RETRIEVAL
AUGMENTED
GENERATION

About the paper

- First posted May 22, 2020 on arXiv
- By researchers from Facebook AI Research, University College London, and New York University
- Widely cited and influential across industry (Microsoft, Google, Amazon, NVIDIA adopt RAG-style systems)

Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks

Patrick Lewis[†‡], Ethan Perez[⋆],

Aleksandra Piktus[†], Fabio Petroni[†], Vladimir Karpukhin[†], Naman Goyal[†], Heinrich Küttler[†],

Mike Lewis[†], Wen-tau Yih[†], Tim Rocktäschel[†‡], Sebastian Riedel[†‡], Douwe Kiela[†]

[†]Facebook AI Research; [‡]University College London; [⋆]New York University;
plewis@fb.com

Abstract

Large pre-trained language models have been shown to store factual knowledge in their parameters, and achieve state-of-the-art results when fine-tuned on downstream NLP tasks. However, their ability to access and precisely manipulate knowledge is still limited, and hence on knowledge-intensive tasks, their performance lags behind task-specific architectures. Additionally, providing provenance for their decisions and updating their world knowledge remain open research problems. Pre-trained models with a differentiable access mechanism to explicit non-parametric memory can overcome this issue, but have so far been only investigated for extractive downstream tasks. We explore a general-purpose fine-tuning recipe for retrieval-augmented generation (RAG) — models which combine pre-trained parametric and non-parametric memory for language generation. We introduce RAG models where the parametric memory is a pre-trained seq2seq model and the non-parametric memory is a dense vector index of Wikipedia, accessed with a pre-trained neural retriever. We compare two RAG formulations, one which conditions on the same retrieved passages across the whole generated sequence, and another which can use different passages per token. We fine-tune and evaluate our models on a wide range of knowledge-intensive NLP tasks and set the state of the art on three open domain QA tasks, outperforming parametric seq2seq models and task-specific retrieve-and-extract architectures. For language generation tasks, we find that RAG models generate more specific, diverse and factual language than a state-of-the-art parametric-only seq2seq baseline.

1 Introduction

Pre-trained neural language models have been shown to learn a substantial amount of in-depth knowledge from data [47]. They can do so without any access to an external memory, as a parameterized implicit knowledge base [51, 52]. While this development is exciting, such models do have downsides: They cannot easily expand or revise their memory, can't straightforwardly provide insight into their predictions, and may produce "hallucinations" [38]. Hybrid models that combine parametric memory with non-parametric (i.e., retrieval-based) memories [20, 26, 48] can address some of these issues because knowledge can be directly revised and expanded, and accessed knowledge can be inspected and interpreted. REALM [20] and ORQA [31], two recently introduced models that combine masked language models [8] with a differentiable retriever, have shown promising results,

34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada.

In this presentation

1. What is RAG?
2. Approaches
3. Experiments / Results
4. Applications
5. Strengths / Weaknesses
6. Related Works

When discussing downsides of pre-trained neural language models

They cannot easily expand or revise their memory, can’t straightforwardly provide insight into their predictions, and may produce “hallucinations”

generating plausible yet nonfactual content

What is RAG?

