From Matching to Generation: A Survey on Generative Information Retrieval

Xiaoxi Li, Jiajie Jin, Yujia Zhou, Yuyao Zhang, Peitian Zhang, Yutao Zhu, and Zhicheng Dou

Abstract—Information Retrieval (IR) systems are crucial tools for users to access information, widely applied in scenarios like search engines, question answering, and recommendation systems. Traditional IR methods, based on similarity matching to return ranked lists of documents, have been reliable means of information acquisition, dominating the IR field for years. With the advancement of pre-trained language models, generative information retrieval (GenIR) has emerged as a novel paradigm, gaining increasing attention in recent years. Currently, research in GenIR can be categorized into two aspects: generative document retrieval (GR) and reliable response generation. GR leverages the generative model's parameters for memorizing documents, enabling retrieval by directly generating relevant document identifiers without explicit indexing. Reliable response generation, on the other hand, employs language models to directly generate the information users seek, breaking the limitations of traditional IR in terms of document granularity and relevance matching, offering more flexibility, efficiency, and creativity, thus better meeting practical needs. This paper aims to systematically review the latest research progress in GenIR. We will summarize the advancements in GR regarding model training, document identifier, incremental learning, downstream tasks adaptation, multi-modal GR and generative recommendation, as well as progress in reliable response generation in aspects of internal knowledge memorization, external knowledge augmentation, generating response with citations and personal information assistant. We also review the evaluation, challenges and future prospects in GenIR systems. This review aims to offer a comprehensive reference for researchers in the GenIR field, encouraging further development in this area.

Index Terms—Generative Information Retrieval; Generative Document Retrieval; Reliable Response Generation

1 Introduction

In today's digital landscape, information retrieval (IR) systems are crucial for navigating the vast sea of online information. From using search engines such as Google [1], Bing [2], and Baidu [3], to engaging with question-answering or dialogue systems like ChatGPT [3] and Bing Chat [4], and discovering content via recommendation platforms like Amazon [5] and YouTube [6], IR technologies are integral to our everyday online experiences. These systems are not only reliable but also play a key role in spreading knowledge and ideas globally.

Traditional IR systems primarily rely on sparse retrieval methods based on word-level matching. These methods, which include Boolean Retrieval [7], BM25 [8], SPLADE [9], and UniCOIL [10], establish connections between vocabulary and documents, offering high retrieval efficiency and robust system performance. With the rise of deep learning, dense retrieval methods such as DPR [11] and ANCE [12], based on the bidirectional encoding representations from the BERT model [13], capture the deep semantic information of documents, significantly improving retrieval precision. Although these methods have achieved leaps in accuracy, they rely on large-scale document indices [14, 15] and cannot be optimized in an end-to-end way. Moreover, when people search for information, what they really need is a precise and reliable answer. This document ranking list-based IR approach still requires users to spend time summarizing their required answers, which is not ideal enough for information seeking [16].

With the development of Transformer-based pre-trained language models such as T5 [17], BART [18], and GPT [19], they have demonstrated their strong text generation capabilities. In recent years, large language models (LLMs) have brought about revolutionary changes in the field of AI-generated content (AIGC) [20, 21]. Based on large pre-training corpora and advanced training techniques like RLHF [22], LLMs [3, 23–25] have made significant progress in natural language tasks, such as dialogue [3, 26] and question answering [27, 28]. The rapid development of LLMs is transforming IR systems, giving rise to a new paradigm of generative information retrieval (GenIR), which achieves IR goals through generative approaches.

As envisioned by Metzler et al. [16], in order to build an IR system that can respond like a domain expert, the system should not only provide accurate responses but also include source citations to improve credibility and transparency of the results. To achieve this, GenIR models must possess both sufficient memorized knowledge and the ability to recall the associations between knowledge and corresponding documents. Current research in GenIR is primarily focused on two main patterns: (1) generative document retrieval (GR), which involves retrieving documents by generating their identifiers; and (2) reliable response generation, which entails directly generating user-centric responses through strategies that enhance their reliability. These two patterns are the central topics of this survey.

Generative document retrieval, a new retrieval paradigm based on generative models, is garnering increasing attention. This approach leverages the parametric memory of

Github Repository: https://github.com/RUC-NLPIR/GenIR-Survey

Zhicheng Dou is the corresponding author. All authors are from Gaoling School of Artificial Intelligence, Renmin University of China.
Contact E-mail: xiaoxi_li@ruc.edu.cn, dou@ruc.edu.cn.

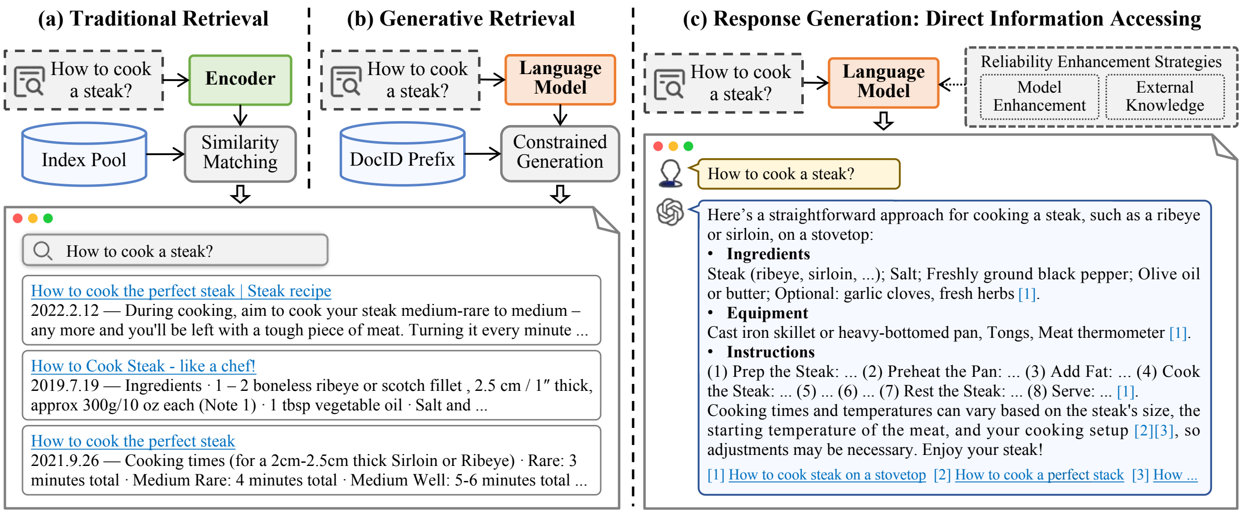

Fig. 1. Exploring IR Evolution: From Traditional to Generative Methods - This diagram illustrates the shift from traditional similarity-based document matching (a) to GenIR techniques. Current GenIR methods can be categorized into two types: generative retrieval (b), which retrieves documents by directly generating relevant DocIDs constrained by a DocID prefix tree; and response generation (c), which directly generates reliable and user-centric answers.

generative models to directly generate document identifiers (DocIDs) related to the documents [29–32]. Figure 1 illustrates this transition, where traditional IR systems match queries to documents based on indexed database (Figure 1(a)), while generative methods use language models to retrieve by directly generating relevant document identifiers (Figure 1(b)). Specifically, GR assigns a unique identifier to each document, which can be numeric-based or text-based, and then trains a generative retrieval model to learn the mapping from queries to the relevant document DocIDs. This allows the model to index documents using its internal parameters. During inference, GR models use constrained beam search to limit the generated DocIDs to be valid within the corpus, ranking them based on generation probability to produce a ranked list of DocIDs. This eliminates the need for large-scale document indexes in traditional retrieval methods, enabling end-to-end training of the retrieval model.

Recent studies on generative retrieval have delved into model training and structure [30, 31, 33, 34], document identifier design [29, 30, 35, 36], continual learning on dynamic corpora [37–39], downstream task adaptation [40–42], multi-modal generative retrieval [43–45], and generative recommender systems [46–48]. The progress in GR is shifting retrieval systems from matching to generation. It has also led to the emergence of workshops [49], progress and challenges discussions [50], and tutorials [51]. However, there is currently no comprehensive review that systematically organizes the research, challenges, and prospects of this emerging field.

Reliable response generation is also a promising direction in the IR field, offering user-centric and accurate answers that directly meet users' needs. Since LLMs are particularly adept at following instructions [20], capable of generating customized responses, and can even cite the knowledge sources [52, 53], making direct response generation a new and intuitive way to access information [54–56]. As illustrated in Figure 1, the generative approach marks a significant shift from traditional IR systems, which return a ranked list of documents (as shown in Figure 1(a,b)). Instead, response generation methods (depicted in Figure 1(c)) offer a more dynamic form of information access by directly generating detailed, user-centric responses, thereby providing a richer and more immediate understanding of the information need behind the users' queries.

However, the responses generated by language models may not always be reliable. They have the potential to generate irrelevant answers [57], contradict factual information [58, 59], provide outdated data [60], or generate toxic content [61, 62]. Consequently, these limitations render them unsuitable for many scenarios that require accurate and up-to-date information. To address these challenges, the academic community has developed strategies across four key aspects: enhancing internal knowledge [63–71]; augmenting external knowledge [52, 72–78]; generating responses with citation [52, 79–82]; and improving personal information assistance [83–86]. Despite these efforts, there is still a lack of a systematic review that organizes the existing research under this new paradigm of generative information access.

This review will systematically review the latest research progress and future developments in the field of GenIR, as shown in Figure 2, which displays the classification of research related to the GenIR system. We will introduce background knowledge in Section 2, generative document retrieval technologies in Section 3, direct information accessing with generative language models in Section 4, evaluation in Section 5, current challenges and future directions in Section 6, respectively. Section 7 will summarize the content of this review. This article is the first to systematically organize the research, evaluation, challenges and prospects of generative IR, while also looking forward to the potential and importance of GenIR's future development. Through this review, readers will gain a deep understanding of the latest progress in developing GenIR systems and how it shapes the future of information access.

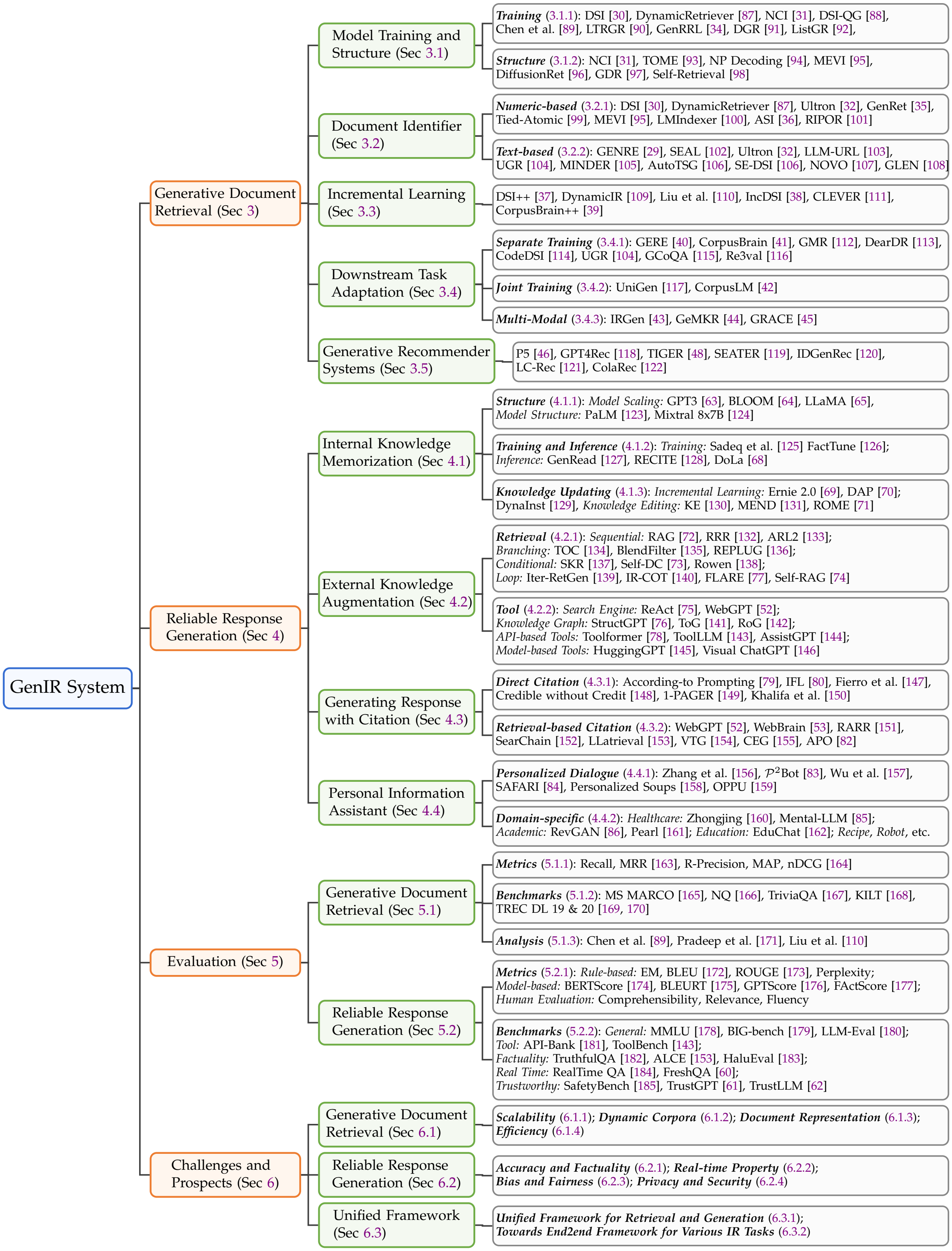

Fig. 2. Taxonomy of research on generative information retrieval: investigating generative document retrieval, reliable response generation, evaluation, challenges and prospects.

2 Background and Preliminaries

Information retrieval techniques aim at efficiently obtaining, processing, and understanding information from massive data. Technological advancements have driven the evolution of retrieval methods, from traditional keyword-based sparse retrieval, to deep learning-based dense retrieval, and further to recently emerged generative retrieval and large language models. Each advancement enhances retrieval accuracy and efficiency, catering to the complex and diverse query needs of users.

2.1 Traditional Information Retrieval

Sparse Retrieval. In the field of traditional information retrieval, sparse retrieval techniques implement fast and accurate document retrieval through the inverted index method. Inverted indexing technology maps each unique term to a list of all documents containing that term, providing an efficient means for information retrieval in large document collections. Among these methods, TF-IDF (Term Frequency-Inverse Document Frequency) [186] is a particularly important statistical tool used to assess the importance of a word in a document collection, thereby widely applied in various traditional retrieval systems.

The core of sparse retrieval technology lies in evaluating the relevance between documents and user queries. Specifically, given a document collection $\mathcal{D}$ and a user query q, traditional information retrieval systems identify and retrieve information by calculating the relevance $\mathcal{R}$ between document d and query q. This relevance evaluation typically relies on the similarity measure between document d and query q, as shown below:

$$\mathcal{R}(q, d) = \sum_{t \in q \cap d} \text{tf-idf}(t, d) \cdot \text{tf-idf}(t, q), \tag{1}$$

where t represents the terms common to both query q and document d, and tf-idf(t, d) and tf-idf(t, q) represent the TF-IDF weights of term t in document d and query q, respectively. Although sparse retrieval methods like TF-IDF [186] and BM25 [8] excel at fast retrieval, it struggles with complex queries involving synonyms, specialized terms, or context, as term matching and TF-IDF may not fully meet users' information needs [187].

Dense Retrieval. The advent of pre-trained language models like BERT [13] has revolutionized information retrieval, leading to the development of dense retrieval methods, like DPR [11], ANCE [12], E5 [188], SimLM [189]. Unlike traditional sparse retrieval, these methods leverage Transformer-based encoders to create dense vector representations for both queries and documents. This approach enhances the capability to grasp the underlying semantics, thereby improving retrieval accuracy.

The core of dense retrieval lies in converting documents and queries into vector representations. Given document d and query q and their vector representations $\mathbf{v}_q$, each document d is transformed into a dense vector $\mathbf{v}_d$ through a pre-trained language model, similarly, query q is transformed into vector $\mathbf{v}_q$. Specifically, we can use encoder functions $E_d(\cdot)$ and $E_q(\cdot)$ to represent the encoding process for documents and queries, respectively:

$$\mathbf{v}_d = E_d(d), \quad \mathbf{v}_q = E_q(q), \tag{2}$$

where $E_d(\cdot)$ and $E_q(\cdot)$ can be the same pre-trained model or different models optimized for specific tasks.

Dense retrieval methods evaluate relevance by calculating the similarity between the query vector and document vector, which can be measured by cosine similarity, expressed as follows:

$$\mathcal{R}(q, d) = \cos(\mathbf{v}_q, \mathbf{v}_d) = \frac{\mathbf{v}_q \cdot \mathbf{v}_d}{|\mathbf{v}_q||\mathbf{v}_d|}, \tag{3}$$

where $\mathbf{v}_q \cdot \mathbf{v}_d$ represents the dot product of query vector $\mathbf{v}_q$ and document vector $\mathbf{v}_d$, and $|\mathbf{v}_q|$ and $|\mathbf{v}_d|$ respectively represent the magnitudes of the query vector and document vector. Finally, documents are ranked based on these similarity scores to identify the most relevant ones for the user.

2.2 Generative Retrieval

With significant progress of language models, generative retrieval [16, 30] has emerged as a new direction in the field of information retrieval. Unlike traditional index-based retrieval methods, generative retrieval relies on pre-trained generative models, such as the T5 [17] and BART [18], to directly generate document identifiers (DocIDs) related to the query, thereby achieving end-to-end retrieval without relying on large-scale pre-built document indices.

Formally, given a query q, a generative retrieval model aims to directly generate the document identifier d' related to q. The model can be represented as a conditional probability distribution, where the probability of generating document identifier d' given query q can be expressed as:

$$\mathcal{R}(q, d) = P(d'|q;\theta) = \prod_{i=1}^{T} P(d'_i|d'_{<i}, q; \theta), \tag{4}$$

where T is the length of the generated document identifier d', d'_i represents the token at position i, $d'_{<i}$ represents the sequence of tokens generated before position i, and θ represents the model parameters.

Generative retrieval models demonstrate flexibility and deep semantic understanding capabilities. Compared to traditional methods, generative retrieval achieves end-to-end optimization of the retrieval process by directly generating DocIDs. Additionally, it reduces reliance on external indexing, lowering the system's demand for storage resources. However, its performance and efficiency are limited by model capacity and computational resources, indicating that future research might focus on finding more efficient model architectures and training methods to better balance performance, resource consumption, and response speed.

2.3 Large Language Models

The evolution of Large Language Models (LLMs) marks a significant leap in natural language processing (NLP), rooted from early statistical and neural network-based language models [190]. These models, through pre-training on vast text corpora, learned deep semantic features of language, greatly enriching the understanding of text. Subsequently, generative language models, most notably the GPT series [19, 63, 191], significantly improved text generation and understanding capabilities with improved model size and number of parameters.

LLMs can be mainly divided into two categories: encoder-decoder models and decoder-only models. Encoder-decoder models, like T5 [17] and BART [18], convert input text into vector representations through their encoder, then the decoder generates output text based on these representations. This model perspective treats various NLP tasks as text-to-text conversion problems, solving them through text generation. On the other hand, decoder-only models, like the GPT [19] and GPT-2 [191], rely entirely

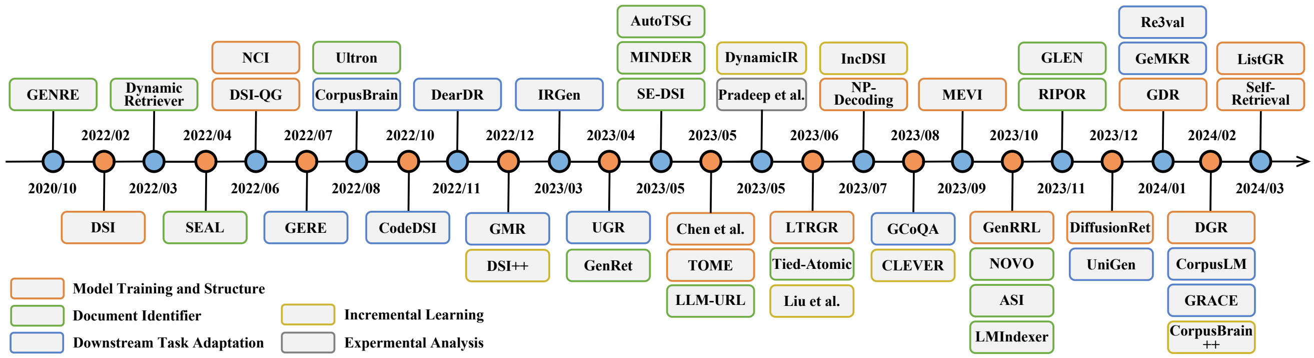

Fig. 3. Timeline of research in generative retrieval: focus on model training and structure, document identifier design, incremental learning and downstream task adaptation.

on the Transformer decoder, generating text step by step through the self-attention mechanism. The introduction of GPT-3 [63], with its 175 billion parameters, marked a significant milestone in this field and led to the creation of models like InstructGPT [192], Falcon [193], PaLM [123] and LLaMA [65]. These models, all using a decoder-only architecture, trained on large-scale datasets, have shown astonishing language processing capabilities [20].

In the field of information retrieval, LLMs utilize two main strategies: In-Context Learning (ICL) and parameter-efficient fine-tuning [190]. ICL leverages the knowledge pre-trained in the model to understand and handle new tasks, requiring only a series of relevant examples and task descriptions to be provided to the model, enabling it to address specific information retrieval tasks without additional training [63]. This method's flexibility and cost-effectiveness are extremely high, allowing the model to quickly adapt to various scenarios. On the other hand, parameter-efficient fine-tuning, such as LoRA [194], PEFT [195] and QLoRA [196] technologies, adjusts a very small part of the model's parameters to adapt to specific tasks, avoiding the need to retrain the entire model. These techniques, by fine-tuning key parts of the model, significantly reduce reliance on computational resources while maintaining performance.

Incorporating these strategies, LLMs have demonstrated exceptional potential across various information retrieval tasks, including but not limited to query rewriting, document retrieval, result re-ranking, document reading comprehension, and agent-based search mechanisms [190, 197]. The application of LLMs extends beyond enhancing the relevance of search outcomes; they are instrumental in directly generating the precise information sought by users. This capability marks a significant step towards a new era of generative information retrieval. In this era, the retrieval process is not solely about locating existing information but also about creating new content that meets the specific needs of users. This feature is especially advantageous in situations where users might not know how to phrase their queries or when they are in search of complex and highly personalized information, scenarios where traditional matching-based methods fall short.

3 Generative Document Retrieval: From Similarity-based Matching to Generating Document Identifiers

In recent advancements in AIGC, generative retrieval (GR) has emerged as an promising approach in the field of information retrieval, garnering increasing interest from the academic community. Figure 3 showcases a timeline of the GR methods. Initially, GENRE [29] proposed to retrieve entities by generating their unique names through constrained beam search via a pre-built entity prefix tree, achieving advanced entity retrieval performance. Subsequently, Metzler et al. [16] envisioned a model-based information retrieval framework aiming to combine the strengths of traditional document retrieval systems and pre-trained language models to create systems capable of providing expert-quality answers in various domains.

Following their lead, a diverse range of methods including DSI [30], DynamicRetriever [87], SEAL [102], NCI [31], etc., have been developed, with a continuously growing body of work. These methods explore various aspects such as model training and architectures, document identifiers, incremental learning, task-specific adaptation, and generative recommendations. Figure 4 presents an overview of the GR system and we'll provide an in-depth discussion of each associated challenge in the following sections.

3.1 Model Training and Structure

One of the core components of GR is the model training and structure, aiming to enhance the model's ability to memorize documents in the corpus.

3.1.1 *Model Training*

To effectively train generative models for indexing documents, the standard approach is to train the mapping from queries to relevant DocIDs, based on standard sequence-to-sequence (seq2seq) training methods, as described in Equation (2). This method has been widely used in numerous GR research works, such as DSI [30], NCI [31], SEAL [102], etc. Moreover, a series of works have proposed various model training methods tailored for GR tasks to further enhance GR retrieval performance, such as sampling documents or generating queries based on document content to serve as pseudo queries for data augmentation; or including training objectives for document ranking.

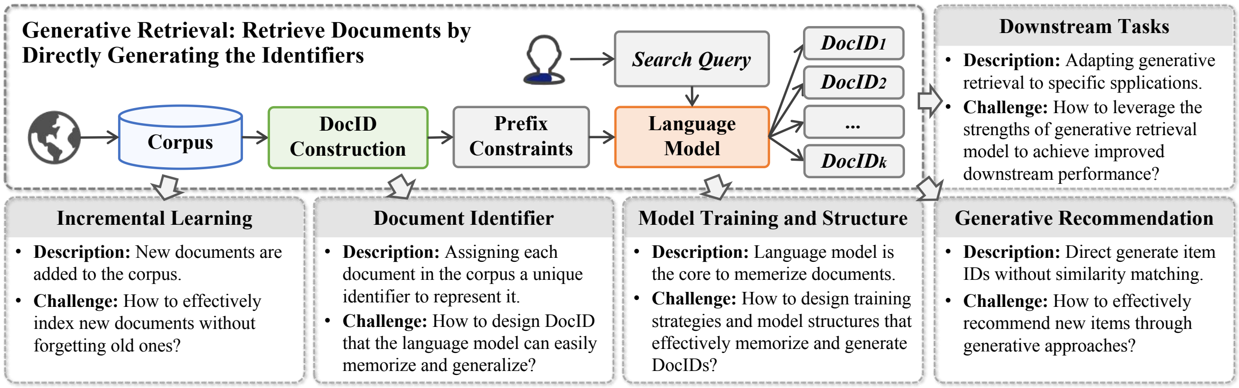

Fig. 4. A conceptual framework for a generative retrieval system, with a focus on challenges in incremental learning, identifier construction, model training and structure, and integration with downstream tasks and recommendation systems.

Specifically, DSI [30] proposed two training strategies: one is "indexing", that is, training the model to associate document tokens with their corresponding DocIDs, where DocIDs are pre-built based on documents in corpus, which will be discussed in detail in Section 3.2; the other is "retrieval", using labeled query-DocID pairs to fine-tune the model. Notably, DSI was the first to realize a differentiable search index based on the Transformer [198] structure, showing good performance in web search [165] and question answering [166] scenarios.

In the same era, DynamicRetriever [87], also based on the encoder-decoder model, constructed a model-based IR system by initializing the encoder with a pre-trained BERT [13]. Besides, DynamicRetriever utilizes passages, sampled terms and N-grams to serve as pseudo queries to enhance the model's memorization of DocIDs. Formally, the training methods can be summarized as follows:

$$\text{Sampled Document}: d_{s_i} \longrightarrow \text{DocID}, i \in \{1, ..., k_{d_s}\}, \quad (5)$$
$$\text{Labeled Query}: q_i \longrightarrow \text{DocID}, i \in \{1, ..., k_q\}, \quad (6)$$

where d_{s_i} and q_i denote each of the k_{d_s} sampled document text and each of the k_q labeled query for the corresponding DocID, respectively.

Following DSI, the NCI [31] model was trained using a combination of labeled query-document pairs and augmented pseudo query-document pairs. Specifically, NCI proposed two strategies: one using the DocT5Query [199] model as a query generator, generating pseudo queries for each document in the corpus through beam search; the other strategy directly uses the document as a query, as stated in Equation (5). Similarly, DSI-QG [88] also proposed using a query generator to enhance training data, establishing a bridge between indexing and retrieval in DSI. This data augmentation method has been proven in subsequent works to be an effective method to enhance the model's memorization for DocIDs, which can be expressed as follows:

$$\text{Pseudo Query}: q_{s_i} \longrightarrow \text{DocID}, i \in \{1, ..., k_{q_s}\}, \quad (7)$$

where q_{s_i} represents each of the k_{q_s} generated pseudo query for the corresponding DocID.

Additionally, a series of methods focus on further optimizing the ranking capability of GR models. Chen et al. [200] aimed at understanding the DSI model and its retrieval effectiveness, noting that the DSI model has flaws in exclusivity, completeness, and relevance ordering, proposing a multi-task distillation method to improve retrieval quality without changing the model structure, thereby obtaining better indexing and ranking capabilities. Meanwhile, LTRGR [90] proposed to directly learn how to rank paragraphs with GR models, not just generating paragraph identifiers. This framework is first trained with a sequence-to-sequence loss, then further trained with a ranking loss, allowing the model to learn ranking at the document level.

Subsequently, [34] pointed out the two main challenges in current GR methods: (i) the gap between token-level probability optimization and document-level relevance estimation; (ii) the overemphasis on top-1 results while neglecting the quality of the overall ranking list. To solve these problems, [34] proposed GenRRL, which improves the overall ranking quality through reinforcement learning with relevance feedback, aligning token-level DocID generation with document-level relevance estimation after seq2seq supervised fine-tuning.

Moreover, [91] introduced DGR, which enhances GR through knowledge distillation. Specifically, DGR uses a cross-encoder as a teacher model, providing fine-grained passage ranking supervision signals, and then optimizes the GR model with a distilled RankNet loss, effectively transferring the capabilities of advanced ranking models to the GR model. ListGR [92] also focuses on enhancing the ranking ability of GR models, by defining positional conditional probabilities, emphasizing the importance of the generation order of each DocID in the list. In addition, ListGR employs relevance calibration that adjusts the generated list of DocIDs to better align with the labeled ranking list. See Table 1 for a detailed comparison of GR methods.

3.1.2 Model Structure

Basic generative retrieval models mostly use pre-trained encoder-decoder structured generative models, such as T5 [17] and BART [18], fine-tuned for the DocID generation task. To better adapt to the GR task, researchers have proposed a series of specifically designed model structures [31, 93–98].

