

An Overview of Language Models for Knowledge Retrieval

with particular attention to REALM

Rocco Tripodi
tripodi@di.uniroma1.it



SAPIENZA
UNIVERSITÀ DI ROMA



Consolidator Grant
MOUSSE No. 726487

**SAPIENZA
NLP**



Outline

Introduction of language models for knowledge retrieval

REALM – architecture

REALM – experiments

New datasets and models – KILT and RAG

Conclusion

Introduction

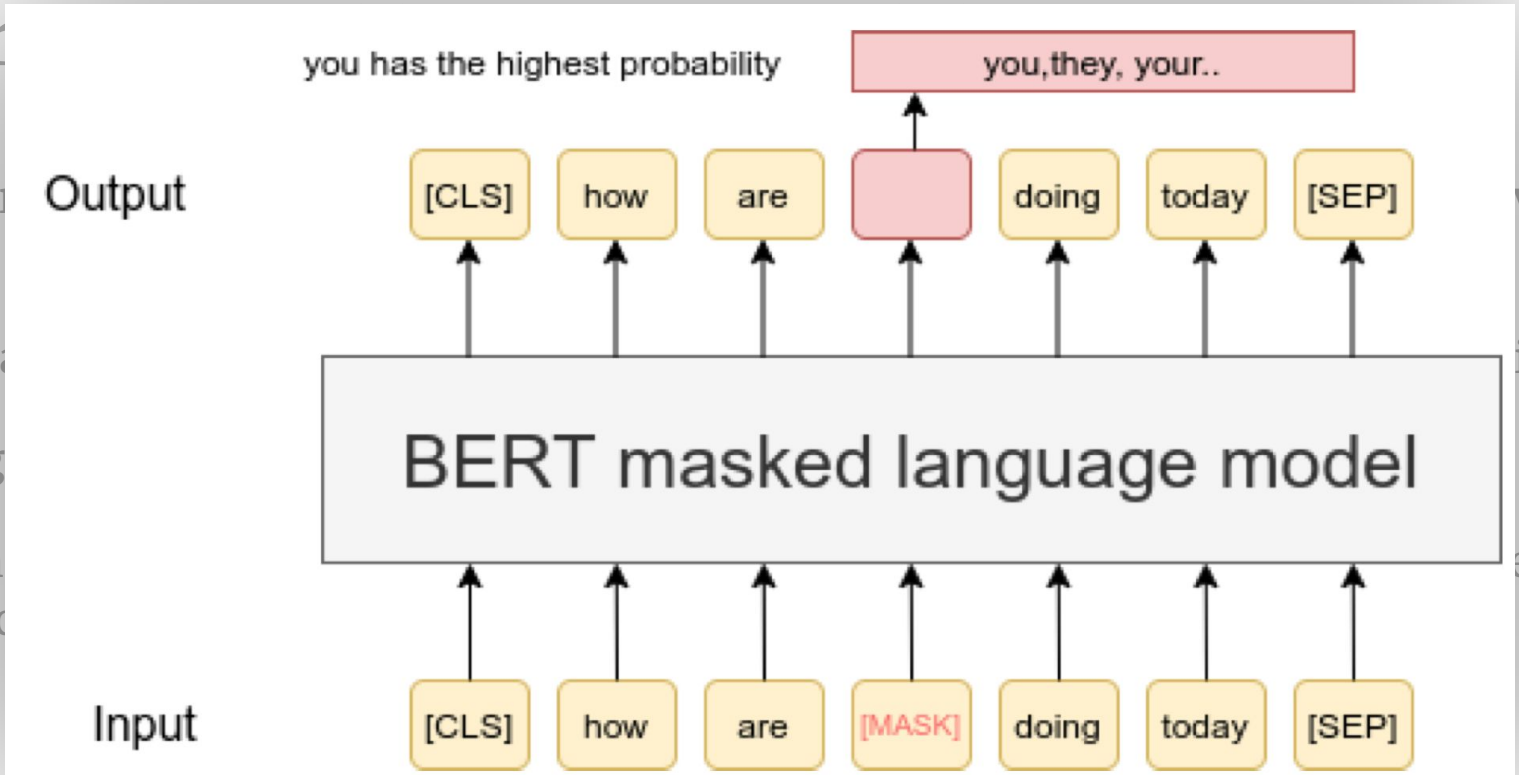


Pre-trained language models

Pre-trained language models such as GPT and BERT have significantly advanced the state of art in a wide range of NLP tasks.