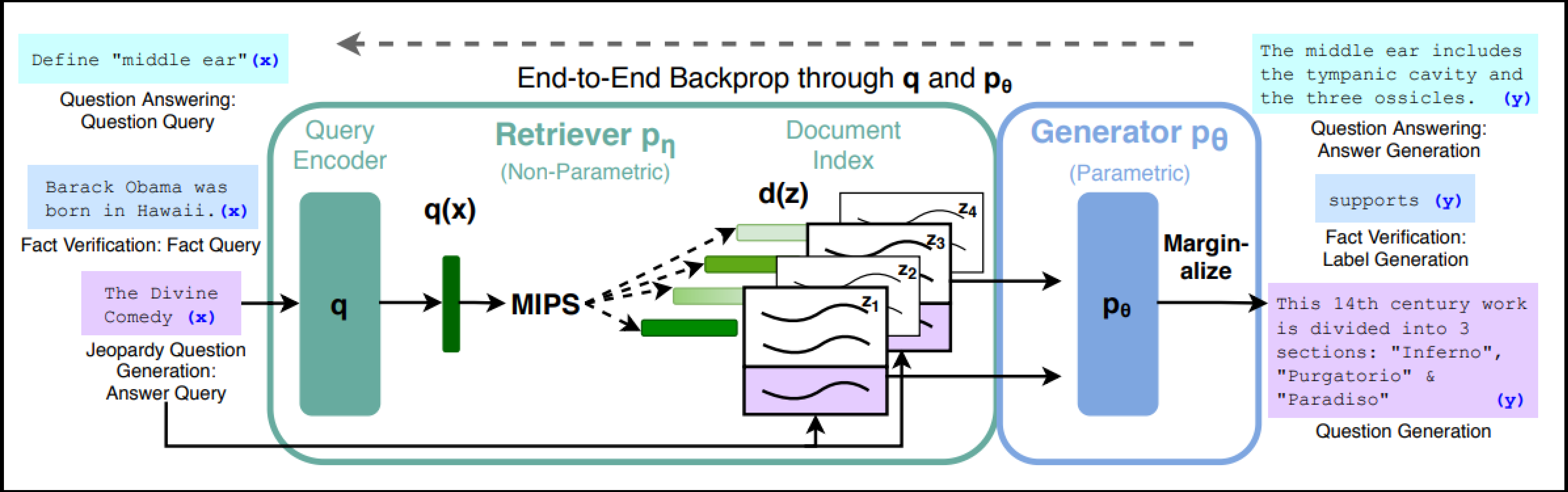
Models which combine pre-trained parametric and non-parametric memory for language generation

Retrieval → Fetching relevant information from a stored DB

Augmented → Enriching by adding extra context to response

Generation → Producing text (or other content) from a model

Overview



User Prompt

A dense vector index of Wikipedia, accessed with a pre-trained neural retriever 'BERT' using Maximum Inner Product Search (MIPS) to find top-K document

A pre-trained seq2seq transformer 'BART'

Output

Retriever

A dense vector index of Wikipedia, accessed with a pre-trained neural retriever

Goal: Retrieve **k documents** most relevant to a user **query** x

- Each document z and the query x are **encoded separately** into vectors
    - $\mathbf{d}(z) = \text{BERT}_d(z)$ —a **document encoder**
    - $\mathbf{q}(x) = \text{BERT}_q(x)$ —a **query encoder**
- Find the **match score** given by **the dot product** of **d** and **q**

