Retrieval-augmented Generation across Heterogeneous Knowledge

Wenhao Yu
University of Notre Dame, USA
wyu1@nd.edu

Abstract

Retrieval-augmented generation (RAG) methods have been receiving increasing attention from the NLP community and achieved state-of-the-art performance on many NLP downstream tasks. Compared with conventional pre-trained generation models, RAG methods have remarkable advantages such as easy knowledge acquisition, strong scalability, and low training cost. Although existing RAG models have been applied to various knowledge-intensive NLP tasks, such as open-domain QA and dialogue systems, most of the work has focused on retrieving unstructured text documents from Wikipedia. In this paper, I first elaborate on the current obstacles to retrieving knowledge from a single-source homogeneous corpus. Then, I demonstrate evidence from both existing literature and my experiments, and provide multiple solutions on retrieval-augmented generation methods across heterogeneous knowledge.

1 Introduction

In recent years, large pre-trained language models (PLMs), such as T5 (Raffel et al., 2020) and GPT-3 (Brown et al., 2020), have revolutionized the field of natural language processing (NLP), achieving remarkable performance on various downstream tasks (Qiu et al., 2020). These PLMs have learned a substantial amount of in-depth knowledge from the pre-training corpus (Petroni et al., 2019), so they can predict the outputs on downstream tasks without access to any external memory or raw text, as a parameterized implicit knowledge base (Roberts et al., 2020). The way of fine-tuning PLMs using only *input-output* pairs of target data is often referred to as *close-book* setting (Petroni et al., 2019).

While this development is exhilarating, such large-scale PLMs still suffer from the following drawbacks: (i) They are usually trained offline, making the model agnostic to the latest information, e.g., asking a chat-bot trained from 2011-2018 about COVID-19 (Yu et al., 2022b). (ii) They make predictions by only "looking up information" stored in its parameters, leading to inferior interpretability (Shuster et al., 2021). (iii) They are mostly trained on general domain corpora, making them less effective on domain-specific tasks (Gururangan et al., 2020). (iv) Their pre-training phase can be prohibitively expensive for academic research groups, limiting the model pre-training to only a few industry labs (Izsak et al., 2021).

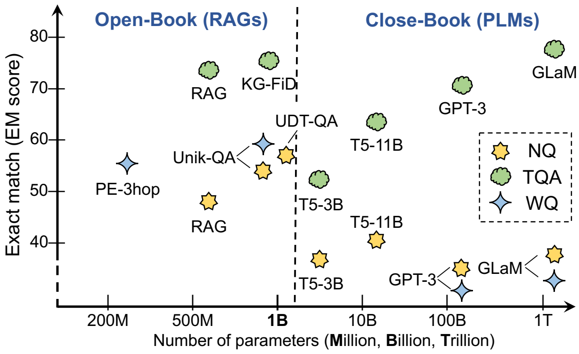

Figure 1: The RAG methods significantly outperform large-scale PLMs on three open-domain QA tasks while trained with much fewer parameters than PLMs.

The solution that seems obvious at first glance is to allow language models free access to open-world resources, such as encyclopedias and books. The way of augmenting the input of PLMs with external information is often referred to as *open-book* setting (Mihaylov et al., 2018). A prominent method in the open-book setting is retrieval-augmented generation (RAG) (Lewis et al., 2020b; Yu et al., 2022c), a new learning paradigm that fuses PLMs and traditional IR techniques, which has achieved state-of-the-art performance in many knowledge-intensive NLP tasks (Petroni et al., 2021). Compared with large-scale PLMs counterparts, e.g., GPT-3, the RAG model has some remarkable ad-

* This is a thesis proposal paper presented at the student research workshop (SRW) at NAACL 2022 in Seattle, USA.

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
July 10-15, 2022 ©2022 Association for Computational Linguistics

vantages: (i) The knowledge is not implicitly stored in model parameters, but is explicitly acquired in a plug-and-play manner, leading to great scalability; (ii) Instead of generating from scratch, the model generates outputs based on some retrieved references, which eases the difficulty of text generation.

