Retrieval-Augmented Knowledge Integration into Language Models: A Survey

Yuxuan Chen[1] **Daniel Röder**[1] **Justus-Jonas Erker**[2,1]
Leonhard Hennig[1] **Philippe Thomas**[1] **Sebastian Möller**[1] **Roland Roller**[1]
[1]German Research Center for Artificial Intelligence (DFKI)
[2] UKP Lab, TU Darmstadt & Hessian.Ai
[1]{*yuxuan.chen, daniel.roeder, leonhard.hennig, philippe.thomas, sebastian.moeller, roland.roller*}*@dfki.de*
[2]*justus-jonas.erker@tu-darmstadt.de*

Abstract

This survey analyses how external knowledge can be integrated into language models in the context of retrieval-augmentation. The main goal of this work is to give an overview of: (1) Which external knowledge can be augmented? (2) Given a knowledge source, how to retrieve from it and then integrate the retrieved knowledge? To achieve this, we define and give a mathematical formulation of retrieval-augmented knowledge integration (RAKI). We discuss *retrieval* and *integration* techniques separately in detail, for each of the following knowledge formats: knowledge graph, tabular and natural language.

1 Introduction

In natural language processing (NLP), external knowledge or information refers to information that is not explicitly present in the language model (LM) input yet helpful for LMs to produce target output (Zhu et al., 2022). Traditional methods to integrate knowledge, especially those before large language models (LLMs) (Touvron et al., 2023; Chowdhery et al., 2023), include pre-training over a knowledge corpus (Beltagy et al., 2019; Huang et al., 2019; Chalkidis et al., 2020), and fine-tuning in the domain that the knowledge is concerned with (Huang et al., 2019). Despite improved performance of the resulting models (Yin et al., 2022), such methods typically require (re-)training on the whole (without filtering) knowledge. This is not efficient, as the ever-growing size of language models (Chowdhery et al., 2023) raises hardware and energy issues (Bannour et al., 2021; Treviso et al., 2023) of applying these training-intensive methods originally proposed for smaller models.

As an alternative to traditional pre-training and fine-tuning to integrate knowledge into LLMs, retrieval-augmented (RA) methods (Karpukhin et al., 2020; Yu et al., 2023) have become more and more popular in recent years. RA methods leverage pre-trained *internal* knowledge already parameterized in LMs as well as retrieved *external* knowledge (Lewis et al., 2020). In the setting of retrieval augmentation, LMs access for instance only the most relevant, top-k retrieved items without seeing the entire external sources, thus enabling efficiency (Cai et al., 2022). Previous works also demonstrate decoupling knowledge and language model can lead to better adaptability (Long et al., 2023), straightforward knowledge edit (Zheng et al., 2023; Ovadia et al., 2023) and improved explainability (Samarinas et al., 2021).

To track the research intersection of retrieving knowledge to augment LMs, we study the topic of retrieval-augmented knowledge integration (RAKI) in this survey. In RAKI, the retrieval base is some specific external knowledge (Baek et al., 2023b) (e.g. a knowledge graph or a set of Wikipedia articles), where the knowledge is typically written by experts and thus enjoys higher factuality than general texts. This survey is mainly based on recent (2018-2024) publications (See Appendix A.1, A.2 for more details of literature). Inspired by Hu et al. (2024), we categorize the published works in this line of research based on the format of knowledge source: knowledge graph, tabular and natural language. For each knowledge source, we start by introducing the source format using the annotations proposed in Section 2. Then, we discuss in detail the retrieval and integration techniques proposed in the reviewed methods. Finally, we point out the challenges of RAKI and list some relevant work to deal with them. We would like to point out that this survey aims to focus on (pure) NLP and does not consider work on vision (Yang et al., 2021; Lin and Byrne, 2022) or audio (Zhao et al., 2023a).

2 Preliminaries

In the following, we briefly introduce retrieval-augmented generation (RAG) and then define and formulate retrieval-augmented knowledge integration (RAKI).

Retrieval-augmented generation is first proposed by Lewis et al. (2020), where world knowledge is retrieved from a vector index constructed over Wikipedia articles and then sent to a seq2seq (Sutskever et al., 2014) model for generation. More formally, given an input-output pair (x, y) from a generation task, retrieval-augmented generation aims to generate the target output y conditioned on the input x and an accessible document set $\mathcal{Z}$ for reference (Lewis et al., 2020; Yu, 2022).

Retrieval-augmented knowledge integration Baek et al. (2023b) uses the term *knowledge augmentation* to address the practice of retrieving knowledge for language models. In this work, we adopt the term *retrieval-augmented knowledge integration* (RAKI) for better clarification, since we would like to avoid confusion with non-retrieval based knowledge-integration methods, as mentioned in Section 1, that involve heavy pre-training or fine-tuning. RAKI also follows the first-retrieve-then-infer paradigm as in RAG, and we identify the differences as follows: (1) RAG, by its nature, deals with generation tasks, while RAKI is compatible with classification tasks as well, i.e. y being a class label (Yu et al., 2023). (2) RAG typically retrieves general documents for generation, while RAKI further specifies certain knowledge sources (e.g. an external knowledge graph) as retrieval base for better factuality (Baek et al., 2023b).

Definition The setting of RAKI can then be formulated as follows: Given a user input x from task $\mathcal{T}$ and a specific knowledge source (to be discussed in Section 3), we denote y as target output and $\mathcal{K}$ as whole knowledge from the source. RAKI consists of two components (Baek et al., 2023b): (1) a retriever $\mathcal{R}$ which selects a subset $\mathcal{K}'$ from knowledge $\mathcal{K}$:

$$\mathcal{K}' = \mathcal{R}(x; \mathcal{K}), \tag{1}$$

where normally $|K'| \ll |K|$ in this *retrieval* step; (2) a language model $\mathcal{M}$ targeted for task $\mathcal{T}$. $\mathcal{M}$ takes both the input x and the retrieved knowledge $\mathcal{K}'$ for prediction:

$$y' = \mathcal{M}(x; \mathcal{K}'). \tag{2}$$

This step is referred to as *integration*. Due to the growing in-context reasoning skills (Brown et al., 2020; Chen, 2023) of language models, prompting (Schick and Schütze, 2021; Liu et al., 2023b) has become the go-to paradigm to integrate external knowledge. In prompting, the retrieved $\mathcal{K}'$ is formulated as text to be inserted into a prompt containing x (Baek et al., 2023b; Zhang et al., 2023c). Then the formulated prompt is sent to LMs for generation. Besides augmentation via prompts, this survey also discusses non-prompting techniques to integrate retrieved $\mathcal{K}'$, which are often based on LMs as encoders to produce representations of x and $\mathcal{K}'$ (e.g. in Section 3.1.2 and Section 3.2.2).

In the following, we use the definitions and notations above to discuss retrieval and integration in detail for the cases of $\mathcal{K}$ specified as knowledge graph (Section 3.1), tabular (Section 3.2) and natural language (Section 3.3).

3 Different Knowledge Sources as $\mathcal{K}$

We cover two structured knowledge: graph-based (*knowledge graph*) and row-based (*tabular*), as well as unstructured knowledge (*natural language*).

3.1 Knowledge Graph

Knowledge graphs (KGs) store rich factual knowledge of things, especially relational information by its graph structure. A KG can be defined as:

$$\mathcal{K} := (E, R), \tag{3}$$

where E is the set of entity nodes, and each edge $r \in R$ is a relation that connects a head entity e_h and a tail entity e_t in the graph (Wang et al., 2019). The corresponding 3-element tuple (e_h, r, e_t) is then referred to as a triple.

Table 1 in Appendix presents an overview of the KGs applied in the literature related to retrieval-based knowledge integration. Table 2 in Appendix summarizes the application of these KGs, showing that retrieving KGs can help with knowledge-intensive tasks such as knowledge graph question answering (Baek et al., 2023a). The entity-centered nature of KGs also makes them suitable for information extraction tasks such as named entity recognition (Zhang et al., 2023a; Fu et al., 2023) and relation classification (Fu et al., 2023).

3.1.1 Graph Retrieval

The goal of graph retrieval is to extract a subgraph $\mathcal{K}'$ of $\mathcal{K}$ given input x. Subgraph $\mathcal{K}'$ can be represented as a list of top-k retrieved triples (Andrus et al., 2022; Baek et al., 2023b; Fu et al., 2023):

$$\mathcal{K}' = \mathcal{R}(x; \mathcal{K}) = \{(e_{hi}, r_i, e_{ti})\}_{i=1}^{k}, \tag{4}$$

where e_{hi}, r_i and e_{ti} denote the head entity, the relation and the tail entity in the i-th triple.

Some previous work (Zhang et al., 2023a) requires only entity information such as entity descriptions from the knowledge graph. The resulting subgraph is then a list of entities without relations:

$$\mathcal{K}' = \{e_i\}_{i=1}^{k}. \quad (5)$$

In both cases, entity retrieval can usually be the first step. Therefore, we next introduce entity retrieval first, and then triple retrieval.

Entity retrieval Entity retrieval finds the most relevant entity candidates that match input x, as described in Equation 5. Linked *entity IDs* and recognized *entity names* are intuitive features for entity retrieval, requiring an additional entity recognition (Akbik et al., 2019) or entity linking (De Cao et al., 2021) procedure over x before retrieval.