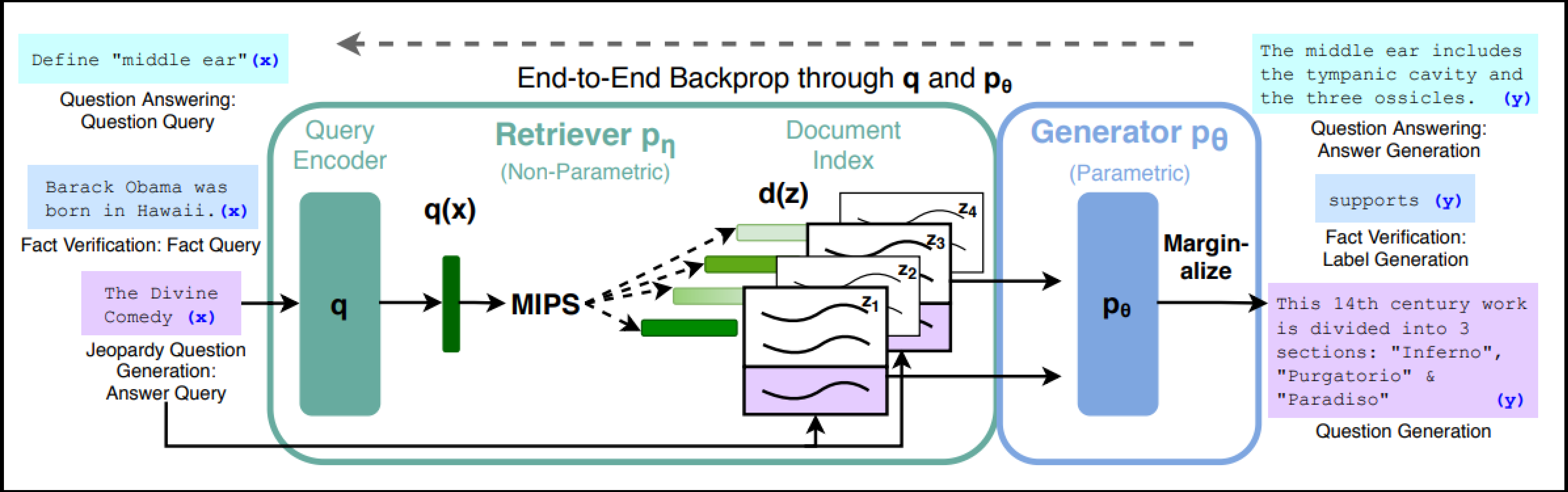
Generator

A pre-trained seq2seq transformer 'BART'

Goal: generates the answer, **attending** to the encoded representation of both x **and** z

- concatenate input x with the retrieved content z when generating from BART
- 'BART' was pre-trained using a denoising objective and a variety of different noising functions

Training



query encoder BERT$_q$

generator (BART)

Given a fine-tuning training corpus of input/output, the authors try to minimize the negative marginal log-likelihood of each target, using stochastic gradient descent with Adam

Approaches: Sequence vs Token

Different ways to produce a distribution over generated text

Sequence vs Token

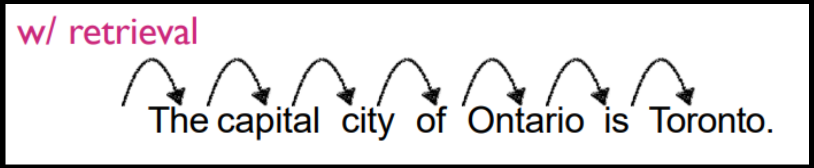
RAG-Sequence Model

The model uses

the same document(s)

to predict each target token

- more **coherence to one source**, easier **attribution**, often **cheaper**
- not able to combine multiple docs within a single answer.



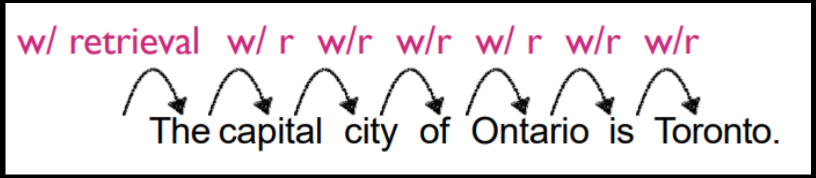
RAG-Token Model

The model uses

different document(s)

to predict each target token

- more **flexibility**
- but more **compute** and potentially **source-switching within a sentence**, which can complicate citation/attribution



Experiments

Experiment with RAG in a wide range of knowledge-intensive tasks

Use a single Wikipedia dump for non-parametric knowledge source

Experiments

Wikipedia snapshot (Dec 2018), cut into ~21 million chunks of ~100 words each.

Every chunk is embedded (turned into a vector) and stored

For each input, the retriever pulls the **top-k** passages

Tasks and Metrics

1. Open-domain QA = Did the predicted answer **exactly** match the gold text string?
2. Abstractive QA = **Answer** doesn’t exist in Wiki, compare **output** with **reference**
3. Jeopardy Question Generation = the model is given the **answer** and must write the **clue**
4. Fact Verification = Give **claim**, the model must retrieve **evidence** to classify whether the claim is true, false, or unverifiable

Results

When comparing results for RAG along with state-of-the-art models

"The state of the art" is a phrase that refers to the most advanced, sophisticated, or modern stage of development in a particular field, such as technology, science, or a specific skill at a given time

(1) Open-domain QA = what % of predictions exactly match the gold answer

Table 1: Open-Domain QA Test Scores. For TQA, left column uses the standard test set for Open-Domain QA, right column uses the TQA-Wiki test set. See Appendix D for further details.