Using self-supervised objective functions such as Masked Language Modelling they are able to produce contextualized representation of words that can be used in different downstream tasks.

These objectives consist mainly in masking one or more of words in a sentence and asking the model to predict those masked words given the other words in sentence.



Language models as knowledge retrievers

Besides containing linguistic information these models demonstrated to store also factual knowledge in their parameters (Petroni et al. 2019).

One approach for extracting such information consists in querying LMs using natural language prompts su as:

“Barack Obama was born in __”

“The __ is the currency of the United Kingdom”

the word assigned the highest probability in the blank will be returned as the answer

Language models as knowledge retrievers

Petroni et al. (2019) introduced the LAnguage Model Analysis (**LAMA**) probe.

It consists of a set of knowledge sources, each comprised of a set of facts, organized as a triples (subject, relation, object).

LMs know these facts if they can predict the correct object in manually created prompts such as:

“Dante was born in ____”

Despite not fine-tuning the performances of BERT on LAMA are quite high!

Language models as knowledge retrievers

Since the way in which LMs are queried can influence their predictions, Jiang et al. (2020) introduced automatic strategies to construct templates based on mining and paraphrasing techniques.

They demonstrated that diversifying the prompts is beneficial (+8 points).

However, their ability to access and manipulate knowledge with these techniques is still limited, and hence on knowledge-intensive tasks their performance lags behind task-specific architectures.

The knowledge is stored implicitly in the parameters of the model.

REALM



REALM – Retrieval-Augmented Language Model

In contrast to models that store knowledge in their parameters, this approach explicitly exposes the role of world knowledge by asking the model to decide what knowledge to retrieve and use during inference.

Before making each prediction, the language model uses the retriever to collect documents from a large corpus such as Wikipedia, and then attends over those documents to help inform its prediction.

REALM is a retrieve then predict generative model

The knowledge retriever is trained in an unsupervised manner, using masked language modeling as the learning signal and backpropagating through the retrieval step that considers millions of documents.

Guu, K., et al., Retrieval Augmented Language Model Pre-Training, ICML, 2020

REALM

In contrast to explicitly exposing knowledge to the model

Before making documents from documents to be retrieved

The knowledge language model step that consists

REALM is fine-tuned

Guu, K., et al., Retrieval

Textual knowledge corpus (\mathcal{Z})

retrieve

Neural Knowledge Retriever $\sim p_{\theta}(z|x)$

Retrieved document

The pyramidion on top allows for less material higher up the pyramid. (z)

Query and document

[CLS] The [MASK] at the top of the pyramid
[SEP] The pyramidion on top allows for less material higher up the pyramid. (x, z)

Knowledge-Augmented Encoder $\sim p_{\phi}(y|x, z)$

Answer

[MASK] = pyramidion (y)