As for **entity IDs**: Fu et al. (2023) employ TagMe (Ferragina and Scaiella, 2010) to detect and link entity mentions in x. TagMe provides linked entities as their IDs from Wikipedia, thus enabling Fu et al. (2023) to find exact match in the Wiki-based KG Wikidata5M (Wang et al., 2021).

As for **entity names**: Li et al. (2023) use a large language model Codex (Chen et al., 2021) to extract entity names of interest automatically. The authors design a text-to-logic template "*Question:* $\{x\}$ *Logic Form:* {logic form containing target retrieved entities}", and provide few-shot examples of user query and corresponding logical forms for in-context learning. Given input x, the last element in the logical language generated by Codex is extracted as the entity name of interest. To deal with a multiple-choice QA task, Lv et al. (2020) identify[1] potential entities both in question and in all five answer candidates, and find their matches in ConceptNet (Speer et al., 2017). Zhang et al. (2023a) train a binary classifier (Su et al., 2022) to identify potential entity mentions. Then for each positive span as a potential entity, Zhang et al. (2023a) use the tool ElasticSearch[2] for its best matches in Wikidata (Vrandečić and Krötzsch, 2014). Shu et al. (2022) also employs span classifiers as mention detection models, but followed by an extra alias mapping tool (Gabrilovich et al., 2013) to obtain better candidate entities for each potential mention.

Other features such as **n-gram** have also been studied for entity retrieval. In this case, a preceding entity detection step is not required before querying the KG. Young et al. (2018) and Li et al. (2022) enumerate n-grams out of input x, and then retrieve by checking if an n-gram is an exact entity entry in the KG. Bian et al. (2021) adapt similar settings to the task of multiple-choice question answering (QA), requiring exact match of n-grams between concept words from ConceptNet (Speer et al., 2017), and question and answer candidates from the task.

Triple retrieval As described in Equation 4, triple retrieval finds the most relevant triples (e_h, r, e_t) as KG facts for final augmentation.

(1) **Triple retrieval from retrieved entities.** A simple and intuitive solution is to base on the result of the above-mentioned entity retrieval: given candidate entities $\{e_i\}$ resulted from entity retrieval, this solution retrieves triples that contain a candidate entity (i.e. from $\{e_i\}$) either as head or tail (Fu et al., 2023; Young et al., 2018; Li et al., 2022; Zhang et al., 2023a; Baek et al., 2023b):

$$\mathcal{K}' = \{(e_h, r, e_t) \in \mathcal{K} | e_h \text{ or } e_t \in \{e_i\}\}. \quad (6)$$

Since retrieved entities $\{e_i\}$ are considered relevant to the input x, and triples in $\mathcal{K}'$ explicitly involve at least one retrieved entity in $\{e_i\}$, these triples are supposed to be relevant to x as well. Note that Equation 6 only includes triples that are directly connected to a retrieved entity, i.e. 1-hop away. To tackle problems that require multi-hop reasoning over graph, Feng et al. (2020) and Bian et al. (2021) further consider triples within a specified maximum distance from retrieved entities.

(2) **Triple retrieval from triple semantics.** One problem with such triple retrieval based on explicit entity-retrieval is, that not all triples involving retrieved entities are necessarily relevant to input x. Therefore, an alternative is the triple retrieval without prerequisite entity retrieval. In the course of that, a promising direction is to model relation r (or (e_h, r, e_t)) and x directly. Most work in this direction study language models as shared encoder for x and verbalized relation r. They for instance reformulate r or (e_h, r, e_t) in natural language. That enables pre-computable representations (Oguz et al., 2022) of relational knowledge before retrieval. Andrus et al. (2022), for instance, verbalize KG triples into natural language by joining e_h, r, e_t with space and making necessary adjustments such as adding an auxiliary verb if r does not contain a verb, or adding the article *the*. The resulting verbalization is treated as a KG fact and denoted as $v(e_h, r, e_t)$.

[1] Their entity identification tool is not explicitly given.
[2] https://www.elastic.co/

In the case of a question answering task, Andrus et al. (2022) retrieve the KG fact with the minimum edit distance from x as top-1 relevant:

$$\mathcal{K}' = (e'_h, r', e'_t) = \underset{(e_h, r, e_t) \in \mathcal{K}}{\arg\min}\ dist(x, v(e_h, r, e_t)). \tag{7}$$

For story completion though, Andrus et al. (2022) apply Sentence-BERT (Reimers and Gurevych, 2019) to embed x and KG facts. The KG fact with the maximum cosine similarity from x is retrieved. Baek et al. (2023a) also follow this first-verbalize-then-embed methodology, but apply MPNet (Song et al., 2020) as the shared encoder.

To summarize this retrieval subsection (Section 3.1.1), Table 3 in Appendix presents the discussed retrieval methods (both entity and triple).

3.1.2 Subgraph Integration

With the selected graph knowledge from graph retrieval (described in Section 3.1.1), the final step is to augment the input x with retrieved subgraph $\mathcal{K}'$ for task $\mathcal{T}$, given as:

$$y' = \mathcal{M}(x; \{(e_{hi}, r_i, e_{ti})\}_{i=1}^{k}), \tag{8}$$

or alternatively

$$y' = \mathcal{M}(x; \{e_i\}_{i=1}^{k}) \tag{9}$$

when only entity information is required (Zhang et al., 2023a) to perform task $\mathcal{T}$. Based on the form of $\mathcal{K}'$ when augmented to the language model, we discuss $\mathcal{K}'$ represented as hard, discrete natural language *prompts* and soft, continuous *embeddings*.

Prompt-based integration Table 4 (See Appendix) presents the prompts employed in prior work of knowledge graph integration. In prompt-based settings, knowledge is inserted as text into a language model. A simple implementation is to append (Li et al., 2022; Fu et al., 2023) or prepend (Baek et al., 2023a,b) the retrieved triple(s) 'as is' to the input x, preserving the triple-structure of $\mathcal{K}'$. Triples can also be augmented with task instruction (e.g. *Below are the facts ...*) (Baek et al., 2023a) or special tokens to highlight recognized entities (Fu et al., 2023) before concatenation with input.

Other works transform triples to natural phrases, to make the inserted knowledge more similar to input. The easiest way is to manually design a mapping from relation names to a descriptive natural language (NL) (Lv et al., 2020; Bian et al., 2021; Zhang et al., 2023a), which will finally connect the head and tail entities in the prompt. For example, Bian et al. (2021) suggest mapping the relation *Synonym* to NL *is the same as*, so to reformulate the triple (*Problem, Synonym, Challenge*) as descriptive *Problem is the same as Challenge*.

Due to the advanced capability of LLMs of understanding and paraphrasing knowledge, even rewriting prompts (Wu et al., 2023; Zhu et al., 2023), some prior work studies the possibility of reformulating the retrieved KG triple with a language model. Bian et al. (2021) discuss paraphrase- and retrieval-based reformulation of KG triples. They send the mapping-based descriptions (e.g. *Problem is the same as Challenge*) to an encoder-decoder LM to generate top decoded paraphrases. Besides, they also use the mapping-based descriptions to retrieve Wikipedia texts for retrieval-based descriptions. Bian et al. (2021) also point out that concatenation of the three types of reformulation (i.e. mapping-based, paraphrase-based and retrieval-based) delivers better performance than using any single type. Wu et al. (2023) adopt ChatGPT to paraphrase KG triples to free-form texts. Andrus et al. (2022) and Li et al. (2023) provide few-shot triple-to-text examples in user input to assist GPT models with paraphrase generation.

Embedding integration In embedding-based KG integration , the retrieved entities $\{e_i\}_{i=1}^{k}$ are explicitly embedded (denoted as E) before sending them to the language model:

$$y' = \mathcal{M}(x; \{E_{(e_{hi}, r_i, e_{ti})}\}_{i=1}^{k}) \tag{10}$$

in the case of relations, and

$$y' = \mathcal{M}(x; \{E_{e_i}\}_{i=1}^{k}) \tag{11}$$

in the case of entities.

To integrate **relation embeddings**, Young et al. (2018) apply an LSTM to encode each retrieved triple r (such as *incomnia, IsA, sleep_problem*) and candidate response (such as *A cup of milk could help you sleep.*) in dialogue task. Bi-linear products of the encodings are then used to compute activation for each possible response. As for **entity embeddings**, Fu et al. (2023) evaluate entity embeddings of retrieved entities from various knowledge-intensive pre-trained LMs (Peters et al., 2019; Zhang et al., 2019). They point out the challenge of integrating multiple knowledge via embeddings (Fu et al., 2023), that it is hard to simply add embeddings from different entities and models at a time without losing much information in each embedding.

3.2 Tabular

A tabular is a row-based format to store knowledge efficiently, with each row representing one entry:

$$\mathcal{K} := \{r_i\}_{i=1} = \{(a_{i1}, a_{i2}, \cdots, a_{iM})\}_{i=1} \quad (12)$$

Each row r_i is a tabular item, normally describing an entity or event. $a_{i1}, a_{i2}, \cdots, a_{iM}$ are M attributes of the i-th row, which can be given as text (e.g. entity description) or numerical values. Prior works also discuss the case of $\mathcal{K}$ being multiple tables (Herzig et al., 2021; Li et al., 2021).