	Model	NQ	TQA	WQ	CT
Closed Book	T5-11B [52]	34.5	- /50.1	37.4	-
	T5-11B+SSM[52]	36.6	- /60.5	44.7	-
Open Book	REALM [20]	40.4	- / -	40.7	46.8
	DPR [26]	41.5	**57.9**/ -	41.1	50.6
	RAG-Token	44.1	55.2/66.1	**45.5**	50.0
	RAG-Seq.	**44.5**	56.8/**68.0**	45.2	**52.2**

- RAG sets a new state of the art
- RAG enjoys strong results without expensive, specialized training
- RAG can generate correct answers even when the correct answer is not in any retrieved document

(2) Abstractive QA = **Answer** doesn't exist in Wiki, compare **output** with **reference**

Table 2: Generation and classification Test Scores. MS-MARCO SotA is [4], FEVER-3 is [68] and FEVER-2 is [57] *Uses gold context/evidence. Best model without gold access underlined.

Model	Jeopardy B-1	Jeopardy QB-1	MSMARCO R-L	MSMARCO B-1	FVR3 Label Acc.	FVR2 Label Acc.
SotA	-	-	**49.8***	**49.9***	**76.8**	**92.2***
BART	15.1	19.7	38.2	41.6	64.0	81.1
RAG-Tok.	**17.3**	**22.2**	40.1	41.5	72.5	<u>89.5</u>
RAG-Seq.	14.7	21.4	<u>40.8</u>	<u>44.2</u>		

- RAG approaches state-of-the-art model performance
- Beating BART
- Report **fewer hallucinations** and **more factual** outputs than BART

(3) Jeopardy Question Generation = the model is given the **answer** and must write the **clue**

Table 4: Human assessments for the Jeopardy Question Generation Task.

	Factuality	Specificity
BART better	7.1%	16.8%
RAG better	**42.7%**	**37.4%**
Both good	11.7%	11.8%
Both poor	17.7%	6.9%
No majority	20.8%	20.1%

- Humans prefer RAG for factuality
- Evaluators also find RAG generations to be more specific by a large margin

(4) Fact Verification = Give **claim**, the model must retrieve **evidence** to classify whether the claim is true, false, or unverifiable

Table 2: Generation and classification Test Scores. MS-MARCO SotA is [4], FEVER-3 is [68] and FEVER-2 is [57] *Uses gold context/evidence. Best model without gold access underlined.

Model	Jeopardy B-1	Jeopardy QB-1	MSMARCO R-L	MSMARCO B-1	FVR3 Label Acc.	FVR2 Label Acc.
SotA	-	-	**49.8***	**49.9***	**76.8**	**92.2***
BART	15.1	19.7	38.2	41.6	64.0	81.1
RAG-Tok.	**17.3**	**22.2**	40.1	41.5	72.5	<u>89.5</u>
RAG-Seq.	14.7	21.4	<u>40.8</u>	<u>44.2</u>		

Table 3: Examples from generation tasks. RAG models generate more specific and factually accurate responses. '?' indicates factually incorrect responses, * indicates partially correct responses.

Task	Input	Model	Generation
MS-MARCO	define middle ear	BART	?The middle ear is the part of the ear between the middle ear and the nose.
		RAG-T	The middle ear is the portion of the ear internal to the eardrum.
		RAG-S	The middle ear includes the tympanic cavity and the three ossicles.
	what currency needed in scotland	BART	The currency needed in Scotland is Pound sterling.
		RAG-T	Pound is the currency needed in Scotland.
		RAG-S	The currency needed in Scotland is the pound sterling.
Jeopardy Question Gener -ation	Washington	BART	?This state has the largest number of counties in the U.S.
		RAG-T	It's the only U.S. state named for a U.S. president
		RAG-S	It's the state where you'll find Mount Rainier National Park
	The Divine Comedy	BART	*This epic poem by Dante is divided into 3 parts: the Inferno, the Purgatorio & the Purgatorio
		RAG-T	Dante's "Inferno" is the first part of this epic poem
		RAG-S	This 14th century work is divided into 3 sections: "Inferno", "Purgatorio" & "Paradiso"