Although the RAG models have been widely used in the existing literature, most of the work has focused on retrieving unstructured text from general domain corpus, e.g., Wikipedia. However, the performance is often limited by the coverage of only one certain knowledge. For example, only a finite portion of questions could be answered from Wikipedia passages in many open-domain QA datasets, while the remaining could only rely on the input question because no supportive documents could be retrieved (Oguz et al., 2022). In this paper, I first elaborate on the current obstacles to retrieving knowledge from a single-source homogeneous corpus. Then, I demonstrate several pieces of evidence from both existing literature and my own experiments, and provide multiple potential solutions on retrieval-augmented generation methods across heterogeneous knowledge.

2 Background

I will first provide a formal definition of the RAG framework and list necessary notations. RAG aims to predict the output y based on the source input x (x, y are from a corpus $\mathcal{D}$), while a document reference set $\mathcal{Z}$ is accessible (e.g., Wikipedia). Besides, the association between a document $z \in \mathcal{Z}$ and the tuple $(x, y) \in \mathcal{D}$ is not necessarily known, though it could be provided by human annotations (Dinan et al., 2019) or weakly supervised signals (Karpukhin et al., 2020).

Overall, a general RAG framework has two major components: (i) a document retriever and (ii) a text generator, as shown in Figure 2. The objective of the RAG is to train a model to maximize the likelihood of y given x and $\mathcal{Z}$, In practice, $\mathcal{Z}$ often contains millions of documents, rendering enumeration over z impossible. Therefore, the first step of RAG is to leverage a document retriever, e.g., DPR (Karpukhin et al., 2020), to narrow down the search to a handful of relevant documents. The retriever takes x and $\mathcal{Z}$ as input and yields relevance scores $\{s_1, \cdots, s_K\}$ of the top-K documents $Z = \{z_{(1)}, \cdots, z_{(K)}\}$. Then, the second step of RAG is to use a text generator, e.g., BART (Lewis et al., 2020a) and T5 (Raffel et al., 2019), to produce desired output y by taking both input x and retrieved document set Z as conditions.

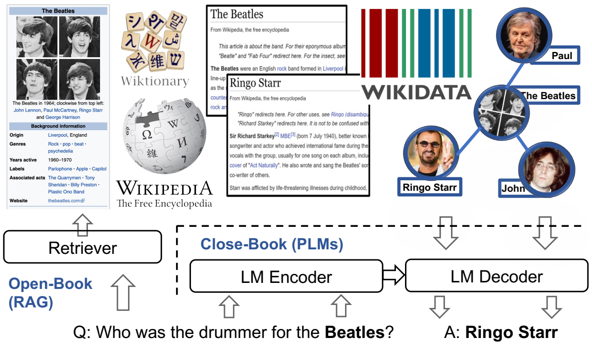

Figure 2: Compared with PLMs, RAG models directly seeks knowledge (e.g., texts, tables and KGs) from external information sources to help answer questions.

Document Retriever. A neural document retriever typically employs two independent encoders like BERT (Devlin et al., 2019) to encode the query and the document separately, and estimates their relevance by computing a single similarity score between two encoded representations. For example, in DPR (Karpukhin et al., 2020), the documents Z and context queries x are mapped into the same dense embedding space. The relevance score $s(x, z)$ for each document z is computed as the vector inner product between document embedding h_z and query embedding h_x, i.e., $s(x, z) = h_x^T \times h_z$.

Text Generator. It can use any encoder-decoder framework, such as BART (Lewis et al., 2020a) and T5 (Raffel et al., 2019). The model takes input sequence, as well as the support documents to generate the desired output. A naive method for combining the input sequence with the support documents is to concatenate them sequentially (Lewis et al., 2020a). However, this method suffers from the input sequence length limitation and high computation cost. FiD (Izacard and Grave, 2021) processed passages independently in the encoder, performed attention over all the retrieved passages, which demonstrated state-of-the-art performance on many knowledge-intensive NLP tasks.