3.2.1 Tabular Retrieval

Tabular retrieval can be performed on three levels: (1) **Retrieve relevant tables** from a collection of tables (Herzig et al., 2021; Li et al., 2021). (2) **Retrieve relevant rows** from a table, which describes the standard setting in table-QA (Wan et al., 2023). (3) **Retrieve relevant blocks** from relevant rows, by removing less important columns (Wan et al., 2023). The goal of tabular retrieval is to find the most relevant table blocks (i.e. *sub-tabular*):

$$\mathcal{K}' := \{(a_{ij_1}, a_{ij_2}, \cdots, a_{ij_m})\}_{i=1}^{k} \quad (13)$$

where $j_1, \cdots, j_m$ are involved columns.

(First-)Retrieval Retrieval based on neural representations have been adapted to tabular tasks since the success of deep passage retrieval (Karpukhin et al., 2020) over text. Herzig et al. (2021) employ TaPas (Herzig et al., 2020), a BERT (Devlin et al., 2019) model pre-trained with weak supervison for table parsing. For a table-QA task, both the question x and the table $T \in \mathcal{K}$ are encoded by TaPas, where the table T is textualized by concatenating the cell contents left-to-right, row by row. The top-k tables yielding maximum inner product with x at [CLS] token are retrieved. Instead of simply concatenating cells (Herzig et al., 2021; Oguz et al., 2022) for encoding tabular data, Wan et al. (2023) and Shi et al. (2023) rewrite each cell into "(column, value)" text, and concatenate this semi-structured text of each row into a textual sequence.

Refinement of tabular retrieval $\mathcal{K}'$ from the first retrieval can still contain redundant information, e.g. less relevant rows from a retrieved table in a multi-table setting. Park et al. (2023) further refine the retriever setup by adding a reranking module after retrieval, to score each retrieved block $b \in \mathcal{K}'$. The relevance score is given by the output distribution of T5 (Raffel et al., 2020) over *Rel* (relevance) and *Nonrel* (non-relevance) from the prompt "*query: {q} block: {b} relevant:* ". While this reranking technique aims to filter out less relevant rows from $\mathcal{K}'$, Wan et al. (2023) propose to filter out columns: by applying a shared LM to encode x and rows given by a sequence of (attribute, value) pairs. The top-k rows are retrieved through maximum inner product search (Mussmann and Ermon, 2016). Irrelevant columns are removed by leveraging the encodings of x, $\mathcal{K}$ and previously retrieved rows. To further enrich augmentation, Zhong et al. (2022) perform an extra retrieval step over natural language sources for an informative passage and reformulate this tabular task to table-text task (Li et al., 2021). This passage is then sent with retrieved table cells for final answer.

3.2.2 Sub-Tabular Integration

Prompt-based integration Given the top-k rows $\mathcal{K}' = \{r_i\}_{i=1}^{k}$ from previous tabular retrieval, the most studied technique to integrate them is to textualize $\mathcal{K}'$ and insert them into a prompt.

Herzig et al. (2021) and Zhong et al. (2022) formulate the prompt learning problem as *extractive QA*, by restricting the final output to be an exact span from retrieved table $\mathcal{K}'$. As suggested in Devlin et al. (2019), they add a multi-layer perception on top of the LM and train the model to predict the start and end position correctly from textualized $\mathcal{K}'$ in the prompt. Li et al. (2021) and Wan et al. (2023) regard the problem as a *generative QA* task, where normally a seq2seq LM is trained to generate the expected response.

Embedding integration To tackle very long contexts from retrieved tabulars $\{r_i\}_{i=1}^{k}$ and original user input x, some works integrate encodings instead of text forms of tabular. Oguz et al. (2022), Park et al. (2023) and Shi et al. (2023) utilize an encoder-decoder where each retrieved row r_i is textualized and then converted by the encoder into a contextualized embedding $E_i := Enc(x||r_i)$, where "$||$" concatenates a retrieved tabular row r_i and the user input x. x denotes a question in a QA task (Park et al., 2023) or current conversation context in a dialogue system (Shi et al., 2023). Finally, the concatenation of $\{E_i\}_{i=1}^{k}$ is sent to the decoder to generate an answer (Park et al., 2023) or next response (Shi et al., 2023).

3.3 Natural Language

While the previous sections describe incorporating structured information, most RAG systems retrieve natural language (NL) documents, mainly because there is more knowledge available in text form than in structured form such as knowledge graph, and converting text to knowledge graph is challenging (Melnyk et al., 2022).

Formally, we define a natural language (NL) source to be the composite of text resources:

$$\mathcal{K} := \{D_i\}, \tag{14}$$

where each D_i is a document consisting of a sequence of tokens. While text is widely considered as *unstructured* (Hu et al., 2024; Mo et al., 2022), some works see that text can be *semi-structured*, because of the sentence and paragraph structure (Ruan et al., 2022) by its nature, as well as handcrafted structural clues (Arivazhagan et al., 2023) such as headings and meta information. Despite their differences in structure, unstructured and semi-structured texts are predominately treated equally in the reader stage following the concatenation and/or compression of retrieved texts.

NL-based RAG systems like LangChain (Chase, 2022) and LlamaIndex (Liu, 2022) usually incorporate the following steps: (1) preparation including chunking and indexing, (2) (first-)retrieval, (3) re-ranking and (4) generation. Respectively, in this RAKI survey, we will describe (1), (2) and (3) in Section 3.3.1 (*NL retrieval*) and final prediction/generation in Section 3.3.2 (*NL integration*).

3.3.1 Natural Language Retrieval

Similar to graph and tabular retrieval, the goal of natural language retrieval is to get top-k text documents from $\mathcal{K}$ given the input *query* x, normally by using the scoring function of the retriever $\mathcal{R}$:

$$\mathcal{K}' = \mathcal{R}(x; \mathcal{K}) = \{D_i\}_{i=1}^{k}. \tag{15}$$

Preparation Retrieval systems for natural language start with the collection of text features, including *chunking* and *indexing*. (1) **Chunking**: Since language models as retrievers have limited context size (e.g. 512 in BERT (Devlin et al., 2019)), documents might need to be split into smaller chunks. Choosing when to split a text into chunks without losing surrounding information is a difficult problem (Chen et al., 2023). While libraries like LangChain have several techniques that split based on textual features like ending paragraphs, many approaches employ strides (overlapping text spans) (Wu and Mooney, 2022; Ram et al., 2023) to prevent incomplete information. In the case of semi-structured text, structural information such as title and meta information can be utilized in text/chunk preparation. Arivazhagan et al. (2023), for instance, proposes to first filter relevant documents based on abstracts and table of contents before considering passage snippets. (2) **Indexing** then computes and stores features of each chunk for fast retrieval. The features to be indexed depend on the applied retriever $\mathcal{R}$, which will be discussed in the following paragraph.

(First-)Retrieval Choosing a suitable retriever $\mathcal{R}$ for one's setting comes with the following considerations: While **sparse retrieval** such as TF-IDF is straightforward and easy to compute, **dense retrieval** based on dense embeddings proves substantial effectiveness (Arabzadeh et al., 2021), especially when the query x and the document D_i have limited common lexicon (Karpukhin et al., 2020). In RAG systems (Lewis et al., 2020; Chase, 2022), two dense retrieval approaches are mainly applied:

(1) **Bi-encoder** is normally a Transformer model that can produce text-level embeddings (Reimers and Gurevych, 2019): Document embeddings $E(D_i)$ are pre-computed offline during indexing, while query embedding $E(x)$ is computed at inference. Embedding query and document separately (Lewis et al., 2020) by bi-encoder allows inner-product search within $\mathcal{O}(|\mathcal{K}|)$ time, but results in weak interaction between query and documents (Erker et al., 2024) since bi-encoder was query-unaware when embedding documents.

(2) **Cross-encoder** directly models the relevance between query and documents, and produces a score $S(x, D_i) \in [0, 1]$ for each candidate document D_i at inference, which is slow given a large $\mathcal{K}$. Despite the cross-encoders can be substantially better than dense retrievers (Wang et al., 2022a), the computational cost makes cross-encoder only applicable to small datasets (Reimers and Gurevych, 2019) or as a re-ranking model (See next paragraph) based on first-retrieval results (Zhou et al., 2023b).

Re-ranking Re-ranking bridges the gap between the two encoders (Glass et al., 2022; Ma et al., 2023): First, a bi-encoder is employed in a previous first-retrieval to quickly filter a (larger than k) set $\overline{\mathcal{K}}$ of candidate documents. Then in re-ranking, a cross-encoder encodes x with each document D_i in $\overline{\mathcal{K}}$ and yields a ranking score $S(x; \overline{D_i})$ to get the final k results.

Besides the retrieve-then-rerank technique, other approaches have been proposed to achieve query-document interaction or computational efficiency. ColBERT (Khattab and Zaharia, 2020) introduce a late interaction method based on the contextualized tokens of BERT that computes dot-product between multiple query vectors and multiple document vectors. PolyEncoders and PreTTR (MacAvaney et al., 2020) pre-compute representations offline and used self-attentive aggregators on top of these representations. Liu et al. (2024) sequentially feed all retrieved $\mathcal{K}'$ alongside x through an accordingly fine-tuned LLM, resulting in a binary classification of their relevance. Similarly, Asai et al. (2024) and Jeong et al. (2024) propose an extended framework where an LLM predicts special tokens in the text indicating both the relevance of external knowledge.