For the semantic structured DocID proposed by DSI [30], NCI [31] designed a Prefix-Aware Weight-Adaptive (PAWA)

TABLE 1. Comparisons of representative generative retrieval methods, focusing on document identifier, training data augmentation, and training objective.

Model	Document Identifier			Training Data Augmentation		Training Objective		
	State	Data Type	Order	Sample Doc	Doc2Query	Seq2seq	DocID	Ranking
GENRE [29]	Static	Text	Sequence	-	-	✓	-	-
DSI [30]	Static	Numeric	Sequence	✓	-	✓	-	-
DynamicRetriever [87]	Static	Numeric	Sequence	✓	-	✓	-	-
SEAL [102]	Static	Text	Sequence	✓	-	✓	-	-
DSI-QG [88]	Static	Numeric	Sequence	-	✓	✓	-	-
NCI [31]	Static	Numeric	Sequence	✓	✓	✓	-	-
Ultron [32]	Static	Numeric/Text	Sequence	✓	✓	✓	-	-
CorpusBrain [41]	Static	Text	Sequence	✓	-	✓	-	-
GenRet [35]	Learnable	Numeric	Sequence	-	✓	✓	✓	-
AutoTSG [106]	Static	Text	Set	-	✓	✓	-	-
SE-DSI [201]	Static	Text	Sequence	✓	-	✓	-	-
Chen et al. [200]	Static	Numeric	Sequence	✓	✓	✓	-	✓
LLM-URL [103]	Static	Text	Sequence	-	-	-	-	-
MINDER [105]	Static	Text	Sequence	-	✓	✓	-	-
LTRGR [90]	Static	Text	Sequence	-	✓	✓	-	✓
NOVO [107]	Learnable	Text	Set	✓	-	-	✓	-
GenRRL [34]	Static	Text	Sequence	-	✓	✓	-	✓
LMIndexer [100]	Learnable	Numeric	Sequence	-	✓	✓	✓	-
ASI [36]	Learnable	Numeric	Sequence	-	✓	✓	✓	-
RIPOR [101]	Learnable	Numeric	Sequence	-	✓	✓	✓	✓
GLEN [108]	Learnable	Text	Sequence	-	✓	✓	✓	✓
DGR [91]	Static	Text	Sequence	-	✓	✓	-	✓
ListGR [92]	Static	Numeric	Sequence	-	✓	✓	-	✓

decoder structure. By perceiving the hierarchical prefix of DocID and adjusting the weights at different positions, this decoder can capture the semantic hierarchy of DocIDs. Later, TOME [93] proposed a method that decomposes the GR task into two stages, first generating text paragraphs related to the query through an additional model, then using the GR model to generate the URL related to that paragraph, thereby improving the accuracy of model retrieval.

To overcome the capacity limitations of GR models, NP-Decoding [94] proposed a decoding method that uses non-parametric contextualized word embeddings (as external memory) instead of traditional word embeddings as the input to the decoder. This allows the GR model not only to utilize its own parameter space but also to access and use information in contextualized word embeddings.

Combining the strengths of seq2seq generative models and dual-encoder retrieval models, MEVI [95] leverages Residual Quantization (RQ) [202] to organize documents into hierarchical clusters, treating these clusters as the generative model's output targets. This approach enables MEVI to efficiently retrieve candidate clusters via the generative model, followed by precise document retrieval within clusters using Approximate Nearest Neighbor (ANN) search methods. This not only boosts recall but also ensures the efficiency of low-latency services.

Similarly, Generative Dense Retrieval (GDR) [97] integrates generative and dense retrieval's benefits in a two-phase process. Initially, it uses generative retrieval to match queries with document clusters broadly, optimizing for interaction depth and memory efficiency. Then, it applies dense retrieval for accurate, cluster-specific document searches, enhancing both recall and scalability. This synergy provides an efficient solution for document retrieval by combining broad matching capabilities with precise searching within clusters.

Moerover, DiffusionRet [96] proposed a two-stage enhanced generative retrieval structure based on diffusion models. First, a sequence-to-sequence diffusion model (SeqDiffuSeq [203]) is used to generate a pseudo-document from a query, a process similar to semantic filling of text, where the generated pseudo-document is similar to real documents in length, format, and content, rich in semantic information; secondly, another sequence-to-sequence model is used for generative retrieval based on N-gram, similar to SEAL [102] during inference, generating sets of N-grams occurring in the corpus based on the FM-Index [204].

Self-Retrieval [98] proposed an architecture that fully integrates indexing, retrieval, and evaluation into a single large language model. By successively generating natural language indices and document segments, and performing self-evaluation to score and rank the generated documents.

3.2 Document Identifiers

Another essential component of generative retrieval is document representation, also known as document identifiers (DocIDs), which act as the target outputs for the GR model. Accurate document representations are crucial as they enable the model to more effectively memorize document information, leading to enhanced retrieval performance. Table 1 provides a detailed comparison of the states, data types, and order of DocIDs across numerous GR methods.

In the following sections, we will explore the design of DocIDs from two categories: numeric-based and text-based.

3.2.1 Numeric-based Identifiers

Initially, DSI [30] introduced three numeric DocIDs to represent documents: (1) Unstructured Atomic DocID: a unique integer identifier is randomly assigned to each document, containing no structure or semantic information. (2) Naively Structured String DocID: treating random integers as divisible strings, implementing character-level DocID decoding to replace large softmax output layers. (3) Semantically Structured DocID: introducing semantic structure through hierarchical k-means method, allowing semantically similar documents to share prefixes in their identifiers, effectively reducing the search space. Concurrently, DynamicRetriever [87] also built a model-based IR system based on unstructured atomic DocID.

Subsequently, Ultron [32] encoded documents into a latent semantic space using BERT [13], and compressed vectors into a smaller semantic space via Product Quantization (PQ) [205, 206], preserving semantic information. Each document's PQ code serves as its semantic identifier, with additional digits added to ensure the uniqueness of each DocID if duplicates exist.

Unlike previous static DocIDs, GenRet [35] proposed learnable document representations, encoding documents into short discrete DocIDs using a discrete autoencoder. GenRet transforms documents into DocIDs through an encoder, then reconstructs documents from DocIDs using a decoder, trained to minimize reconstruction error. Furthermore, GenRet optimized the model through progressive training and diversity clustering.

To ensure that DocID embeddings can reflect document content, Tied-Atomic [99] and MEVI [95] proposed to enhance DocID embeddings to mirror document content accurately. Tied-Atomic links document text with token embeddings and employs contrastive loss for DocID creation, merging generative and dense retrieval's expressiveness and efficiency. MEVI clusters documents using Residual Quantization (RQ) [202], forming semantically similar clusters for efficient search. It utilizes dual-tower and seq2seq model embeddings for a balanced performance in large-scale document retrieval.

LMIndexer [100] and ASI [36] learned optimal DocIDs through semantic indexing, with LMIndexer using a reparameterization mechanism for unified optimization, facilitating efficient retrieval by aligning semantically similar documents under common DocIDs. ASI extends this by establishing an end-to-end retrieval framework, incorporating semantic loss functions and reparameterization to enable joint training. This approach ensures that documents with similar content are assigned similar DocIDs, enhancing retrieval accuracy and efficiency.

RIPOR [101] treats the GR model as a dense encoder to generate document representations by encoding document content and using a start token for decoding. It then splits these representations into vectors via RQ [202], creating unique DocID sequences. Furthermore, RIPOR implements a prefix-guided ranking optimization, increasing relevance scores for prefixes of pertinent DocIDs through margin decomposed pairwise loss during decoding. This ensures higher scores for relevant DocID prefixes in beam search steps, enhancing search precision.

In summary, numeric-based document representations can utilize the embeddings of dense retrievers, obtaining semantically meaningful DocID sequences through methods such as k-means, PQ [205], and RQ [202]; they can also combine encoder-decoder GR models with bi-encoder DR models to achieve complementary advantages [95, 99].

3.2.2 Text-based Identifiers

Text-based DocIDs have the inherent advantage of effectively leveraging the strong capabilities of pre-trained language models and offering better interpretability.

The most straightforward text-based identifier is the document title, which requires each title to uniquely represent a document in the corpus, otherwise, it would not be possible to accurately retrieve a specific document. The Wikipedia corpus used in the KILT [168] benchmark, due to its well-regulated manual annotation, has a unique title corresponding to each document. Thus, GENRE [29], based on the title as DocID and leveraging the generative model BART [18] and pre-built DocID prefix, achieved superior retrieval performance across 11 datasets in KILT. Following GENRE, GERE [40], CorpusBrain [41], Re3val [116], and CorpusBrain++ [39] also based their work on title DocIDs for Wikipedia-based tasks. Notably, LLM-URL [103] directly generated URLs using ChatGPT prompts, achieving commendable performance after removing invalid URLs.

However, in the web search scenario [165], document titles in the corpus often have significant duplication and many meaningless titles, making it unfeasible to use titles alone as DocIDs. Thus, Ultron [32] effectively addressed this issue by combining URLs and titles as DocIDs, uniquely identifying documents through keywords in web page URLs and titles.

To increase the flexibility of DocIDs, SEAL [102] proposed a sub-string identifier, representing documents with any N-grams within them. Using FM-Index (a compressed full-text sub-string index) [204], SEAL could generate N-grams present in the corpus to retrieve all documents containing those N-grams, scoring and ranking documents based on the frequency of N-grams in each document and the importance of N-grams in the query. Following SEAL, various GR models [90, 91, 104, 105] also utilized sub-string DocIDs and FM-Index during inference.

For a more comprehensive representation of documents, MINDER [105] proposed multi-view identifiers, including generated pseudo queries from document content via DocT5Query [199], titles, and sub-strings. However, compared to single DocID methods, multi-DocID GR methods required more memory use and inference time. This multi-DocID approach was also used in LTRGR [90] and DGR [91]. Concurrently, SE-DSI [201] also demonstrated the effectiveness of using pseudo queries alone as DocIDs.

Unlike the sequential DocIDs described earlier, AutoTSG [106] proposed a term set-based document representation, using keywords extracted from titles and content, rather than predefined sequences, allowing for retrieval of the target document as long as the generated term set is included in the document's identifiers. During inference, AutoTSG designed a constrained greedy search to ensure each term generated stepwise was an effective identifier belonging to some document.

Text-based identifiers can also be learnable. Similarly based on term-sets, NOVO [107] proposed learnable continuous N-grams constituting term-set DocIDs. Through the Denoising Query Modeling (DQM) task, the model learned to generate queries from documents with noise, thereby implicitly learning to filter out document N-grams more relevant to queries. NOVO also utilized the retrieval task to update the semantic representations of document identifiers, i.e., improving the document's semantic representation by updating N-gram embeddings.

Later, GLEN [108] designs dynamic lexical DocIDs and is trained through a two-phase index learning strategy. Firstly, the keyword-based DocID assignment phase, GLEN defines DocIDs by extracting keywords from documents using self-supervised signals and learns them. Secondly, the ranking-based DocID refinement phase, GLEN learns dynamic DocIDs by directly integrating query-document relevance through two loss functions. During inference, GLEN utilizes collision-free inference, ranking documents using DocID weights without incurring additional overhead.

3.3 Incremental Learning on Dynamic Corpora

Prior studies have focused on generative retrieval from static document corpora. However, in reality, the documents available for retrieval are continuously updated and expanded. To address this challenge, researchers have developed a range of methods to optimize GR models for adapting to dynamic corpora.

At first, DSI++ [37] aims to address the incremental learning challenges within dynamic document corpora encountered by DSI [30]. DSI++ modifies the training by optimizing flat loss basins through the Sharpness-Aware Minimization (SAM) optimizer, thereby stabilizing the learning process of the model. It also introduces generative memory by employing DocT5Query [199] to generate pseudo queries for documents in the existing corpus as training data augmentation, mitigating the forgetting issue of GR models.

DynamicIR [109] proposes a task framework based on StreamingQA [207] benchmark for evaluating IR models within dynamically updated corpora. Through experimental analysis, DynamicIR revealed that GR systems are superior in adapting to evolving knowledge, handling temporally informed data, and are more efficient in terms of memory, indexing time, and FLOPs compared to dense retrieval systems. Concurrently, for the more challenging Out-of-Distribution (OOD) scenarios, [110] explored the robustness of several representative GR and DR models from three perspectives: query variation, unforeseen query types, and unforeseen tasks. The findings indicate a generally poorer performance of GR models in OOD scenarios, highlighting the need for further enhancement.

Addressing the scenario of real-time addition of new documents, such as news or scientific literature IR systems, IncDSI [38] views the addition of new documents as a constrained optimization problem to find optimal representations for the new documents. This approach aims to (1) ensure new documents can be correctly retrieved by their relevant queries, and (2) maintain the retrieval performance of existing documents unaffected. IncDSI manages to add each document within approximately 20-50ms, significantly reducing the time and computational resources required compared to full model retraining, while maintaining competitive retrieval performance.

CLEVER [111], based on Product Quantization (PQ) [205], proposes Incremental Product Quantization (IPQ) for generating PQ codes as DocIDs for documents. Compared to traditional PQ methods, IPQ designs two adaptive thresholds to update only a subset of centroids instead of all, maintaining the indices of updated centroids constant. This method reduces computational costs and allows the system to adapt flexibly to new documents. To mitigate forgetting, CLEVER employs a memory-enhanced learning mechanism, maintaining a dynamic memory bank to store example documents similar to the new ones.

CorpusBrain++ [39] introduces the KILT++ benchmark for continuously updated KILT [168] tasks and designs a dynamic architecture paradigm with a backbone-adapter structure. By fixing a shared backbone model to provide basic retrieval capabilities while introducing task-specific adapters to incrementally learn new documents for each task, it effectively avoids catastrophic forgetting. During training, CorpusBrain++ generates pseudo queries for new document sets and continues to pre-train adapters for specific tasks. Moreover, it employs document clustering based on semantic similarity and a retraining strategy to maintain memory of older documents by revisiting them.

3.4 Downstream Task Adaption

Generative retrieval methods, apart from addressing retrieval tasks individually, have been tailored to various downstream generative tasks. These include fact verification [208], entity linking [209], open-domain QA [166], dialogue [210], slot filling [211], among others, as well as knowledge-intensive tasks [168], code [212], conversational QA [213], and multi-modal retrieval scenarios [214], demonstrating superior performance and efficiency. These methods are discussed in terms of separate training, joint training, and multi-modal generative retrieval.

3.4.1 *Separate Training*

For fact verification tasks [208], which involve determining the correctness of input claims, GERE [40] proposed using an encoder-decoder-based GR model to replace traditional indexing-based methods. Specifically, GERE first utilizes a claim encoder to encode input claims, and then generates document titles related to the claim through a title decoder to obtain candidate sentences for corresponding documents. Finally, an evidence decoder generates evidence sentence identifiers, resulting in improvements in time, memory consumption, and performance.

For Knowledge-Intensive Language Tasks (KILT) [168], CorpusBrain [41] introduced three pre-training tasks to enhance the model's understanding of query-document relationships at various granularities: Internal Sentence Selection, Leading Paragraph Selection, and Hyperlink Identifier Prediction. Similarly, UGR [104] proposed using different granularities of N-gram DocIDs to adapt to various downstream tasks, unifying different retrieval tasks into a single generative form. UGR achieves this by letting the GR model learn prompts specific to tasks, generating corresponding document, passage, sentence, or entity identifiers.

In multi-hop retrieval tasks, which require iterative document retrievals to gather adequate evidence for answering a query, GMR [112] proposed a generative approach. GMR employs language model memory and multi-hop memory to train the model, enabling it to memorize the target corpus and simulate real retrieval scenarios through constructing pseudo multi-hop query data, achieving dynamic stopping and efficient performance in multi-hop retrieval tasks.

DearDR [113] is an efficient data-centric self-regressive document retrieval method. It utilizes remote supervision and self-supervised learning techniques, using Wikipedia page titles and hyperlinks as training data. The model samples sentences from Wikipedia documents as input and trains a self-regressive model to decode page titles or hyperlinks, or both, without the need for manually labeled data. The model achieves zero-shot implementation of Wikipedia-based fact verification tasks and further optimizes performance through fine-tuning.

CodeDSI [114] is an end-to-end generative code search method that directly maps queries to pre-stored code samples' DocIDs instead of generating new code. Similar to DSI [30], it includes indexing and retrieval stages, learning to map code samples and real queries to their respective DocIDs. CodeDSI explores different DocID representation strategies, including direct and clustered representation, as well as numerical and character representations, showing superior performance compared to traditional methods on 1K and 10K scales, with numerical DocIDs performing better than alphabetic ones.

GCoQA [115] is a generative retrieval method for conversational QA systems. Unlike traditional methods relying on bi-encoder architecture and similarity matching, GCoQA directly generates DocIDs for passage retrieval. This method focuses on key information in the dialogue context at each decoding step, achieving more precise and efficient passage retrieval and answer generation, thereby improving retrieval performance and overall system efficiency.

Re3val [116] proposes a retrieval framework combining generative reordering and reinforcement learning. It first reorders retrieved page titles using context information obtained from a dense retriever, then optimizes the reordering using the REINFORCE algorithm to maximize rewards generated by constrained decoding. By improving page title reordering and context selection, Re3val achieves more accurate information retrieval.

3.4.2 Joint Training

The methods in the previous section involve separately training generative retrievers and downstream task generators. However, due to the inherent nature of GR models as generative models, a natural advantage lies in their ability to be jointly trained with downstream generators to obtain a unified model for retrieval and generation tasks.

UniGen [117] proposes a unified generation framework to integrate retrieval and question answering tasks, bridging the gap between query input and generation targets using connectors generated by large language models. UniGen employs shared encoders and task-specific decoders for retrieval and question answering, introducing iterative enhancement strategies to continuously improve the performance of both tasks. It demonstrates superior performance on both web search and question answering tasks.

Later, CorpusLM [42] introduces a unified language model that integrates GR, closed-book generation, and retrieval-augmented generation to handle various knowledge-intensive tasks. The model adopts a multi-task learning approach and introduces ranking-guided DocID decoding strategies and continuous generation strategies to improve retrieval and generation performance. In addition, CorpusLM designs a series of auxiliary DocID understanding tasks to deepen the model's understanding of DocID semantics. Experimental results validate the effectiveness and potential of CorpusLM (T5 [17] and Llama2 [23] variants) for knowledge-intensive language tasks.

3.4.3 Multi-modal Generative Retrieval

Generative retrieval methods can also leverage multi-modal data such as text, images, etc., to achieve end-to-end multi-modal retrieval.

IRGen [43] transforms image retrieval problems into generative problems, predicting relevant discrete visual tokens, i.e., image identifiers, through a seq2seq model given a query image. Its core innovation lies in its semantic image tokenizer, which converts global image features into short sequences capturing high-level semantic information. Unlike traditional methods that handle feature extraction and ANN search separately, IRGen achieves end-to-end differentiable search, optimizing directly from the final retrieval target, thereby enhancing retrieval accuracy and efficiency.

GeMKR [44] combines LLMs' generation capabilities with visual-text features, designing a generative knowledge retrieval framework. It first guides multi-granularity visual learning using object-aware prefix tuning techniques to align visual features with LLMs' text feature space, achieving cross-modal interaction. GeMKR then employs a two-step retrieval process: generating knowledge clues closely related to the query and then retrieving corresponding documents based on these clues. It aims to improve knowledge retrieval efficiency and accuracy in multi-modal scenarios.

The GRACE framework [44] achieves generative cross-modal retrieval method by assigning unique identifier strings to images and training multi-modal large language models (MLLMs) [215] to memorize the association between images and their identifiers. The training process includes (1) learning to memorize images and their corresponding identifiers, and (2) learning to generate the target image identifiers from textual queries. GRACE explores various types of image identifiers, including strings, numbers, semantics, identifiers, and atomic identifiers, to adapt to different memory and retrieval requirements.

3.5 Generative Recommender Systems

Recommendation systems, as an integral part of the information retrieval domain, are currently undergoing a paradigm shift from discriminative models to generative models. Generative recommendation systems do not require the computation of ranking scores for each item followed by database indexing, but instead accomplish item recommendations through the direct generation of IDs. In this section, several seminal works, including P5 [46], GPT4Rec [118], TIGER [48], SEATER [119], IDGenRec [120], LC-Rec [121] and ColaRec [122], are summarized to outline the development trends in generative recommendations.

P5 [46] transforms various recommendation tasks into different natural language sequences, designing a universal, shared framework for recommendation completion. This method, by setting unique training objectives, prompts, and prediction paradigms for each recommendation domain's downstream tasks, serves well as a backbone model, accomplishing various recommendation tasks through generated text. This approach demonstrates the viability and flexibility of generative models in recommendation systems.

Benefiting from the GPT architecture's outstanding performance in natural language processing, especially in text generation, GPT4Rec [118] takes users' historical interaction sequences as input, then generates multiple queries using beam search. They finally use these queries to complete the recommendation task by searching for items with the BM25 search engine.

In generative retrieval, effective indexing identifiers have been proven to significantly enhance the performance of generative methods. Similarly, TIGER [48] initially learns a residual quantized autoencoder to generate semantically informative indexing identifiers for different items. It then trains a transformer-based encoder-decoder model with this semantically informative indexing identifier sequence to generate item identifiers for recommending the next item based on historical sequences.

SEATER [119] designs a balanced k-ary tree-structured indexes, using a constrained k-means clustering method to recursively cluster vectors encoded from item texts, obtaining equal-length identifiers. Compared to the method proposed by DSI [30], this balanced k-ary tree index maintains semantic consistency at every level. It then trains a Transformer-based encoder-decoder model and enhances the semantics of each level of indexing through contrastive learning and multi-task learning.

IDGenRec [120] innovatively combines generative recommendation systems with large language models by using human language tokens to generate unique, concise, semantically rich and platform-agnostic texual identifiers for recommended items. The framework includes a text ID generator trained on item metadata with a diversified ID generation algorithm, and an alternating training strategy that optimizes both the ID generator and the LLM-based recommendation model for improved performance and accuracy in sequential recommendations. The zero-shot performance of IDGenRec is comparable to, or even surpasses, certain traditional recommendation models that rely on supervised training, highlighting its potential as a foundational model for recommendation systems.

Focusing solely on semantic information and overlooking the collaborative filtering information under the recommendation context might limit the further development of generative models. Therefore, after generating semantic indexing identifiers similar to TIGER using a residual quantized autoencoder with uniform semantic mapping, LC-Rec [121] also engages in a series of alignment tasks, including sequential item prediction, explicit index-language alignment, and recommendation-oriented implicit alignment. Based on the learned item identifiers, it integrates semantic and collaborative information, enabling large language models to better adapt to sequence recommendation tasks.

ColaRec [122] integrates collaborative filtering signals and content information by deriving generative item identifiers from a pretrained recommendation model and representing users via aggregated item content. Then it uses an item indexing generation loss and contrastive loss to align content-based semantic spaces with collaborative interaction spaces, enhancing the model's ability to recommend items in an end-to-end framework.

4 Reliable Response Generation: Direct Information Accessing with Generative Language Models

The rapid advancement of large language models has positioned them as a novel form of IR system, capable of generating reliable responses directly aligned with users' informational needs. This not only saves the time users would otherwise spend on collecting and integrating information but also provides personalized, user-centric answers tailored to individual users.

However, challenges remain in creating a grounded system that delivers faithful answers, such as hallucination, prolonged inference time, and high operational costs. This section will discuss strategies for constructing a faithful GenIR system, focusing on both optimizing the model internally and enhancing it with external knowledge.

4.1 Internal Knowledge Memorization

To develop an user-friendly and reliable IR system, the generative model should be equipped with comprehensive internal knowledge. Optimization of the backbone generative model can be categorized into three aspects: structural enhancements, training strategies, and inference techniques.

4.1.1 Model Structure

With the advent of generative models, a variety of methods have been introduced to improve model structure and enhance generative reliability. We aim to discuss the crucial technologies contributing to this advancement in this subsection.

Model Scaling Model parameter scaling is a pivotal factor influencing performance. Contemporary language models predominantly employ the Transformer architecture, which has been observed that scaling both the model parameters and the training data enhances the model's capacity to retain knowledge and capabilities [216]. For instance, in the GPT [19, 63, 191, 217] series and LLaMA [23, 65] family, models with larger parameter sizes tend to perform better on diverse downstream tasks. These include few-shot learning, language understanding, and generation [123]. In addition, scaling the model contributes to improved instruction-following capabilities [218], enabling a more adept comprehension of user intent and generating responses that better satisfy user requests.

Model integration Model integration is an effective and intuitive method to enhance the reliability of generated outputs by capitalizing on diverse strengths inherent in

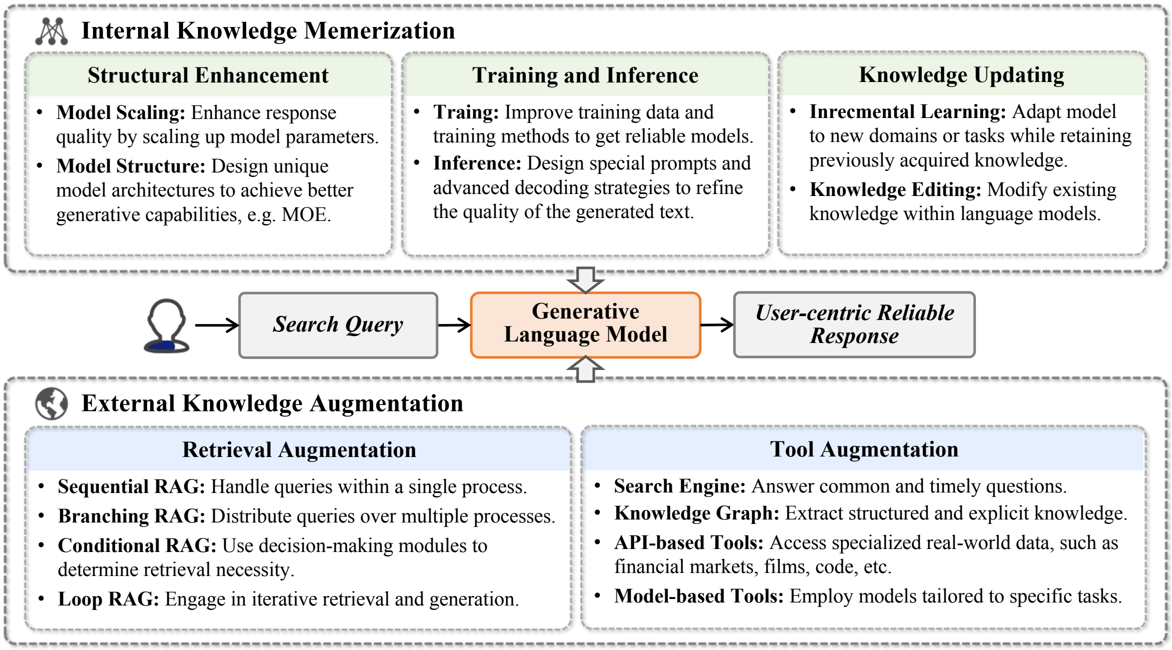

Fig. 5. An Illustration of strategies for enhancing language models to generate user-centric and reliable responses, including model internal knowledge memorization and external knowledge augmentation.

various models. This diversity arises from differences in training data and architectural frameworks. The predominant approach to model integration is the 'Mixture of Experts' (MoE) [219], which utilizes a gating mechanism to selectively activate sections of network parameters during inference. MoE facilitates the amalgamation of the strengths of multiple expert models, greatly increasing the effective parameters of the model without inflating inference costs. Consequently, this leads to enhanced performance [124, 220–222]. This method also boasts impressive scalability: the overall efficacy of the MoE model is augmented in tandem with the expanding parameter volume and the number of expert models [223].

In contrast to the MoE mechanism that involves training multiple expert models and a gating system from scratch, the LLM-Blender framework [224] employs a ranker and a fuser module to organize and merge answers from diverse LLMs. This approach is applicable to a wide array of large-scale models, including black-box types; however, it presents challenges associated with high deployment costs.

4.1.2 Training and Inference

In the model training stage, methods to enhance the reliability of answers can be categorized into two aspects: training data optimization and training methods optimization.

Training Data Optimization The quality of training data substantially affects the reliability of model outputs. Noise, misinformation, and incomplete information can disrupt the learning process of models, leading to hallucation and other issues. Many studies have focused on the processing of model training data. [225] used GPT3.5 to artificially create textbooks filled with examples and language descriptions as training data. They minorly fine-tuned the model with a small amount of data, resulting in significant improvements on downstream tasks. LIMA [226] used dialogues from community forums to construct a small-scale fine-tuning dataset, enhancing the model's conversation capabilities during the alignment phase.

Concerning the fact that a large part of training data comes from automatically crawled internet page data, which contains many redundancies, some works have attempted to deduplicate the training dataset. Using a combination of suffix array [227] and MinHash [228] techniques, Lee et al. [66] designed an approximate matching algorithm to identify redundant content originating from the same source, hence reducing the proportion of answers the model directly reproduces from training data.

Training Methods Optimization In addition to conventional training methods, additional techniques have been proposed to improve the factuality of model outputs. Given that the autoregressive training objective of language models can lead to blind imitation of training data, MixCL [229] incorporated contrastive learning into the training objective of language models. It used an external knowledge base to identify correct knowledge snippets, incorporated contrastive learning loss function to reduce the generation probability of tokens in incorrect knowledge snippets, thereby enhancing model reliability.

CaliNet [67] utilized a contrastive method to assess erroneous knowledge learned by the model and fine-tuned the parameters of the FFN layer to rectify these errors. FactTune [126] incorporated factuality assessment during the RLHF phase. It used several automatic evaluation methods, including Factscore [177], to rank the factuality of model outputs. Consequently, it utilized the DPO [230] method to teach the model factuality preference ranking.