3 Proposed Work

3.1 Background and Motivation

Despite achieving remarkable performance, previous efforts of retrieval-augmented generation (RAG) works mainly exploit only a single-source homogeneous knowledge retrieval space, i.e., Wikipedia passages (Karpukhin et al., 2020; Lewis et al., 2020b; Petroni et al., 2021; Izacard and

Grave, 2021; Yu et al., 2022a). However, their model performance might be limited by the coverage of only one certain knowledge. For example, only a finite portion of questions can be answered from the Wikipedia passages in many open-domain QA datasets, while the remaining can only rely on the input query because no supportive documents can be retrieved (Oguz et al., 2022). Since much useful information cannot be fulfilled based on Wikipedia alone, a natural solution is to expand the retrieval corpus from Wikipedia to the entire World Wide Web (WWW). However, suffering from the long-tail issue and the cost of a massive workforce, it is not wise to improve the coverage by expanding the number of entries in a single-source knowledge (Piktus et al., 2021; Lazaridou et al., 2022). For example, as shown in Table 1, increasing the retrieval space from Wikipedia (22M documents) to the web-scale corpus CCNet (906M documents) even hurts model performance on NQ and HotpotQA datasets. This is most likely due to the lower quality (where quality could mean truthfulness, objectivity, lack of harmful content, source reliability, etc) of the web corpus, compared with the Wikipedia corpus (Piktus et al., 2021).

Instead of expanding the number of entries in a single-source knowledge, an alternative solution is resorting to heterogeneous knowledge sources. This is also in line with our human behavior of answering questions that often seek a variety of knowledge learned from different sources. Therefore, grounding generation across heterogeneous knowledge sources is a natural solution to improve knowledge coverage and have more room to select appropriate knowledge. It is worth mentioning that no knowledge type can always perform the best. The most suitable knowledge depends on the case, in which multiple knowledge might need to be combined for answering one question.

3.2 Evidence from Existing Literature

There are several studies in the existing literature that combine multiple knowledge to enhance language models, such as augmenting commonsense reasoning with knowledge graphs (Yu et al., 2022d), and introducing multi-modal visual features to enhance emotional dialogue (Liang et al., 2022). However, most of them use aligned knowledge from different sources (e.g., graph-text pairs, image-text pairs), without retrieving knowledge from a large-scale heterogeneous corpus.

Table 1: With a larger corpus of unstructured text retrieval – CCNet, the model performs even worse than retrieving from Wikipedia alone on the NQ and HotpotQA datasets. The model used in the table is DPR+FiD.

No.	Source	# docs	NQ	TQA	HotpotQA
1	Wikipedia	22M	**51.4**	71.0	**36.9**
2	CCNet	906M	48.6	**73.1**	31.6

‘

Table 2: Exact match (EM-score) of retrieving heterogeneous knowledge for three open-domain QA benchmarks. The model used in the table is DPR+FiD.

No.	Knowledge type			Dataset		
	Text	Table	KG	NQ	TQA	WebQ
1	√			49.0	64.0	50.6
2		√		36.0	34.5	41.0
3			√	27.9	35.4	55.2
4	√	√		**54.1**	**65.1**	50.2
5	√	√	√	54.0	64.1	**57.8**

The most relevant works to this proposal are UniK-QA (Oguz et al., 2022) and PLUG (Li et al., 2021). In UniK-QA, Oguz et al. (2022) proposed to retrieve information from a merged corpus of structured (i.e., KG triples), semi-structured (i.e., tables) and unstructured data (i.e., text passages) for open-domain QA (Oguz et al., 2022). Their experiments were conducted on multiple open-domain QA benchmark datasets, including NaturalQuestions (NQ) (Kwiatkowski et al., 2019), TriviaQA (TQA) (Joshi et al., 2017) and WebQuestions (WebQ) (Berant et al., 2013).