3.3.2 Natural Language Integration

The integration of NL in RAG systems follows the retrieve-then-read paradigm (Lewis et al., 2020), where a small set of relevant context documents is retrieved and subsequently used alongside the question to generate an informed response. In this survey of RAKI, we generalize retrieval augmentation to generation and classification tasks, and also cover embedding-based methods for integration. Therefore, natural language integration can be categorized into the following three cases:

(1) **Prompt integration for generation**, by concatenating retrieved documents $\mathcal{K}' = \{D_i\}_{i=1}^k$ and combining with query x in a prompt (Lewis et al., 2020; Guu et al., 2020; Wang et al., 2022b; Cai et al., 2023):

$$y' = \mathcal{M}(prompt(x, D_1||D_2||\cdots||D_k)), \quad (16)$$

where $\mathcal{M}$ is the (generative) language model for final output and $prompt(\cdot)$ denotes the template that includes all its variables in a prompt.

(2) **Embedding integration for generation**, by processing query-document pairs separately:

$$E_i = Enc(x||D_i), i = 1, \cdots, k, \quad (17)$$

and combining the intermediate encodings in a final decoding stage (Izacard and Grave, 2021; Hofstätter et al., 2023; Zhang et al., 2023b):

$$y' = Dec(x||E_1||E_2||\cdots||E_k), \quad (18)$$

where Enc and Dec denote a LM encoder and decoder. The fusion of query x and encodings $\{E_i\}_{i=1}^k$ during decoding stage mitigates the risk of exceeding the input context length.

(3) **Embedding integration for classification**, by embedding retrieved documents $\{D_i\}_{i=1}^k$ as features in a kNN model (Khandelwal et al., 2020; Drozdov et al., 2022). The prediction is based on the majority vote or nearest neighbor over supervised labels of $\{D_i\}_{i=1}^k$.

4 Challenges & Outlook

Here we summarize some challenges of retrieval-augmented knowledge integration techniques, followed by an outlook of the RAKI framework.

Necessity of external knowledge In this survey, our definition in Section 2 and the many included works dive into retrieving and augmenting external knowledge, without questioning before retrieval if external knowledge is necessary. We discern two methodologies in identifying the need for external information during the pre-retrieval stage:

(1) *Passively*, by relying on self-consistency decoding techniques (Wang et al., 2023; Zhao et al., 2023b; Li et al., 2024). For example, Wang et al. (2023) allows to quantify the uncertainty associated with the use of parametric knowledge. By employing a non-zero temperature to ensure diversity, multiple generations are sampled and compared for similarity in the final output. If a set of answers yields a significant deviation above a threshold, it indicates substantial uncertainty, necessitating the introduction of external knowledge.

(2) *Actively*, by guiding the language model to generate special tokens as assessment of retrieved information (Asai et al., 2024; Jeong et al., 2024), or employing a separate model to score the need for external knowledge (Liu et al., 2024; Chen et al., 2024). For example, Chen et al. (2024) uses ChatGPT to score generated knowledge (based on *internal*, parameterized knowledge of LM) against retrieved passages (*external*) in a QA task. They find out for time-sensitive questions, external information is prioritized, while non-time-sensitive ones prompt comparison between generated and retrieved knowledge to determine the best source.

Prediction consistency with knowledge RAKI formulated in Section 2 does not verify if LM predictions reflect knowledge. To address this issue, Sun et al. (2023) utilize an LLM discriminator framework to ensure consistent citations by prompting about various aspects of the generation: (1) whether the cited source supports the claim, (2) whether any of the retrieved documents support the claim, and (3) whether the cited set of documents is *minimal*. Here *minimal* refers to the document set not containing any documents that are unnecessary for supporting the claim. Asai et al. (2024) and Jeong et al. (2024) again apply their special token generation scheme (discussed in Section 3.3.1 for reranking) to predict whether the generated claim is fully supported by the retrieved knowledge.

Multi-step reasoning For simplicity of modelling, we formulate the RAKI problem in Section 2 as single pass. Apart from the single-pass pipeline, multi-step reasoning frameworks leverage multiple retrieve-and-read cycles. This approach facilitates the construction of coherent reasoning chains, enabling the system to address complex questions effectively (Liu et al., 2024, 2023a; Wang et al., 2024; Li et al., 2024; Zhou et al., 2023a). We summarize two primary approaches to integrating knowledge into reasoning frameworks: (1) *Knowledge as a tool for verifying and refining reasoning steps post-creation* (Li et al., 2024; Zhao et al., 2023b; Wang et al., 2024). For example, Zhao et al. (2023b) improve factuality during Chain-of-Thought (CoT) generation (Wei et al., 2022) by integrating an optional RAG stage, where an uninformed CoT chain undergoes self-consistency tests (Wang et al., 2023). Failing chains are refined by verifying questions for each step, retrieving relevant information, and creating a new knowledge-informed rationale. Based on this refined CoT rationale the final answer is corrected.

(2) *Knowledge retrieval as an integral part of creating informed reasoning steps.* Liu et al. (2023a) propose a framework for multi-step reasoning where questions are sequentially decomposed. A central component of this framework is an agent LLM delegating the answering process. This agent is tasked with determining whether to decompose a query further into sub-questions and deciding whether to retrieve external knowledge or answer internally for each step. Once enough information is collected, the LLM provides a final answer, ensuring grounded reasoning without the need for post-reasoning verification.

Outlook As can be seen from the above mentioned challanges and solutions, research in retrieval-augmented knowledge integration has witnessed a growing role of LLMs. Besides the generation (integration) step where LLMs are good fits for by their nature, LLMs can also serve in the retrieval step, as retriever itself (Gao et al., 2023) or as discriminator to assess the quality of retrieval (Liu et al., 2024). Beyond the retrieve-and-integrate framework of RAKI, LLMs bring several enrichment steps which are not discussed in Section 3, such as knowledge extraction (Xu et al., 2023) and consistency verification (Asai et al., 2024).

5 Related Work

Survey of surveys Recent surveys show the paradigm shift from traditional knowledge integration to retrieval augmentation: Wei et al. (2021) and Hu et al. (2024) provide an overview on different pre-training and fine-tuning techniques of knowledge enhancement, organized by different knowledge formats. Hu et al. (2024) cover retrieval-augmented methods also but restrict the source of retrieval to be text and the task to be natural language generation. Mialon et al. (2023) compare various retrieval augmentation methods over textual documents. Pan et al. (2024) narrow the source of knowledge to knowledge graphs (KGs). Ling et al. (2023) survey different methods to apply LLMs in a specialized domain, including retrieving explicit domain information for in-context learning. Zhao et al. (2023a) focus on the topic of multi-modal (such as vision and audio) retrieval-augmented generation (RAG) but also discuss structured knowledge for four tasks such as knowledge-grounded dialogue. Gao et al. (2023) and Hu and Lu (2024) both provide a short introduction of unstructured and structured data for augmentation, with a focus on available datasets/corpus. To our knowledge, there is still no comprehensive survey that studies both structured and unstructured sources and describes respective NLP techniques accordingly.

6 Conclusion

This survey paper studies recent works that augment language models by retrieving external knowledge sources. We categorize research in retrieval-augmented knowledge integration (RAKI) into three sections, according to knowledge format: knowledge graph, tabular, and natural language. Besides a comprehensive collection of knowledge retrieval and integration approaches, we also point out the limitations and challenges of current RAKI. We hope this survey could (1) help researchers who are looking for a technical-intensive overview and (2) encourage future work to improve current RAKI.

Limitations

Collecting papers for this survey using search engines (e.g. Google Scholar and dblp) is very challenging, mainly because: (1) It is infeasible to enumerate all possible search words to approach every potential paper of our interest. For example, we include *knowledge augmentation/integration/enhancement* in the search word list (See Appendix A.1 for complete list of search words), as well as their variants with suffix changes (e.g. *knowledge augment/-ed*). These words would still leave out a paper using *knowledge augmenting* or *we fuse knowledge*. (2) Each search engine has its own drawbacks (Appendix A.1 presents a detailed comparison of our employed search engines): e.g. ACL Anthology supports full-text search but mainly includes publications from *CL venues; dblp covers most venues but only supports search over title. Therefore, a relevant non-*CL publication might have been left out if its title does not match one of our specified search words.

We would also like to point out that this survey is focused on the methodological part of RAKI rather than performance. The idea of retrieval-augmentation is general and can thus be applied to a great variety of NLP tasks. Therefore, it makes limited sense to compare scores reported by papers that conduct different tasks.

Ethics Statement

In this survey, we (1) formulate the problem setting of RAKI and (2) collect, explain and analyse searched literature. As for (1), we try to make formulation objective by giving a general mathematical definition.

As for (2), we make the paper selection criteria public in Appendix A.1. As shown in Appendix A.2, 51.8% of the included papers are accepted at *CL venues, which require a mandatory ethics review since 2022. While we cannot ensure the absence of ethical issues in the selected papers from prior *CL and other venues (especially arXiv), we ensure the explanations and findings in this survey are presented objectively.

Acknowledgments

We would like to thank the anonymous reviewers for their valuable comments and feedback on the paper. This work has been supported by the Federal Joint Committee of Germany (Gemeinsamer Bundesausschuss) as part of the project ADBoard (01VSF21047).

References

Alan Akbik, Tanja Bergmann, and Roland Vollgraf. 2019. Pooled contextualized embeddings for named entity recognition. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 724–728, Minneapolis, Minnesota. Association for Computational Linguistics.

