[10] Nicola De Cao, Gautier Izacard, Sebastian Riedel, and Fabio Petroni. 2021. Autoregressive Entity Retrieval. In *ICLR*.
[11] Susan Dumais, Michele Banko, Eric Brill, Jimmy Lin, and Andrew Ng. 2002. Web Question Answering: Is More Always Better?. In *SIGIR*. 291–298.
[12] Jianfeng Gao, Xiaodong He, and Jian-Yun Nie. 2010. Clickthrough-based Translation Models for Web Search: From Word Models to Phrase Models. In *CIKM*.
[13] Jiafeng Guo, Yixing Fan, Liang Pang, Liu Yang, Qingyao Ai, Hamed Zamani, Chen Wu, W Bruce Croft, and Xueqi Cheng. 2020. A Deep Look into Neural Ranking Models for Information Retrieval. *IPM* 57, 6 (2020), 102067.
[14] Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. 2020. Retrieval Augmented Language Model Pre-training. In *ICML*. 3929–3938.
[15] Po-Sen Huang, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry Heck. 2013. Learning Deep Structured Semantic Models for Web Search Using Clickthrough Data. In *CIKM*. 2333–2338.
[16] Tom Kenter and Maarten de Rijke. 2015. Short Text Similarity with Word Embeddings. In *CIKM*. 1411–1420.
[17] Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural Questions: A Benchmark for Question Answering Research. *Transactions of the Association for Computational Linguistics* 7 (2019), 452–466.
[18] Hyunji Lee, Jaeyoung Kim, Hoyeon Chang, Hanseok Oh, Sohee Yang, Vladimir Karpukhin, Yi Lu, and Minjoon Seo. 2023. Nonparametric Decoding for Generative Retrieval. In *Findings of the ACL 2023*. 12642–12661.
[19] Hyunji Lee, Sohee Yang, Hanseok Oh, and Minjoon Seo. 2022. Generative Multi-hop Retrieval. In *EMNLP*. 1417–1436.
[20] Yongqi Li, Nan Yang, Liang Wang, Furu Wei, and Wenjie Li. 2023. Learning to Rank in Generative Retrieval. *arXiv preprint arXiv:2306.15222* (2023).
[21] Yongqi Li, Nan Yang, Liang Wang, Furu Wei, and Wenjie Li. 2023. Multiview Identifiers Enhanced Generative Retrieval. In *ACL*. 6636–6648.
[22] Shichen Liu, Fei Xiao, Wenwu Ou, and Luo Si. 2017. Cascade Ranking for Operational E-commerce Search. In *KDD*. 1557–1565.
[23] Yu-An Liu, Ruqing Zhang, Jiafeng Guo, Wei Chen, and Xueqi Cheng. 2023. On the Robustness of Generative Retrieval Models: An Out-of-Distribution Perspective. In *Gen-IR@SIGIR*.
[24] Irina Matveeva, Chris Burges, Timo Burkard, Andy Laucius, and Leon Wong. 2006. High Accuracy Retrieval with Multiple Nested Ranker. In *SIGIR*. 437–444.
[25] 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* (2022).
[26] Donald Metzler, Yi Tay, Dara Bahri, and Marc Najork. 2021. Rethinking Search: Making Domain Experts Out of Dilettantes. *SIGIR Forum* 55, 1 (2021), 1–27.
[27] Bhaskar Mitra, Eric Nalisnick, Nick Craswell, and Rich Caruana. 2016. A Dual Embedding Space Model for Document Ranking. *arXiv preprint arXiv:1602.01137* (2016).
[28] Usama Nadeem, Noah Ziems, and Shaoen Wu. 2022. CodeDSI: Differentiable Code Search. *arXiv preprint arXiv:2210.00328* (2022).
[29] Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. MS MARCO: A Human Generated MAchine Reading COmprehension Dataset. In *Workshop on Cognitive Computation: Integrating Neural and Symbolic Approaches*.
[30] Thong Nguyen and Andrew Yates. 2023. Generative Retrieval as Dense Retrieval. In *Gen-IR@SIGIR*.
[31] 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?. In *Gen-IR@SIGIR*.