The results in the first three lines in Table 2 highlight the limitation of current state-of-the-art open-domain QA models which use only one information source. Among the three types of knowledge sources, text-only methods perform best on NQ and TQA datasets, and KG-only methods perform best on WebQ datasets. This is because most of the questions in WebQ are collected from Freebase. The results in the last two lines show that adding semi-structured and structured information sources significantly improves the performance over text-only models on NQ and TQA datasets. This indicates tables and knowledge graph triples contain valuable knowledge which is either absent in the unstructured texts or harder to extract from them.

It is worth mentioning that knowledge heterogeneity can be defined not only by the format of knowledge data (i.e., structured and unstructured knowledge), but also by the scope of knowledge data (i.e., encyclopedic and common-

Table 3: Commonly used knowledge sources.

	Unstructured	(Semi-)structured
Encyclopedic knowledge	Wikipedia, AMiner	Wikidata, Freebase
Commonsense knowledge	ConceptNet, CSKG, Atomic	OMCS, ARC, Wiktionary

Table 4: Accuracy of retrieving heterogeneous knowledge for commonsense reasoning over entity tasks.

No.	Knowledge source		Dataset	
	Commonsense	Encyclopedia	CREAK	CSQA2.0
1		√	86.55	59.28
2	√		82.28	58.23
3	√	√	**87.57**	**60.49**

sense knowledge). Table 3 shows common knowledge sources under two categories. In addition of combining structured and unstructured knowledge, combining encyclopedic and commonsense knowledge also brings benefits for many NLP tasks, such as commonsense reasoning over entities. Some preliminary experiments were conducted on CREAK (Onoe et al., 2021) and CSQA2.0 (Talmor et al., 2021) datasets. CREAK is a dataset of human-authored English claims about entities that are either true or false, such as "Harry Potter can teach classes on how to fly on a broomstick *(True)*." The model is supposed to bridge fact-checking about entities with commonsense inferences. An entity fact relevant to this statement, *"Harry Potter is a wizard and is skilled at riding a broomstick"*, can be retrieved from Wikipedia. A commonsense knowledge, *"if you are good at a skill you can teach others how to do it"*, can be retrieved from the *ATOMIC* (Sap et al., 2019). By leveraging both commonsense knowledge and encyclopedic knowledge in the first-step retrieval, as shown in Table 4, the RAG model can achieve superior performance than only using either of them.

3.3 Proposed Solutions

As mentioned above, heterogeneous knowledge is often required when solving open-domain QA and many other knowledge-intensive NLP tasks. One natural assumption is to expand knowledge sources and add more data to increase the coverage of relevant contexts, thereby improving the end-to-end performance. In this section, I will present three potential solutions for grounding generation across heterogeneous knowledge.

3.3.1 Homogenize Different Knowledge to a Unified Knowledge Representation

The first solution is to homogenize different knowledge source data into a unified data format – unstructured text. This transformation will then require only one retriever, enable relevance comparison across different types of data, and offer textual knowledge to easily augment the input of generation models by concatenation. Table 3 shows some commonly used knowledge sources. For example, semi-structured tables and structured knowledge graph triples can be converted into the unstructured text by template-based methods (Bosselut et al., 2019; Oguz et al., 2022) or neural data-to-text methods (Wang et al., 2021; Nan et al., 2021).

First, the template-based method is easy to implement and requires no training process. For example, a relation triplet in a knowledge graph consists of subject, predicate, and object. It can be serialized by concatenating the surface form of the three elements to be a sequence of words. Besides, a table can also be hierarchically converted into text format: first, concatenate cell values of each row separated by commas; then combine these rows' text forms delimited by semicolons. Although the template-based method is simple but may suffer from incorrect syntax and incomplete semantics. On the contrary, the neural graph-to-text and table-to-text generation methods rely on pre-trained language models that may ensure syntax correctness and semantic completeness. Once either type of the methods converts the structured and semi-structured data to unstructured text, a dense retriever model such as DPR (Karpukhin et al., 2020) can be used to index all of them and retrieve relevant knowledge. The reader model will concatenate the retrieved text with original input and compute full attention over the entire representations through a T5 (Raffel et al., 2020) decoder. This unified knowledge index allows the models to learn knowledge of various formats and scopes of data, and the model can simultaneously retrieve information from a unified index of multiple knowledge sources to improve the knowledge coverage.