Berkeley R Andrus, Yeganeh Nasiri, Shilong Cui, Benjamin Cullen, and Nancy Fulda. 2022. Enhanced story comprehension for large language models through dynamic document-based knowledge graphs. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(10):10436–10444.

Negar Arabzadeh, Xinyi Yan, and Charles L. A. Clarke. 2021. Predicting efficiency/effectiveness trade-offs for dense vs. sparse retrieval strategy selection. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, CIKM '21, page 2862–2866, New York, NY, USA. Association for Computing Machinery.

Manoj Ghuhan Arivazhagan, Lan Liu, Peng Qi, Xinchi Chen, William Yang Wang, and Zhiheng Huang. 2023. Hybrid hierarchical retrieval for open-domain question answering. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 10680–10689, Toronto, Canada. Association for Computational Linguistics.

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2024. Self-RAG: Learning to retrieve, generate, and critique through self-reflection. In *The Twelfth International Conference on Learning Representations*.

Tom Ayoola, Joseph Fisher, and Andrea Pierleoni. 2022. Improving entity disambiguation by reasoning over a knowledge base. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2899–2912, Seattle, United States. Association for Computational Linguistics.

Jinheon Baek, Alham Fikri Aji, and Amir Saffari. 2023a. Knowledge-augmented language model prompting for zero-shot knowledge graph question answering. In *Proceedings of the 1st Workshop on Natural Language Reasoning and Structured Explanations (NLRSE)*, pages 78–106, Toronto, Canada. Association for Computational Linguistics.

Jinheon Baek, Soyeong Jeong, Minki Kang, Jong Park, and Sung Hwang. 2023b. Knowledge-augmented language model verification. In *Proceedings of the 2023 Conference on Empirical Methods in Natural*

Language Processing, pages 1720–1736, Singapore. Association for Computational Linguistics.

Nesrine Bannour, Sahar Ghannay, Aurélie Névéol, and Anne-Laure Ligozat. 2021. Evaluating the carbon footprint of NLP methods: a survey and analysis of existing tools. In *Proceedings of the Second Workshop on Simple and Efficient Natural Language Processing*, pages 11–21, Virtual. Association for Computational Linguistics.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A pretrained language model for scientific text. 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)*, pages 3615–3620, Hong Kong, China. Association for Computational Linguistics.

Ning Bian, Xianpei Han, Bo Chen, and Le Sun. 2021. Benchmarking knowledge-enhanced commonsense question answering via knowledge-to-text transformation. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 12574–12582. AAAI Press.

Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, SIGMOD '08, page 1247–1250, New York, NY, USA. Association for Computing Machinery.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Deng Cai, Yan Wang, Lemao Liu, and Shuming Shi. 2022. Recent advances in retrieval-augmented text generation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '22, page 3417–3419, New York, NY, USA. Association for Computing Machinery.

Mingzhu Cai, Siqi Bao, Xin Tian, Huang He, Fan Wang, and Hua Wu. 2023. Query enhanced knowledge-intensive conversation via unsupervised joint modeling. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1730–1745, Toronto, Canada. Association for Computational Linguistics.

Erik Cambria, Soujanya Poria, Rajiv Bajpai, and Bjoern Schuller. 2016. SenticNet 4: A semantic resource for sentiment analysis based on conceptual primitives. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2666–2677, Osaka, Japan. The COLING 2016 Organizing Committee.

Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. 2020. LEGAL-BERT: The muppets straight out of law school. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2898–2904, Online. Association for Computational Linguistics.

Harrison Chase. 2022. LangChain.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Pondé de Oliveira Pinto, Jared Kaplan, Harrison Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Joshua Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. 2021. Evaluating large language models trained on code. *CoRR*, abs/2107.03374.

Tong Chen, Hongwei Wang, Sihao Chen, Wenhao Yu, Kaixin Ma, Xinran Zhao, Hongming Zhang, and Dong Yu. 2023. Dense X retrieval: What retrieval granularity should we use? *CoRR*, abs/2312.06648.

Wenhu Chen. 2023. Large language models are few(1)-shot table reasoners. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 1120–1130, Dubrovnik, Croatia. Association for Computational Linguistics.

Yuhan Chen, Shuqi Li, and Rui Yan. 2024. FlexiQA: Leveraging LLM's evaluation capabilities for flexible knowledge selection in open-domain question answering. In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 56–66, St. Julian's, Malta. Association for Computational Linguistics.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. 2023. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113.

Nicola De Cao, Gautier Izacard, Sebastian Riedel, and Fabio Petroni. 2021. Autoregressive entity retrieval. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Zhendong Dong, Qiang Dong, and Changling Hao. 2010. HowNet and its computation of meaning. In *Coling 2010: Demonstrations*, pages 53–56, Beijing, China. Coling 2010 Organizing Committee.

Andrew Drozdov, Shufan Wang, Razieh Rahimi, Andrew McCallum, Hamed Zamani, and Mohit Iyyer. 2022. You can't pick your neighbors, or can you? when and how to rely on retrieval in the kNN-LM. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 2997–3007, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Justus-Jonas Erker, Florian Mai, Nils Reimers, Gerasimos Spanakis, and Iryna Gurevych. 2024. Triple-encoders: Representations that fire together, wire together. *CoRR*, abs/2402.12332.

Yanlin Feng, Xinyue Chen, Bill Yuchen Lin, Peifeng Wang, Jun Yan, and Xiang Ren. 2020. Scalable multi-hop relational reasoning for knowledge-aware question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1295–1309, Online. Association for Computational Linguistics.

Paolo Ferragina and Ugo Scaiella. 2010. TAGME: on-the-fly annotation of short text fragments (by wikipedia entities). In *Proceedings of the 19th ACM Conference on Information and Knowledge Management, CIKM 2010, Toronto, Ontario, Canada, October 26-30, 2010*, pages 1625–1628. ACM.

Peng Fu, Yiming Zhang, Haobo Wang, Weikang Qiu, and Junbo Zhao. 2023. Revisiting the knowledge injection frameworks. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10983–10997, Singapore. Association for Computational Linguistics.

Evgeniy Gabrilovich, Michael Ringgaard, and Amarnag Subramanya. 2013. Facc1: Freebase annotation of clueweb corpora, version 1 (release date 2013-06-26, format version 1, correction level 0).

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Qianyu Guo, Meng Wang, and Haofen Wang. 2023. Retrieval-augmented generation for large language models: A survey. *CoRR*, abs/2312.10997.

Michael Glass, Gaetano Rossiello, Md Faisal Mahbub Chowdhury, Ankita Naik, Pengshan Cai, and Alfio Gliozzo. 2022. Re2G: Retrieve, rerank, generate. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2701–2715, Seattle, United States. Association for Computational Linguistics.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. Realm: retrieval-augmented language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20. JMLR.org.

Jonathan Herzig, Thomas Müller, Syrine Krichene, and Julian Eisenschlos. 2021. Open domain question answering over tables via dense retrieval. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 512–519, Online. Association for Computational Linguistics.

Jonathan Herzig, Pawel Krzysztof Nowak, Thomas Müller, Francesco Piccinno, and Julian Eisenschlos. 2020. TaPas: Weakly supervised table parsing via pre-training. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4320–4333, Online. Association for Computational Linguistics.

Sebastian Hofstätter, Jiecao Chen, Karthik Raman, and Hamed Zamani. 2023. Fid-light: Efficient and effective retrieval-augmented text generation. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2023, Taipei, Taiwan, July 23-27, 2023*, pages 1437–1447. ACM.

Linmei Hu, Zeyi Liu, Ziwang Zhao, Lei Hou, Liqiang Nie, and Juanzi Li. 2024. A survey of knowledge enhanced pre-trained language models. *IEEE Transactions on Knowledge and Data Engineering*, 36(4):1413–1430.

Yucheng Hu and Yuxing Lu. 2024. Rag and rau: A survey on retrieval-augmented language model in natural language processing. *arXiv preprint arXiv:2404.19543*.

Kexin Huang, Jaan Altosaar, and Rajesh Ranganath. 2019. Clinicalbert: Modeling clinical notes and predicting hospital readmission. *CoRR*, abs/1904.05342.

Yongfeng Huang, Yanyang Li, Yichong Xu, Lin Zhang, Ruyi Gan, Jiaxing Zhang, and Liwei Wang. 2023. MVP-tuning: Multi-view knowledge retrieval with prompt tuning for commonsense reasoning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13417–13432, Toronto, Canada. Association for Computational Linguistics.

Gautier Izacard and Edouard Grave. 2021. Leveraging passage retrieval with generative models for open domain question answering. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 874–880, Online. Association for Computational Linguistics.

Minbyul Jeong, Jiwoong Sohn, Mujeen Sung, and Jaewoo Kang. 2024. Improving medical reasoning through retrieval and self-reflection with retrieval-augmented large language models. *CoRR*, abs/2401.15269.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.

Urvashi Khandelwal, Omer Levy, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. 2020. Generalization through memorization: Nearest neighbor language models. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Omar Khattab and Matei Zaharia. 2020. Colbert: Efficient and effective passage search via contextualized late interaction over bert. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '20, page 39–48, New York, NY, USA. Association for Computing Machinery.

Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N. Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick van Kleef, Sören Auer, and Christian Bizer. 2015. Dbpedia - A large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web*, 6(2):167–195.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474. Curran Associates, Inc.