Apart from enhancing the internal knowledge reliability of the model during the training phase, the inference stage significantly impacts the reliability of answers. The overall inference process consists of two parts: the inputting of corresponding instructions by the user to the model and the decoding of the corresponding response tokens by the model. The approach to increase generation reliability during the inference stage can also be divided into two parts: prompting engineering and decoding strategy.

Prompt Engineering Prompting method plays a vital role in guiding the model. A well-designed prompt can better promote the model's internal capabilities to provide more accurate answers. The Chain-of-Thought (CoT) [231] prompting method guides the model to explicitly decompose the question into a reasoning chain during the decoding process and to reason out the answer step by step according to the reasoning chain. This method can improve the response accuracy for each sub-question, grounding the final question on the most accurate intermediate step, thus enhancing the reliability of the answer. Further, CoT-SC [232] builds upon this by sampling multiple answers and choosing the most consistent one as the final answer. The Tree of Thoughts [233] expands the single reasoning path of CoT to multiple paths and synthesizes the reasoning outcomes of these paths to arrive at the final answer.

The Chain-of-Verification (CoVE) [234] introduces a self-reflection mechanism within the prompts of LLM. The prompt causes the LLM to generate a draft response, and then formulates a validation plan for each statement within the response to check for factual inaccuracies. If errors are found, corrections are carried out, effectively enhancing the factual accuracy of the response.

Besides, some prompting methods enhance the grounding of answers on more accurate content by prompting the model to output relevant internal knowledge. For example, RECITE [128] and GenRead [127] utilize a sampling method to prompt the model to output certain relevant knowledge fragments, which are then used to bolster the question-answering process.

Decoding Strategy Decoding strategies are another critical factor influencing the reliability of model-generated responses. An appropriate decoding method can maintain the reliability and diversity of a model's response. In common gready search or beam search approaches, models may produce responses that are high in general applicability but lack diversity, compromising the quality of the generated content. Nucleus sampling [235] is proposed to tackle this issue. By sampling within a set probability range for tokens, it can ensure better diversity for each generated token, and achieve results that balance variety and reliability. Building on this, Lee et al. introduced Factual-Nucleus Sampling [236], which employs a dynamic threshold that decays for subsequent tokens, ensuring that later parts of a generated sentence are not adversely affected by earlier parts that may lack factual content. Wan et al. [237] proposed a faithfulness-aware decoding method to enhance the faithfulness of the beam-search approach. This method incorporates a Ranker to reorder the generated beam sequences and a lookahead method to avoid selecting tokens that may lead to unfaithfulness in further generation, thereby improving the faithfulness of the results.

Apart from directly modifying the decoding method, numerous studies have influenced the model's decoding distribution by utilizing or modifying the information in the hidden layers. DoLa [68] uses the distributional differences between the hidden layer and the output layer as the distribution for the next token, capturing factual knowledge or key terms that are newly learned by the output layer relative to the base network, hence increasing the likelihood of generating these terms. Inference-Time Intervention (ITI) [238] focuses on identifying attention heads highly correlated with response correctness and adjusting their orientations, then moderating the activation of these attention heads. This method achieves more truthful generation with minimal interference to the model.

In practical applications, prompts often contain additional textual content to provide extra information, which may conflict with the model's inherent knowledge, leading to hallucinations. Shi et al. [239] proposed the CAD method, which compares the output distributions before and after the addition of extra information. This reduces the weight given to the model's own knowledge, thus avoiding conflicts between the two that might result in factual inaccuracies.

4.1.3 Knowledge Updating

In real-life scenarios, information is constantly evolving, and therefore, the GenIR system needs to continuously acquire the latest knowledge to meet users' information needs. Since the model's knowledge storage is limited, knowledge updating is necessary to ensure more reliable generated responses. In this section, we will discuss existing methods for knowledge updating from two perspectives: incremental learning and knowledge editing.

Incremental Learning ◦ Incremental learning refers to the ability of machine learning models to continuously learn new skills and tasks while retaining previously acquired knowledge [240–243]. In the GenIR system, it is crucial to enable the language model to memorize the latest information while preventing the forgetting of previous knowledge.

One approach is *Incremental Pre-training*, which does not rely on supervised data but continues pre-training on continuously updated corpora to alleviate catastrophic forgetting. For example, Baidu proposed the ERNIE 2.0 framework [69], which enhances language understanding through continuous multi-task learning. Jang et al. [244] introduced the concept of Continual Knowledge Learning (CKL) to explore how LLMs can update and retain knowledge in the face of rapidly changing world knowledge. They also created new benchmarks and evaluation metrics such as FUAR. Cossu et al. [245] studied continual pre-training for language and vision and found that self-supervised or unsupervised pre-training methods are more effective in retaining previous knowledge compared to supervised learning. Additionally, Ke et al. [70] proposed Domain Adaptive Pre-training (DAP-training) to improve the model's adaptability to new domains while preventing the forgetting of previously learned knowledge using techniques like soft masking and contrastive learning. For domain-specific model construction, Xie et al. [246] introduced FinPythia-6.9B, an efficient continual pre-training method specifically designed for constructing large-scale language models in the financial domain.

On the other hand, *Incremental Fine-tuning* involves training the model using only labeled data. For instance, Progressive Prompts [247] creates new soft prompts for each new task and concatenates them in the order of their occurrence, effectively transferring knowledge and mitigating forgetting. DynaInst [129] improves the lifelong learning performance of pre-trained language models using parameter regularization and experience replay. It employs dynamic instance selection and task selection mechanisms to optimize learning efficiency under resource-constrained settings. Jang et al. [248] challenge traditional multi-task prompt fine-tuning methods by fine-tuning expert language models on individual tasks. Suhr et al. [249] adopt a feedback-driven continual learning approach applicable to instruction-following agents. In human-agent interaction, users control the agent using natural language and convert feedback into immediate rewards through contextual bandits, further optimizing the learning process. Furthermore, O-LoRA [250] demonstrates its superiority in the field of continual learning by learning new tasks in low-rank subspaces while maintaining orthogonality between these subspaces, significantly reducing interference between tasks. Peng et al. [251] propose a scalable language model that dynamically adjusts parameters based on different task requirements, effectively reducing the forgetting of previous task knowledge during new task learning.

Knowledge Editing. Knowledge editing refers to the process of modifying and updating existing knowledge within language models [242, 252], which is distinct from incremental learning that focuses on adapting to new domains or tasks. Knowledge editing methods, by editing the weights or layers of a model, can correct erroneous facts and incorporate new knowledge. This makes it an important technology before deploying GenIR systems. There are primarily three paradigms for internal knowledge editing within language models: adding trainable parameters, locate-then-edit, and meta-learning.

One method of *Adding Trainable Parameters* is by integrating new single neurons (patches) in the final feed-forward neural network (FFN) layer, such as T-Patcher [253] and CaliNET [67]. These neurons serve as trainable parameters that introduce new errors and adjust the model's behavior. Alternatively, discrete code-book modules are introduced in the middle layers of the language model, as in GRACE [254], to adjust and correct information.

Moerover, the *Locate-then-Edit* method first identifies the parameters corresponding to specific knowledge and then updates these targeted parameters directly. Common techniques involve identifying key-value pairs in the FFN matrix, known as "knowledge neurons," and updating them [255]. Techniques like ROME [71] use causal mediation analysis to pinpoint areas needing editing, and MEMIT [256] builds on ROME to implement synchronized editing in various scenarios. Moreover, methods such as PMET [257] employ attention mechanisms for editing, while BIRD [258] has introduced a bidirectional inverse relation modeling approach.

Meta Learning, which is another paradigm, uses hyper-networks to generate the necessary updates for model editing. KE (Knowledge Editor) [130] predicts weight updates for each data point using a hyper-network. MEND [131], by taking low-order decomposition of gradients as input, learns to rapidly edit language models to enhance performance. Additionally, MALMEN [259] manages to separate the computations of hyper-networks and language models, facilitating the editing of multiple facts under a limited memory budget. These meta-learning mechanisms enable models to swiftly adapt to new knowledge and tasks and perform the necessary edits.

4.2 External Knowledge Augmentation

Although large language models have demonstrated significant effectiveness in response generation, issues such as susceptibility to hallucinations, difficulty to handle in-domain knowledge and challenges with knowledge updating persist. Augmenting the model's generative process with external knowledge sources can serves as an effectively address to tackle these issues. Based on the form of external knowledge employed, these approaches can be classified into retrieval augmentation and tool augmentation.

4.2.1 Retrieval Augmentation

Retrieval-Augmented Generation (RAG) enhance the response of generative models by combining them with a retrieval mechanism [72, 260, 261]. By querying a large collection of documents, information that is relevant to the input query can be fetched and integrated into input of the generative model. RAG enables generative models to be grounded in existing reliable knowledge, significantly improving the reliability of model generation. Typically, a RAG method involves a retriever and a generator. Based on the interaction flow between these two, RAG methods can be divided into four categories [262].

Sequential RAG: Sequential RAG operates on a linear progression, where the retriever first retrieves relevant information and the generator utilizes information to directly complete the response generation process.

The basic form of sequential RAG is a "Retrieve-Read" framework [132]. Early works complete the response generation process through joint training of retriever and generator [72, 263, 264] or separate training [260]. Although it has achieved good results, there is a need for pre-training of the language model, which is expensive and not conducive to generalization. The In-Context RALM [265] solves this problem by directly using the retrieval document as input, utilizing the model's in-context learning capability to understand the document without any training.

With the widespread adoption of large language models, most subsequent works is built on the foundation of a frozen generator. AAR method [266] fine-tuned a general retriever to adapt to the information acquisition preferences of the generative model. LLM-embedder [267] using reward produced by LLM to train an embedding model dedicated to retrieval augmentation. ARL2 [133] leverages LLM to annotate relevance score in training set, and train a retriever using contrastive learning.

Several works introduce pre-retrieval and post-retrieval process [262] into the sequential pipeline to enhance the overall efficiency. In pre-retrieval process, the RRR model [132] introduces a rewriter module before the retriever. The rewriter is trained by utilizing generator's feedback and can enable the retrieval system to provide more suitable information for generation.

In post-retrieval process, information compressors are proposed to filter out irrelevant content from documents, avoiding misleading generator's response. [268–270] RECOMP [271] introduce both abstractive and extractive compressors to generate concise summary for retrieved documents. LLMLingua [272] calculates the importance of each token based on the perplexity provided by the generative model, while retaining important tokens. Furthermore, LongLLMLingua [273]introduces query-aware compression on its basis and rerank the retrieved documents based on the calculated importance score to alleviate the loss in the middle phenomenon [268]. PRCA [274] employs reinforcement learning to train a text compressor adaptable to black-box LLMs and various retrievers, functioning as a versatile plug-in for multiple scenarios.

Branching RAG: In Branching RAG framework, the input query is processed across multiple pipelines, and each pipeline may involve the entire process in the sequential pipeline. The outputs from all pipelines merged to form the final response. Compared to the sequential RAG process, this approach allows for finer-grained handling of the query or retrieval results.

In pre-retrieval stage, TOC [134] applies few-shot prompting to recursively decompose complex, ambiguous questions into clear, disambiguated sub-questions. These sub-questions are structured in a tree structure, with relevant documents retrieved for each. Using all valid question nodes, a long-form answer covering all sub-questions is generated. BlendFilter [135] leverages prompts to enhance the original query through both internal and external knowledge. The augmented queries are then used to retrieve related documents, which are subsequently merged. When facing complex questions from users, this type of method can break down the questions and then answer them, improving the comprehensiveness of the response.

In post-retrieval stage, REPLUG [136] inputs each retrieved document along with the query into a generator to obtain a predicted probability distribution. These individual distributions are then amalgamated to yield the ultimate predictive probability distribution. GenRead [127] prompts LLM to generate related documents on its own while utilizing the retriever to retrieve documents. These two sets of documents are then merged as input. The versatility in designing prompts for LLM to create related documents allows for a broader coverage of content, thus enhancing the likelihood of meeting the user's query requirements more effectively.

Conditional RAG: The Conditional RAG framework adapts to various query types through distinct processes, improving the system's flexibility. Since there can be knowledge conflict between the knowledge from retrieved documents and the generator's own knowledge, RAG's effectiveness isn't consistent across all scenarios. To address this, common conditional RAG methods include a decision-making module that determines whether to engage the retrieval process for each specific query.

SKR [137] builds a binary classification model trained on a dataset compiled from questions that LLMs can and cannot answer. This model is leveraged in inference stage to discern whether a given query should utilize retrieval. Labeling for the training data is gathered by directly prompting the model to ascertain if external knowledge is required for generation.

Self-DC [73] employs the confidence score of the model's response to determine the necessity of retrieval. Based on the confidence score, queries are categorized into three groups: unknown, uncertainty, and known. Queries deemed unknown are processed through a sequential RAG pipeline, while those with uncertainty are broken down into sub-questions to generate answers.

Rowen [138] introduces a multilingual detection module that transforms the original question into semantically equivalent perturbed questions and collects their responses. The decision to retrieve is then based on measuring the consistency across these responses.

Loop RAG: Loop RAG involves deep interactions between the retriever and generator components. Owing to multi-turn retrieval and generation processes, accompanied by comprehensive interactions, it excels at handling complex and diverse input queries, yielding superior results in response generation.

ITER-RETGEN [139] introduce an iterative retrieval and generation framework. For a given query, it initially performs retrieval-augmented generation, retrieving relevant documents and then generate an answer. Subsequently, it undertakes generation-augmented retrieval, where it continues to retrieve based on the content generated in the previous step, and synthesizes text using the newly retrieved documents. This process of alternating the two steps is repeated a fixed number of times to produce the final answer. IR-COT [140] has a similar overall procedure to ITER-RETGEN, but its iteration pause is contingent on the model's own generative process.

FLARE [77] introduces a strategy of concurrent retrieval as responses are being generated, in contrast to conducting retrieval only after a complete response has been produced. For each new sentence generated, the framework evaluates the need for retrieval based on the LLM's confidence score for that sentence. If needed, it formulates a query based on the sentence to retrieve relevant information, and then regenerates the sentence. This method dynamically supplements the LLM with the necessary information, enhancing the reliability of the generated content. COG [275] models the generation process as a continual retrieval of segments from an external corpus and subsequent copying. The generator's primary role therein is to produce certain conjunction words to preserve sentence fluency. Self-RAG [74] adds special tokens into the vocabulary to allow the generator to decide whether to retrieve, the importance of the retrieved document, and whether to perform a critique.

Some works focus on deconstructing complex inquiries into sub-questions, addressing these individually to produce a more dependable response. [276] guides LLM to decompose complex questions into sub-questions, responds to each individually using retrieved results, and ultimately synthesizes the answers to all sub-questions to form the final response. Building upon this, RET-Robust [277] incorporates a NLI model to evaluate whether the retrieved documents can substantiate the answers to sub-questions, thereby minimizing the risk of the LLM being misled by irrelevant information.

4.2.2 Tool Augmentation

Although retrieval-augmented techniques have significantly improved upon the blind spots of generator's self-knowledge, these methods struggle with the rapid and flexible update of information since they rely on the existence of information within an external corpus of documents. Tool augmentation, on the other hand, excels in addressing this issue by invoking various tools that allow for the timely acquisition and usage of the latest data, including finance, news, and more. Moreover, tool augmentation expands the scope of responses a model can offer, such as language translation, image generation, and other tasks, to more comprehensively meet users' information retrieval needs. There are four categories of tools that can be utilized to construct a more reliable information retrieval system:

Search Engine: Common search engine tools like Google Search and Bing Search help answer frequent and time-sensitive queries effectively. Self-Ask [276] initially decomposes complex questions into multiple sub-questions, then uses search engine to answer each sub-question, and finally generating a comprehensive answer to the complex question. ReAct [75] embeds search engine calls into the model's reasoning process, allowing the generative model to determine when to make calls and what queries to input for more flexible reasoning. New Bing can automatically search relevant information from Bing based on user input, yielding reliable and detailed answers, including citation annotations in the generated content.

Some works have also built advanced conversational systems based on tools like search engines. Internet-Augmented Generation [278] enhances the quality of conversational replies by using search engine during conversations. LaMDA [26] and BlenderBot [279] combine search engines with conversational agents, constantly accessing internet information to enrich conversation factualness. WebGPT [52] and WebCPM [28] directly teache models to perform human-like browser operations by generating commands such as Search, Click, and Quote, facilitating the automated retrieval and acquisition of information.

Knowledge Graph (KG): Compared to search engine, KG is particularly useful for extracting structured, explicit knowledge. Relevant knowledge from a knowledge graph can be extracted and used as a prompt input to enhance the generative process [141]. StructGPT [76] introduced an iterative reading-and-reasoning framework where the model can access a knowledge graph through a well-designed interface, continually acquiring information and reasoning until an answer is obtained. RoG [142] generates plausible reasoning paths based on a KG before executing each path in parallel and integrating the outcomes for a final answer. The concept behind ToG [141] is similar to RoG, but instead of pre-planning reasoning paths, it allows the model to explore potential entities and links first and then reason, continuously assessing the feasibility of the reasoning paths.

API-based Tools: An important part of the tools is the real-world APIs, which enable the model to obtain information from specific data sources, such as real-time stock information, movie services, code interpreters and so on. However, the multitude and diversity of APIs, coupled with the adherence to certain operational protocols, make the teaching of API usage to models a focal point of this area. Toolformer [78] trains language models in self-supervised manner to automatically call APIs when needed. It commences by utilizing prompts to generate API calls within the text, subsequently executes these calls, and filters out ineffective ones based on the execution results to form the final dataset. By employing standard language modeling objectives on this dataset, it is possible to train a model capable of autonomously invoking APIs across various downstream tasks without losing its inherent language modeling capabilities.

RestGPT [280] has formulated a comprehensive framework for prompting LLMs to invoke RESTful APIs, comprising an online planner, an API selector, and an executor. ToolLLM [143] capitalizes on a substantial corpus of scraped APIs to build a dataset for fine-tuning. Moreover, Gorilla [281] introduces an information retriever to provide the model with reference API documentation, which facilitates the teaching of retrieval-based information utilization during fine-tuning. ToolkenGPT [282] incorporates each tool as a new token into the vocabulary. During the training process, the model learns the representation of each token, enabling it to invoke APIs in an inference phase as naturally as generating regular text. Beyond learning to invoke APIs, CREATOR [283] proposes a framework to prompt models to write code based on actual problems as a new tool implementation. The generated tools function through a code interpreter and have demonstrated impressive outcomes on complex mathematical reasoning tasks.

A part of the work additionally supports multimodal inputs, further broadening the application scope of the models. AssistGPT [144] offers a comprehensive framework that includes modules such as Planner, Executor, Inspector, and Learner, utilizing both language and code to enable more intricate inference processes. ViperGPT [284], by feeding CodeX with user queries and visual API information, generates corresponding Python code to invoke APIs, successfully completing complex visual tasks.

Model-based Tools: With the swift expansion of diverse AI communities(i.e., Huggingface, ModelScope, Github), various types of AI models have become readily accessible for use, serving as a pivotal tool in enhancing generative retrieval systems. These AI models encompass a wide array of tasks each accompanied by comprehensive model descriptions and usage examples. Owing to their extensive training on specific tasks, these models often exhibit exceptional performance in those tasks.

HuggingGPT [145] employs ChatGPT as a controller to deconstruct user queries into a sequence of tasks, subsequently determining which models to invoke for task execution. Similarly, Visual ChatGPT [146] integrates a visual foundation model with Large Language Models (LLMs), leveraging ChatGPT as a prompt manager to mobilize various Visual Foundation Models, like BLIP and ControlNet. It is adept at processing users' image-based requests and is more efficient compared to multi-modal models.

4.3 Generating Response with Citation

To build a reliable GenIR system, generating responses with citations is a promising approach [16, 285, 286]. Citations allow users to clearly understand the source of each piece of knowledge in the response, not only enhancing trust in the GenIR system but also facilitating its widespread adoption. Existing methods can be divided into directly generating responses with citations and using a retrieval module to enhance the generated content.

4.3.1 Direct Generating Response with Citation

This method uses the model's intrinsic memory to generate source citations without relying on the retrieval module.

Model Intrinsic Knowledge. Leveraging the capabilities of the language model itself, according-to prompting [79] guides LLMs to more accurately cite information from the pre-training data to reduce the generation of false information. By adding phrases like "according to Wikipedia" in the model's prompts, the model is guided to cite the corresponding knowledge sources.

Furthermore, to improve the quality of citations, several methods are proposed. Iterative Feedback Learning (IFL) [80] first uses a critique model to assess the generated text, then provides targeted feedback based on the assessment results, guiding LLMs to iteratively improve their performance. Through this method, the IFL approach can effectively enhance the accuracy of citations, content correctness, and linguistic fluency of LLMs, while maintaining a high standard of information accuracy. In addition, Fierro et al. [147] proposed a plan-based approach, defining the generation plan as a series of questions that serve as a blueprint for generating content and its organization. They introduced two attribution models utilizing the blueprint: an abstract model, where questions are generated from scratch; and an extractive model, where questions are directly copied from the input. Experiments show that planning consistently improves the quality of citations.

Incorporating Generative Retrieval. As envisioned by Metzler et al. [16], to build an expert-level reliable IR system, allowing the model to directly generate responses with citations is a promising approach. Users do not need to search for answers from a list of returned documents like traditional IR systems but can directly receive reliable responses tailored to their information needs. Moreover, the cited document is also generated by the model through the generative retrieval approach described in Section 3, directly generating corresponding DocIDs.

Utilizing generative retrieval, 1-PAGER [149] combines answer generation and evidence retrieval. 1-PAGER gradually generating N-gram DocIDs through constrained decoding using FM-Index [204], thereby partitioning the retrieval corpus, selecting documents, and generating response step by step. 1-PAGER is comparable to existing retrieval-then-read methods in retrieval and answer accuracy and superior to pure closed-book QA models because it attributes predictions to specific evidence corpora. It provides a new scheme for integrating retrieval into seq2seq response generation.

Recently, [150] proposes a source-aware training method. Specifically, they teach the model to associate DocIDs with knowledge during pre-training, then provide citations of supporting evidence during instruction tuning. Experiments have shown that this method can effectively achieve knowledge attribution of pre-training data, enhancing the verifiability of LLMs, and providing a promising research direction for building a trustworthy GenIR system.

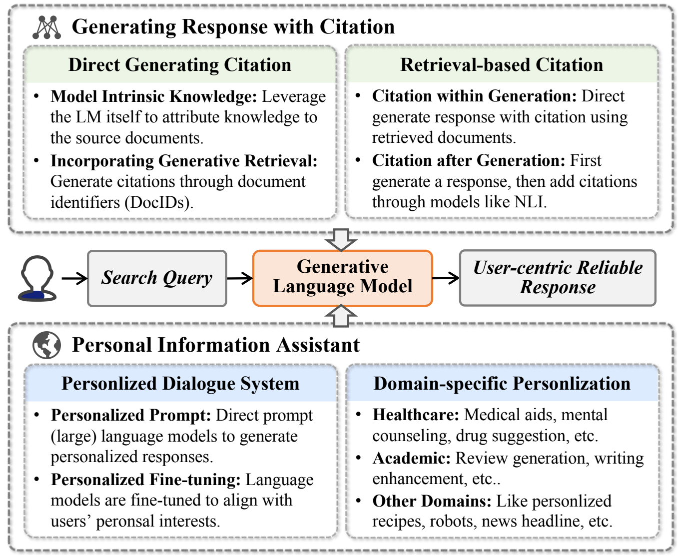

Fig. 6. Other notable approaches for building a reliable and user-centric GenIR system, including generating response with citation and personal information assistant.

4.3.2 Retrieval-based Response with Citation

To enhance the accuracy of citations, several methods have been developed based on retrieval techniques to fetch relevant documents, thereby improving the quality of responses with embedded citations.

Citation within Generation. Following retrieval, models directly generate responses that include citations. Initially, systems like WebGPT [52], LaMDA [26], and WebBrain [53] were developed. These methods utilize web pages or Wikipedia to construct large-scale pre-training datasets, teaching models how to generate responses with citations.

Subsequently, more advanced strategies for citation generation were proposed. For instance, Search-in-the-Chain (SearChain) [152] introduces a method where it first generates a reasoning chain (Chain-of-Query, CoQ) through prompts to LLMs. The retrieval module interacts with each node of the CoQ for verification and completion tasks. After the interaction, SearChain performs tracing operations, generating the entire reasoning process and marking citations for each step of the inference.

Following retrieval and answer generation, LLatrieval [81] suggests continuously improving the retrieval results through an iterative updating process, verifying whether the retrieved documents can adequately support the generated answers until the verification is satisfied. Later, AGREE [287] utilizes a Natural Language Inference (NLI) model to verify the consistency between the LLM-generated answers and the retrieved documents, employing a Test-Time Adaptation (TTA) strategy that allows LLMs to actively search and cite the most current information during the generation process, thus enhancing the accuracy and reliability of the responses. VTG [154] integrates an evolved memory system and a dual-layer validator for generating verifiable text, combining long-term and short-term memories to adapt to dynamically changing content focuses and using an NLI model to evaluate the logical support between claims and potential evidences.

Based on the graph of thoughts (GoT) [288], HGOT [289] improves context learning in retrieval-augmented settings by constructing a hierarchical GoT. This method leverages the LLM's planning capabilities to break down complex queries into smaller sub-queries and introduces a scoring mechanism to assess the quality of retrieved paragraphs.

Employing reinforcement learning, Huang et al. [290] introduce a fine-grained reward mechanism to train language models, allocating specific rewards for each generated sentence and citation, thus teaching models how to accurately cite external information sources. This approach uses rejection sampling and reinforcement learning algorithms to enhance model performance in generating citation-inclusive text through localized and specialized reward signals. APO [82] reimagines the task of attributive text generation as a preference learning problem, automatically generating a large number of preference data pairs to reduce the cost of manual annotation. Through progressive preference optimization and experience replay techniques, this method reinforces the model's preference signals at a fine-grained level, while avoiding overfitting and text degradation caused by automatic data generation.

Citation after Generation. This approach involves models first generating a response, then adding citations through models like NLI. RARR [151] improves attributability by automatically finding external evidence for the language model's output and post-editing to correct content while preserving the original output as much as possible. This method combines a few training examples, LLM, and retrieval to enhance its attribution capabilities without altering the existing model. Following that, PURR [291] employs an unsupervised learning method by allowing LLMs to generate text noise themselves, then training an editor to eliminate this noise, achieving an efficient editing process. PURR leverages the generative capabilities of language models to create training data, which not only improves attribution performance but also significantly speeds up the generation process. CEG [155] searches for supporting documents related to generated content and uses an NLI-based citation generation module to ensure each statement is supported by citations. "Attribute First, then Generate" [292] decomposes the generation process, first selecting relevant source text details and then generating based on these details, achieving localized attributability in text generation. This method offers a finer granularity of attribution, ensuring high-quality text with each sentence supported by a clear source, greatly reducing the workload of manual fact-checking.

4.4 Personal Information Assistant

The core of the GenIR system is the user, so understanding user intent is crucial. Researchers have explored various methods like personalized search [293–296], dialogue [83, 156, 297] and recommender [298–300] systems to explore the users' interests. Specifically, personalized information assistants aims to better understand users' personalities and preferences, generating personalized responses to better meet their information needs. This section reviews the progress in research on personalized dialogue and domain-specific personalization.

4.4.1 Personalized Dialogue System

To better understand user needs, researchers have explored two main approaches: personalized prompt design and model fine-tuning.

Personalized Prompt. For personalized prompt design, Liu et al. [298] and Dai et al. [299] input users' interaction history and rating history into ChatGPT [3] for in-context learning, effectively generating personalized responses. LaMP [301] enhances the language model's personalized output by retrieving personalized history from user profiles. Using long-term history, [302] designs prompts describing users' long-term interests, needs, and goals for input into LLMs. BookGPT [300] uses LLM prompts, interactive querying methods, and result verification frameworks to obtain personalized book recommendations. PerSE [303] infers preferences from several reviews by a specific reviewer and provides personalized evaluations for new story inputs.

Using prompt rewriting, [304] proposes a method combining supervised and reinforcement learning to better generate responses from frozen LLMs. Similarly, [305] rewrites user input prompts using extensive user text-to-image interaction history to align better with expected visual outputs.

Personalized Fine-tuning. This line of work focuses on fine-tuning models for personalized response generation. To generate more engaging dialogues based on users' personalities and interests, Zhang et al. [156] introduced a dataset with 5 million personas for dialogues, Persona-Chat, to train models to produce more personalized and appealing conversations. Mazaré et al. [306] created a dataset of over 700 million conversations extracted from Reddit, demonstrating the effectiveness of training dialogue models on a large-scale personal profile dataset. $\mathcal{P}^2$Bot [83] generates more personalized and consistent dialogues by simulating the perception of personalities between conversation participants. DHAP [297] designs a novel Transformer model structure to automatically learn implicit user profiles from users' dialogue history without explicit personal information. Wu et al. [157] propose a generative segmentation memory network to integrate diverse personal information. Fu et al. [307] developed a variational approach to model the relationship between personal memory and knowledge selection, with a bidirectional learning mechanism allowing mutual learning between personal memory fragments and knowledge selection.

Using reinforcement learning, Cheng et al. [308] collected a domain-specific preference (DSP) dataset and proposed a three-stage reward model learning scheme, including base language model training, general preference fine-tuning, and customized preference fine-tuning. Jang et al. [158] developed a method called "Personalized Soups," first optimizing multiple policy models with different preferences independently using PPO [309], then dynamically combining these models' parameters during the inference stage.