3.3.2 Multi-virtual Hops Retrieval over Heterogeneous Knowledge

Retrieved data are expected to bridge the gap between inputs and outputs of generation models. In other words, retrievers are trained to provide information that is found with the inputs as queries and related to the outputs. Ideally, they find the output-related information just once. However, that may actually take multiple hops of retrieval across knowledge sources. Thus, the second solution is to iteratively retrieve knowledge from different sources. Regarding an entity, encyclopedic knowledge usually contains its attribute information (e.g., age, duration), while commonsense knowledge includes universally recognized facts in human's daily life. For example, the entity "soup" in Wikipedia is described as "a primarily liquid food, generally served warm or hot, made by combining ingredients of meat or vegetables with stock, milk, or water"; and in the OMCS corpus (Singh et al., 2002), it contains a well-known fact "soup and salad can be a healthy lunch". Therefore, to answer the question "What are the common ingredients in a healthy lunch?", the encyclopedic corpus and commonsense corpus can provide complementary knowledge that should be both leveraged.

Besides, it also might be necessary to first read a subset of the corpus to extract the useful information, and then further retrieve information from other knowledge sources. For example, given input q, it may take k steps, each step retrieving data d_i from source $s_i \in \mathcal{S}$ with an incremental query $q_i = q \oplus d_1 \oplus \cdots \oplus d_{i-1}$ $(i \leq k)$ until the final d_k contains the information that can directly augment the generation of outputs o. Here $\mathcal{S}$ includes various sources such as text corpora, tables, and knowledge graphs. To achieve this, however, the primary challenge for training such a multi-hop retriever is that it cannot observe any intermediate document for supervision but only the final output. So, the multi-virtual hops retrieval (MVHL) needs to perform multi-hop retrieval without any intermediate signal. I will discuss two promising designs as below. First, the MVHL approach will dynamically determine when the multi-hops retrieval finishes. I denote the relevance score between query q_i and data d_i from source s_i by $r(d_i; q_i, s_i)$. The search continues at the i-th step, if $r(d_i; q_i, s_i) > r(d_i; q_{i-1}, s_{i-1} \cup s_i)$; because d_i brings new relevant information that was not able to be retrieved at the $(i-1)$-th step or any previous steps. Second, the MVHL can use sequential models instead of heuristics to control the multi-hops search. The search is expected to finish at step i, when the relevance between the retrieved data d_i and output o, which can be computed by BERTScore (Zhang et al., 2020), achieves a local maximum. In order to model the relationship between this target relevance $r_o(d_i)$ and the retrieval score $r(d_i; q_i, s_i)$, a straightforward solution is to train a multi-hop retriever with only the output o using a fixed number of hops K (5 or 10) and use the validation set to choose the best model. With that model, I can observe the K-length series of r and r_o, and train an RNN model that predicts $r_o(d_k)$ based on the first k elements in the r series. The search terminates when the predicted r_o decreases.

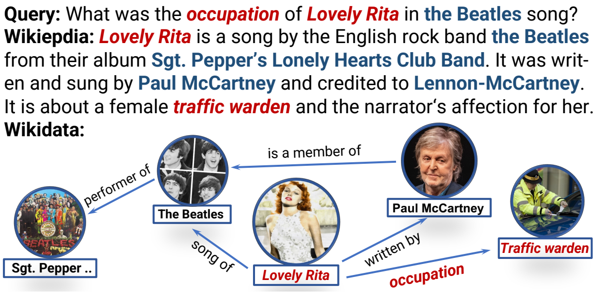
Query: What was the ***occupation*** of ***Lovely Rita*** in **the Beatles** song?
Wikiepdia: ***Lovely Rita*** is a song by the English rock band **the Beatles** from their album **Sgt. Pepper's Lonely Hearts Club Band**. It was writen and sung by **Paul McCartney** and credited to **Lennon-McCartney**. It is about a female ***traffic warden*** and the narrator's affection for her.
Wikidata:

Figure 3: Reasoning over retrieved documents on structured knowledge provides explicit knowledge grounding to help answer questions. For example, in WebQ, 46.9%/56.1% of the questions can be solved by one/two-hop neighbors on the query-document subgraph.