Alexander Hanbo Li, Patrick Ng, Peng Xu, Henghui Zhu, Zhiguo Wang, and Bing Xiang. 2021. Dual reader-parser on hybrid textual and tabular evidence for open domain question answering. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4078–4088, Online. Association for Computational Linguistics.

Tianle Li, Xueguang Ma, Alex Zhuang, Yu Gu, Yu Su, and Wenhu Chen. 2023. Few-shot in-context learning on knowledge base question answering. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6966–6980, Toronto, Canada. Association for Computational Linguistics.

Xingxuan Li, Ruochen Zhao, Yew Ken Chia, Bosheng Ding, Shafiq Joty, Soujanya Poria, and Lidong Bing. 2024. Chain-of-knowledge: Grounding large language models via dynamic knowledge adapting over heterogeneous sources. In *The Twelfth International Conference on Learning Representations*.

Yu Li, Baolin Peng, Yelong Shen, Yi Mao, Lars Liden, Zhou Yu, and Jianfeng Gao. 2022. Knowledge-grounded dialogue generation with a unified knowledge representation. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 206–218, Seattle, United States. Association for Computational Linguistics.

Weizhe Lin and Bill Byrne. 2022. Retrieval augmented visual question answering with outside knowledge. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11238–11254, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Chen Ling, Xujiang Zhao, Jiaying Lu, Chengyuan Deng, Can Zheng, Junxiang Wang, Tanmoy Chowdhury, Yun Li, Hejie Cui, Xuchao Zhang, Tianjiao Zhao, Amit Panalkar, Wei Cheng, Haoyu Wang, Yanchi Liu, Zhengzhang Chen, Haifeng Chen, Chris White, Quanquan Gu, Carl Yang, and Liang Zhao. 2023. Beyond one-model-fits-all: A survey of domain specialization for large language models. *CoRR*, abs/2305.18703.

Chang Liu, Xiaoguang Li, Lifeng Shang, Xin Jiang, Qun Liu, Edmund Lam, and Ngai Wong. 2023a. Gradually excavating external knowledge for implicit complex question answering. In *Findings of the Association for Computational Linguistics: EMNLP*

2023, pages 14405–14417, Singapore. Association for Computational Linguistics.

Jerry Liu. 2022. LlamaIndex.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023b. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Comput. Surv.*, 55(9).

Weijie Liu, Peng Zhou, Zhe Zhao, Zhiruo Wang, Qi Ju, Haotang Deng, and Ping Wang. 2020. K-BERT: enabling language representation with knowledge graph. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 2901–2908. AAAI Press.

Yanming Liu, Xinyue Peng, Xuhong Zhang, Weihao Liu, Jianwei Yin, Jiannan Cao, and Tianyu Du. 2024. RA-ISF: learning to answer and understand from retrieval augmentation via iterative self-feedback. *CoRR*, abs/2403.06840.

Quanyu Long, Wenya Wang, and Sinno Pan. 2023. Adapt in contexts: Retrieval-augmented domain adaptation via in-context learning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6525–6542, Singapore. Association for Computational Linguistics.

Shangwen Lv, Daya Guo, Jingjing Xu, Duyu Tang, Nan Duan, Ming Gong, Linjun Shou, Daxin Jiang, Guihong Cao, and Songlin Hu. 2020. Graph-based reasoning over heterogeneous external knowledge for commonsense question answering. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 8449–8456. AAAI Press.

Yubo Ma, Yixin Cao, Yong Hong, and Aixin Sun. 2023. Large language model is not a good few-shot information extractor, but a good reranker for hard samples! In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10572–10601, Singapore. Association for Computational Linguistics.

Sean MacAvaney, Franco Maria Nardini, Raffaele Perego, Nicola Tonellotto, Nazli Goharian, and Ophir Frieder. 2020. Efficient document re-ranking for transformers by precomputing term representations. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '20. ACM.

Igor Melnyk, Pierre Dognin, and Payel Das. 2022. Knowledge graph generation from text. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 1610–1622, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Grégoire Mialon, Roberto Dessi, Maria Lomeli, Christoforos Nalmpantis, Ramakanth Pasunuru, Roberta Raileanu, Baptiste Roziere, Timo Schick, Jane Dwivedi-Yu, Asli Celikyilmaz, Edouard Grave, Yann LeCun, and Thomas Scialom. 2023. Augmented language models: a survey. *Transactions on Machine Learning Research*. Survey Certification.

Lingbo Mo, Zhen Wang, Jie Zhao, and Huan Sun. 2022. Knowledge transfer between structured and unstructured sources for complex question answering. In *Proceedings of the Workshop on Structured and Unstructured Knowledge Integration (SUKI)*, pages 55–66, Seattle, USA. Association for Computational Linguistics.

Stephen Mussmann and Stefano Ermon. 2016. Learning and inference via maximum inner product search. In *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 2587–2596, New York, New York, USA. PMLR.

Barlas Oguz, Xilun Chen, Vladimir Karpukhin, Stan Peshterliev, Dmytro Okhonko, Michael Schlichtkrull, Sonal Gupta, Yashar Mehdad, and Scott Yih. 2022. UniK-QA: Unified representations of structured and unstructured knowledge for open-domain question answering. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1535–1546, Seattle, United States. Association for Computational Linguistics.

Oded Ovadia, Menachem Brief, Moshik Mishaeli, and Oren Elisha. 2023. Fine-tuning or retrieval? comparing knowledge injection in llms. *CoRR*, abs/2312.05934.

Shirui Pan, Linhao Luo, Yufei Wang, Chen Chen, Jiapu Wang, and Xindong Wu. 2024. Unifying large language models and knowledge graphs: A roadmap. *IEEE Transactions on Knowledge and Data Engineering*, pages 1–20.

Eunhwan Park, Sung-Min Lee, Daeryong Seo, Seonhoon Kim, Inho Kang, and Seung-Hoon Na. 2023. RINK: reader-inherited evidence reranker for table-and-text open domain question answering. In *Thirty-Seventh AAAI Conference on Artificial Intelligence, AAAI 2023, Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence, IAAI 2023, Thirteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2023, Washington, DC, USA, February 7-14, 2023*, pages 13446–13456. AAAI Press.

Matthew E. Peters, Mark Neumann, Robert Logan, Roy Schwartz, Vidur Joshi, Sameer Singh, and Noah A. Smith. 2019. Knowledge enhanced contextual word representations. 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), pages 43–54, Hong Kong, China. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67.

Ori Ram, Yoav Levine, Itay Dalmedigos, Dor Muhlgay, Amnon Shashua, Kevin Leyton-Brown, and Yoav Shoham. 2023. In-context retrieval-augmented language models. *Transactions of the Association for Computational Linguistics*, 11:1316–1331.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. 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)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Qian Ruan, Malte Ostendorff, and Georg Rehm. 2022. HiStruct+: Improving extractive text summarization with hierarchical structure information. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1292–1308, Dublin, Ireland. Association for Computational Linguistics.

Chris Samarinas, Wynne Hsu, and Mong Li Lee. 2021. Improving evidence retrieval for automated explainable fact-checking. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Demonstrations*, pages 84–91, Online. Association for Computational Linguistics.

Timo Schick and Hinrich Schütze. 2021. Exploiting cloze-questions for few-shot text classification and natural language inference. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 255–269, Online. Association for Computational Linguistics.

Tianyuan Shi, Liangzhi Li, Zijian Lin, Tao Yang, Xiaojun Quan, and Qifan Wang. 2023. Dual-feedback knowledge retrieval for task-oriented dialogue systems. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6566–6580, Singapore. Association for Computational Linguistics.

Yiheng Shu, Zhiwei Yu, Yuhan Li, Börje Karlsson, Tingting Ma, Yuzhong Qu, and Chin-Yew Lin. 2022. TIARA: Multi-grained retrieval for robust question answering over large knowledge base. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8108–8121, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. Mpnet: Masked and permuted pre-training for language understanding. In *Advances in Neural Information Processing Systems*, volume 33, pages 16857–16867. Curran Associates, Inc.

Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge.

Jianlin Su, Ahmed Murtadha, Shengfeng Pan, Jing Hou, Jun Sun, Wanwei Huang, Bo Wen, and Yunfeng Liu. 2022. Global pointer: Novel efficient span-based approach for named entity recognition. *CoRR*, abs/2208.03054.

Hao Sun, Hengyi Cai, Bo Wang, Yingyan Hou, Xiaochi Wei, Shuaiqiang Wang, Yan Zhang, and Dawei Yin. 2023. Towards verifiable text generation with evolving memory and self-reflection. *CoRR*, abs/2312.09075.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurélien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models. *CoRR*, abs/2302.13971.

Marcos Treviso, Ji-Ung Lee, Tianchu Ji, Betty van Aken, Qingqing Cao, Manuel R. Ciosici, Michael Hassid, Kenneth Heafield, Sara Hooker, Colin Raffel, Pedro H. Martins, André F. T. Martins, Jessica Zosa Forde, Peter Milder, Edwin Simpson, Noam Slonim, Jesse Dodge, Emma Strubell, Niranjan Balasubramanian, Leon Derczynski, Iryna Gurevych, and Roy Schwartz. 2023. Efficient methods for natural language processing: A survey. *Transactions of the Association for Computational Linguistics*, 11:826–860.

Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: a free collaborative knowledgebase. *Commun. ACM*, 57(10):78–85.