[32] Shashank Rajput, Nikhil Mehta, Anima Singh, Raghunandan H Keshavan, Trung Vu, Lukasz Heldt, Lichan Hong, Yi Tay, Vinh Q. Tran, Jonah Samost, et al. 2023. Recommender Systems with Generative Retrieval. *arXiv preprint arXiv:2305.05065* (2023).
[33] 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*.
[34] Daniel E. Rose and Danny Levinson. 2004. Understanding User Goals in Web Search. In *WWW*.
[35] Tetsuya Sakai, Daisuke Ishikawa, Noriko Kando, Yohei Seki, Kazuko Kuriyama, and Chin-Yew Lin. 2011. Using Graded-relevance Metrics for Evaluating Community QA Answer Selection. In *WSDM*. 187–196.
[36] Chirag Shah and Emily M Bender. 2022. Situating search. In *ACM SIGIR Conf. on Human Information Interaction and Retrieval*. 221–232.
[37] Weiwei Sun, Lingyong Yan, Zheng Chen, Shuaiqiang Wang, Haichao Zhu, Pengjie Ren, Zhumin Chen, Dawei Yin, Maarten de Rijke, and Zhaochun Ren. 2023. Learning to Tokenize for Generative Retrieval. *arXiv preprint arXiv:2304.04171* (2023).
[38] Yubao Tang, Ruqing Zhang, Jiafeng Guo, Jiangui Chen, Zuowei Zhu, Shuaiqiang Wang, Dawei Yin, and Xueqi Cheng. 2023. Semantic-Enhanced Differentiable Search Index Inspired by Learning Strategies. In *KDD*.
[39] Yi Tay, Vinh Q. Tran, Mostafa Dehghani, Jianmo Ni, Dara Bahri, Harsh Mehta, Zhen Qin, Kai Hui, Zhe Zhao, Jai Gupta, Tal Schuster, William W. Cohen, and Donald Metzler. 2022. Transformer Memory as a Differentiable Search Index. In *NeurIPS*, Vol. 35. 21831–21843.
[40] James Thorne. 2022. Data-efficient Autoregressive Document Retrieval for Fact Verification. In *Workshop on SENLP*.
[41] Yujing Wang, Yingyan Hou, Haonan Wang, Ziming Miao, Shibin Wu, Hao Sun, Qi Chen, Yuqing Xia, Chengmin Chi, Guoshuai Zhao, Zheng Liu, Xing Xie, Hao Sun, Weiwei Deng, Qi Zhang, and Mao Yang. 2022. A Neural Corpus Indexer for Document Retrieval. In *NeurIPS*, Vol. 35. 25600–25614.
[42] Zihan Wang, Yujia Zhou, Yiteng Tu, and Zhicheng Dou. 2023. NOVO: Learnable and Interpretable Document Identifiers for Model-Based IR. In *CIKM*.
[43] Caiming Xiong, Victor Zhong, and Richard Socher. 2017. DCN+: Mixed Objective And Deep Residual Coattention for Question Answering. In *ICLR*.
[44] Soyoung Yoon, Chaeeun Kim, Hyunji Lee, Joel Jang, and Minjoon Seo. 2023. Continually Updating Generative Retrieval on Dynamic Corpora. *arXiv preprint arXiv:2305.18952* (2023).
[45] Peitian Zhang, Zheng Liu, Yujia Zhou, Zhicheng Dou, and Zhao Cao. 2023. Term-Sets Can Be Strong Document Identifiers For Auto-Regressive Search Engines. *arXiv preprint arXiv:2305.13859* (2023).
[46] Yujia Zhou, Jing Yao, Zhicheng Dou, Ledell Wu, Peitian Zhang, and Ji-Rong Wen. 2022. Ultron: An Ultimate Retriever on Corpus with a Model-based Indexer. *arXiv preprint arXiv:2208.09257* (2022).
[47] Yu-Jia Zhou, Jing Yao, Zhi-Cheng Dou, Ledell Wu, and Ji-Rong Wen. 2023. DynamicRetriever: A Pre-trained Model-based IR System Without an Explicit Index. *Machine Intelligence Research* 20, 2 (2023), 276–288.
[48] Shengyao Zhuang, Houxing Ren, Linjun Shou, Jian Pei, Ming Gong, Guido Zuccon, and Daxin Jiang. 2023. Bridging the Gap between Indexing and Retrieval for Differentiable Search Index with Query Generation. In *Gen-IR@SIGIR*.