3.3.3 Reasoning over Retrieved Documents Based on Structured Knowledge

Traditional reader modules typically concatenate the input query and retrieved documents sequentially, and then feed them into a pre-trained generation model, such as T5. Although the token-level attention can *implicitly* learning some relational patterns between the input query and retrieved documents, it does not fully utilize the structured knowledge that can provide more *explicit* grounding. As shown in Figure 3, the relational information between important entities in the input query (i.e., Lovely Rita) and the retrieved documents (i.e., traffic warden) may require reasoning over structured knowledge that is not explicitly stated in the context. So, the third solution is to perform multi-hop reasoning on structured knowledge, e.g., Wikidata, to learn relational patterns between the input query and retrieved documents. In this way, the representation of retrieved documents is further enriched by structured knowledge. To perform knowledge reasoning over retrieved documents, the idea is to first extract a query-document subgraph since direct reasoning on the entire knowledge graph is intractable. Entities on the subgraph can be mapped by given hyperlinks in Wikipedia passages. Then, a multi-relational graph encoder iteratively updates the representation of each entity node by aggregating information from its neighboring nodes and edges. Then, the embedded node and relation representations, as well as the query and document representations, are then fused into the reader model.

Acknowledgements

Many thanks to my doctoral supervisor Dr. Meng Jiang and my doctoral thesis committee members Dr. Nitesh Chawala, Dr. David Chiang, Dr. Heng Ji, Dr. Scott Yih for giving me constructive suggestions. My research is mainly funded by National Science Foundation (NSF) grants IIS-1849816, CCF-1901059, IIS-2119531, and IIS-2142827.

References

Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. 2013. Semantic parsing on freebase from question-answer pairs. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Antoine Bosselut, Hannah Rashkin, Maarten Sap, Chaitanya Malaviya, Asli Celikyilmaz, and Yejin Choi. 2019. Comet: Commonsense transformers for automatic knowledge graph construction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems (Neurips)*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL)*.

Emily Dinan, Stephen Roller, Kurt Shuster, Angela Fan, Michael Auli, and Jason Weston. 2019. Wizard of wikipedia: Knowledge-powered conversational agents. In *International Conference on Learning Representations (ICLR)*.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *58th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Gautier Izacard and Édouard 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 (EACL)*.

Peter Izsak, Moshe Berchansky, and Omer Levy. 2021. How to train bert with an academic budget. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. 2017. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL)*.

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.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics (TACL)*.

Angeliki Lazaridou, Elena Gribovskaya, Wojciech Stokowiec, and Nikolai Grigorev. 2022. Internet-augmented language models through few-shot prompting for open-domain question answering. *arXiv preprint arXiv:2203.05115*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020a. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Annual Meeting of the Association for Computational Linguistics (ACL)*.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020b. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems (Neruips)*.

Yu Li, Baolin Peng, Yelong Shen, Yi Mao, Lars Liden, Zhou Yu, and Jianfeng Gao. 2021. Knowledge-grounded dialogue generation with a unified knowledge representation. *arXiv preprint arXiv:2112.07924*.

Yunlong Liang, Fandong Meng, Ying Zhang, Yufeng Chen, Jinan Xu, and Jie Zhou. 2022. Emotional conversation generation with heterogeneous graph neural network. *Artificial Intelligence*.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. Can a suit of armor conduct electricity? a new dataset for open book question answering. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Linyong Nan, Dragomir Radev, Rui Zhang, Amrit Rau, Abhinand Sivaprasad, Chiachun Hsieh, Xiangru Tang, Aadit Vyas, Neha Verma, Pranav Krishna, et al. 2021. Dart: Open-domain structured data record to text generation. In *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL)*.