Fanqi Wan, Weizhou Shen, Ke Yang, Xiaojun Quan, and Wei Bi. 2023. Multi-grained knowledge retrieval for end-to-end task-oriented dialog. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11196–11210, Toronto, Canada. Association for Computational Linguistics.

Kexin Wang, Nandan Thakur, Nils Reimers, and Iryna Gurevych. 2022a. GPL: Generative pseudo labeling for unsupervised domain adaptation of dense retrieval. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2345–2360, Seattle, United States. Association for Computational Linguistics.

Shuohang Wang, Yichong Xu, Yuwei Fang, Yang Liu, Siqi Sun, Ruochen Xu, Chenguang Zhu, and Michael Zeng. 2022b. Training data is more valuable than you think: A simple and effective method by retrieving from training data. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3170–3179, Dublin, Ireland. Association for Computational Linguistics.

Xiaozhi Wang, Tianyu Gao, Zhaocheng Zhu, Zhengyan Zhang, Zhiyuan Liu, Juanzi Li, and Jian Tang. 2021. KEPLER: A unified model for knowledge embedding and pre-trained language representation. *Transactions of the Association for Computational Linguistics*, 9:176–194.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Zihao Wang, Kwunping Lai, Piji Li, Lidong Bing, and Wai Lam. 2019. Tackling long-tailed relations and uncommon entities in knowledge graph completion. 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)*, pages 250–260, Hong Kong, China. Association for Computational Linguistics.

Zihao Wang, Anji Liu, Haowei Lin, Jiaqi Li, Xiaojian Ma, and Yitao Liang. 2024. RAT: retrieval augmented thoughts elicit context-aware reasoning in long-horizon generation. *CoRR*, abs/2403.05313.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Xiaokai Wei, Shen Wang, Dejiao Zhang, Parminder Bhatia, and Andrew O. Arnold. 2021. Knowledge enhanced pretrained language models: A compreshensive survey. *CoRR*, abs/2110.08455.

Jialin Wu and Raymond Mooney. 2022. Entity-focused dense passage retrieval for outside-knowledge visual question answering. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8061–8072, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yike Wu, Nan Hu, Sheng Bi, Guilin Qi, Jie Ren, Anhuan Xie, and Wei Song. 2023. Retrieve-rewrite-answer: A kg-to-text enhanced llms framework for knowledge graph question answering. *CoRR*, abs/2309.11206.

Bo Xu, Yong Xu, Jiaqing Liang, Chenhao Xie, Bin Liang, Wanyun Cui, and Yanghua Xiao. 2017. Cn-dbpedia: A never-ending chinese knowledge extraction system. In *Advances in Artificial Intelligence: From Theory to Practice - 30th International Conference on Industrial Engineering and Other Applications of Applied Intelligent Systems, IEA/AIE 2017, Arras, France, June 27-30, 2017, Proceedings, Part II*, volume 10351 of *Lecture Notes in Computer Science*, pages 428–438. Springer.

Derong Xu, Wei Chen, Wenjun Peng, Chao Zhang, Tong Xu, Xiangyu Zhao, Xian Wu, Yefeng Zheng, and Enhong Chen. 2023. Large language models for generative information extraction: A survey. *CoRR*, abs/2312.17617.

Xingyi Yang, Muchao Ye, Quanzeng You, and Fenglong Ma. 2021. Writing by memorizing: Hierarchical retrieval-based medical report generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5000–5009, Online. Association for Computational Linguistics.

Da Yin, Li Dong, Hao Cheng, Xiaodong Liu, Kai-Wei Chang, Furu Wei, and Jianfeng Gao. 2022. A survey of knowledge-intensive nlp with pre-trained language models. *CoRR*, abs/2202.08772.

Tom Young, Erik Cambria, Iti Chaturvedi, Hao Zhou, Subham Biswas, and Minlie Huang. 2018. Augmenting end-to-end dialogue systems with commonsense knowledge. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 4970–4977. AAAI Press.

Guoxin Yu, Lemao Liu, Haiyun Jiang, Shuming Shi, and Xiang Ao. 2023. Retrieval-augmented few-shot text classification. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 6721–6735, Singapore. Association for Computational Linguistics.

Wenhao Yu. 2022. Retrieval-augmented generation across heterogeneous knowledge. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Student Research Workshop*, pages 52–58, Hybrid: Seattle, Washington + Online. Association for Computational Linguistics.

Yuming Zhang, Hongyu Li, Yongwei Zhang, Shanshan Jiang, and Bin Dong. 2023a. SRCB at SemEval-2023 task 2: A system of complex named entity recognition with external knowledge. In *Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023)*, pages 671–678, Toronto, Canada. Association for Computational Linguistics.

Yunxiang Zhang, Muhammad Khalifa, Lajanugen Logeswaran, Moontae Lee, Honglak Lee, and Lu Wang. 2023b. Merging generated and retrieved knowledge for open-domain QA. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4710–4728, Singapore. Association for Computational Linguistics.

Zhebin Zhang, Xinyu Zhang, Yuanhang Ren, Saijiang Shi, Meng Han, Yongkang Wu, Ruofei Lai, and Zhao Cao. 2023c. IAG: Induction-augmented generation framework for answering reasoning questions. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 1–14, Singapore. Association for Computational Linguistics.

Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. 2019. ERNIE: Enhanced language representation with informative entities. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1441–1451, Florence, Italy. Association for Computational Linguistics.

Ruochen Zhao, Hailin Chen, Weishi Wang, Fangkai Jiao, Xuan Long Do, Chengwei Qin, Bosheng Ding, Xiaobao Guo, Minzhi Li, Xingxuan Li, and Shafiq Joty. 2023a. Retrieving multimodal information for augmented generation: A survey. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 4736–4756, Singapore. Association for Computational Linguistics.

Ruochen Zhao, Xingxuan Li, Shafiq Joty, Chengwei Qin, and Lidong Bing. 2023b. Verify-and-edit: A knowledge-enhanced chain-of-thought framework. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5823–5840, Toronto, Canada. Association for Computational Linguistics.

Ce Zheng, Lei Li, Qingxiu Dong, Yuxuan Fan, Zhiyong Wu, Jingjing Xu, and Baobao Chang. 2023. Can we edit factual knowledge by in-context learning? In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4862–4876, Singapore. Association for Computational Linguistics.

Wanjun Zhong, Junjie Huang, Qian Liu, Ming Zhou, Jiahai Wang, Jian Yin, and Nan Duan. 2022. Reasoning over hybrid chain for table-and-text open domain question answering. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI 2022, Vienna, Austria, 23-29 July 2022*, pages 4531–4537. ijcai.org.

Andy Zhou, Kai Yan, Michal Shlapentokh-Rothman, Haohan Wang, and Yu-Xiong Wang. 2023a. Language agent tree search unifies reasoning acting and planning in language models. *CoRR*, abs/2310.04406.

Yucheng Zhou, Tao Shen, Xiubo Geng, Chongyang Tao, Can Xu, Guodong Long, Binxing Jiao, and Daxin Jiang. 2023b. Towards robust ranker for text retrieval. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 5387–5401, Toronto, Canada. Association for Computational Linguistics.

Chenguang Zhu, Yichong Xu, Xiang Ren, Bill Yuchen Lin, Meng Jiang, and Wenhao Yu. 2022. Knowledge-augmented methods for natural language processing. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts*, pages 12–20, Dublin, Ireland. Association for Computational Linguistics.

Yutao Zhu, Huaying Yuan, Shuting Wang, Jiongnan Liu, Wenhan Liu, Chenlong Deng, Zhicheng Dou, and Ji-Rong Wen. 2023. Large language models for information retrieval: A survey. *CoRR*, abs/2308.07107.

A Appendix

A.1 Literature Search Setup

Search words The search words we used are listed below[3]:

- retriev-e/-al augment/-ed/-ion
- knowledge retriev-e/-al
- open domain/book
- knowledge inject-ed/–ion
- knowledge augment/-ed/-ion
- knowledge enhance/-ed/-ment
- knowledge integrat-ed/-ion

Search engines We first considered the following four search engines: ACL Anthology, dblp, Google Scholar and Semantic Scholar. We summarized the pros and cons as follows after conducting some probation searches.

(1) *ACL Anthology* is the only one among the four that supports full-time search. *However*, it does not include most non-*CL publications.

(2) *dblp* supports partial match, so a word stem such as augment can also match augmentation and augmented, which greatly reduces our workload. *However*, it searches only over titles.

(3) *Google Scholar* searches over title and abstract, and also supports partial match as dblp. *However*, one paper can have duplicate items which require handcraft to de-duplicate.

(4) *Semantic Scholar* also searches over title and abstract as Google Scholar. *However*, applying its built-in filter (year, conference, etc.) can wrongly lead to only very few results.

Search pipeline We use dblp and Google Scholar for literature search, since their pros and cons are complementary. Our search pipeline is defined as follows:

(1) We search on dblp and then Google Scholar the search words listed in the previous section.

(2) For all our searches, we filter those from after 2017 since this survey model-wise focuses on Tranformer-based language models.

(3) All search results are manually filtered based on their relevance to retrieval-augmented knowledge integration. For example, papers that match *knowledge injection* need to be further checked to contain retrieval-related content to be eligible.

(4) Finally, we de-duplicate results from Google Scholar and dblp. According to the ACL author guidelines[4] that referred version should be prioritized over preprints, we only keep the refereed version (mostly from dblp) of an accepted publication.