Using retrieval-enhanced methods, LAPDOG [310] retrieves relevant information from story documents to enhance personal profiles and generate better personalized responses. To effectively utilize multiple knowledge sources, SAFARI [84] leverages LLMs' capabilities in planning, understanding, and integrating knowledge under supervised and unsupervised training settings, effectively generating responses consistent with character settings and knowledge-enhanced. Inspired by writing education, Li et al. [311] proposed a multi-stage, multi-task framework to teach LLMs to generate personalized responses, including retrieval, ranking, summarization, synthesis, and generation. For subjective tasks, [312] studied the superior performance of personalized fine-tuning in subjective text perception tasks compared to non-personalized models.

To achieve a personalized information assistant for every user, OPPU [159] uses personalized PEFT [195] to store user-specific behavioral patterns and preferences, showing superior performance in handling changes in user behavior, modeling users with different activity levels, and among different PEFT methods.

For multimodal scenarios, PMG [313] proposes a personalized multi-modal generation method that transforms user behavior into natural language, allowing LLMs to understand and extract user preferences.

4.4.2 Domain-specific Personalization

Understanding users' personalized information needs, the GenIR system has broad applications across various domains such as healthcare, academia, education, recipes, etc.

Healthcare. In AI-assisted healthcare, personalization plays a crucial role. Liu et al. [314] utilize few-shot tuning to process and infer based on time-series physiological and behavioral data. Zhang et al. [315] implement specific medical diagnosis identification in databases and prospective diagnostic assistance using prompts from ChatGPT [3] and GPT-4 [217]. Subsequently, Yang et al. [316] propose an LLM for traditional Chinese medicine called Zhongjing, based on LLaMA [65], which undergoes a complete training process from continued pre-training, supervised fine-tuning to reinforcement learning with human feedback (RLHF) [22]. Abbasian et al. [317] introduce an open-source LLM-based conversational health agent framework called openCHA, which collects necessary information through specific actions and generates personalized responses. MedAgents [318] propose a multidisciplinary collaboration (MC) framework where LLM-based agents engage in multi-round cooperative discussions through role-playing to enhance the model's expertise and reasoning capabilities.

For mental healthcare, Mental-LLM [85] presents a new framework for using LLMs to predict mental health from social media text data, with prompting-based and finetuning-based methods for real-time monitoring and prediction of psychological issues such as depression and anxiety. Lai et al. [319] introduce a psychological consultation aid called Psy-LLM, combining pre-trained LLMs with real psychologist Q&As and a large corpus of psychological articles.

For medication suggestions, Liu et al. [160] propose a framework called PharmacyGPT for generating personalized patient groups, formulating medication plans, and predicting patient outcomes.

Academic. In the academic domain, RevGAN [86] can automatically generate controllable and personalized user reviews based on users' emotional tendencies and stylistic information. For writing assistants, Porsdam et al. [320] explore the personalized enhancement of academic writing using LLMs like GPT-3 [63], showing higher quality in format, style, overall quality, and novelty after training with published academic works of three authors. Similarly, to address the lack of personalized outputs for author communication styles and expertise in current LLM outputs, Mysore et al. [161] propose Pearl, a personalized LLM writing assistant trained on data selected from users' historical documents to optimize personalized text generation; they also develop a new KL divergence training objective to help retrievers more accurately track document contributions to personalized generation.

Education. In the education domain, Cui et al. [321] propose a new adaptive and personalized exercise generation method that dynamically adjusts exercise difficulty to match students' learning progress by combining knowledge tracing and controlled text generation. EduChat [162] learns education-specific functionalities, such as article evaluation and emotional support, through pre-training on educational corpora and fine-tuning on customized instructions, addressing the issues of delayed knowledge updates and lack of educational expertise traditionally faced by LLMs in the educational sector.

Other Domains. For recipe generation tasks, traditional methods fail to consider users' personal tastes and unfamiliar dishes. To address this, Majumder et al. [322] propose a personalized generation model based on users' historical recipe consumption. Using an encoder-decoder structure, the model utilizes users' past recipe data to generate responses, enhancing the personalization and adaptability of the outputs. Subsequently, for personalized headline generation, Zhang et al. [323] simulate users' interests based on their news browsing history and generate news headlines based on these interests. Salemi et al. [301] propose the LaMP benchmark, which includes various personalized generation tasks such as personalized news headline generation, personalized academic title generation, personalized email subject generation, and personalized tweet rewriting. Additionally, for personalized assistance with home cleaning robots in determining the correct storage locations for items, TidyBot [324] proposes using LLMs to generalize from a small number of user-provided examples to infer broadly applicable user preference rules.

5 Evaluation

This section will provide a range of evaluation metrics and benchmarks for generative information retrieval methods, along with analysis and discussions on their performance.

5.1 Evaluation for Generative Document Retrieval

5.1.1 Metrics

In this section, we will discuss several core metrics for evaluating GR methods, including Recall [325], R-Precision [325], Mean Reciprocal Rank (MRR) [163], Mean Average Precision (MAP) [325], and Normalized Discounted Cumulative Gain (nDCG) [164]. These metrics provide different perspectives on the effectiveness of a GR system, including its accuracy, efficiency, and the relevance of its results.

Recall [325] is a metric that measures the proportion of relevant documents retrieved by the search system. For a given cutoff point k, the recall Recall@k is defined as:

$$\text{Recall@}k = \frac{1}{|Q|}\sum_{q=1}^{|Q|}\frac{ret_{q,k}}{rel_q}, \tag{8}$$

where $|Q|$ is the number of queries in the set, $ret_{q,k}$ is the number of relevant documents retrieved for the q-th query within the top k results, and rel_q is the total number of relevant documents for the q-th query.

R-Precision [325] measures the precision at the rank position R, which corresponds to the number of relevant documents for a given query q. It is calculated as:

$$\text{R-Precision} = \frac{ret_{q,R}}{rel_q}, \tag{9}$$

where $ret_{q,R}$ is the number of relevant documents retrieved within the top R positions, and R is equivalent to rel_q.

MRR [163] reflects the average rank position of the first relevant document returned in the search results. It is computed as follows:

$$\text{MRR} = \frac{1}{|Q|}\sum_{q=1}^{|Q|}\frac{1}{\text{rank}_q}, \tag{10}$$

where rank_q is the rank of the first relevant document returned for the q-th query.

MAP [325] calculates the average precision across multiple queries. It considers the exact position of all relevant documents and is calculated using the following formula:

$$\text{MAP} = \frac{1}{|Q|}\sum_{q=1}^{|Q|}\left(\frac{1}{rel_q}\sum_{k=1}^{n_q}\text{P@}k \times I(q,k)\right), \tag{11}$$

where P@k is the precision at cutoff k, and $I(q,k)$ is an indicator function that is 1 if the document at position k is relevant to the q-th query and 0 otherwise.

nDCG [164] takes into account not only the relevance of the documents returned but also their positions in the result list, which is defined by:

$$\text{DCG@}k = \sum_{i=1}^{k}\frac{2^{\text{rel}_i}-1}{\log_2(i+1)}, \tag{12}$$

$$\text{nDCG@}k = \frac{\text{DCG@}k}{\text{IDCG@}k}, \tag{13}$$

where rel_i represents the graded relevance of the i-th document, DCG@k is the discounted cumulative gain, and IDCG@k represents the maximum possible DCG@k.

5.1.2 Benchmarks

Evaluating the effectiveness of GR methods relies on high-quality and challenging benchmark datasets. Here are several benchmark datasets that are widely used in the field:

MS MARCO [165] (Microsoft Machine Reading Comprehension) is a large dataset for evaluating machine reading comprehension, retrieval, and question-answering capabilities in web search scenarios, containing two benchmarks: document ranking and passage ranking, with a total of 3.2 million documents and 8.8 million passages. It is compiled from real user queries extracted from Microsoft Bing's search logs, each accompanied by annotated relevant documents. This dataset covers a diverse range of question types and document genres, aiming to assess the performance of GR systems in complex web search scenarios.

NQ [166] (Natural Questions) is a question-answering dataset introduced by Google, using Wikipedia as the corpus, which includes 3.2 million documents, each document being a Wikipedia page. It contains numerous natural user queries and their corresponding answers extracted from web pages in Google search results, which could be utilzed to evaluate the retrieval performance of GR systems in addressing real-world questions.

TriviaQA [167] is a QA dataset that includes a large number of factual questions and answers from various sources, including Wikipedia and web-based QA forums, totaling about 0.66M evidence documents. TriviaQA aims to assess the generalization ability of information retrieval systems across different domains and question types.

KILT [168] (Knowledge Intensive Language Tasks) is a comprehensive benchmark dataset integrating 5 categories of knowledge-intensive tasks, including fact checking (FEVER [208]), entity linking (AIDA CoNLL-YAGO [209], WNED-WIKI [326], WNED-CWEB [326]), slot filling (T-REx [327], Zero Shot RE [211]), open-domain QA (Natural Questions [166], HotpotQA [328], TriviaQA [167], ELI5 [329]), and dialogue (Wizard of Wikipedia [210]). The KILT benchmark also uses Wikipedia as its corpus, comprising 5.9 million wiki pages, which aims to evaluate the effectiveness of information retrieval systems in handling complex language tasks that require extensive background knowledge.

TREC Deep Learning Track 2019 & 2020 [169, 170] focus on using deep learning to enhance the efficiency of information retrieval, with primary tasks including document and passage ranking. These evaluation campaigns use the MS MARCO dataset to simulate real-world search queries, providing a standardized environment for assessing different retrieval techniques.

It is noteworthy that, due to the challenge of indexing large-scale document corpus by GR models, some methods like [30–32, 87–89, 100] have adopted the approach of evaluating retrieval performance on smaller subsets of documents, ranging from 10K to 300K, along with their corresponding training and test pairs.

5.1.3 Analysis

In addition to the benchmarks and metrics for evaluating the performance of GR methods, there is a series of works that have conducted detailed analyses and discussions to study the behavior of GR models.

To understand the performance of DSI [30] in text retrieval, Chen et al. [89] examines uniqueness, completeness, and relevance ordering. These respectively reflect the system's ability to distinguish between different documents, retrieve all relevant documents, and accurately rank documents by relevance. Experimental analysis find that DSI excels in remembering the mapping from pseudo queries to DocIDs, indicating a strong capability to recall specific DocIDs from particular queries. However, the study also pointed out DSI's deficiency in distinguishing relevant documents from random ones, negatively impacting its retrieval effectiveness.

Later, Pradeep et al. [171] conduct the first comprehensive experimental study on GR techniques over large document sets, such as the 8.8M MS MARCO passages, which results in significant computational demands. It was found that among all the techniques examined, using generated pseudo queries to augment training data remains the only effective method on large document corpus. The strongest result in the experiments was achieved by using a training task that only utilized synthetic queries to Naive DocIDs, expanding the model to T5-XL (3B parameters) to achieve an MRR@10 of 26.7. Surprisingly, increasing the parameters to T5 XXL (11B) in the same setup did not improve performance but rather led to a decline. These findings suggest that more research and in-depth analysis are needed in the GR field, and possibly additional improvements to the paradigm, to fully leverage larger language models.

For out-of-distribution (OOD) robustness of GR models, Liu et al. [110] investigate three aspects: query variations, new query types, and new tasks. Their study showed that all types of retrieval models suffer from performance drops with query variations, indicating sensitivity to query quality and structure. However, when dealing with new query types and tasks, GR models showed different levels of adaptability, with pre-training enhancing their flexibility. The research highlights the critical need for OOD robustness in GR models for dealing with ever-changing real-world information sources.

5.2 Evaluation for Response Generation

5.2.1 Metrics

Evaluating the quality of generated responses involves multiple aspects, including accuracy, fluency, relevance, etc. In this section, we'll introduce the main metrics for evaluating reliable response generation, categorized into rule-based, model-based, and human evaluation metrics.

Rule-based Metrics. Exact Match (EM) is a straightforward evaluation method requiring the model's output to be completely identical to the reference answer at the word level. This full character-level matching is stringent, often used in tasks requiring precise and concise answers, such as question answering systems, e.g., NQ [166], TriviaQA [167], SQuAD [330], etc. It simply calculates the ratio of perfectly matched instances to the total number of instances.

For the generation of longer text sequences, BLEU [172] is a common metric initially used to evaluate the quality of machine translation. It compares the similarity between the model's output and a set of reference texts by calculating the overlap of n-grams, thereby deriving a score. This method assumes that high-quality generation should have a high lexical overlap with the labeled answer. Optimized from BLEU, METEOR [331] is an alignment-based metric that considers not only exact word matches but also synonyms and stem matches. Additionally, METEOR introduces considerations for word order and syntactic structure to better assess the fluency and consistency of the generated text.

ROUGE [173], also a commonly used metric for evaluating longer texts, by measuring the extent of overlap in words, sentences, n-grams, and so forth, between the generated text and a collection of reference texts. It focuses on recall, meaning it evaluates how much of the information in the reference text is covered by the generated text. ROUGE comes in various forms, including ROUGE-N, which evaluates based on n-gram overlap, and ROUGE-L, which considers the longest common subsequence, catering to diverse evaluation requirements.

Perplexity (PPL) is a metric for evaluating the performance of language models, defined as the exponentiation of the average negative log-likelihood, reflecting the model's average predictive ability for a given corpus of text sequences. The lower the perplexity, the stronger the model's predictive ability. Specifically, given a sequence of words $W = w_1, w_2, \ldots, w_N$, where N is the total number of words in the sequence, PPL can be expressed as:

$$\mathrm{PPL}(W) = \exp\left\{-\frac{1}{N}\sum_{i=1}^{N}\log p(w_i|w_{<i})\right\}, \tag{14}$$

where $p(w_i|w_{<i})$ represents the pre-trained language model's probability of predicting the i-th word w_i given the previous words $w_{<i}$.

Model-based Metrics. With the rise of pre-trained language models, a series of model-based evaluation metrics have emerged. These metrics utilize neural models to capture the deep semantic relationships between texts.

Unlike traditional rule-based metrics, BERTScore [174] utilizes the contextual embeddings of BERT [13] to capture the deep semantics of words, evaluating the similarity between candidate and reference sentences through the cosine similarity of embeddings. BERTScore employs a greedy matching strategy to optimize word-level matching and uses optional inverse document frequency weighting to emphasize important words, ultimately providing a comprehensive evaluation through a combination of recall, precision, and F1 score. BERTScore captures not only surface lexical overlap but also a deeper understanding of the semantic content of sentences.

Similarly based on BERT [13], BLEURT [175] designed multiple pre-training tasks, enhancing the model's ability to recognize textual differences with millions of synthetic training pairs. These pre-training tasks include automatic evaluation metrics (such as BLEU [172], ROUGE [173], and BERTScore [174]), back-translation likelihood, textual entailment, etc. Each task provides different signals to help the model learn how to evaluate the quality of text generation.

BARTScore [332], based on the pre-trained seq2seq generative model BART [18], treats the evaluation of generated text as a text generation problem. Specifically, BARTScore determines the quality of text based on the transition probability between the generated text and reference text. BARTScore does not require additional parameters or labeled data and can flexibly evaluate generated text from multiple perspectives (such as informativeness, fluency, factuality, etc.) and further enhance evaluation performance through text prompts or fine-tuning for specific tasks.

FActScore [177] focuses on the factual accuracy of each independent information point in long texts. It calculates a score representing factual accuracy by decomposing the text into atomic facts and verifying whether these facts are supported by reliable knowledge sources. This method provides a more detailed evaluation than traditional binary judgments and can be implemented efficiently and accurately through human evaluation and automated models (combining retrieval and powerful language models).

GPTScore [176] is a flexible, multi-faceted evaluation tool that allows users to evaluate text using natural language instructions without the need for complex training processes or costly annotations. GPTScore constructs an evaluation protocol dynamically through task specification and aspect definition and utilizes the zero-shot capability of pre-trained language models to evaluate text quality, optionally using demonstration samples to improve evaluation accuracy.

Human Evaluation Metrics. Human evaluation is an important method for assessing the performance of language models, especially in complex tasks where automated evaluation tools struggle to provide accurate assessments. Compared to rule-based and model-based metrics, human evaluation is more accurate and reliable in real-world applications. This evaluation method requires human evaluators (such as experts, researchers, or everyday users) to provide comprehensive assessments of the model-generated content based on their intuition and knowledge.

Human evaluation measures the quality of language model outputs by integrating multiple assessment criteria, following [333]: Accuracy [334] primarily evaluates the correctness of information and its correspondence with facts; Relevance [335] focuses on whether the model's output is pertinent to the specific context and user query; Fluency [336] examines whether the text is coherent, natural, and facilitates smooth communication with users; Safety [337] scrutinizes whether the content may lead to potential adverse consequences or harm. These indicators collectively provide a comprehensive assessment of the model's performance in real-world settings, ensuring its effectiveness and applicability.

5.2.2 Benchmarks and Analysis

In this section, we explore various benchmarks for evaluating the performance of language models in generating reliable responses. These benchmarks assess language understanding, factual accuracy, reliability, and the ability to provide timely information.

General Evaluation. To comprehensively assess the language models' understanding capabilities across a wide range of scenarios, MMLU [178] utilizes a multiple-choice format covering 57 different tasks, from basic mathematics to American history, computer science, and law. This benchmark spans evaluations in humanities, social science, and science, technology, engineering, and mathematics, providing a comprehensive and challenging test. It has been widely used in the evaluation of Large Language Models (LLMs) in recent years [23, 25, 65].

Furthermore, BIG-bench [179] introduces a large-scale and diverse benchmark designed to measure and understand the capabilities and limitations of LLMs across a broad range of tasks. Including 204 tasks contributed by 450 authors from 132 institutions, it covers areas such as linguistics, mathematics, and common sense reasoning. It focuses on tasks beyond the capabilities of language models, exploring how model performance and societal biases evolve with scale and complexity.

LLM-Eval [180] offers a unified multi-dimensional automatic evaluation method for open-domain dialogue of LLMs, eliminating the need for manual annotation. The performance of LLM-Eval across various datasets demonstrates its effectiveness, efficiency, and adaptability, improving over existing evaluation methods. The research also analyzes the impact of different LLMs and decoding strategies on the evaluation outcomes, underscoring the importance of selecting suitable LLMs and decoding strategies.

For Chinese, C-Eval [338] aims to comprehensively evaluate LLMs' advanced knowledge and reasoning capabilities in the Chinese context. It is based on a multiple-choice format, covering four difficulty levels and 52 different academic fields from secondary school to professional levels. C-Eval also introduces C-Eval Hard, a subset containing highly challenging subjects to test the models' advanced reasoning capabilities. Through evaluating state-of-the-art English and Chinese LLMs, C-Eval reveals areas where current models still fall short in handling complex tasks, guiding the development and optimization of Chinese LLMs.

Tool Evaluation. To assess the ability of language models to utilize tools, API-Bank [181] provides a comprehensive evaluation framework containing 73 APIs and 314 tool usage dialogs, along with a rich training dataset of 1,888 dialogs covering 1,000 domains to improve LLMs' tool usage capabilities. Experiments show that different LLMs perform variably in tool usage, highlighting their strengths and areas for improvement.

Later, ToolBench [143] developed a comprehensive framework including a dataset and evaluation tools to facilitate and assess the ability of LLMs to use over 16,000 real-world APIs. It enhances reasoning capabilities by automatically generating diverse instruction and API usage scenario paths, introducing a decision tree based on depth-first search. ToolBench significantly enhances LLMs' performance in executing complex instructions and in their ability to generalize to unseen APIs. ToolLLaMA, an LLM fine-tuned from LLaMA [65], exhibits remarkable zero-shot capabilities and performance comparable to state-of-the-art LLMs like ChatGPT [3].

Factuality Evaluation. TruthfulQA [182] measures the truthfulness of language models in answering questions. This benchmark consists of 817 questions covering 38 categories, including health, law, finance, and politics. This evaluation reveals that, even in optimal conditions, the truthfulness of model responses only reaches 58%, in stark contrast to human performance at 94%. Moreover, they proposed an automated evaluation metric named GPT-judge, which classifies the truthfulness of answers by fine-tuning the GPT-3 [63] model, achieving 90-96% accuracy in predicting human evaluations.

HaluEval [183] is a benchmark for evaluating LLM illusions, constructed using a dataset containing 35K illusion samples, employing a combination of automated generation and manual annotation. This provides effective tools and methods for assessing and enhancing large language models' capabilities in identifying and reducing illusions. For Chinese scenarios, HalluQA [339] designs 450 meticulously selected adversarial questions to assess the illusion phenomenon in Chinese LLMs, covering multiple domains and reflecting Chinese culture and history, identifying two main types of illusions: imitative falsehoods and factual errors.

To evaluate the ability of LLMs to generate answers with cited text, ALCE [153] builds an end-to-end system for retrieving relevant text passages and generating answers with citations. ALCE contains three datasets, covering different types of questions, and evaluates the generated text's quality from 'fluency', 'correctness', and 'citation quality' dimensions, combining human evaluation to verify the effectiveness of the evaluation metrics. The experimental results show that while LLMs excel at generating fluent text, there is significant room for improvement in ensuring content factual correctness and citation quality, especially on the ELI5 dataset where the best model was incomplete in citation support half of the time.

Real-Time Evaluation. RealTime QA [184] created a dynamic question-and-answer platform that regularly releases questions and evaluates systems weekly to ask and answer questions about the latest events or information. It challenges the static assumption of traditional QA datasets aiming for immediate application. Experiments based on LLMs like GPT-3 and T5 found that models could effectively update their generated results based on newly retrieved documents. However, when the retrieved documents failed to provide sufficient information, models tended to return outdated answers.

Furthermore, FreshQA [60] evaluates large language models' performance in challenges involving time-sensitive and erroneous premise questions by creating a new benchmark containing questions of this nature. Evaluating various open and closed-source LLMs revealed significant limitations in handling questions involving rapidly changing knowledge and erroneous premises. Based on these findings, the study proposed a simple in-context learning method, FreshPrompt, significantly improving LLMs' performance on FreshQA by integrating relevant and up-to-date information sourced from search engines into the prompt.

Safety, Ethic, and Trustworthiness. To comprehensively evaluate the safety of LLMs, SafetyBench [185] implements an efficient and accurate evaluation of LLMs' safety through 11,435 multiple-choice questions covering 7 safety categories in multiple languages (Chinese and English). The diversity of question types and the broad data sources ensure rigorous testing of LLMs in various safety-related scenarios. Comparing the performance of 25 popular LLMs, SafetyBench revealed GPT-4's significant advantage and pointed out the areas where current models need improvements in safety to promote the rapid development of safer LLMs.

For ethics, TrustGPT [61] aims to assess LLMs' ethical performance from toxicity, bias, and value alignment, three key dimensions. The benchmark uses predefined prompt templates based on social norms to guide LLMs in generating content and employs multiple metrics to quantitatively assess the toxicity, bias, and value consistency of these contents. Experimental analysis revealed that even the most advanced LLMs still have significant issues and potential risks in these ethical considerations.

For trustworthiness, TrustLLM [62] explores principles and benchmarks including truthfulness, safety, fairness, robustness, privacy, and machine ethics across six dimensions. Extensive experiments, including assessing 16 mainstream LLMs' performance on 30 datasets, found that trustworthiness usually positively correlates with functional effectiveness. While proprietary models typically outperform open-source models in trustworthiness, some open-source models like Llama2 showed comparable high performance.

These benchmarks provide important tools and metrics for evaluating and improving the capabilities of language models, contributing to the development of more accurate, reliable, safe, and timely GenIR systems. For further understanding of the evaluation works, [59, 333, 340] offer more detailed introductions.

6 Challenges and Prospects

This section discusses the key challenges faced in the fields of generative document retrieval and reliable response generation, as well as potential directions for future research.

6.1 Challenges on Generative Document Retrieval

6.1.1 Scalability Issues

As extensively studied by [171], generative retrieval demonstrates significantly lower retrieval accuracy compared to dense retrieval when handling million-level document corpora in web search scenarios. Merely increasing the model size does not yield stable performance improvements. However, GR outperforms dense retrieval in document collections smaller than 300K, posing a question: What impedes GR methods from scaling to large document sizes? This issue encompasses several aspects:

Training Data. Current LLMs are pre-trained on huge datasets ranging from hundreds of billions to several trillion tokens, covering vast knowledge sources such as the internet, books, and news articles, consuming substantial computational power [20]. They are then extensively fine-tuned with high-quality, human-annotated data to achieve substantial generalization capabilities [17, 18, 65, 192]. In contrast, generative retrieval (GR) models often begin with a pre-trained language model and are fine-tuned on labeled data comprising ¡query, DocID¿ pairs, which does not sufficiently prepare them to fully grasp GR tasks. For numeric-based DocIDs, the models, having not encountered these numbers in their pre-training phase, tend to rote memorize the DocIDs seen during training, struggling to predict unseen ones effectively. Similarly, if text-based DocIDs fail to precisely represent the documents, the model also tend to rote learning.

A potential solution is to create a large-scale pre-training dataset for generative retrieval on a general corpus, possibly including a variety of common DocIDs such as URLs, titles, and numerical sequences. We can utilize instructions to distinguish generation targets for various DocIDs. Then we can pre-train a Transformer-based GR model from scratch, the model can understand generative retrieval across diverse domains. This method could bridge the gap between language model pre-training data and GR tasks, enhancing the generalization ability of GR models across different corpora.

Training Method. As described in Section 3.1.1, existing training methods explore various training objectives, including seq2seq training, learning DocID, and ranking capabilities. Other methods involve knowledge distillation [200], reinforcement learning [34], etc. Is there a better training method to enable GR models to master generating DocID ranking lists? For example, RLHF [22] has been effectively used to train LLMs [23, 192], though at a high cost. Exploring RLHF in the GR field is also worthwhile.

Model Structure. As discussed in Section 3.1.2, most current GR models are based on encoder-decoder Transformers structures [30–32], such as T5 [17] and BART [18]. Some GR methods like CorpusLM [42] have experimented with a decoder-only structure of the LLM Llama2 [23], requiring more training computational power but not significantly improving performance. Research is needed to determine which structure is more suitable for generative retrieval. Additionally, whether increasing model and data size could lead to emergent phenomena similar to those observed in LLMs [341, 342] is also a promising research direction.

6.1.2 Handling Dynamic Corpora

Real-world applications often involve dynamically changing corpora, such as the web and news archives, where incremental learning is essential. However, for language models, indexing new documents inevitably leads to forgetting old ones, posing a challenge for GR systems. Existing methods like DSI++ [37], IncDSI [38], CLEVER [111], and CorpusBrain++ [39] propose solutions such as experience replay, constrained optimization, incremental product quantization, and continual generative pre-training frameworks to address incremental learning issues. Yet, these methods have their specific applicable scenarios, and more effective and universally applicable incremental learning strategies remain a key area for exploration.

6.1.3 Document Identifier

Accurately representing a document with high-quality DocIDs is crucial for generative retrieval.

For example, the KILT dataset based on the Wikipedia corpus, which includes 5.9 million documents, demonstrates optimistic retrieval performance for GR methods using titles as DocIDs [29, 41, 42]. This is because each document in Wikipedia has a unique manually annotated title that represents the core entity discussed in that page. However, in the web search scenario, such as in the MS MARCO dataset [165], many documents lack a unique title, are overlapping, and the titles do not accurately represent the core content of the documents. Thus, GR performance significantly declines in the MS MARCO corpus of 8.8 million passages.

Therefore, how to construct high-quality titles (or other types of DocIDs) in general corpora, similar to those in Wikipedia, that not only accurately represent documents but also are lightweight, is a critical factor for implementing GR methods and warrants in-depth research.

Text or Numeric? As discussed in Section 3.2, current methods include text-based and numeric-based DocIDs, each with their advantages and disadvantages. Text-based DocIDs effectively leverage the linguistic capabilities of pre-trained generative language models and offer better interpretability. Numeric-based DocIDs can utilize dense retriever embeddings to obtain semantic DocID sequences; they can also complement dense retrievers to achieve synergistic benefits.

However, to ensure good generalization ability of GR models without extensive pre-training, it is essential to utilize the inherent pre-trained parameters of the model. Coherent textual DocIDs can naturally leverage this aspect, but they also need to capture key document semantics and maintain linguistic sequence characteristics. Numeric DocIDs, however, do not offer this advantage. Thus, as mentioned in 6.1.1, extensive pre-training is necessary to enable models to fully understand the meanings behind these numerical strings, which is a costly endeavor.

Do We Need a Unique ID for Each Document? Most current GR methods use a unique DocID to uniquely identify a document. However, as the number of documents in a corpus increases, maintaining a unique DocID becomes increasingly challenging. Even if a unique DocID is maintained, it is difficult to differentiate significantly from other DocIDs semantically, leading to reduced retrieval precision. Some methods, such as using sub-string as DocIDs [102, 104], have proven effective. These methods utilize the FM-Index [204] to ensure the generated sub-string exists in the corpus and use the number of generated sub-strings in different documents to rank documents, demonstrating good performance and generalization ability.

However, since this method is based on FM-Index, its inference latency is high, which is an issue that needs addressing. Furthermore, exploring other more efficient alternatives to FM-Index and even considering not using constrained search but freely generating a DocID sequence followed by a lightweight matching and scoring module to efficiently return a document ranking list are also worthy of exploration.

6.1.4 Efficiency Concerns

Current GR methods generally rely on constrained beam search to generate multiple DocID sequences during inference, resulting in high latency. This is particularly severe when returning 100 or more documents, with latencies reaching several hundred milliseconds [31], which is unacceptable for low-latency IR systems. Therefore, designing more efficient inference methods is crucial. To reduce inference latency, the length of the DocID sequence should not be too long; 16 tokens or fewer is an efficient range. This necessitates designing DocIDs that are precise and concise enough to represent documents while maintaining performance and improving efficiency. Additionally, developing more efficient decoding strategies is a valuable research direction for the future.