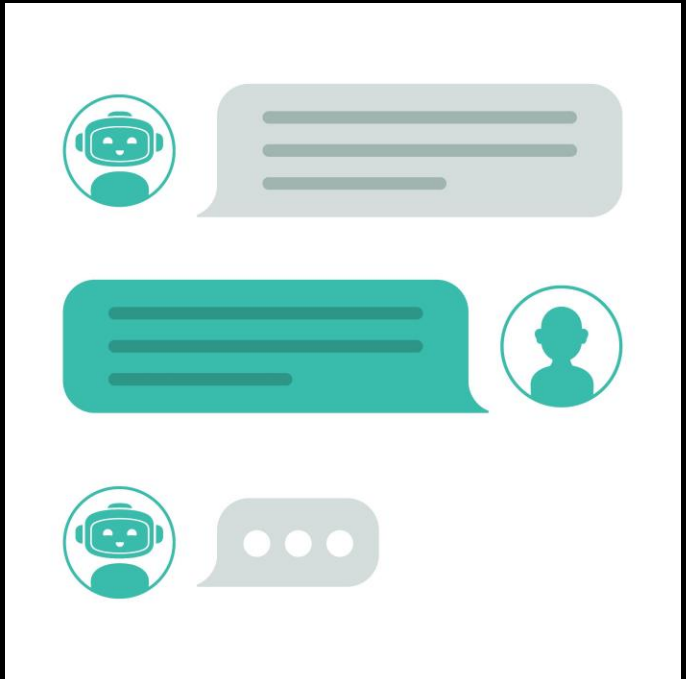
- Humans prefer RAG for factuality
- Evaluators also find RAG generations to be more specific by a large margin

Applications

What can you do with RAG?

Examples Applications

- Business knowledge assistant
- Customer support & help centers
- Legal/contract review
- Product Information retriever



Strength / Weakness of Paper

Strengths and Weaknesses of the paper

Strength / Weakness of Paper

Strengths

- Novelty
- Well-organized, content is easy to follow
- Extensive demonstration of Experiments and Results
- Provide instructions on how to reproduce experiments

Weaknesses

- Only use one set of pre-trained models for the component ‘BERT’ and ‘BART’
- Only use one knowledge base

Related Works

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-T., Rocktäschel, T., Riedel, S., & Kiela, D. (2020). Retrieval-Augmented Generation for knowledge-intensive NLP tasks. *Advances in Neural Information Processing Systems (NeurIPS 2020).* https://proceedings.neurips.cc/paper_files/paper/2020/file/6b493230205f780e1bc26945df7481e5-Paper.pdf

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-T., Rocktäschel, T., Riedel, S., & Kiela, D. (2020). *Appendices for "Retrieval-Augmented Generation for knowledge-intensive NLP tasks"* [Supplementary material]. NeurIPS 2020. https://proceedings.neurips.cc/paper_files/paper/2020/file/6b493230205f780e1bc26945df7481e5-Supplemental.pdf

Yang, J., Liu, Z., Li, C., Sun, G., & Xie, X. (2023). Longtriever: A pre-trained long text encoder for dense document retrieval. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP 2023)* (pp. 3655–3665). Association for Computational Linguistics. https://aclanthology.org/2023.emnlp-main.223.pdf

Wikipedia contributors. (n.d.). *MIPS architecture.* In *Wikipedia, The Free Encyclopedia.* Retrieved September 22, 2025, from https://en.wikipedia.org/wiki/MIPS_architecture

Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., & Zettlemoyer, L. (2019). *BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension.* arXiv. https://doi.org/10.48550/arXiv.1910.13461

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). *BERT: Pre-training of deep bidirectional transformers for language understanding.* arXiv. https://doi.org/10.48550/arXiv.1810.04805

[KodeKloud]. (2025, Aug 13). *RAG explained for beginners* [Video]. YouTube. https://www.youtube.com/watch?v=_HQ2H_0Ayy

Martin, L. J. (2025, October 2). *Retrieval-augmented generation* [PDF slides]. https://laramartin.net/interactive-fiction-class/slides/25-10-02_RAG.pdf