[3]Note that some words have variants: For example, *augmentation* and *augmented* for *augment*. Therefore, we need 6 separate searches for *retriev-e/-al augment/-ed/-ion*.

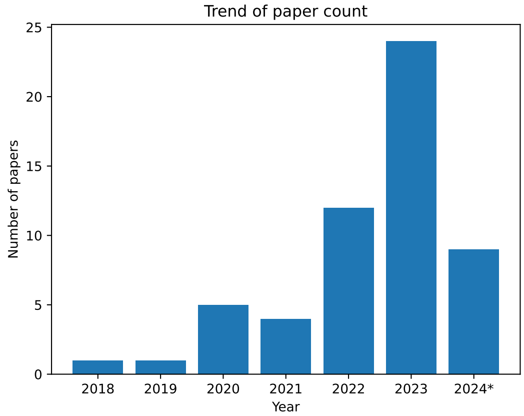

Figure 1: Number of analysed papers per year. 2024* only counts papers by April 2024.

A.2 Statistics of Literature

Statistics over years Our literature search resulted in 56 papers of RAKI, among which 1 from 2018, 1 from 2019, 5 from 2020, 4 from 2021, 12 from 2022, 24 from 2023 and 9 from 2024 (until April 2024). The trend of paper counts by year is given by Figure 1.

Statistics over venues To get an overview of which venues publish the most works, we sort the venues by the number of their accepted papers in the resulted literature search:

- EMNLP (**11**): 8 from main + 3 from findings.
- arXiv (**10**).
- ACL (**10**): 8 from main + 2 from workshops.
- NAACL (**6**): 4 from main + 1 from finding + 1 from workshop.
- AAAI (**5**).
- ICLR (**4**).
- NeurIPS (**2**).
- TKDE (**2**).
- EACL (**2**): 1 from main + 1 from finding.
- Other venues (**5**): **1** from ICML, IJCAI, SIGIR, TACL and TMLR each.

Statistics of knowledge formats Among the 56 analysed papers, 19 are from knowledge graph, 8 from tabular and 32 from natural language. Note that the sum here exceeds 56 since a paper can involve more than one knowledge sources (Oguz et al., 2022; Mo et al., 2022; Hu and Lu, 2024).

[4]https://acl-org.github.io/policies/submission

Knowledge graph $\mathcal{K}$	**Domain**	**Language**	**#Nodes**	**Example of triple** (e_h, r, e_t)
Freebase (Bollacker et al., 2008)	General	English	-	(*Richard Feynman*, `Profession`, *Physicist*)
Wikidata (Vrandečić and Krötzsch, 2014)	General	Multilingual	15.8M	(*Douglas Adams*, `educated_at`, *St John's College*)
DBPedia (Lehmann et al., 2015)	General	Multilingual	3.7M	(*Berlin*, `capital_of`, *Province of Brandenburg*)
SenticNet (Cambria et al., 2016)	Concept	Multilingual	50K	(*person*, `Desires`, *eat*)
ConceptNet (Speer et al., 2017)	Concept	Multilingual	79.9K	(*ConceptNet*, `is_a`, *semantic network*)
Wikidata5M (Wang et al., 2021)	General	English	4.6M	(*Johannes Kepler*, `occupation`, *astronomer*)
HowNet (Dong et al., 2010)	Concept	Chinese, English	–	(*doctor*, `hypernym`, *human*)
CN-DBpedia (Xu et al., 2017)	General	Chinese	9M	(知识图谱*KG*, 也称`alias`, 科学知识图谱*Sci KG*)
MedicalKG (Liu et al., 2020)	Medicine	Chinese	–	(彩超*ultrasound*, 类别`hypernym`, 检查*treatment*)

Table 1: Overview of some knowledge graphs applied in retrieval-augmentation literature. #Nodes denotes the number of entities in the knowledge graph. Regarding example triples from non-English knowledge graphs (i.e. CN-DBpedia and MedicalKG), their English translations are appended to each element in the triples. The number of nodes of HowNet is not directly given in the original paper (Dong et al., 2010), and Liu et al. (2020) use a refined version of HowNet with 52,576 triples. The Freebase (Bollacker et al., 2008) paper gives its number of triples to be 125M without giving the number of nodes. MedicalKG (Liu et al., 2020) has 13,864 triples.

Knowledge graph $\mathcal{K}$	**Target task** $\mathcal{T}$
Freebase (Bollacker et al., 2008)	QA (Oguz et al., 2022)
DBPedia (Lehmann et al., 2015)	Dialogue Generation (Li et al., 2022)
SenticNet (Cambria et al., 2016)	Open-Domain Response Selection (Young et al., 2018)
ConceptNet (Speer et al., 2017)	QA (Lv et al., 2020; Bian et al., 2021; Huang et al., 2023)
Wikidata (Vrandečić and Krötzsch, 2014)	KGQA (Baek et al., 2023a), NER (Zhang et al., 2023a), ED (Ayoola et al., 2022)
Wikidata5M (Wang et al., 2021)	Entity Typing (Fu et al., 2023), Relation Classicification (Fu et al., 2023)
CN-DBpedia (Xu et al., 2017), HowNet (Dong et al., 2010), MedicalKG (Wang et al., 2021)	NER (Fu et al., 2023)

Table 2: Previous work to retrieve knowledge graphs for specific target tasks. The left column lists the external knowledge graphs. The right column presents the target tasks together with retrieval-augmented papers conducting the tasks. QA: Question Answering. KGQA: Knowledge Graph Question Answering. NER: Named Entity Recognition. ED: Entity Disambiguation.

Previous work	**Feature for retrieval**	**Level**	**Selection criterion**
Fu et al. (2023)	Entity ID (from TagMe)	Entity	Exact match
Li et al. (2023)	Entity name (from in-context learning)	Entity	Exact match
Lv et al. (2020)	Entity name (from mention detection)	Entity	Exact match
Zhang et al. (2023a)	Entity name (from global pointer (Su et al., 2022))	Entity	Best match from ES
Shu et al. (2022)	Entity name (from mention detection + alias mapping)	Entity	Exact match
Young et al. (2018); Bian et al. (2021)	n-gram	Entity	Exact n-gram match
Andrus et al. (2022) (QA)	Edit distance	Triple	Min. edit distance
Andrus et al. (2022) (story completion)	sBERT (Reimers and Gurevych, 2019) embeddings	Triple	Max. cosine similarity
Oguz et al. (2022)	DPR (Karpukhin et al., 2020) embeddings	Triple	Max. cosine similarity
Baek et al. (2023a)	MPNet (Song et al., 2020) embeddings	Triple	—

Table 3: Overview of prior graph retrieval methods of retrieval-based knowledge graph augmentation. ES: Elastic-Search. sBERT: Sentence-BERT. (Baek et al., 2023a) does not explicitly give the criterion score over embeddings.

Previous work	Prompt template	Knowledge $\mathcal{K}'$ to fill-in
w/o reformulation		
Li et al. (2022)	USER: Who is *Michael F. Phelps*? KG: $\{\mathcal{K}'\}$.	<Michael F. Phelps, occupation, Swimmer>
Fu et al. (2023)	Who is **Michael F. Phelps**? $\{\mathcal{K}'\}$.	(Michael F. Phelps occupation Swimmer)
Baek et al. (2023a,b)	Below are facts that might be meaningful to answer the given question: $\{\mathcal{K}'\}$. Question: Who is *Michael Phelps*? Answer:	(Michael F. Phelps, occupation, Swimmer)
Reformulation with relation-NL mapping		
Lv et al. (2020)	$\{\mathcal{K}'\}$. <SEP> Who is *Michael F. Phelps*?	Michael F. Phelps has occupation swimmer.
Bian et al. (2021)	$\{\mathcal{K}'\}$ [SEP] Who is *Michael F. Phelps*? A.lawyer ǀ B. businessman ǀ C. swimmer [SEP]	Michael F. Phelps has occupation swimmer.
Reformulation by LMs		
Bian et al. (2021)	$\{\mathcal{K}'\}$ [SEP] Who is *Michael F. Phelps*? A.lawyer ǀ B. businessman ǀ C. swimmer [SEP]	Michael F. Phelps is a swimmer. (*paraphrase based*)
Bian et al. (2021)	$\{\mathcal{K}'\}$ [SEP] Who is *Michael F. Phelps*? A.lawyer ǀ B. businessman ǀ C. swimmer [SEP]	Phelps (born June 30, 1985) is an American former swimmer. (*retrieval based*)
Wu et al. (2023)	Below are the facts that might be relevant to answer the question:$\{\mathcal{K}'\}$. Question: Who is *Michael F. Phelps*? Answer:	Michael F. Phelps is a swimmer by profession. (*paraphrase by GPT-3.5*)
Andrus et al. (2022)	Story: -. Useful Information: $\{\mathcal{K}'\}$. Question: Who is *Michael F. Phelps*? Answer:	Michael F. Phelps is professionally involved in swimming. (*paraphrase by GPT-3.5*)

Table 4: Overview of prompts to augment graph. Prompts are concluded into three categories based on reformulation. Assume entity *Michael F. Phelps* is recognized in the question *Who is Michael F. Phelps* during retrieval and marked as italic. The knowledge is given by (Baek et al., 2023b): (*Michael F. Phelps, occupation, Swimmer*). Due to availability of models, we employ GPT-3.5 (instead of GPT-3 used in Andrus et al. (2022)) to generate paraphrase.