6.2 Challenges on Reliable Response Generation

6.2.1 Improving Accuracy and Factuality

In GenIR systems, ensuring content accuracy and factuality is crucial. To achieve this, as mentioned in Section 4, there are two main areas of improvement:

Internal Knowledge Memorization. Firstly, training stronger generative models is critical for building reliable GenIR systems. Various commercial LLMs continue to progress, utilizing vast training data and computational resources, but exploring better model structures is also worthwhile. Recent research such as Retentive Networks [343], Mamba [344], and others, have shown potential to challenge the performance and efficiency of Transformers [198]. However, whether these can scale and truly surpass Transformer-based LLMs in generation quality is still an open question. Moreover, what types of training data and methods can consistently produce models capable of generating high-quality, reliable text also deserve thorough investigation and summary. The mechanisms by which language models recall knowledge during inference are not yet clear and need to be fully understood to better serve user information needs.

External Knowledge Enhancement. As described in Section 4.2.1, retrieval-augmented generation is an effective method widely applied in LLMs. However, there is still room for improvement. For example, whether inserting retrieved documents directly into generative models via prompts is the best method, or if there are better ways, such as inputting embeddings [345], needs exploration. Additionally, whether models can autonomously decide whether to perform retrieval [73, 346], and when in the generation process to perform it [77], are topics worth further exploration.

Tool-augmented generation, as discussed in Section 4.2.2, is also a popular method for endowing LLMs with fine-grained world knowledge and performing complex tasks. Recent research has raised questions, such as "Should tools always be used?" [347]. More specifically, whether the performance improvements brought by using tools justify the extra computational costs incurred during model training or the inference costs during testing. Existing work mainly focuses on task accuracy, but studying the cost-effectiveness of these methods is also a valuable topic.

6.2.2 *Real-time Properties of GenIR Systems*

Timeliness is critical for GenIR systems, as well as traditional IR systems, to provide users with the most up-to-date information. However, since the knowledge of pre-trained generative models is fixed after training, methods like retrieval and tool augmentation are needed to acquire new external knowledge. Research on real-time knowledge acquisition remains limited, making it a valuable area for investigation.

Moreover, continually relying on outdated knowledge from language models is inadequate, as models cannot comprehend the significance of given contexts or backgrounds in the current era, thus reducing the reliability of the generated content. Therefore, updating the information in language models while avoiding the forgetting of existing knowledge, such as through continual learning [240, 348], knowledge editing [242, 252, 349, 350], etc., is a topic worth further exploring.

6.2.3 *Bias and Fairness*

Since LLMs are often trained on large, unfiltered datasets, GenIR systems may propagate stereotypes and biases present in the data regarding race, culture, and other aspects [351]. Researchers have explored various methods to enhance the fairness of generated content during training data selection, training methods, generation techniques, and rewriting phases. However, biases have not been eradicated and require a thorough understanding of the mechanisms by which generative models produce biases, to design methods to solve them and build fair GenIR systems that further the practical application of GenIR.

6.2.4 *Privacy and Security*

Firstly, the content generated by GenIR systems risks plagiarism [352, 353]. For instance, studies such as [354, 355] indicate that pre-trained language models can reproduce large segments of their training data, leading to inadvertent plagiarism and causing academic dishonesty or copyright issues. On one hand, legal regulations regarding the copyright of AI-generated content will gradually emerge and evolve. On the other hand, technical research aimed at reducing plagiarism by generative models, such as generating text with correct citations [16, 285, 356], is a promising research direction for reliable GenIR that has received increasing attention in recent years.

Moreover, due to the unclear mechanisms of memory and generation in pre-trained language models, GenIR systems inevitably return unsafe content. For example, [354, 357] show that when attacked, LLMs may return private information of users seen in training data. Therefore, understanding the mechanisms by which LLMs recall training data and designing effective defense mechanisms to enhance security are crucial for the widespread use of GenIR systems. Additionally, developing effective detection methods for content generated by LLMs is essential for enhancing the security of GenIR systems [358].

6.3 Unified Framework

This article discusses two mainstream forms of GenIR: generative document retrieval and reliable response generation. However, each approach has its advantages and limitations. Generative document retrieval still returns a list of documents, whereas the reliable response generation model itself cannot effectively capture document-level relationships. Therefore, integrating these two approaches is a promising research direction.

6.3.1 *Unified Framework for Retrieval and Generation*

Given that both generative retrieval and downstream generation tasks can be based on generative language models, could a single model perform both retrieval and generation tasks? Indeed, it could.

Current attempts, such as UniGen [117], use a shared encoder and two decoders for GR and QA tasks respectively, and show superior performance on small-scale retrieval and QA datasets. However, they struggle to generalize across multiple downstream tasks and to integrate with powerful LLMs. Additionally, CorpusLM [42] uses a multi-task training approach to obtain a universal model for GR, QA, and RAG. Yet, merely merging training data does not significantly improve retrieval and generation performance, and CorpusLM remains limited to the Wikipedia corpus. Facing a broader internet corpus presents significant challenges.

In the future, can we construct a large search model (LSM) that allows an LLM to have the capability to generate DocIDs and reliable responses autonomously? Even LSM could decide when to generate DocIDs to access the required knowledge before continuing generation. Unlike the large search model defined in [359], which unifies models beyond the first-stage retrieval (such as re-ranking, snippet, and answer models), we aim to integrate the first-stage retrieval as well, enabling the LSM to fully understand the meaning of retrieval and its connection with various downstream generation tasks.

6.3.2 Towards End-to-End Framework for Various IR Tasks

As envisioned by Metzler et al. [16], propose an expert-level corpus model that not only possesses linguistic capabilities but also understands document-level DocIDs and knows the sources of its own knowledge. Such a model could not only solve the issue of hallucinations common in traditional language models but could also generate texts with references pointing to the source documents, thus achieving a reliable end-to-end GenIR model. By understanding DocIDs and knowledge sources, this end-to-end system could also perform additional IR tasks, such as returning the main content of a document given its DocID or returning other related document DocIDs, as well as enabling multi-lingual and multi-modal retrieval.

Current methods, as discussed in this GenIR survey, primarily focus on generative document retrieval (GR) and response generation as separate entities. GR models excel at comprehending document identifiers at the document-level, while downstream models demonstrate powerful task generation capabilities. However, existing methods face challenges when it comes to effectively integrating these two generative abilities, limiting the overall performance and effectiveness of the GenIR system. The integration of these generative abilities in a seamless and efficient manner remains a key challenge in the field.

In the future, we can design training methods that align knowledge and DocIDs and construct high-quality training datasets for generating answers with references, to train such an end-to-end GenIR model. Achieving this goal remains challenging and requires the collaborative efforts of researchers to contribute to building the next generation of GenIR systems.

7 Conclusion

In this survey, we explore the latest research developments, evaluations, current challenges, and future directions in generative information retrieval (GenIR). We discuss two main directions in the GenIR field: generative document retrieval (GR) and reliable response generation. Specifically, we systematically review the progress of GR covering model training, document identifier design, incremental learning, adaptability to downstream tasks, multi-modal GR, and generative recommendation systems; as well as advancements in reliable response generation in terms of internal knowledge memorization, external knowledge enhancement, generating responses with citations, and personal information assistance. Additionally, we have sorted out the existing evaluation methods and benchmarks for GR and response generation. We organize the current limitations and future directions of GR systems, addressing scalability, handling dynamic corpora, document representation, and efficiency challenges. Furthermore, we identify challenges in reliable response generation, such as accuracy, real-time capabilities, bias and fairness, privacy, and security. We propose potential solutions and future research directions to tackle these challenges. Finally, we also envision a unified framework, including unified retrieval and generation tasks, and even building an end-to-end framework capable of handling various information retrieval tasks. Through this review, we hope to provide a comprehensive reference for researchers in the GenIR field to further promote the development of this area.

References

[1] Google, "Google," *https://www.google.com*, 2023.

[2] Microsoft, "Bing," *https://www.bing.com*, 2023.

[3] OpenAI, "Introducing chatgpt," *https://openai.com/blog/chatgpt*, 2022.

[4] Microsoft, "Bing chat," *https://www.bing.com/new*, 2023.

[5] Amazon, "Amazon," *https://www.amazon.com*, 2023.

[6] Google, "Youtube," *https://www.youtube.com*, 2023.

[7] G. Salton, E. A. Fox, and H. Wu, "Extended boolean information retrieval," *Commun. ACM*, vol. 26, no. 11, pp. 1022–1036, 1983. [Online]. Available: https://doi.org/10.1145/182.358466

[8] S. E. Robertson and H. Zaragoza, "The probabilistic relevance framework: BM25 and beyond," *Found. Trends Inf. Retr.*, vol. 3, no. 4, pp. 333–389, 2009. [Online]. Available: https://doi.org/10.1561/1500000019

[9] T. Formal, B. Piwowarski, and S. Clinchant, "SPLADE: sparse lexical and expansion model for first stage ranking," in *SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021*, F. Diaz, C. Shah, T. Suel, P. Castells, R. Jones, and T. Sakai, Eds. ACM, 2021, pp. 2288–2292. [Online]. Available: https://doi.org/10.1145/3404835.3463098

[10] J. Lin and X. Ma, "A few brief notes on deepimpact, coil, and a conceptual framework for information retrieval techniques," *CoRR*, vol. abs/2106.14807, 2021. [Online]. Available: https://arxiv.org/abs/2106.14807

[11] V. Karpukhin, B. Oguz, S. Min, P. Lewis, L. Wu, S. Edunov, D. Chen, and W.-t. Yih, "Dense passage retrieval for open-domain question answering," in *EMNLP*, 2020, pp. 6769–6781.

[12] L. Xiong, C. Xiong, Y. Li, K.-F. Tang, J. Liu, P. N. Bennett, J. Ahmed, and A. Overwijk, "Approximate nearest neighbor negative contrastive learning for dense text retrieval," in *ICLR*, 2020.

[13] J. D. M.-W. C. Kenton and L. K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *NAACL-HLT*, 2019, pp. 4171–4186.

[14] M. Douze, A. Guzhva, C. Deng, J. Johnson, G. Szilvasy, P. Mazaré, M. Lomeli, L. Hosseini, and H. Jégou, "The faiss library," *CoRR*, vol. abs/2401.08281, 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2401.08281

[15] Y. A. Malkov and D. A. Yashunin, "Efficient and robust approximate nearest neighbor search

using hierarchical navigable small world graphs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 4, pp. 824–836, 2020. [Online]. Available: https://doi.org/10.1109/TPAMI.2018.2889473

[16] D. Metzler, Y. Tay, D. Bahri, and M. Najork, "Rethinking search: making domain experts out of dilettantes," in *ACM SIGIR Forum*, vol. 55, no. 1. ACM New York, NY, USA, 2021, pp. 1–27.

[17] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *J. Mach. Learn. Res.*, vol. 21, pp. 140:1–140:67, 2020. [Online]. Available: http://jmlr.org/papers/v21/20-074.html

[18] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer, "Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," in *ACL*, 2020, pp. 7871–7880.

[19] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever *et al.*, "Improving language understanding by generative pre-training," 2018.

[20] W. X. Zhao, K. Zhou, J. Li, T. Tang, X. Wang, Y. Hou, Y. Min, B. Zhang, J. Zhang, Z. Dong, Y. Du, C. Yang, Y. Chen, Z. Chen, J. Jiang, R. Ren, Y. Li, X. Tang, Z. Liu, P. Liu, J. Nie, and J. Wen, "A survey of large language models," *CoRR*, vol. abs/2303.18223, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2303.18223

[21] Y. Cao, S. Li, Y. Liu, Z. Yan, Y. Dai, P. S. Yu, and L. Sun, "A comprehensive survey of ai-generated content (AIGC): A history of generative AI from GAN to chatgpt," *CoRR*, vol. abs/2303.04226, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2303.04226

[22] P. F. Christiano, J. Leike, T. B. Brown, M. Martic, S. Legg, and D. Amodei, "Deep reinforcement learning from human preferences," in *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan, and R. Garnett, Eds., 2017, pp. 4299–4307. [Online]. Available: https://proceedings.neurips.cc/paper/2017/hash/d5e2c0adad503c91f91df240d0cd4e49-Abstract.html

[23] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale *et al.*, "Llama 2: Open foundation and fine-tuned chat models," *arXiv preprint arXiv:2307.09288*, 2023.

[24] J. Bai, S. Bai, Y. Chu, Z. Cui, K. Dang, X. Deng, Y. Fan, W. Ge, Y. Han, F. Huang, B. Hui, L. Ji, M. Li, J. Lin, R. Lin, D. Liu, G. Liu, C. Lu, K. Lu, J. Ma, R. Men, X. Ren, X. Ren, C. Tan, S. Tan, J. Tu, P. Wang, S. Wang, W. Wang, S. Wu, B. Xu, J. Xu, A. Yang, H. Yang, J. Yang, S. Yang, Y. Yao, B. Yu, H. Yuan, Z. Yuan, J. Zhang, X. Zhang, Y. Zhang, Z. Zhang, C. Zhou, J. Zhou, X. Zhou, and T. Zhu, "Qwen technical report," *CoRR*, vol. abs/2309.16609, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2309.16609

[25] A. Q. Jiang, A. Sablayrolles, A. Mensch, C. Bamford, D. S. Chaplot, D. de Las Casas, F. Bressand, G. Lengyel, G. Lample, L. Saulnier, L. R. Lavaud, M. Lachaux, P. Stock, T. L. Scao, T. Lavril, T. Wang, T. Lacroix, and W. E. Sayed, "Mistral 7b," *CoRR*, vol. abs/2310.06825, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2310.06825

[26] R. Thoppilan, D. D. Freitas, J. Hall, N. Shazeer, A. Kulshreshtha, H. Cheng, A. Jin, T. Bos, L. Baker, Y. Du, Y. Li, H. Lee, H. S. Zheng, A. Ghafouri, M. Menegali, Y. Huang, M. Krikun, D. Lepikhin, J. Qin, D. Chen, Y. Xu, Z. Chen, A. Roberts, M. Bosma, Y. Zhou, C. Chang, I. Krivokon, W. Rusch, M. Pickett, K. S. Meier-Hellstern, M. R. Morris, T. Doshi, R. D. Santos, T. Duke, J. Soraker, B. Zevenbergen, V. Prabhakaran, M. Diaz, B. Hutchinson, K. Olson, A. Molina, E. Hoffman-John, J. Lee, L. Aroyo, R. Rajakumar, A. Butryna, M. Lamm, V. Kuzmina, J. Fenton, A. Cohen, R. Bernstein, R. Kurzweil, B. A. y Arcas, C. Cui, M. Croak, E. H. Chi, and Q. Le, "Lamda: Language models for dialog applications," *CoRR*, vol. abs/2201.08239, 2022. [Online]. Available: https://arxiv.org/abs/2201.08239

[27] X. Liu, H. Lai, H. Yu, Y. Xu, A. Zeng, Z. Du, P. Zhang, Y. Dong, and J. Tang, "Webglm: Towards an efficient web-enhanced question answering system with human preferences," in *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD 2023, Long Beach, CA, USA, August 6-10, 2023*, A. K. Singh, Y. Sun, L. Akoglu, D. Gunopulos, X. Yan, R. Kumar, F. Ozcan, and J. Ye, Eds. ACM, 2023, pp. 4549–4560. [Online]. Available: https://doi.org/10.1145/3580305.3599931

[28] Y. Qin, Z. Cai, D. Jin, L. Yan, S. Liang, K. Zhu, Y. Lin, X. Han, N. Ding, H. Wang, R. Xie, F. Qi, Z. Liu, M. Sun, and J. Zhou, "WebCPM: Interactive web search for Chinese long-form question answering," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, A. Rogers, J. Boyd-Graber, and N. Okazaki, Eds. Toronto, Canada: Association for Computational Linguistics, Jul. 2023, pp. 8968–8988. [Online]. Available: https://aclanthology.org/2023.acl-long.499

[29] N. D. Cao, G. Izacard, S. Riedel, and F. Petroni, "Autoregressive entity retrieval," in *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. [Online]. Available: https://openreview.net/forum?id=5k8F6UU39V

[30] Y. Tay, V. Tran, M. Dehghani, J. Ni, D. Bahri, H. Mehta, Z. Qin, K. Hui, Z. Zhao, J. P. Gupta, T. Schuster, W. W. Cohen, and D. Metzler, "Transformer memory as a differentiable search index," in *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, Eds., 2022. [Online]. Available: http://papers.nips.cc/paper_files/paper/2022/hash/892840a6123b5ec99ebaab8be1530fba-Abstract-Conference.html

[31] Y. Wang, Y. Hou, H. Wang, Z. Miao, S. Wu, H. Sun, Q. Chen, Y. Xia, C. Chi, G. Zhao, Z. Liu, X. Xie, H. A. Sun, W. Deng, Q. Zhang, and M. Yang, "A neural corpus indexer for document retrieval," *CoRR*, vol. abs/2206.02743, 2022.

[32] Y. Zhou, J. Yao, Z. Dou, L. Wu, P. Zhang, and J. Wen, "Ultron: An ultimate retriever on corpus with a model-based indexer," *CoRR*, vol. abs/2208.09257, 2022.

[33] Y. Zhou, J. Yao, L. Wu, Z. Dou, and J.-R. Wen, "Webultron: An ultimate retriever on webpages under the model-centric paradigm," *IEEE Transactions on Knowledge and Data Engineering*, 2023.

[34] Y. Zhou, Z. Dou, and J. Wen, "Enhancing generative retrieval with reinforcement learning from relevance feedback," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, H. Bouamor, J. Pino, and K. Bali, Eds. Association for Computational Linguistics, 2023, pp. 12 481–12 490. [Online]. Available: https://aclanthology.org/2023.emnlp-main.768

[35] W. Sun, L. Yan, Z. Chen, S. Wang, H. Zhu, P. Ren, Z. Chen, D. Yin, M. de Rijke, and Z. Ren, "Learning to tokenize for generative retrieval," *CoRR*, vol. abs/2304.04171, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2304.04171

[36] T. Yang, M. Song, Z. Zhang, H. Huang, W. Deng, F. Sun, and Q. Zhang, "Auto search indexer for end-to-end document retrieval," in *Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023*, H. Bouamor, J. Pino, and K. Bali, Eds. Association for Computational Linguistics, 2023, pp. 6955–6970. [Online]. Available: https://aclanthology.org/2023.findings-emnlp.464

[37] S. V. Mehta, J. Gupta, Y. Tay, M. Dehghani, V. Q. Tran, J. Rao, M. Najork, E. Strubell, and D. Metzler, "Dsi++: Updating transformer memory with new documents," *arXiv preprint arXiv:2212.09744*, 2022.

[38] V. Kishore, C. Wan, J. Lovelace, Y. Artzi, and K. Q. Weinberger, "Incdsi: Incrementally updatable document retrieval," in *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, ser. Proceedings of Machine Learning Research, A. Krause, E. Brunskill, K. Cho, B. Engelhardt, S. Sabato, and J. Scarlett, Eds., vol. 202. PMLR, 2023, pp. 17 122–17 134. [Online]. Available: https://proceedings.mlr.press/v202/kishore23a.html

[39] J. Guo, C. Zhou, R. Zhang, J. Chen, M. de Rijke, Y. Fan, and X. Cheng, "Corpusbrain++: A continual generative pre-training framework for knowledge-intensive language tasks," *arXiv preprint arXiv:2402.16767*, 2024.

[40] J. Chen, R. Zhang, J. Guo, Y. Fan, and X. Cheng, "Gere: Generative evidence retrieval for fact verification," *arXiv preprint arXiv:2204.05511*, 2022.

[41] J. Chen, R. Zhang, J. Guo, Y. Liu, Y. Fan, and X. Cheng, "Corpusbrain: Pre-train a generative retrieval model for knowledge-intensive language tasks," in *Proceedings of the 31st ACM International Conference on Information & Knowledge Management, Atlanta, GA, USA, October 17-21, 2022*, M. A. Hasan and L. Xiong, Eds. ACM, 2022, pp. 191–200. [Online]. Available: https://doi.org/10.1145/3511808.3557271

[42] X. Li, Z. Dou, Y. Zhou, and F. Liu, "Towards a unified language model for knowledge-intensive tasks utilizing external corpus," *CoRR*, vol. abs/2402.01176, 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2402.01176

[43] Y. Zhang, T. Zhang, D. Chen, Y. Wang, Q. Chen, X. Xie, H. Sun, W. Deng, Q. Zhang, F. Yang, M. Yang, Q. Liao, and B. Guo, "Irgen: Generative modeling for image retrieval," *CoRR*, vol. abs/2303.10126, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2303.10126

[44] X. Long, J. Zeng, F. Meng, Z. Ma, K. Zhang, B. Zhou, and J. Zhou, "Generative multi-modal knowledge retrieval with large language models," *CoRR*, vol. abs/2401.08206, 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2401.08206

[45] Y. Li, W. Wang, L. Qu, L. Nie, W. Li, and T. Chua, "Generative cross-modal retrieval: Memorizing images in multimodal language models for retrieval and beyond," *CoRR*, vol. abs/2402.10805, 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2402.10805

[46] S. Geng, S. Liu, Z. Fu, Y. Ge, and Y. Zhang, "Recommendation as language processing (RLP): A unified pretrain, personalized prompt & predict paradigm (P5)," in *RecSys '22: Sixteenth ACM Conference on Recommender Systems, Seattle, WA, USA, September 18 - 23, 2022*, J. Golbeck, F. M. Harper, V. Murdock, M. D. Ekstrand, B. Shapira, J. Basilico, K. T. Lundgaard, and E. Oldridge, Eds. ACM, 2022, pp. 299–315. [Online]. Available: https://doi.org/10.1145/3523227.3546767

[47] W. Wang, X. Lin, F. Feng, X. He, and T. Chua, "Generative recommendation: Towards next-generation recommender paradigm," *CoRR*, vol. abs/2304.03516, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2304.03516

[48] S. Rajput, N. Mehta, A. Singh, R. H. Keshavan, T. Vu, L. Heldt, L. Hong, Y. Tay, V. Q. Tran, J. Samost, M. Kula, E. H. Chi, and M. Sathiamoorthy, "Recommender systems with generative retrieval," in *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, Eds., 2023. [Online]. Available: http://papers.nips.cc/paper_files/paper/2023/hash/20dcab0f14046a5c6b02b61da9f13229-Abstract-Conference.html

[49] G. Bénédict, R. Zhang, and D. Metzler, "Gen-ir@ sigir 2023: The first workshop on generative information retrieval," in *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2023, pp. 3460–3463.

[50] Y. Fan, Y. Tang, J. Chen, R. Zhang, and J. Guo, "生成式信息检索前沿进展与挑战 (challenges and advances in generative information retrieval)," in *Proceedings of the 22nd Chinese National Conference on Computational Linguistics (Volume 2: Frontier Forum)*, 2023, pp. 57–66.

[51] Y. Tang, R. Zhang, J. Guo, and M. de Rijke, "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 2023, Beijing, China, November 26-28, 2023*, Q. Ai, Y. Liu, A. Moffat, X. Huang, T. Sakai, and J. Zobel, Eds. ACM, 2023, pp. 294–297. [Online]. Available: https://doi.org/10.1145/3624918.3629547

[52] R. Nakano, J. Hilton, S. Balaji, J. Wu, L. Ouyang, C. Kim, C. Hesse, S. Jain, V. Kosaraju, W. Saunders, X. Jiang, K. Cobbe, T. Eloundou, G. Krueger, K. Button, M. Knight, B. Chess, and J. Schulman, "Webgpt: Browser-assisted question-answering with human feedback," *CoRR*, vol. abs/2112.09332, 2021. [Online]. Available: https://arxiv.org/abs/2112.09332

[53] H. Qian, Y. Zhu, Z. Dou, H. Gu, X. Zhang, Z. Liu, R. Lai, Z. Cao, J. Nie, and J. Wen, "Webbrain: Learning to generate factually correct articles for queries by grounding on large web corpus," *CoRR*, vol. abs/2304.04358, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2304.04358

[54] M. Sallam, N. A. Salim, A. B. Al-Tammemi, M. M. Barakat, D. Fayyad, S. Hallit, H. Harapan, R. Hallit, and A. Mahafzah, "Chatgpt output regarding compulsory vaccination and covid-19 vaccine conspiracy: A descriptive study at the outset of a paradigm shift in online search for information," *Cureus*, vol. 15, 2023. [Online]. Available: https://api.semanticscholar.org/CorpusID:256897987

[55] L. Gienapp, H. Scells, N. Deckers, J. Bevendorff, S. Wang, J. Kiesel, S. Syed, M. Frobe, G. Zucoon, B. Stein, M. Hagen, and M. Potthast, "Evaluating generative ad hoc information retrieval," *ArXiv*, vol. abs/2311.04694, 2023. [Online]. Available: https://api.semanticscholar.org/CorpusID:265050661

[56] R. W. White, "Tasks, copilots, and the future of search: A keynote at SIGIR 2023," *SIGIR Forum*, vol. 57, no. 2, pp. 4:1–4:8, 2023. [Online]. Available: https://doi.org/10.1145/3642979.3642985

[57] W. R. Hersh, "Search still matters: Information retrieval in the era of generative AI," *CoRR*, vol. abs/2311.18550, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2311.18550

[58] Z. Ji, N. Lee, R. Frieske, T. Yu, D. Su, Y. Xu, E. Ishii, Y. J. Bang, A. Madotto, and P. Fung, "Survey of hallucination in natural language generation," *ACM Computing Surveys*, vol. 55, no. 12, pp. 1–38, 2023.

[59] L. Huang, W. Yu, W. Ma, W. Zhong, Z. Feng, H. Wang, Q. Chen, W. Peng, X. Feng, B. Qin, and T. Liu, "A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions," *CoRR*, vol. abs/2311.05232, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2311.05232

[60] T. Vu, M. Iyyer, X. Wang, N. Constant, J. W. Wei, J. Wei, C. Tar, Y. Sung, D. Zhou, Q. V. Le, and T. Luong, "Freshllms: Refreshing large language models with search engine augmentation," *CoRR*, vol. abs/2310.03214, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2310.03214

[61] Y. Huang, Q. Zhang, P. S. Yu, and L. Sun, "Trustgpt: A benchmark for trustworthy and responsible large language models," *CoRR*, vol. abs/2306.11507, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2306.11507

[62] L. Sun, Y. Huang, H. Wang, S. Wu, Q. Zhang, C. Gao, Y. Huang, W. Lyu, Y. Zhang, X. Li, Z. Liu, Y. Liu, Y. Wang, Z. Zhang, B. Kailkhura, C. Xiong, C. Xiao, C. Li, E. P. Xing, F. Huang, H. Liu, H. Ji, H. Wang, H. Zhang, H. Yao, M. Kellis, M. Zitnik, M. Jiang, M. Bansal, J. Zou, J. Pei, J. Liu, J. Gao, J. Han, J. Zhao, J. Tang, J. Wang, J. Mitchell, K. Shu, K. Xu, K. Chang, L. He, L. Huang, M. Backes, N. Z. Gong, P. S. Yu, P. Chen, Q. Gu, R. Xu, R. Ying, S. Ji, S. Jana, T. Chen, T. Liu, T. Zhou, W. Wang, X. Li, X. Zhang, X. Wang, X. Xie, X. Chen, X. Wang, Y. Liu, Y. Ye, Y. Cao, and Y. Zhao, "Trustllm: Trustworthiness in large language models," *CoRR*, vol. abs/2401.05561, 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2401.05561

[63] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell *et al.*, "Language models are few-shot learners," *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020.

[64] T. L. Scao, A. Fan, C. Akiki, E. Pavlick, S. Ilic, D. Hesslow, R. Castagné, A. S. Luccioni, F. Yvon, M. Gallé, J. Tow, A. M. Rush, S. Biderman, A. Webson, P. S. Ammanamanchi, T. Wang, B. Sagot, N. Muennighoff, A. V. del Moral, O. Ruwase, R. Bawden, S. Bekman, A. McMillan-Major, I. Beltagy, H. Nguyen, L. Saulnier, S. Tan, P. O. Suarez, V. Sanh, H. Laurençon, Y. Jernite, J. Launay, M. Mitchell, C. Raffel, A. Gokaslan, A. Simhi, A. Soroa, A. F. Aji, A. Alfassy, A. Rogers, A. K. Nitzav, C. Xu, C. Mou, C. Emezue, C. Klamm, C. Leong, D. van Strien, D. I. Adelani, and et al., "BLOOM: A 176b-parameter open-access multilingual language model," *CoRR*, vol. abs/2211.05100, 2022. [Online]. Available: https://doi.org/10.48550/arXiv.2211.05100

[65] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar *et al.*, "Llama: Open and efficient foundation language models," *arXiv preprint arXiv:2302.13971*, 2023.

[66] K. Lee, D. Ippolito, A. Nystrom, C. Zhang, D. Eck, C. Callison-Burch, and N. Carlini, "Deduplicating training data makes language models better," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, S. Muresan, P. Nakov, and A. Villavicencio, Eds. Association for Computational Linguistics, 2022, pp. 8424–8445. [Online]. Available: https://doi.org/10.18653/v1/2022.acl-long.577

[67] Q. Dong, D. Dai, Y. Song, J. Xu, Z. Sui, and L. Li, "Calibrating factual knowledge in pretrained language models," in *Findings of the Association for Computational Linguistics: EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, Y. Goldberg, Z. Kozareva, and Y. Zhang, Eds. Association for Computational Linguistics, 2022, pp. 5937–5947. [Online]. Available: https://doi.org/10.18653/v1/2022.findings-emnlp.438

[68] Y. Chuang, Y. Xie, H. Luo, Y. Kim, J. R. Glass, and P. He, "Dola: Decoding by contrasting layers improves factuality in large language models," *CoRR*, vol. abs/2309.03883, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2309.03883

[69] Y. Sun, S. Wang, Y. Li, S. Feng, H. Tian, H. Wu, and H. Wang, "Ernie 2.0: A continual pre-training framework for language understanding," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, no. 05, 2020, pp. 8968–8975.