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. *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*.

Yasumasa Onoe, Michael JQ Zhang, Eunsol Choi, and Greg Durrett. 2021. Creak: A dataset for commonsense reasoning over entity knowledge. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Neurips)*.

Fabio Petroni, Aleksandra Piktus, Angela Fan, Patrick Lewis, Majid Yazdani, Nicola De Cao, James Thorne, Yacine Jernite, Vladimir Karpukhin, Jean Maillard, et al. 2021. 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 (NAACL)*.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? 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)*.

Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Dmytro Okhonko, Samuel Broscheit, Gautier Izacard, Patrick Lewis, Barlas Oğuz, Edouard Grave, Wen-tau Yih, et al. 2021. The web is your oyster–knowledge-intensive nlp against a very large web corpus. *arXiv preprint arXiv:2112.09924*.

Xipeng Qiu, Tianxiang Sun, Yige Xu, Yunfan Shao, Ning Dai, and Xuanjing Huang. 2020. Pre-trained models for natural language processing: A survey. In *Science China Technological Sciences*. Springer.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*.

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. In *Journal of Machine Learning Research*.

Adam Roberts, Colin Raffel, and Noam Shazeer. 2020. How much knowledge can you pack into the parameters of a language model? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5418–5426.

Maarten Sap, Ronan Le Bras, Emily Allaway, Chandra Bhagavatula, Nicholas Lourie, Hannah Rashkin, Brendan Roof, Noah A Smith, and Yejin Choi. 2019. Atomic: An atlas of machine commonsense for if-then reasoning. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*.

Kurt Shuster, Spencer Poff, Moya Chen, Douwe Kiela, and Jason Weston. 2021. Retrieval augmentation reduces hallucination in conversation. In *Findings of Empirical Methods in Natural Language Processing*.

Push Singh, Thomas Lin, Erik T Mueller, Grace Lim, Travell Perkins, and Wan Li Zhu. 2002. Open mind common sense: Knowledge acquisition from the general public. In *International Conferences on the Move to Meaningful Internet Systems*. Springer.

Alon Talmor, Ori Yoran, Ronan Le Bras, Chandra Bhagavatula, Yoav Goldberg, Yejin Choi, and Jonathan Berant. 2021. Commonsenseqa 2.0: Exposing the limits of ai through gamification. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Neurips)*.

Luyu Wang, Yujia Li, Ozlem Aslan, and Oriol Vinyals. 2021. Wikigraphs: A wikipedia text-knowledge graph paired dataset. In *Proceedings of the Fifteenth Workshop on Graph-Based Methods for Natural Language Processing (TextGraphs-15)*.

Donghan Yu, Chenguang Zhu, Yuwei Fang, Wenhao Yu, Shuohang Wang, Yichong Xu, Xiang Ren, Yiming Yang, and Michael Zeng. 2022a. Kg-fid: Infusing knowledge graph in fusion-in-decoder for open-domain question answering. *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.

Wenhao Yu, Chenguang Zhu, Yuwei Fang, Donghan Yu, Shuohang Wang, Yichong Xu, Michael Zeng, and Meng Jiang. 2022b. Dict-bert: Enhancing language model pre-training with dictionary. *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.

Wenhao Yu, Chenguang Zhu, Zaitang Li, Zhiting Hu, Qingyun Wang, Heng Ji, and Meng Jiang. 2022c. A survey of knowledge-enhanced text generation. In *ACM Computing Survey (CSUR)*.

Wenhao Yu, Chenguang Zhu, Lianhui Qin, Zhihan Zhang, Tong Zhao, and Meng Jiang. 2022d. Diversifying content generation for commonsense reasoning with mixture of knowledge graph experts. In *Annual Meeting of the Association for Computational Linguistics (ACL)*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.