[70] Z. Ke, Y. Shao, H. Lin, T. Konishi, G. Kim, and B. Liu, "Continual pre-training of language models," *arXiv preprint arXiv:2302.03241*, 2023.

[71] K. Meng, D. Bau, A. Andonian, and Y. Belinkov, "Locating and editing factual associations in GPT," in *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, Eds., 2022. [Online]. Available: http://papers.nips.cc/paper_files/paper/2022/hash/6f1d43d5a82a37e89b0665b33bf3a182-Abstract-Conference.html

[72] P. S. H. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W. Yih, T. Rocktäschel, S. Riedel, and D. Kiela, "Retrieval-augmented generation for knowledge-intensive NLP tasks," in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., 2020. [Online]. Available: https://proceedings.neurips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html

[73] H. Wang, B. Xue, B. Zhou, T. Zhang, C. Wang, G. Chen, H. Wang, and K. fai Wong, "Self-dc: When to retrieve and when to generate? self divide-and-conquer for compositional unknown questions," 2024.

[74] A. Asai, Z. Wu, Y. Wang, A. Sil, and H. Hajishirzi, "Self-rag: Learning to retrieve, generate, and critique through self-reflection," *CoRR*, vol. abs/2310.11511, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2310.11511

[75] S. Yao, J. Zhao, D. Yu, I. Shafran, K. R. Narasimhan, and Y. Cao, "React: Synergizing reasoning and acting in language models," in *NeurIPS 2022 Foundation Models for Decision Making Workshop*, 2022. [Online]. Available: https://openreview.net/forum?id=tvI4u1ylcqs

[76] J. Jiang, K. Zhou, Z. Dong, K. Ye, X. Zhao, and J.-R. Wen, "StructGPT: A general framework for large language model to reason over structured data," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, H. Bouamor, J. Pino, and K. Bali, Eds. Singapore: Association for Computational Linguistics, Dec. 2023, pp. 9237–9251. [Online]. Available: https://aclanthology.org/2023.emnlp-main.574

[77] Z. Jiang, F. F. Xu, L. Gao, Z. Sun, Q. Liu, J. Dwivedi-Yu, Y. Yang, J. Callan, and G. Neubig, "Active retrieval augmented generation," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, H. Bouamor, J. Pino, and K. Bali, Eds. Association for Computational Linguistics, 2023, pp. 7969–7992. [Online]. Available: https://aclanthology.org/2023.emnlp-main.495

[78] T. Schick, J. Dwivedi-Yu, R. Dessì, R. Raileanu, M. Lomeli, L. Zettlemoyer, N. Cancedda, and T. Scialom, "Toolformer: Language models can teach themselves to use tools," *arXiv preprint arXiv:2302.04761*, 2023.

[79] O. Weller, M. Marone, N. Weir, D. J. Lawrie, D. Khashabi, and B. V. Durme, ""according to . . . ": Prompting language models improves quoting from pre-training data," in *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2024 - Volume 1: Long Papers, St. Julian's, Malta, March 17-22, 2024*, Y. Graham and M. Purver, Eds. Association for Computational Linguistics, 2024, pp. 2288–2301. [Online]. Available: https://aclanthology.org/2024.eacl-long.140

[80] D. Lee, T. Whang, C. Lee, and H. Lim, "Towards reliable and fluent large language models: Incorporating feedback learning loops in QA systems," *CoRR*, vol. abs/2309.06384, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2309.06384

[81] X. Li, C. Zhu, L. Li, Z. Yin, T. Sun, and X. Qiu, "Llatrieval: Llm-verified retrieval for verifiable generation," *CoRR*, vol. abs/2311.07838, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2311.07838

[82] D. Li, Z. Sun, B. Hu, Z. Liu, X. Hu, X. Liu, and M. Zhang, "Improving attributed text generation of large language models via preference learning," *CoRR*, vol. abs/2403.18381, 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2403.18381

[83] Q. Liu, Y. Chen, B. Chen, J. Lou, Z. Chen, B. Zhou, and D. Zhang, "You impress me: Dialogue generation via mutual persona perception," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, D. Jurafsky, J. Chai, N. Schluter, and J. R. Tetreault, Eds. Association for Computational Linguistics, 2020, pp. 1417–1427. [Online]. Available: https://doi.org/10.18653/v1/2020.acl-main.131

[84] H. Wang, M. Hu, Y. Deng, R. Wang, F. Mi, W. Wang, Y. Wang, W. Kwan, I. King, and K. Wong, "Large language models as source planner for personalized knowledge-grounded dialogues," in *Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023*, H. Bouamor, J. Pino, and K. Bali, Eds. Association for Computational Linguistics, 2023, pp. 9556–9569. [Online]. Available: https://doi.org/10.18653/v1/2023.findings-emnlp.641

[85] X. Xu, B. Yao, Y. Dong, H. Yu, J. Hendler, A. K. Dey, and D. Wang, "Leveraging large language models for mental health prediction via online text data," *arXiv preprint arXiv:2307.14385*, 2023.

[86] P. Li and A. Tuzhilin, "Towards controllable and personalized review generation," in *Proceedings of*

the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019, K. Inui, J. Jiang, V. Ng, and X. Wan, Eds. Association for Computational Linguistics, 2019, pp. 3235–3243. [Online]. Available: https://doi.org/10.18653/v1/D19-1319

[87] Y. Zhou, J. Yao, Z. Dou, L. Wu, and J. Wen, "Dynamicretriever: A pre-trained model-based IR system without an explicit index," *Mach. Intell. Res.*, vol. 20, no. 2, pp. 276–288, 2023. [Online]. Available: https://doi.org/10.1007/s11633-022-1373-9

[88] S. Zhuang, H. Ren, L. Shou, J. Pei, M. Gong, G. Zuccon, and D. Jiang, "Bridging the gap between indexing and retrieval for differentiable search index with query generation," *CoRR*, vol. abs/2206.10128, 2022. [Online]. Available: https://doi.org/10.48550/arXiv.2206.10128

[89] X. Chen, Y. Liu, B. He, L. Sun, and Y. Sun, "Understanding differential search index for text retrieval," in *Findings of the Association for Computational Linguistics: ACL 2023, Toronto, Canada, July 9-14, 2023*, A. Rogers, J. L. Boyd-Graber, and N. Okazaki, Eds. Association for Computational Linguistics, 2023, pp. 10 701–10 717. [Online]. Available: https://doi.org/10.18653/v1/2023.findings-acl.681

[90] Y. Li, N. Yang, L. Wang, F. Wei, and W. Li, "Learning to rank in generative retrieval," *CoRR*, vol. abs/2306.15222, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2306.15222

[91] Y. Li, Z. Zhang, W. Wang, L. Nie, W. Li, and T.-S. Chua, "Distillation enhanced generative retrieval," *arXiv preprint arXiv:2402.10769*, 2024.

[92] Y. Tang, R. Zhang, J. Guo, M. de Rijke, W. Chen, and X. Cheng, "Listwise generative retrieval models via a sequential learning process," *ACM Transactions on Information Systems*, 2024.

[93] R. Ren, W. X. Zhao, J. Liu, H. Wu, J. Wen, and H. Wang, "TOME: A two-stage approach for model-based retrieval," pp. 6102–6114, 2023. [Online]. Available: https://doi.org/10.18653/v1/2023.acl-long.336

[94] H. Lee, J. Kim, H. Chang, H. Oh, S. Yang, V. Karpukhin, Y. Lu, and M. Seo, "Nonparametric decoding for generative retrieval," in *Findings of the Association for Computational Linguistics: ACL 2023, Toronto, Canada, July 9-14, 2023*, A. Rogers, J. L. Boyd-Graber, and N. Okazaki, Eds. Association for Computational Linguistics, 2023, pp. 12 642–12 661. [Online]. Available: https://doi.org/10.18653/v1/2023.findings-acl.801

[95] H. Zhang, Y. Wang, Q. Chen, R. Chang, T. Zhang, Z. Miao, Y. Hou, Y. Ding, X. Miao, H. Wang *et al.*, "Model-enhanced vector index," *arXiv preprint arXiv:2309.13335*, 2023.

[96] S. Qiao, X. Liu, and S. Na, "Diffusionret: Diffusion-enhanced generative retriever using constrained decoding," in *Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023*, H. Bouamor, J. Pino, and K. Bali, Eds. Association for Computational Linguistics, 2023, pp. 9515–9529. [Online]. Available: https://aclanthology.org/2023.findings-emnlp.638

[97] P. Yuan, X. Wang, S. Feng, B. Pan, Y. Li, H. Wang, X. Miao, and K. Li, "Generative dense retrieval: Memory can be a burden," *CoRR*, vol. abs/2401.10487, 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2401.10487

[98] Q. Tang, J. Chen, B. Yu, Y. Lu, C. Fu, H. Yu, H. Lin, F. Huang, B. He, X. Han *et al.*, "Self-retrieval: Building an information retrieval system with one large language model," *arXiv preprint arXiv:2403.00801*, 2024.

[99] T. Nguyen and A. Yates, "Generative retrieval as dense retrieval," *CoRR*, vol. abs/2306.11397, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2306.11397

[100] B. Jin, H. Zeng, G. Wang, X. Chen, T. Wei, R. Li, Z. Wang, Z. Li, Y. Li, H. Lu *et al.*, "Language models as semantic indexers," *arXiv preprint arXiv:2310.07815*, 2023.

[101] H. Zeng, C. Luo, B. Jin, S. M. Sarwar, T. Wei, and H. Zamani, "Scalable and effective generative information retrieval," *CoRR*, vol. abs/2311.09134, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2311.09134

[102] M. Bevilacqua, G. Ottaviano, P. S. H. Lewis, S. Yih, S. Riedel, and F. Petroni, "Autoregressive search engines: Generating substrings as document identifiers," in *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, Eds., 2022. [Online]. Available: http://papers.nips.cc/paper_files/paper/2022/hash/cd88d62a2063fdaf7ce6f9068fb15dcd-Abstract-Conference.html

[103] N. Ziems, W. Yu, Z. Zhang, and M. Jiang, "Large language models are built-in autoregressive search engines," in *Findings of the Association for Computational Linguistics: ACL 2023, Toronto, Canada, July 9-14, 2023*, A. Rogers, J. L. Boyd-Graber, and N. Okazaki, Eds. Association for Computational Linguistics, 2023, pp. 2666–2678. [Online]. Available: https://doi.org/10.18653/v1/2023.findings-acl.167

[104] J. Chen, R. Zhang, J. Guo, M. de Rijke, Y. Liu, Y. Fan, and X. Cheng, "A unified generative retriever for knowledge-intensive language tasks via prompt learning," in *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2023, Taipei, Taiwan, July 23-27, 2023*, H. Chen, W. E. Duh, H. Huang, M. P. Kato, J. Mothe, and B. Poblete, Eds. ACM, 2023, pp. 1448–1457. [Online]. Available: https://doi.org/10.1145/3539618.3591631

[105] Y. Li, N. Yang, L. Wang, F. Wei, and W. Li, "Multiview identifiers enhanced generative retrieval," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, A. Rogers, J. L. Boyd-Graber, and

N. Okazaki, Eds. Association for Computational Linguistics, 2023, pp. 6636–6648. [Online]. Available: https://doi.org/10.18653/v1/2023.acl-long.366

[106] P. Zhang, Z. Liu, Y. Zhou, Z. Dou, and Z. Cao, "Term-sets can be strong document identifiers for auto-regressive search engines," *CoRR*, vol. abs/2305.13859, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2305.13859

[107] Z. Wang, Y. Zhou, Y. Tu, and Z. Dou, "NOVO: learnable and interpretable document identifiers for model-based IR," in *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management, CIKM 2023, Birmingham, United Kingdom, October 21-25, 2023*, I. Frommholz, F. Hopfgartner, M. Lee, M. Oakes, M. Lalmas, M. Zhang, and R. L. T. Santos, Eds. ACM, 2023, pp. 2656–2665. [Online]. Available: https://doi.org/10.1145/3583780.3614993

[108] S. Lee, M. Choi, and J. Lee, "GLEN: generative retrieval via lexical index learning," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, H. Bouamor, J. Pino, and K. Bali, Eds. Association for Computational Linguistics, 2023, pp. 7693–7704. [Online]. Available: https://aclanthology.org/2023.emnlp-main.477

[109] S. Yoon, C. Kim, H. Lee, J. Jang, and M. Seo, "Exploring the practicality of generative retrieval on dynamic corpora," 2023. [Online]. Available: https://api.semanticscholar.org/CorpusID:258967398

[110] Y. Liu, R. Zhang, J. Guo, W. Chen, and X. Cheng, "On the robustness of generative retrieval models: An out-of-distribution perspective," *CoRR*, vol. abs/2306.12756, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2306.12756

[111] J. Chen, R. Zhang, J. Guo, M. de Rijke, W. Chen, Y. Fan, and X. Cheng, "Continual learning for generative retrieval over dynamic corpora," in *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, 2023, pp. 306–315.

[112] H. Lee, S. Yang, H. Oh, and M. Seo, "Generative multi-hop retrieval," in *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, Y. Goldberg, Z. Kozareva, and Y. Zhang, Eds. Association for Computational Linguistics, 2022, pp. 1417–1436. [Online]. Available: https://doi.org/10.18653/v1/2022.emnlp-main.92

[113] J. Thorne, "Data-efficient autoregressive document retrieval for fact verification," *CoRR*, vol. abs/2211.09388, 2022. [Online]. Available: https://doi.org/10.48550/arXiv.2211.09388

[114] U. Nadeem, N. Ziems, and S. Wu, "Codedsi: Differentiable code search," *CoRR*, vol. abs/2210.00328, 2022. [Online]. Available: https://doi.org/10.48550/arXiv.2210.00328

[115] Y. Li, N. Yang, L. Wang, F. Wei, and W. Li, "Generative retrieval for conversational question answering," *Inf. Process. Manag.*, vol. 60, no. 5, p. 103475, 2023. [Online]. Available: https://doi.org/10.1016/j.ipm.2023.103475

[116] E. Song, S. Kim, H. Lee, J. Kim, and J. Thorne, "Re3val: Reinforced and reranked generative retrieval," *CoRR*, vol. abs/2401.16979, 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2401.16979

[117] X. Li, Y. Zhou, and Z. Dou, "Unigen: A unified generative framework for retrieval and question answering with large language models," in *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2014, February 20-27, 2024, Vancouver, Canada*, M. J. Wooldridge, J. G. Dy, and S. Natarajan, Eds. AAAI Press, 2024, pp. 8688–8696. [Online]. Available: https://doi.org/10.1609/aaai.v38i8.28714

[118] J. Li, W. Zhang, T. Wang, G. Xiong, A. Lu, and G. Medioni, "Gpt4rec: A generative framework for personalized recommendation and user interests interpretation," *arXiv preprint arXiv:2304.03879*, 2023.

[119] Z. Si, Z. Sun, J. Chen, G. Chen, X. Zang, K. Zheng, Y. Song, X. Zhang, and J. Xu, "Generative retrieval with semantic tree-structured item identifiers via contrastive learning," *CoRR*, vol. abs/2309.13375, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2309.13375

[120] J. Tan, S. Xu, W. Hua, Y. Ge, Z. Li, and Y. Zhang, "Towards llm-recsys alignment with textual id learning," *arXiv preprint arXiv:2403.19021*, 2024.

[121] B. Zheng, Y. Hou, H. Lu, Y. Chen, W. X. Zhao, and J.-R. Wen, "Adapting large language models by integrating collaborative semantics for recommendation," *arXiv preprint arXiv:2311.09049*, 2023.

[122] Y. Wang, Z. Ren, W. Sun, J. Yang, Z. Liang, X. Chen, R. Xie, S. Yan, X. Zhang, P. Ren, Z. Chen, and X. Xin, "Enhanced generative recommendation via content and collaboration integration," 2024. [Online]. Available: https://api.semanticscholar.org/CorpusID:268723798

[123] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung, C. Sutton, S. Gehrmann, P. Schuh, K. Shi, S. Tsvyashchenko, J. Maynez, A. Rao, P. Barnes, Y. Tay, N. Shazeer, V. Prabhakaran, E. Reif, N. Du, B. Hutchinson, R. Pope, J. Bradbury, J. Austin, M. Isard, G. Gur-Ari, P. Yin, T. Duke, A. Levskaya, S. Ghemawat, S. Dev, H. Michalewski, X. Garcia, V. Misra, K. Robinson, L. Fedus, D. Zhou, D. Ippolito, D. Luan, H. Lim, B. Zoph, A. Spiridonov, R. Sepassi, D. Dohan, S. Agrawal, M. Omernick, A. M. Dai, T. S. Pillai, M. Pellat, A. Lewkowycz, E. Moreira, R. Child, O. Polozov, K. Lee, Z. Zhou, X. Wang, B. Saeta, M. Diaz, O. Firat, M. Catasta, J. Wei, K. Meier-Hellstern, D. Eck, J. Dean, S. Petrov, and N. Fiedel, "Palm: Scaling language modeling with pathways," *J. Mach. Learn. Res.*, vol. 24, pp. 240:1–240:113, 2023. [Online]. Available: http://jmlr.org/papers/v24/22-1144.html

[124] A. Q. Jiang, A. Sablayrolles, A. Roux, A. Mensch, B. Savary, C. Bamford, D. S. Chaplot, D. de Las Casas, E. B. Hanna, F. Bressand, G. Lengyel, G. Bour, G. Lample, L. R. Lavaud, L. Saulnier, M. Lachaux, P. Stock, S. Subramanian, S. Yang, S. Antoniak,

T. L. Scao, T. Gervet, T. Lavril, T. Wang, T. Lacroix, and W. E. Sayed, "Mixtral of experts," *CoRR*, vol. abs/2401.04088, 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2401.04088

[125] N. Sadeq, B. Kang, P. Lamba, and J. J. McAuley, "Unsupervised improvement of factual knowledge in language models," in *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2023, Dubrovnik, Croatia, May 2-6, 2023*, A. Vlachos and I. Augenstein, Eds. Association for Computational Linguistics, 2023, pp. 2952–2961. [Online]. Available: https://doi.org/10.18653/v1/2023.eacl-main.215

[126] O. Ovadia, M. Brief, M. Mishaeli, and O. Elisha, "Fine-tuning or retrieval? comparing knowledge injection in llms," *CoRR*, vol. abs/2312.05934, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2312.05934

[127] W. Yu, D. Iter, S. Wang, Y. Xu, M. Ju, S. Sanyal, C. Zhu, M. Zeng, and M. Jiang, "Generate rather than retrieve: Large language models are strong context generators," *arXiv preprint arXiv:2209.10063*, 2022.

[128] Z. Sun, X. Wang, Y. Tay, Y. Yang, and D. Zhou, "Recitation-augmented language models," *arXiv preprint arXiv:2210.01296*, 2022.

[129] J. Mok, J. Do, S. Lee, T. Taghavi, S. Yu, and S. Yoon, "Large-scale lifelong learning of in-context instructions and how to tackle it," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2023, pp. 12 573–12 589.

[130] N. D. Cao, W. Aziz, and I. Titov, "Editing factual knowledge in language models," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, M. Moens, X. Huang, L. Specia, and S. W. Yih, Eds. Association for Computational Linguistics, 2021, pp. 6491–6506. [Online]. Available: https://doi.org/10.18653/v1/2021.emnlp-main.522

[131] E. Mitchell, C. Lin, A. Bosselut, C. Finn, and C. D. Manning, "Fast model editing at scale," in *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. [Online]. Available: https://openreview.net/forum?id=0DcZxeWfOPt

[132] X. Ma, Y. Gong, P. He, H. Zhao, and N. Duan, "Query rewriting in retrieval-augmented large language models," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, H. Bouamor, J. Pino, and K. Bali, Eds. Singapore: Association for Computational Linguistics, Dec. 2023, pp. 5303–5315. [Online]. Available: https://aclanthology.org/2023.emnlp-main.322

[133] L. Zhang, Y. Yu, K. Wang, and C. Zhang, "Arl2: Aligning retrievers for black-box large language models via self-guided adaptive relevance labeling," 2024.

[134] G. Kim, S. Kim, B. Jeon, J. Park, and J. Kang, "Tree of clarifications: Answering ambiguous questions with retrieval-augmented large language models," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, H. Bouamor, J. Pino, and K. Bali, Eds. Singapore: Association for Computational Linguistics, Dec. 2023, pp. 996–1009. [Online]. Available: https://aclanthology.org/2023.emnlp-main.63

[135] H. Wang, T. Zhao, and J. Gao, "Blendfilter: Advancing retrieval-augmented large language models via query generation blending and knowledge filtering," 2024.

[136] W. Shi, S. Min, M. Yasunaga, M. Seo, R. James, M. Lewis, L. Zettlemoyer, and W. Yih, "REPLUG: retrieval-augmented black-box language models," *CoRR*, vol. abs/2301.12652, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2301.12652

[137] Y. Wang, P. Li, M. Sun, and Y. Liu, "Self-knowledge guided retrieval augmentation for large language models," in *Findings of the Association for Computational Linguistics: EMNLP 2023*, H. Bouamor, J. Pino, and K. Bali, Eds. Singapore: Association for Computational Linguistics, Dec. 2023, pp. 10 303–10 315. [Online]. Available: https://aclanthology.org/2023.findings-emnlp.691

[138] H. Ding, L. Pang, Z. Wei, H. Shen, and X. Cheng, "Retrieve only when it needs: Adaptive retrieval augmentation for hallucination mitigation in large language models," 2024.

[139] Z. Shao, Y. Gong, Y. Shen, M. Huang, N. Duan, and W. Chen, "Enhancing retrieval-augmented large language models with iterative retrieval-generation synergy," 2023.

[140] H. Trivedi, N. Balasubramanian, T. Khot, and A. Sabharwal, "Interleaving retrieval with chain-of-thought reasoning for knowledge-intensive multi-step questions," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, A. Rogers, J. Boyd-Graber, and N. Okazaki, Eds. Toronto, Canada: Association for Computational Linguistics, Jul. 2023, pp. 10 014–10 037. [Online]. Available: https://aclanthology.org/2023.acl-long.557

[141] J. Sun, C. Xu, L. Tang, S. Wang, C. Lin, Y. Gong, H.-Y. Shum, and J. Guo, "Think-on-graph: Deep and responsible reasoning of large language model with knowledge graph," 2023.

[142] L. Luo, Y.-F. Li, G. Haffari, and S. Pan, "Reasoning on graphs: Faithful and interpretable large language model reasoning," in *International Conference on Learning Representations*, 2024.

[143] Y. Qin, S. Liang, Y. Ye, K. Zhu, L. Yan, Y. Lu, Y. Lin, X. Cong, X. Tang, B. Qian, S. Zhao, R. Tian, R. Xie, J. Zhou, M. Gerstein, D. Li, Z. Liu, and M. Sun, "Toolllm: Facilitating large language models to master 16000+ real-world apis," *CoRR*, vol. abs/2307.16789, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2307.16789

[144] D. Gao, L. Ji, L. Zhou, K. Q. Lin, J. Chen, Z. Fan, and M. Z. Shou, "Assistgpt: A general multi-modal assistant that can plan, execute, inspect, and learn," 2023.

[145] Y. Shen, K. Song, X. Tan, D. Li, W. Lu, and Y. Zhuang, "Hugginggpt: Solving AI tasks with chatgpt and its friends in hugging face," in *Advances in Neural Information Processing Systems*

36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023, A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, Eds., 2023. [Online]. Available: http://papers.nips.cc/paper_files/paper/2023/hash/77c33e6a367922d003ff102ffb92b658-Abstract-Conference.html

[146] C. Wu, S. Yin, W. Qi, X. Wang, Z. Tang, and N. Duan, "Visual chatgpt: Talking, drawing and editing with visual foundation models," 2023.

[147] C. Fierro, R. K. Amplayo, F. Huot, N. De Cao, J. Maynez, S. Narayan, and M. Lapata, "Learning to plan and generate text with citations," *arXiv preprint arXiv:2404.03381*, 2024.

[148] D. Peskoff and B. Stewart, "Credible without credit: Domain experts assess generative language models," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, A. Rogers, J. L. Boyd-Graber, and N. Okazaki, Eds. Association for Computational Linguistics, 2023, pp. 427–438. [Online]. Available: https://doi.org/10.18653/v1/2023.acl-short.37

[149] P. Jain, L. Soares, and T. Kwiatkowski, "1-pager: One pass answer generation and evidence retrieval," in *Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023*, H. Bouamor, J. Pino, and K. Bali, Eds. Association for Computational Linguistics, 2023, pp. 14 529–14 543. [Online]. Available: https://doi.org/10.18653/v1/2023.findings-emnlp.967

[150] M. Khalifa, D. Wadden, E. Strubell, H. Lee, L. Wang, I. Beltagy, and H. Peng, "Source-aware training enables knowledge attribution in language models," 2024. [Online]. Available: https://api.semanticscholar.org/CorpusID:268819100

[151] L. Gao, Z. Dai, P. Pasupat, A. Chen, A. T. Chaganty, Y. Fan, V. Y. Zhao, N. Lao, H. Lee, D. Juan, and K. Guu, "RARR: researching and revising what language models say, using language models," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, A. Rogers, J. L. Boyd-Graber, and N. Okazaki, Eds. Association for Computational Linguistics, 2023, pp. 16 477–16 508. [Online]. Available: https://doi.org/10.18653/v1/2023.acl-long.910

[152] S. Xu, L. Pang, H. Shen, X. Cheng, and T. Chua, "Search-in-the-chain: Towards the accurate, credible and traceable content generation for complex knowledge-intensive tasks," *CoRR*, vol. abs/2304.14732, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2304.14732

[153] T. Gao, H. Yen, J. Yu, and D. Chen, "Enabling large language models to generate text with citations," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, H. Bouamor, J. Pino, and K. Bali, Eds. Association for Computational Linguistics, 2023, pp. 6465–6488. [Online]. Available: https://aclanthology.org/2023.emnlp-main.398

[154] H. Sun, H. Cai, B. Wang, Y. Hou, X. Wei, S. Wang, Y. Zhang, and D. Yin, "Towards verifiable text generation with evolving memory and self-reflection," *CoRR*, vol. abs/2312.09075, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2312.09075

[155] W. Li, J. Li, W. Ma, and Y. Liu, "Citation-enhanced generation for llm-based chatbots," *CoRR*, vol. abs/2402.16063, 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2402.16063

[156] S. Zhang, E. Dinan, J. Urbanek, A. Szlam, D. Kiela, and J. Weston, "Personalizing dialogue agents: I have a dog, do you have pets too?" in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, I. Gurevych and Y. Miyao, Eds. Association for Computational Linguistics, 2018, pp. 2204–2213. [Online]. Available: https://aclanthology.org/P18-1205/

[157] Y. Wu, X. Ma, and D. Yang, "Personalized response generation via generative split memory network," in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, K. Toutanova, A. Rumshisky, L. Zettlemoyer, D. Hakkani-Tür, I. Beltagy, S. Bethard, R. Cotterell, T. Chakraborty, and Y. Zhou, Eds. Association for Computational Linguistics, 2021, pp. 1956–1970. [Online]. Available: https://doi.org/10.18653/v1/2021.naacl-main.157

[158] J. Jang, S. Kim, B. Y. Lin, Y. Wang, J. Hessel, L. Zettlemoyer, H. Hajishirzi, Y. Choi, and P. Ammanabrolu, "Personalized soups: Personalized large language model alignment via post-hoc parameter merging," *CoRR*, vol. abs/2310.11564, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2310.11564

[159] Z. Tan, Q. Zeng, Y. Tian, Z. Liu, B. Yin, and M. Jiang, "Democratizing large language models via personalized parameter-efficient fine-tuning," *CoRR*, vol. abs/2402.04401, 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2402.04401

[160] Z. Liu, Z. Wu, M. Hu, B. Zhao, L. Zhao, T. Zhang, H. Dai, X. Chen, Y. Shen, S. Li *et al.*, "Pharmacygpt: The ai pharmacist," *arXiv preprint arXiv:2307.10432*, 2023.

[161] S. Mysore, Z. Lu, M. Wan, L. Yang, S. Menezes, T. Baghaee, E. B. Gonzalez, J. Neville, and T. Safavi, "PEARL: personalizing large language model writing assistants with generation-calibrated retrievers," *CoRR*, vol. abs/2311.09180, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2311.09180

[162] Y. Dan, Z. Lei, Y. Gu, Y. Li, J. Yin, J. Lin, L. Ye, Z. Tie, Y. Zhou, Y. Wang *et al.*, "Educhat: A large-scale language model-based chatbot system for intelligent education," *arXiv preprint arXiv:2308.02773*, 2023.

[163] N. Craswell, "Mean reciprocal rank," in *Encyclopedia of Database Systems*, L. Liu and M. T. Özsu, Eds. Springer US, 2009, p. 1703. [Online]. Available: https://doi.org/10.1007/978-0-387-39940-9_488

[164] K. Järvelin and J. Kekäläinen, "Cumulated gain-based evaluation of IR techniques," *ACM Trans. Inf. Syst.*,

vol. 20, no. 4, pp. 422–446, 2002. [Online]. Available: http://doi.acm.org/10.1145/582415.582418

[165] T. Nguyen, M. Rosenberg, X. Song, J. Gao, S. Tiwary, R. Majumder, and L. Deng, "MS MARCO: A human generated machine reading comprehension dataset," in *Proceedings of the Workshop on Cognitive Computation: Integrating neural and symbolic approaches 2016 co-located with the 30th Annual Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain, December 9, 2016*, ser. CEUR Workshop Proceedings, T. R. Besold, A. Bordes, A. S. d'Avila Garcez, and G. Wayne, Eds., vol. 1773. CEUR-WS.org, 2016. [Online]. Available: https://ceur-ws.org/Vol-1773/CoCoNIPS_2016_paper9.pdf

[166] T. Kwiatkowski, J. Palomaki, O. Redfield, M. Collins, A. Parikh, C. Alberti, D. Epstein, I. Polosukhin, J. Devlin, K. Lee *et al.*, "Natural questions: a benchmark for question answering research," *Transactions of the Association for Computational Linguistics*, vol. 7, pp. 453–466, 2019.

[167] M. Joshi, E. Choi, D. Weld, and L. Zettlemoyer, "TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension," in *ACL*. Vancouver, Canada: Association for Computational Linguistics, Jul. 2017, pp. 1601–1611.

[168] F. Petroni, A. Piktus, A. Fan, P. Lewis, M. Yazdani, N. De Cao, J. Thorne, Y. Jernite, V. Karpukhin, J. Maillard, V. Plachouras, T. Rocktäschel, and S. Riedel, "KILT: a benchmark for knowledge intensive language tasks," in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Online: Association for Computational Linguistics, Jun. 2021, pp. 2523–2544. [Online]. Available: https://aclanthology.org/2021.naacl-main.200

[169] N. Craswell, B. Mitra, E. Yilmaz, D. Campos, and E. M. Voorhees, "Overview of the TREC 2019 deep learning track," *CoRR*, vol. abs/2003.07820, 2020. [Online]. Available: https://arxiv.org/abs/2003.07820

[170] N. Craswell, B. Mitra, E. Yilmaz, and D. Campos, "Overview of the TREC 2020 deep learning track," *CoRR*, vol. abs/2102.07662, 2021. [Online]. Available: https://arxiv.org/abs/2102.07662

[171] R. Pradeep, K. Hui, J. Gupta, Á. D. Lelkes, H. Zhuang, J. Lin, D. Metzler, and V. Q. Tran, "How does generative retrieval scale to millions of passages?" in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, H. Bouamor, J. Pino, and K. Bali, Eds. Association for Computational Linguistics, 2023, pp. 1305–1321. [Online]. Available: https://aclanthology.org/2023.emnlp-main.83

[172] K. Papineni, S. Roukos, T. Ward, and W. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, July 6-12, 2002, Philadelphia, PA, USA*. ACL, 2002, pp. 311–318. [Online]. Available: https://aclanthology.org/P02-1040/

[173] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," in *Annual Meeting of the Association for Computational Linguistics*, 2004. [Online]. Available: https://api.semanticscholar.org/CorpusID:964287

[174] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, and Y. Artzi, "Bertscore: Evaluating text generation with BERT," in *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020. [Online]. Available: https://openreview.net/forum?id=SkeHuCVFDr

[175] T. Sellam, D. Das, and A. Parikh, "BLEURT: Learning robust metrics for text generation," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, Jul. 2020, pp. 7881–7892. [Online]. Available: https://aclanthology.org/2020.acl-main.704

[176] J. Fu, S. Ng, Z. Jiang, and P. Liu, "Gptscore: Evaluate as you desire," *CoRR*, vol. abs/2302.04166, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2302.04166

[177] S. Min, K. Krishna, X. Lyu, M. Lewis, W. Yih, P. W. Koh, M. Iyyer, L. Zettlemoyer, and H. Hajishirzi, "Factscore: Fine-grained atomic evaluation of factual precision in long form text generation," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, H. Bouamor, J. Pino, and K. Bali, Eds. Association for Computational Linguistics, 2023, pp. 12 076–12 100. [Online]. Available: https://aclanthology.org/2023.emnlp-main.741

[178] D. Hendrycks, C. Burns, S. Basart, A. Zou, M. Mazeika, D. Song, and J. Steinhardt, "Measuring massive multitask language understanding," in *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. [Online]. Available: https://openreview.net/forum?id=d7KBjmI3GmQ

[179] A. Srivastava, A. Rastogi, A. Rao, A. A. M. Shoeb, A. Abid, A. Fisch, A. R. Brown, A. Santoro, A. Gupta, A. Garriga-Alonso, A. Kluska, A. Lewkowycz, A. Agarwal, A. Power, A. Ray, A. Warstadt, A. W. Kocurek, A. Safaya, A. Tazarv, A. Xiang, A. Parrish, A. Nie, A. Hussain, A. Askell, A. Dsouza, A. Rahane, A. S. Iyer, A. Andreassen, A. Santilli, A. Stuhlmüller, A. M. Dai, A. La, A. K. Lampinen, A. Zou, A. Jiang, A. Chen, A. Vuong, A. Gupta, A. Gottardi, A. Norelli, A. Venkatesh, A. Gholamidavoodi, A. Tabassum, A. Menezes, A. Kirubarajan, A. Mullokandov, A. Sabharwal, A. Herrick, A. Efrat, A. Erdem, A. Karakas, and et al., "Beyond the imitation game: Quantifying and extrapolating the capabilities of language models," *CoRR*, vol. abs/2206.04615, 2022. [Online]. Available: https://doi.org/10.48550/arXiv.2206.04615

[180] Y. Lin and Y. Chen, "Llm-eval: Unified multi-dimensional automatic evaluation for open-domain conversations with large language models," *CoRR*, vol. abs/2305.13711, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2305.13711

[181] M. Li, Y. Zhao, B. Yu, F. Song, H. Li, H. Yu, Z. Li, F. Huang, and Y. Li, "Api-bank: A comprehensive

benchmark for tool-augmented llms," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, H. Bouamor, J. Pino, and K. Bali, Eds. Association for Computational Linguistics, 2023, pp. 3102–3116. [Online]. Available: https://aclanthology.org/2023.emnlp-main.187

[182] S. Lin, J. Hilton, and O. Evans, "Truthfulqa: Measuring how models mimic human falsehoods," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, S. Muresan, P. Nakov, and A. Villavicencio, Eds. Association for Computational Linguistics, 2022, pp. 3214–3252. [Online]. Available: https://doi.org/10.18653/v1/2022.acl-long.229

[183] J. Li, X. Cheng, X. Zhao, J. Nie, and J. Wen, "Halueval: A large-scale hallucination evaluation benchmark for large language models," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, H. Bouamor, J. Pino, and K. Bali, Eds. Association for Computational Linguistics, 2023, pp. 6449–6464. [Online]. Available: https://aclanthology.org/2023.emnlp-main.397

[184] J. Kasai, K. Sakaguchi, Y. Takahashi, R. L. Bras, A. Asai, X. Yu, D. Radev, N. A. Smith, Y. Choi, and K. Inui, "Realtime QA: what's the answer right now?" in *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, Eds., 2023. [Online]. Available: http://papers.nips.cc/paper_files/paper/2023/hash/9941624ef7f867a502732b5154d30cb7-Abstract-Datasets_and_Benchmarks.html

[185] Z. Zhang, L. Lei, L. Wu, R. Sun, Y. Huang, C. Long, X. Liu, X. Lei, J. Tang, and M. Huang, "Safetybench: Evaluating the safety of large language models with multiple choice questions," *CoRR*, vol. abs/2309.07045, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2309.07045

[186] J. Ramos *et al.*, "Using tf-idf to determine word relevance in document queries," in *Proceedings of the first instructional conference on machine learning*, vol. 242, no. 1. Citeseer, 2003, pp. 29–48.

[187] Y. Luan, J. Eisenstein, K. Toutanova, and M. Collins, "Sparse, dense, and attentional representations for text retrieval," *Trans. Assoc. Comput. Linguistics*, vol. 9, pp. 329–345, 2021. [Online]. Available: https://doi.org/10.1162/tacl_a_00369

[188] L. Wang, N. Yang, X. Huang, B. Jiao, L. Yang, D. Jiang, R. Majumder, and F. Wei, "Text embeddings by weakly-supervised contrastive pre-training," *CoRR*, vol. abs/2212.03533, 2022. [Online]. Available: https://doi.org/10.48550/arXiv.2212.03533

[189] ——, "Simlm: Pre-training with representation bottleneck for dense passage retrieval," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, A. Rogers, J. L. Boyd-Graber, and N. Okazaki, Eds. Association for Computational Linguistics, 2023, pp. 2244–2258. [Online]. Available: https://doi.org/10.18653/v1/2023.acl-long.125

[190] Y. Zhu, H. Yuan, S. Wang, J. Liu, W. Liu, C. Deng, Z. Dou, and J.-R. Wen, "Large language models for information retrieval: A survey," *arXiv preprint arXiv:2308.07107*, 2023.

[191] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever *et al.*, "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.

[192] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. L. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, J. Schulman, J. Hilton, F. Kelton, L. Miller, M. Simens, A. Askell, P. Welinder, P. F. Christiano, J. Leike, and R. Lowe, "Training language models to follow instructions with human feedback," in *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, Eds., 2022. [Online]. Available: http://papers.nips.cc/paper_files/paper/2022/hash/b1efde53be364a73914f58805a001731-Abstract-Conference.html

[193] G. Penedo, Q. Malartic, D. Hesslow, R. Cojocaru, H. Alobeidli, A. Cappelli, B. Pannier, E. Almazrouei, and J. Launay, "The refinedweb dataset for falcon LLM: outperforming curated corpora with web data only," in *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, Eds., 2023. [Online]. Available: http://papers.nips.cc/paper_files/paper/2023/hash/fa3ed726cc5073b9c31e3e49a807789c-Abstract-Datasets_and_Benchmarks.html

[194] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "Lora: Low-rank adaptation of large language models," in *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. [Online]. Available: https://openreview.net/forum?id=nZeVKeeFYf9

[195] N. Ding, Y. Qin, G. Yang, F. Wei, Z. Yang, Y. Su, S. Hu, Y. Chen, C. Chan, W. Chen, J. Yi, W. Zhao, X. Wang, Z. Liu, H. Zheng, J. Chen, Y. Liu, J. Tang, J. Li, and M. Sun, "Parameter-efficient fine-tuning of large-scale pre-trained language models," *Nat. Mac. Intell.*, vol. 5, no. 3, pp. 220–235, 2023. [Online]. Available: https://doi.org/10.1038/s42256-023-00626-4

[196] T. Dettmers, A. Pagnoni, A. Holtzman, and L. Zettlemoyer, "Qlora: Efficient finetuning of quantized llms," *CoRR*, vol. abs/2305.14314, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2305.14314

[197] Y. Zhu, P. Zhang, C. Zhang, Y. Chen, B. Xie, Z. Dou, Z. Liu, and J. Wen, "INTERS: unlocking the power

of large language models in search with instruction tuning," *CoRR*, vol. abs/2401.06532, 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2401.06532

[198] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan, and R. Garnett, Eds., 2017, pp. 5998–6008. [Online]. Available: https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html

[199] R. Nogueira, J. Lin, and A. Epistemic, "From doc2query to docttttquery," *Online preprint*, vol. 6, p. 2, 2019.

[200] X. Chen, Y. Liu, B. He, L. Sun, and Y. Sun, "Understanding differential search index for text retrieval," pp. 10 701–10 717, 2023. [Online]. Available: https://doi.org/10.18653/v1/2023.findings-acl.681

[201] Y. Tang, R. Zhang, J. Guo, J. Chen, Z. Zhu, S. Wang, D. Yin, and X. Cheng, "Semantic-enhanced differentiable search index inspired by learning strategies," in *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD 2023, Long Beach, CA, USA, August 6-10, 2023*, A. K. Singh, Y. Sun, L. Akoglu, D. Gunopulos, X. Yan, R. Kumar, F. Ozcan, and J. Ye, Eds. ACM, 2023, pp. 4904–4913. [Online]. Available: https://doi.org/10.1145/3580305.3599903

[202] J. Martinez, H. H. Hoos, and J. J. Little, "Stacked quantizers for compositional vector compression," *CoRR*, vol. abs/1411.2173, 2014. [Online]. Available: http://arxiv.org/abs/1411.2173

[203] H. Yuan, Z. Yuan, C. Tan, F. Huang, and S. Huang, "Seqdiffuseq: Text diffusion with encoder-decoder transformers," *CoRR*, vol. abs/2212.10325, 2022. [Online]. Available: https://doi.org/10.48550/arXiv.2212.10325

[204] P. Ferragina and G. Manzini, "Opportunistic data structures with applications," in *41st Annual Symposium on Foundations of Computer Science, FOCS 2000, 12-14 November 2000, Redondo Beach, California, USA*. IEEE Computer Society, 2000, pp. 390–398. [Online]. Available: https://doi.org/10.1109/SFCS.2000.892127

[205] H. Jégou, M. Douze, and C. Schmid, "Product quantization for nearest neighbor search," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 1, pp. 117–128, 2011. [Online]. Available: https://doi.org/10.1109/TPAMI.2010.57

[206] T. Ge, K. He, Q. Ke, and J. Sun, "Optimized product quantization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 4, pp. 744–755, 2014. [Online]. Available: https://doi.org/10.1109/TPAMI.2013.240

[207] A. Liska, T. Kociský, E. Gribovskaya, T. Terzi, E. Sezener, D. Agrawal, C. de Masson d'Autume, T. Scholtes, M. Zaheer, S. Young, E. Gilsenan-McMahon, S. Austin, P. Blunsom, and A. Lazaridou, "Streamingqa: A benchmark for adaptation to new knowledge over time in question answering models," in *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, ser. Proceedings of Machine Learning Research, K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvári, G. Niu, and S. Sabato, Eds., vol. 162. PMLR, 2022, pp. 13 604–13 622. [Online]. Available: https://proceedings.mlr.press/v162/liska22a.html

[208] J. Thorne, A. Vlachos, C. Christodoulopoulos, and A. Mittal, "Fever: a large-scale dataset for fact extraction and verification," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 2018, pp. 809–819.

[209] J. Hoffart, M. A. Yosef, I. Bordino, H. Fürstenau, M. Pinkal, M. Spaniol, B. Taneva, S. Thater, and G. Weikum, "Robust disambiguation of named entities in text," in *EMNLP*, 2011, pp. 782–792.

[210] E. Dinan, S. Roller, K. Shuster, A. Fan, M. Auli, and J. Weston, "Wizard of wikipedia: Knowledge-powered conversational agents," in *International Conference on Learning Representations*, 2018.

[211] O. Levy, M. Seo, E. Choi, and L. Zettlemoyer, "Zero-shot relation extraction via reading comprehension," in *CoNLL*, 2017, pp. 333–342.

[212] S. Lu, D. Guo, S. Ren, J. Huang, A. Svyatkovskiy, A. Blanco, C. B. Clement, D. Drain, D. Jiang, D. Tang, G. Li, L. Zhou, L. Shou, L. Zhou, M. Tufano, M. Gong, M. Zhou, N. Duan, N. Sundaresan, S. K. Deng, S. Fu, and S. Liu, "Codexglue: A machine learning benchmark dataset for code understanding and generation," in *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021, December 2021, virtual*, J. Vanschoren and S. Yeung, Eds., 2021. [Online]. Available: https://datasets-benchmarks-proceedings.neurips.cc/paper/2021/hash/c16a5320fa475530d9583c34fd356ef5-Abstract-round1.html

[213] V. Adlakha, S. Dhuliawala, K. Suleman, H. de Vries, and S. Reddy, "Topiocqa: Open-domain conversational question answering with topic switching," *Trans. Assoc. Comput. Linguistics*, vol. 10, pp. 468–483, 2022. [Online]. Available: https://doi.org/10.1162/tacl_a_00471

[214] T. Lin, M. Maire, S. J. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: common objects in context," in *Computer Vision - ECCV 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V*, ser. Lecture Notes in Computer Science, D. J. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds., vol. 8693. Springer, 2014, pp. 740–755. [Online]. Available: https://doi.org/10.1007/978-3-319-10602-1_48

[215] A. Awadalla, I. Gao, J. Gardner, J. Hessel, Y. Hanafy, W. Zhu, K. Marathe, Y. Bitton, S. Y. Gadre, S. Sagawa, J. Jitsev, S. Kornblith, P. W. Koh, G. Ilharco, M. Wortsman, and L. Schmidt, "Openflamingo: An open-source framework for training large autoregressive vision-language models," *CoRR*, vol. abs/2308.01390, 2023. [Online]. Available: https:

//doi.org/10.48550/arXiv.2308.01390

[216] J. Kaplan, S. McCandlish, T. Henighan, T. B. Brown, B. Chess, R. Child, S. Gray, A. Radford, J. Wu, and D. Amodei, "Scaling laws for neural language models," *CoRR*, vol. abs/2001.08361, 2020. [Online]. Available: https://arxiv.org/abs/2001.08361

[217] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat *et al.*, "Gpt-4 technical report," *arXiv preprint arXiv:2303.08774*, 2023.

[218] Y. Qin, K. Song, Y. Hu, W. Yao, S. Cho, X. Wang, X. Wu, F. Liu, P. Liu, and D. Yu, "Infobench: Evaluating instruction following ability in large language models," 2024.

[219] R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E. Hinton, "Adaptive mixtures of local experts," *Neural computation*, vol. 3, no. 1, pp. 79–87, 1991.

[220] W. Fedus, B. Zoph, and N. Shazeer, "Switch transformers: scaling to trillion parameter models with simple and efficient sparsity," *J. Mach. Learn. Res.*, vol. 23, no. 1, jan 2022.

[221] D. Lepikhin, H. Lee, Y. Xu, D. Chen, O. Firat, Y. Huang, M. Krikun, N. Shazeer, and Z. Chen, "Gshard: Scaling giant models with conditional computation and automatic sharding," *CoRR*, vol. abs/2006.16668, 2020. [Online]. Available: https://arxiv.org/abs/2006.16668

[222] N. Du, Y. Huang, A. M. Dai, S. Tong, D. Lepikhin, Y. Xu, M. Krikun, Y. Zhou, A. W. Yu, O. Firat, B. Zoph, L. Fedus, M. Bosma, Z. Zhou, T. Wang, Y. E. Wang, K. Webster, M. Pellat, K. Robinson, K. Meier-Hellstern, T. Duke, L. Dixon, K. Zhang, Q. V. Le, Y. Wu, Z. Chen, and C. Cui, "Glam: Efficient scaling of language models with mixture-of-experts," *CoRR*, vol. abs/2112.06905, 2021. [Online]. Available: https://arxiv.org/abs/2112.06905

[223] A. Clark, D. de Las Casas, A. Guy, A. Mensch, M. Paganini, J. Hoffmann, B. Damoc, B. A. Hechtman, T. Cai, S. Borgeaud, G. van den Driessche, E. Rutherford, T. Hennigan, M. J. Johnson, K. Millican, A. Cassirer, C. Jones, E. Buchatskaya, D. Budden, L. Sifre, S. Osindero, O. Vinyals, J. W. Rae, E. Elsen, K. Kavukcuoglu, and K. Simonyan, "Unified scaling laws for routed language models," *CoRR*, vol. abs/2202.01169, 2022. [Online]. Available: https://arxiv.org/abs/2202.01169

[224] D. Jiang, X. Ren, and B. Y. Lin, "Llm-blender: Ensembling large language models with pairwise comparison and generative fusion," in *Proceedings of the 61th Annual Meeting of the Association for Computational Linguistics (ACL 2023)*, 2023.

[225] S. Gunasekar, Y. Zhang, J. Aneja, C. C. T. Mendes, A. D. Giorno, S. Gopi, M. Javaheripi, P. Kauffmann, G. de Rosa, O. Saarikivi, A. Salim, S. Shah, H. S. Behl, X. Wang, S. Bubeck, R. Eldan, A. T. Kalai, Y. T. Lee, and Y. Li, "Textbooks are all you need," 2023.

[226] C. Zhou, P. Liu, P. Xu, S. Iyer, J. Sun, Y. Mao, X. Ma, A. Efrat, P. Yu, L. Yu, S. Zhang, G. Ghosh, M. Lewis, L. Zettlemoyer, and O. Levy, "LIMA: less is more for alignment," in *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, Eds., 2023. [Online]. Available: http://papers.nips.cc/paper_files/paper/2023/hash/ac662d74829e4407ce1d126477f4a03a-Abstract-Conference.html

[227] U. Manber and G. Myers, "Suffix arrays: a new method for on-line string searches," in *Proceedings of the First Annual ACM-SIAM Symposium on Discrete Algorithms*, ser. SODA '90. USA: Society for Industrial and Applied Mathematics, 1990, p. 319–327.

[228] A. Broder, "On the resemblance and containment of documents," in *Proceedings. Compression and Complexity of SEQUENCES 1997 (Cat. No.97TB100171)*, 1997, pp. 21–29.

[229] W. Sun, Z. Shi, S. Gao, P. Ren, M. de Rijke, and Z. Ren, "Contrastive learning reduces hallucination in conversations," 2022.

[230] R. Rafailov, A. Sharma, E. Mitchell, C. D. Manning, S. Ermon, and C. Finn, "Direct preference optimization: Your language model is secretly a reward model," in *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. [Online]. Available: https://openreview.net/forum?id=HPuSIXJaa9

[231] J. Wei, X. Wang, D. Schuurmans, M. Bosma, E. H. Chi, Q. Le, and D. Zhou, "Chain of thought prompting elicits reasoning in large language models," *CoRR*, vol. abs/2201.11903, 2022. [Online]. Available: https://arxiv.org/abs/2201.11903

[232] X. Wang, J. Wei, D. Schuurmans, Q. Le, E. Chi, S. Narang, A. Chowdhery, and D. Zhou, "Self-consistency improves chain of thought reasoning in language models," 2023.

[233] S. Yao, D. Yu, J. Zhao, I. Shafran, T. L. Griffiths, Y. Cao, and K. Narasimhan, "Tree of Thoughts: Deliberate problem solving with large language models," 2023.

[234] S. Dhuliawala, M. Komeili, J. Xu, R. Raileanu, X. Li, A. Celikyilmaz, and J. Weston, "Chain-of-verification reduces hallucination in large language models," 2023.

[235] N. Lee, W. Ping, P. Xu, M. Patwary, P. Fung, M. Shoeybi, and B. Catanzaro, "Factuality enhanced language models for open-ended text generation," in *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, Eds., 2022. [Online]. Available: http://papers.nips.cc/paper_files/paper/2022/hash/df438caa36714f69277daa92d608dd63-Abstract-Conference.html

[236] ——, "Factuality enhanced language models for open-ended text generation," 2023.

[237] D. Wan, M. Liu, K. McKeown, M. Dreyer, and M. Bansal, "Faithfulness-aware decoding strategies for abstractive summarization," in *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, A. Vlachos and I. Augenstein, Eds. Dubrovnik,

Croatia: Association for Computational Linguistics, May 2023, pp. 2864–2880. [Online]. Available: https://aclanthology.org/2023.eacl-main.210

[238] K. Li, O. Patel, F. Viégas, H. Pfister, and M. Wattenberg, "Inference-time intervention: Eliciting truthful answers from a language model," in *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. [Online]. Available: https://openreview.net/forum?id=aLLuYpn83y

[239] W. Shi, X. Han, M. Lewis, Y. Tsvetkov, L. Zettlemoyer, and S. W. tau Yih, "Trusting your evidence: Hallucinate less with context-aware decoding," 2023.

[240] L. Wang, X. Zhang, H. Su, and J. Zhu, "A comprehensive survey of continual learning: Theory, method and application," *CoRR*, vol. abs/2302.00487, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2302.00487

[241] T. Wu, L. Luo, Y.-F. Li, S. Pan, T.-T. Vu, and G. Haffari, "Continual learning for large language models: A survey," *arXiv preprint arXiv:2402.01364*, 2024.

[242] S. Wang, Y. Zhu, H. Liu, Z. Zheng, C. Chen, and J. Li, "Knowledge editing for large language models: A survey," *CoRR*, vol. abs/2310.16218, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2310.16218

[243] N. Zhang, Y. Yao, B. Tian, P. Wang, S. Deng, M. Wang, Z. Xi, S. Mao, J. Zhang, Y. Ni *et al.*, "A comprehensive study of knowledge editing for large language models," *arXiv preprint arXiv:2401.01286*, 2024.

[244] J. Jang, S. Ye, S. Yang, J. Shin, J. Han, G. Kim, S. J. Choi, and M. Seo, "Towards continual knowledge learning of language models," *arXiv preprint arXiv:2110.03215*, 2021.

[245] A. Cossu, T. Tuytelaars, A. Carta, L. Passaro, V. Lomonaco, and D. Bacciu, "Continual pre-training mitigates forgetting in language and vision," *arXiv preprint arXiv:2205.09357*, 2022.

[246] Y. Xie, K. Aggarwal, and A. Ahmad, "Efficient continual pre-training for building domain specific large language models," *arXiv preprint arXiv:2311.08545*, 2023.

[247] A. Razdaibiedina, Y. Mao, R. Hou, M. Khabsa, M. Lewis, and A. Almahairi, "Progressive prompts: Continual learning for language models," *arXiv preprint arXiv:2301.12314*, 2023.

[248] J. Jang, S. Kim, S. Ye, D. Kim, L. Logeswaran, M. Lee, K. Lee, and M. Seo, "Exploring the benefits of training expert language models over instruction tuning," in *International Conference on Machine Learning*. PMLR, 2023, pp. 14 702–14 729.

[249] A. Suhr and Y. Artzi, "Continual learning for instruction following from realtime feedback," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[250] X. Wang, T. Chen, Q. Ge, H. Xia, R. Bao, R. Zheng, Q. Zhang, T. Gui, and X. Huang, "Orthogonal subspace learning for language model continual learning," *arXiv preprint arXiv:2310.14152*, 2023.

[251] B. Peng, Z. Tian, S. Liu, M. Yang, and J. Jia, "Scalable language model with generalized continual learning," *arXiv preprint arXiv:2404.07470*, 2024.

[252] V. Mazzia, A. Pedrani, A. Caciolai, K. Rottmann, and D. Bernardi, "A survey on knowledge editing of neural networks," *CoRR*, vol. abs/2310.19704, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2310.19704

[253] Z. Huang, Y. Shen, X. Zhang, J. Zhou, W. Rong, and Z. Xiong, "Transformer-patcher: One mistake worth one neuron," in *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. [Online]. Available: https://openreview.net/pdf?id=4oYUGeGBPm

[254] T. Hartvigsen, S. Sankaranarayanan, H. Palangi, Y. Kim, and M. Ghassemi, "Aging with GRACE: lifelong model editing with discrete key-value adaptors," in *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, Eds., 2023. [Online]. Available: http://papers.nips.cc/paper_files/paper/2023/hash/95b6e2ff961580e03c0a662a63a71812-Abstract-Conference.html

[255] D. Dai, L. Dong, Y. Hao, Z. Sui, B. Chang, and F. Wei, "Knowledge neurons in pretrained transformers," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, S. Muresan, P. Nakov, and A. Villavicencio, Eds. Association for Computational Linguistics, 2022, pp. 8493–8502. [Online]. Available: https://doi.org/10.18653/v1/2022.acl-long.581

[256] K. Meng, A. S. Sharma, A. J. Andonian, Y. Belinkov, and D. Bau, "Mass-editing memory in a transformer," in *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. [Online]. Available: https://openreview.net/pdf?id=MkbcAHIYgyS

[257] X. Li, S. Li, S. Song, J. Yang, J. Ma, and J. Yu, "PMET: precise model editing in a transformer," in *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2014, February 20-27, 2024, Vancouver, Canada*, M. J. Wooldridge, J. G. Dy, and S. Natarajan, Eds. AAAI Press, 2024, pp. 18 564–18 572. [Online]. Available: https://doi.org/10.1609/aaai.v38i17.29818

[258] J. Ma, J. Gu, Z. Ling, Q. Liu, and C. Liu, "Untying the reversal curse via bidirectional language model editing," *CoRR*, vol. abs/2310.10322, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2310.10322

[259] C. Tan, G. Zhang, and J. Fu, "Massive editing for large language models via meta learning," *CoRR*, vol. abs/2311.04661, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2311.04661

[260] G. Izacard and E. Grave, "Leveraging passage retrieval with generative models for open domain question answering," *arXiv preprint arXiv:2007.01282*, 2020.

[261] Y. Zhou, Z. Liu, J. Jin, J. Nie, and Z. Dou, "Metacognitive retrieval-augmented large language models," *CoRR*, vol. abs/2402.11626, 2024. [Online].

Available: https://doi.org/10.48550/arXiv.2402.11626
[262] Y. Gao, Y. Xiong, X. Gao, K. Jia, J. Pan, Y. Bi, Y. Dai, J. Sun, Q. Guo, M. Wang, and H. Wang, "Retrieval-augmented generation for large language models: A survey," 2024.
[263] K. Guu, K. Lee, Z. Tung, P. Pasupat, and M. Chang, "Retrieval augmented language model pre-training," in *International conference on machine learning*. PMLR, 2020, pp. 3929–3938.
[264] S. Borgeaud, A. Mensch, J. Hoffmann, T. Cai, E. Rutherford, K. Millican, G. van den Driessche, J. Lespiau, B. Damoc, A. Clark, D. de Las Casas, A. Guy, J. Menick, R. Ring, T. Hennigan, S. Huang, L. Maggiore, C. Jones, A. Cassirer, A. Brock, M. Paganini, G. Irving, O. Vinyals, S. Osindero, K. Simonyan, J. W. Rae, E. Elsen, and L. Sifre, "Improving language models by retrieving from trillions of tokens," in *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, ser. Proceedings of Machine Learning Research, K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvári, G. Niu, and S. Sabato, Eds., vol. 162. PMLR, 2022, pp. 2206–2240. [Online]. Available: https://proceedings.mlr.press/v162/borgeaud22a.html
[265] O. Ram, Y. Levine, I. Dalmedigos, D. Muhlgay, A. Shashua, K. Leyton-Brown, and Y. Shoham, "In-context retrieval-augmented language models," *arXiv preprint arXiv:2302.00083*, 2023.
[266] Z. Yu, C. Xiong, S. Yu, and Z. Liu, "Augmentation-adapted retriever improves generalization of language models as generic plug-in," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, A. Rogers, J. Boyd-Graber, and N. Okazaki, Eds. Toronto, Canada: Association for Computational Linguistics, Jul. 2023, pp. 2421–2436. [Online]. Available: https://aclanthology.org/2023.acl-long.136
[267] P. Zhang, S. Xiao, Z. Liu, Z. Dou, and J. Nie, "Retrieve anything to augment large language models," *CoRR*, vol. abs/2310.07554, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2310.07554
[268] N. F. Liu, K. Lin, J. Hewitt, A. Paranjape, M. Bevilacqua, F. Petroni, and P. Liang, "Lost in the middle: How language models use long contexts," 2023, arXiv:2307.03172.
[269] F. Cuconasu, G. Trappolini, F. Siciliano, S. Filice, C. Campagnano, Y. Maarek, N. Tonellotto, and F. Silvestri, "The power of noise: Redefining retrieval for rag systems," 2024.
[270] J. Jin, Y. Zhu, Y. Zhou, and Z. Dou, "BIDER: bridging knowledge inconsistency for efficient retrieval-augmented llms via key supporting evidence," *CoRR*, vol. abs/2402.12174, 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2402.12174
[271] F. Xu, W. Shi, and E. Choi, "RECOMP: improving retrieval-augmented lms with compression and selective augmentation," *CoRR*, vol. abs/2310.04408, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2310.04408
[272] H. Jiang, Q. Wu, C.-Y. Lin, Y. Yang, and L. Qiu, "LLMLingua: Compressing prompts for accelerated inference of large language models," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Dec. 2023, pp. 13 358–13 376. [Online]. Available: https://aclanthology.org/2023.emnlp-main.825
[273] H. Jiang, Q. Wu, , X. Luo, D. Li, C.-Y. Lin, Y. Yang, and L. Qiu, "Longllmlingua: Accelerating and enhancing llms in long context scenarios via prompt compression," *ArXiv preprint*, vol. abs/2310.06839, 2023. [Online]. Available: https://arxiv.org/abs/2310.06839
[274] H. Yang, Z. Li, Y. Zhang, J. Wang, N. Cheng, M. Li, and J. Xiao, "PRCA: fitting black-box large language models for retrieval question answering via pluggable reward-driven contextual adapter," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, H. Bouamor, J. Pino, and K. Bali, Eds. Association for Computational Linguistics, 2023, pp. 5364–5375. [Online]. Available: https://aclanthology.org/2023.emnlp-main.326
[275] T. Lan, D. Cai, Y. Wang, H. Huang, and X.-L. Mao, "Copy is all you need," in *The Eleventh International Conference on Learning Representations*, 2023. [Online]. Available: https://openreview.net/forum?id=CROlOA9Nd8C
[276] O. Press, M. Zhang, S. Min, L. Schmidt, N. Smith, and M. Lewis, "Measuring and narrowing the compositionality gap in language models," in *Findings of the Association for Computational Linguistics: EMNLP 2023*, H. Bouamor, J. Pino, and K. Bali, Eds. Singapore: Association for Computational Linguistics, Dec. 2023, pp. 5687–5711. [Online]. Available: https://aclanthology.org/2023.findings-emnlp.378
[277] O. Yoran, T. Wolfson, O. Ram, and J. Berant, "Making retrieval-augmented language models robust to irrelevant context," 2023.
[278] M. Komeili, K. Shuster, and J. Weston, "Internet-augmented dialogue generation," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, S. Muresan, P. Nakov, and A. Villavicencio, Eds. Dublin, Ireland: Association for Computational Linguistics, May 2022, pp. 8460–8478. [Online]. Available: https://aclanthology.org/2022.acl-long.579
[279] K. Shuster, J. Xu, M. Komeili, D. Ju, E. M. Smith, S. Roller, M. Ung, M. Chen, K. Arora, J. Lane, M. Behrooz, W. Ngan, S. Poff, N. Goyal, A. Szlam, Y.-L. Boureau, M. Kambadur, and J. Weston, "Blenderbot 3: a deployed conversational agent that continually learns to responsibly engage," 2022.
[280] Y. Song, W. Xiong, D. Zhu, W. Wu, H. Qian, M. Song, H. Huang, C. Li, K. Wang, R. Yao, Y. Tian, and S. Li, "Restgpt: Connecting large language models with real-world restful apis," 2023.
[281] S. G. Patil, T. Zhang, X. Wang, and J. E. Gonzalez, "Gorilla: Large language model connected with massive apis," 2023.
[282] S. Hao, T. Liu, Z. Wang, and Z. Hu, "Toolkengpt: Augmenting frozen language models with massive

tools via tool embeddings," in *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, Eds., 2023. [Online]. Available: http://papers.nips.cc/paper_files/paper/2023/hash/8fd1a81c882cd45f64958da6284f4a3f-Abstract-Conference.html

[283] C. Qian, C. Han, Y. Fung, Y. Qin, Z. Liu, and H. Ji, "CREATOR: Tool creation for disentangling abstract and concrete reasoning of large language models," in *Findings of the Association for Computational Linguistics: EMNLP 2023*, H. Bouamor, J. Pino, and K. Bali, Eds. Singapore: Association for Computational Linguistics, Dec. 2023, pp. 6922–6939. [Online]. Available: https://aclanthology.org/2023.findings-emnlp.462

[284] D. Surís, S. Menon, and C. Vondrick, "Vipergpt: Visual inference via python execution for reasoning," *arXiv preprint arXiv:2303.08128*, 2023.

[285] J. Huang and K. C. Chang, "Citation: A key to building responsible and accountable large language models," *CoRR*, vol. abs/2307.02185, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2307.02185

[286] R. Litschko, M. Müller-Eberstein, R. van der Goot, L. Weber-Genzel, and B. Plank, "Establishing trustworthiness: Rethinking tasks and model evaluation," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, H. Bouamor, J. Pino, and K. Bali, Eds. Association for Computational Linguistics, 2023, pp. 193–203. [Online]. Available: https://doi.org/10.18653/v1/2023.emnlp-main.14

[287] X. Ye, R. Sun, S. Ö. Arik, and T. Pfister, "Effective large language model adaptation for improved grounding," *CoRR*, vol. abs/2311.09533, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2311.09533

[288] M. Besta, N. Blach, A. Kubicek, R. Gerstenberger, M. Podstawski, L. Gianinazzi, J. Gajda, T. Lehmann, H. Niewiadomski, P. Nyczyk, and T. Hoefler, "Graph of thoughts: Solving elaborate problems with large language models," in *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2014, February 20-27, 2024, Vancouver, Canada*, M. J. Wooldridge, J. G. Dy, and S. Natarajan, Eds. AAAI Press, 2024, pp. 17 682–17 690. [Online]. Available: https://doi.org/10.1609/aaai.v38i16.29720

[289] Y. Fang, S. W. Thomas, and X. Zhu, "HGOT: hierarchical graph of thoughts for retrieval-augmented in-context learning in factuality evaluation," *CoRR*, vol. abs/2402.09390, 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2402.09390

[290] C. Huang, Z. Wu, Y. Hu, and W. Wang, "Training language models to generate text with citations via fine-grained rewards," *CoRR*, vol. abs/2402.04315, 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2402.04315

[291] A. Chen, P. Pasupat, S. Singh, H. Lee, and K. Guu, "PURR: efficiently editing language model hallucinations by denoising language model corruptions," *CoRR*, vol. abs/2305.14908, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2305.14908

[292] A. Slobodkin, E. Hirsch, A. Cattan, T. Schuster, and I. Dagan, "Attribute first, then generate: Locally-attributable grounded text generation," *CoRR*, vol. abs/2403.17104, 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2403.17104

[293] Y. Zhou, Z. Dou, and J. Wen, "Encoding history with context-aware representation learning for personalized search," in *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020*, J. X. Huang, Y. Chang, X. Cheng, J. Kamps, V. Murdock, J. Wen, and Y. Liu, Eds. ACM, 2020, pp. 1111–1120. [Online]. Available: https://doi.org/10.1145/3397271.3401175

[294] S. Wang, Z. Dou, J. Yao, Y. Zhou, and J. Wen, "Incorporating explicit subtopics in personalized search," in *Proceedings of the ACM Web Conference 2023, WWW 2023, Austin, TX, USA, 30 April 2023 - 4 May 2023*, Y. Ding, J. Tang, J. F. Sequeda, L. Aroyo, C. Castillo, and G. Houben, Eds. ACM, 2023, pp. 3364–3374. [Online]. Available: https://doi.org/10.1145/3543507.3583488

[295] Y. Zhou, Q. Zhu, J. Jin, and Z. Dou, "Cognitive personalized search integrating large language models with an efficient memory mechanism," *CoRR*, vol. abs/2402.10548, 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2402.10548

[296] W. Liu, Y. Zhou, Y. Zhu, and Z. Dou, "How to personalize and whether to personalize? candidate documents decide," 2023.

[297] Z. Ma, Z. Dou, Y. Zhu, H. Zhong, and J. Wen, "One chatbot per person: Creating personalized chatbots based on implicit user profiles," in *SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021*, F. Diaz, C. Shah, T. Suel, P. Castells, R. Jones, and T. Sakai, Eds. ACM, 2021, pp. 555–564. [Online]. Available: https://doi.org/10.1145/3404835.3462828

[298] J. Liu, C. Liu, R. Lv, K. Zhou, and Y. Zhang, "Is chatgpt a good recommender? A preliminary study," *CoRR*, vol. abs/2304.10149, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2304.10149

[299] S. Dai, N. Shao, H. Zhao, W. Yu, Z. Si, C. Xu, Z. Sun, X. Zhang, and J. Xu, "Uncovering chatgpt's capabilities in recommender systems," in *Proceedings of the 17th ACM Conference on Recommender Systems, RecSys 2023, Singapore, Singapore, September 18-22, 2023*, J. Zhang, L. Chen, S. Berkovsky, M. Zhang, T. D. Noia, J. Basilico, L. Pizzato, and Y. Song, Eds. ACM, 2023, pp. 1126–1132. [Online]. Available: https://doi.org/10.1145/3604915.3610646

[300] A. Zhiyuli, Y. Chen, X. Zhang, and X. Liang, "Bookgpt: A general framework for book recommendation empowered by large language model," *CoRR*, vol. abs/2305.15673, 2023. [Online]. Available: https:

//doi.org/10.48550/arXiv.2305.15673

[301] A. Salemi, S. Mysore, M. Bendersky, and H. Zamani, “Lamp: When large language models meet personalization,” *CoRR*, vol. abs/2304.11406, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2304.11406

[302] K. Christakopoulou, A. Lalama, C. Adams, I. Qu, Y. Amir, S. Chucri, P. Vollucci, F. Soldo, D. Bseiso, S. Scodel, L. Dixon, E. H. Chi, and M. Chen, “Large language models for user interest journeys,” *CoRR*, vol. abs/2305.15498, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2305.15498

[303] D. Wang, K. Yang, H. Zhu, X. Yang, A. Cohen, L. Li, and Y. Tian, “Learning personalized story evaluation,” *CoRR*, vol. abs/2310.03304, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2310.03304

[304] C. Li, M. Zhang, Q. Mei, W. Kong, and M. Bendersky, “Automatic prompt rewriting for personalized text generation,” *CoRR*, vol. abs/2310.00152, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2310.00152

[305] Z. Chen, L. Zhang, F. Weng, L. Pan, and Z. Lan, “Tailored visions: Enhancing text-to-image generation with personalized prompt rewriting,” *CoRR*, vol. abs/2310.08129, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2310.08129

[306] P. Mazaré, S. Humeau, M. Raison, and A. Bordes, “Training millions of personalized dialogue agents,” in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, E. Riloff, D. Chiang, J. Hockenmaier, and J. Tsujii, Eds. Association for Computational Linguistics, 2018, pp. 2775–2779. [Online]. Available: https://doi.org/10.18653/v1/d18-1298

[307] T. Fu, X. Zhao, C. Tao, J. Wen, and R. Yan, “There are a thousand hamlets in a thousand people’s eyes: Enhancing knowledge-grounded dialogue with personal memory,” in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, S. Muresan, P. Nakov, and A. Villavicencio, Eds. Association for Computational Linguistics, 2022, pp. 3901–3913. [Online]. Available: https://doi.org/10.18653/v1/2022.acl-long.270

[308] P. Cheng, J. Xie, K. Bai, Y. Dai, and N. Du, “Everyone deserves A reward: Learning customized human preferences,” *CoRR*, vol. abs/2309.03126, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2309.03126

[309] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, “Proximal policy optimization algorithms,” *CoRR*, vol. abs/1707.06347, 2017. [Online]. Available: http://arxiv.org/abs/1707.06347

[310] Q. Huang, S. Fu, X. Liu, W. Wang, T. Ko, Y. Zhang, and L. H. Y. Tang, “Learning retrieval augmentation for personalized dialogue generation,” in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, H. Bouamor, J. Pino, and K. Bali, Eds. Association for Computational Linguistics, 2023, pp. 2523–2540. [Online]. Available: https://doi.org/10.18653/v1/2023.emnlp-main.154

[311] C. Li, M. Zhang, Q. Mei, Y. Wang, S. A. Hombaiah, Y. Liang, and M. Bendersky, “Teach llms to personalize - an approach inspired by writing education,” *CoRR*, vol. abs/2308.07968, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2308.07968

[312] S. Wozniak, B. Koptyra, A. Janz, P. Kazienko, and J. Kocon, “Personalized large language models,” *CoRR*, vol. abs/2402.09269, 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2402.09269

[313] X. Shen, R. Zhang, X. Zhao, J. Zhu, and X. Xiao, “Pmg: Personalized multimodal generation with large language models,” *arXiv preprint arXiv:2404.08677*, 2024.

[314] X. Liu, D. McDuff, G. Kovacs, I. R. Galatzer-Levy, J. E. Sunshine, J. Zhan, M. Poh, S. Liao, P. D. Achille, and S. N. Patel, “Large language models are few-shot health learners,” *CoRR*, vol. abs/2305.15525, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2305.15525

[315] J. Zhang, K. Sun, A. Jagadeesh, M. Ghahfarokhi, D. Gupta, A. Gupta, V. Gupta, and Y. Guo, “The potential and pitfalls of using a large language model such as chatgpt or GPT-4 as a clinical assistant,” *CoRR*, vol. abs/2307.08152, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2307.08152

[316] S. Yang, H. Zhao, S. Zhu, G. Zhou, H. Xu, Y. Jia, and H. Zan, “Zhongjing: Enhancing the chinese medical capabilities of large language model through expert feedback and real-world multi-turn dialogue,” in *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2014, February 20-27, 2024, Vancouver, Canada*, M. J. Wooldridge, J. G. Dy, and S. Natarajan, Eds. AAAI Press, 2024, pp. 19 368–19 376. [Online]. Available: https://doi.org/10.1609/aaai.v38i17.29907

[317] M. Abbasian, I. Azimi, A. M. Rahmani, and R. C. Jain, “Conversational health agents: A personalized llm-powered agent framework,” *CoRR*, vol. abs/2310.02374, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2310.02374

[318] X. Tang, A. Zou, Z. Zhang, Y. Zhao, X. Zhang, A. Cohan, and M. Gerstein, “Medagents: Large language models as collaborators for zero-shot medical reasoning,” *CoRR*, vol. abs/2311.10537, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2311.10537

[319] T. Lai, Y. Shi, Z. Du, J. Wu, K. Fu, Y. Dou, and Z. Wang, “Psy-llm: Scaling up global mental health psychological services with ai-based large language models,” *CoRR*, vol. abs/2307.11991, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2307.11991

[320] S. Porsdam Mann, B. D. Earp, N. Møller, S. Vynn, and J. Savulescu, “Autogen: A personalized large language model for academic enhancement—ethics and proof of principle,” *The American Journal of Bioethics*, vol. 23, no. 10, pp. 28–41, 2023.

[321] P. Cui and M. Sachan, “Adaptive and personalized exercise generation for online language learning,”

in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, A. Rogers, J. L. Boyd-Graber, and N. Okazaki, Eds. Association for Computational Linguistics, 2023, pp. 10 184–10 198. [Online]. Available: https://doi.org/10.18653/v1/2023.acl-long.567

[322] B. P. Majumder, S. Li, J. Ni, and J. J. McAuley, "Generating personalized recipes from historical user preferences," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, K. Inui, J. Jiang, V. Ng, and X. Wan, Eds. Association for Computational Linguistics, 2019, pp. 5975–5981. [Online]. Available: https://doi.org/10.18653/v1/D19-1613

[323] K. Zhang, G. Lu, G. Zhang, Z. Lei, and L. Wu, "Personalized headline generation with enhanced user interest perception," in *Artificial Neural Networks and Machine Learning - ICANN 2022 - 31st International Conference on Artificial Neural Networks, Bristol, UK, September 6-9, 2022, Proceedings, Part II*, ser. Lecture Notes in Computer Science, E. Pimenidis, P. P. Angelov, C. Jayne, A. Papaleonidas, and M. Aydin, Eds., vol. 13530. Springer, 2022, pp. 797–809. [Online]. Available: https://doi.org/10.1007/978-3-031-15931-2_65

[324] J. Wu, R. Antonova, A. Kan, M. Lepert, A. Zeng, S. Song, J. Bohg, S. Rusinkiewicz, and T. A. Funkhouser, "Tidybot: personalized robot assistance with large language models," *Auton. Robots*, vol. 47, no. 8, pp. 1087–1102, 2023. [Online]. Available: https://doi.org/10.1007/s10514-023-10139-z

[325] M. Zhu, "Recall, precision and average precision," *Department of Statistics and Actuarial Science, University of Waterloo, Waterloo*, vol. 2, no. 30, p. 6, 2004.

[326] Z. Guo and D. Barbosa, "Robust named entity disambiguation with random walks," *Semantic Web*, vol. 9, no. 4, pp. 459–479, 2018.

[327] H. Elsahar, P. Vougiouklis, A. Remaci, C. Gravier, J. Hare, F. Laforest, and E. Simperl, "T-rex: A large scale alignment of natural language with knowledge base triples," in *LREC*, 2018.

[328] Z. Yang, P. Qi, S. Zhang, Y. Bengio, W. Cohen, R. Salakhutdinov, and C. D. Manning, "HotpotQA: A dataset for diverse, explainable multi-hop question answering," in *EMNLP*. Brussels, Belgium: Association for Computational Linguistics, Oct.-Nov. 2018, pp. 2369–2380. [Online]. Available: https://aclanthology.org/D18-1259

[329] A. Fan, Y. Jernite, E. Perez, D. Grangier, J. Weston, and M. Auli, "ELI5: Long form question answering," in *ACL*. Florence, Italy: Association for Computational Linguistics, Jul. 2019, pp. 3558–3567. [Online]. Available: https://aclanthology.org/P19-1346

[330] P. Rajpurkar, R. Jia, and P. Liang, "Know what you don't know: Unanswerable questions for squad," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 2: Short Papers*, I. Gurevych and Y. Miyao, Eds. Association for Computational Linguistics, 2018, pp. 784–789. [Online]. Available: https://aclanthology.org/P18-2124/

[331] S. Banerjee and A. Lavie, "METEOR: an automatic metric for MT evaluation with improved correlation with human judgments," in *Proceedings of the Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization@ACL 2005, Ann Arbor, Michigan, USA, June 29, 2005*, J. Goldstein, A. Lavie, C. Lin, and C. R. Voss, Eds. Association for Computational Linguistics, 2005, pp. 65–72. [Online]. Available: https://aclanthology.org/W05-0909/

[332] W. Yuan, G. Neubig, and P. Liu, "Bartscore: Evaluating generated text as text generation," in *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, M. Ranzato, A. Beygelzimer, Y. N. Dauphin, P. Liang, and J. W. Vaughan, Eds., 2021, pp. 27 263–27 277. [Online]. Available: https://proceedings.neurips.cc/paper/2021/hash/e4d2b6e6fdeca3e60e0f1a62fee3d9dd-Abstract.html

[333] Y. Chang, X. Wang, J. Wang, Y. Wu, K. Zhu, H. Chen, L. Yang, X. Yi, C. Wang, Y. Wang, W. Ye, Y. Zhang, Y. Chang, P. S. Yu, Q. Yang, and X. Xie, "A survey on evaluation of large language models," *CoRR*, vol. abs/2307.03109, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2307.03109

[334] K. Singhal, S. Azizi, T. Tu, S. S. Mahdavi, J. Wei, H. W. Chung, N. Scales, A. K. Tanwani, H. Cole-Lewis, S. Pfohl, P. Payne, M. Seneviratne, P. Gamble, C. Kelly, N. Schärli, A. Chowdhery, P. A. Mansfield, B. A. y Arcas, D. R. Webster, G. S. Corrado, Y. Matias, K. Chou, J. Gottweis, N. Tomasev, Y. Liu, A. Rajkomar, J. K. Barral, C. Semturs, A. Karthikesalingam, and V. Natarajan, "Large language models encode clinical knowledge," *CoRR*, vol. abs/2212.13138, 2022. [Online]. Available: https://doi.org/10.48550/arXiv.2212.13138

[335] M. Zhong, Y. Liu, D. Yin, Y. Mao, Y. Jiao, P. Liu, C. Zhu, H. Ji, and J. Han, "Towards a unified multi-dimensional evaluator for text generation," in *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, Y. Goldberg, Z. Kozareva, and Y. Zhang, Eds. Association for Computational Linguistics, 2022, pp. 2023–2038. [Online]. Available: https://doi.org/10.18653/v1/2022.emnlp-main.131

[336] C. van der Lee, A. Gatt, E. van Miltenburg, S. Wubben, and E. Krahmer, "Best practices for the human evaluation of automatically generated text," in *Proceedings of the 12th International Conference on Natural Language Generation, INLG 2019, Tokyo, Japan, October 29 - November 1, 2019*, K. van Deemter, C. Lin, and H. Takamura, Eds. Association for Computational Linguistics, 2019, pp. 355–368. [Online]. Available: https://aclanthology.org/W19-8643/

[337] J. Ji, M. Liu, J. Dai, X. Pan, C. Zhang, C. Bian, B. Chen,

R. Sun, Y. Wang, and Y. Yang, “Beavertails: Towards improved safety alignment of LLM via a human-preference dataset,” in *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, Eds., 2023. [Online]. Available: http://papers.nips.cc/paper_files/paper/2023/hash/4dbb61cb68671edc4ca3712d70083b9f-Abstract-Datasets_and_Benchmarks.html

[338] Y. Huang, Y. Bai, Z. Zhu, J. Zhang, J. Zhang, T. Su, J. Liu, C. Lv, Y. Zhang, J. Lei, Y. Fu, M. Sun, and J. He, “C-eval: A multi-level multi-discipline chinese evaluation suite for foundation models,” in *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, Eds., 2023. [Online]. Available: http://papers.nips.cc/paper_files/paper/2023/hash/c6ec1844bec96d6d32ae95ae694e23d8-Abstract-Datasets_and_Benchmarks.html

[339] Q. Cheng, T. Sun, W. Zhang, S. Wang, X. Liu, M. Zhang, J. He, M. Huang, Z. Yin, K. Chen, and X. Qiu, “Evaluating hallucinations in chinese large language models,” *CoRR*, vol. abs/2310.03368, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2310.03368

[340] C. Wang, X. Liu, Y. Yue, X. Tang, T. Zhang, J. Cheng, Y. Yao, W. Gao, X. Hu, Z. Qi, Y. Wang, L. Yang, J. Wang, X. Xie, Z. Zhang, and Y. Zhang, “Survey on factuality in large language models: Knowledge, retrieval and domain-specificity,” *CoRR*, vol. abs/2310.07521, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2310.07521

[341] J. Wei, Y. Tay, R. Bommasani, C. Raffel, B. Zoph, S. Borgeaud, D. Yogatama, M. Bosma, D. Zhou, D. Metzler, E. H. Chi, T. Hashimoto, O. Vinyals, P. Liang, J. Dean, and W. Fedus, “Emergent abilities of large language models,” *Trans. Mach. Learn. Res.*, vol. 2022, 2022. [Online]. Available: https://openreview.net/forum?id=yzkSU5zdwD

[342] R. Schaeffer, B. Miranda, and S. Koyejo, “Are emergent abilities of large language models a mirage?” in *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, Eds., 2023. [Online]. Available: http://papers.nips.cc/paper_files/paper/2023/hash/adc98a266f45005c403b8311ca7e8bd7-Abstract-Conference.html

[343] Y. Sun, L. Dong, S. Huang, S. Ma, Y. Xia, J. Xue, J. Wang, and F. Wei, “Retentive network: A successor to transformer for large language models,” *CoRR*, vol. abs/2307.08621, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2307.08621

[344] A. Gu and T. Dao, “Mamba: Linear-time sequence modeling with selective state spaces,” *CoRR*, vol. abs/2312.00752, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2312.00752

[345] H. Yen, T. Gao, and D. Chen, “Long-context language modeling with parallel context encoding,” *CoRR*, vol. abs/2402.16617, 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2402.16617

[346] J. Tan, Z. Dou, Y. Zhu, P. Guo, K. Fang, and J. Wen, “Small models, big insights: Leveraging slim proxy models to decide when and what to retrieve for llms,” *CoRR*, vol. abs/2402.12052, 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2402.12052

[347] Z. Wang, Z. Cheng, H. Zhu, D. Fried, and G. Neubig, “What are tools anyway? A survey from the language model perspective,” *CoRR*, vol. abs/2403.15452, 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2403.15452

[348] T. Wu, M. Caccia, Z. Li, Y. Li, G. Qi, and G. Haffari, “Pretrained language model in continual learning: A comparative study,” in *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. [Online]. Available: https://openreview.net/forum?id=figzpGMrdD

[349] O. Ovadia, M. Brief, M. Mishaeli, and O. Elisha, “Fine-tuning or retrieval? comparing knowledge injection in llms,” *CoRR*, vol. abs/2312.05934, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2312.05934

[350] Y. Yao, P. Wang, B. Tian, S. Cheng, Z. Li, S. Deng, H. Chen, and N. Zhang, “Editing large language models: Problems, methods, and opportunities,” in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, H. Bouamor, J. Pino, and K. Bali, Eds. Association for Computational Linguistics, 2023, pp. 10 222–10 240. [Online]. Available: https://doi.org/10.18653/v1/2023.emnlp-main.632

[351] I. O. Gallegos, R. A. Rossi, J. Barrow, M. M. Tanjim, S. Kim, F. Dernoncourt, T. Yu, R. Zhang, and N. K. Ahmed, “Bias and fairness in large language models: A survey,” *CoRR*, vol. abs/2309.00770, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2309.00770

[352] J. Dien, “Generative artificial intelligence as a plagiarism problem,” p. 108621, 2023.

[353] K. Kenthapadi, H. Lakkaraju, and N. Rajani, “Generative ai meets responsible ai: Practical challenges and opportunities,” in *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2023, pp. 5805–5806.

[354] N. Carlini, F. Tramèr, E. Wallace, M. Jagielski, A. Herbert-Voss, K. Lee, A. Roberts, T. B. Brown, D. Song, Ú. Erlingsson, A. Oprea, and C. Raffel, “Extracting training data from large language models,” in *30th USENIX Security Symposium, USENIX Security 2021, August 11-13, 2021*, M. D. Bailey and R. Greenstadt, Eds. USENIX Association, 2021, pp. 2633–2650. [Online]. Available: https://www.usenix.org/conference/usenixsecurity21/presentation/carlini-extracting

[355] J. Huang, H. Shao, and K. C. Chang, “Are large pre-trained language models leaking your

personal information?" in *Findings of the Association for Computational Linguistics: EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, Y. Goldberg, Z. Kozareva, and Y. Zhang, Eds. Association for Computational Linguistics, 2022, pp. 2038–2047. [Online]. Available: https://doi.org/10.18653/v1/2022.findings-emnlp.148

[356] N. F. Liu, T. Zhang, and P. Liang, "Evaluating verifiability in generative search engines," in *Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023*, H. Bouamor, J. Pino, and K. Bali, Eds. Association for Computational Linguistics, 2023, pp. 7001–7025. [Online]. Available: https://doi.org/10.18653/v1/2023.findings-emnlp.467

[357] N. Carlini, C. Liu, Ú. Erlingsson, J. Kos, and D. Song, "The secret sharer: Evaluating and testing unintended memorization in neural networks," in *28th USENIX Security Symposium, USENIX Security 2019, Santa Clara, CA, USA, August 14-16, 2019*, N. Heninger and P. Traynor, Eds. USENIX Association, 2019, pp. 267–284. [Online]. Available: https://www.usenix.org/conference/usenixsecurity19/presentation/carlini

[358] X. Yang, L. Pan, X. Zhao, H. Chen, L. R. Petzold, W. Y. Wang, and W. Cheng, "A survey on detection of llms-generated content," *CoRR*, vol. abs/2310.15654, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2310.15654

[359] L. Wang, N. Yang, X. Huang, L. Yang, R. Majumder, and F. Wei, "Large search model: Redefining search stack in the era of llms," *SIGIR Forum*, vol. 57, no. 2, pp. 23:1–23:16, 2023. [Online]. Available: https://doi.org/10.1145/3642979.3643006