End-to-end backpropagation

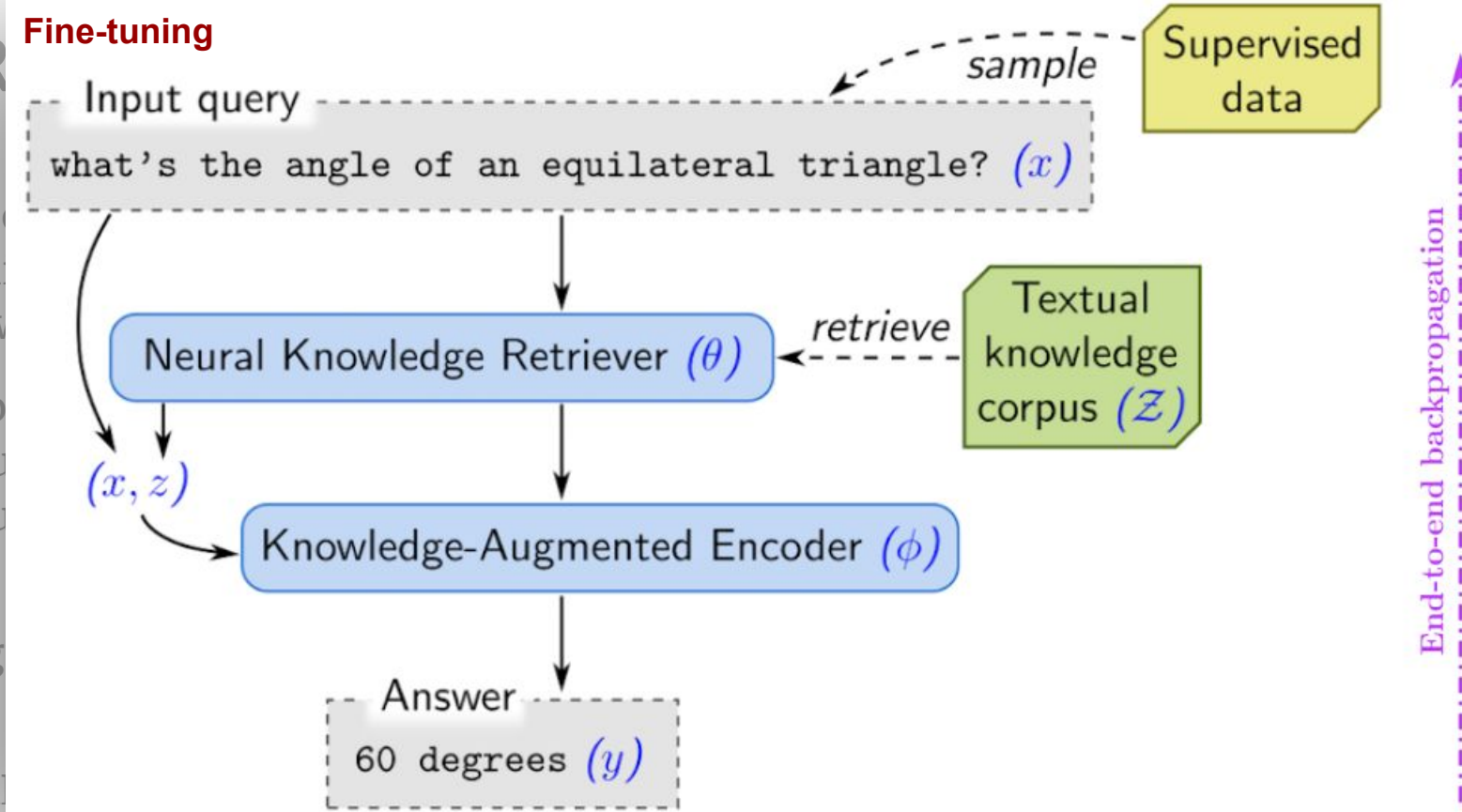
e Model

this approach to decide what

ver to retrieve nds over those

using masked ugh a retrieval

ion Answering



REALM's generative process

For both pre-training and fine-tuning, REALM takes some input \mathbf{x} and learns a distribution $p(\mathbf{y}|\mathbf{x})$ over possible outputs \mathbf{y} .

For pre-training, the task is MLM (salient span masking) \mathbf{x} is a sentence from a pre-training corpus \mathbf{X} with specific tokens masked out, and the model must predict the value of those missing tokens, \mathbf{y} .

For fine-tuning, the task is Open-QA \mathbf{x} is a question and \mathbf{y} is its answer.

Retrieve then predict

REALM decomposes $p(y|x)$ into two steps: **retrieve**, then **predict**.

Given an input x , it first retrieves possibly helpful documents z from a knowledge corpus Z . This is modeled as a sample from the distribution $p(z|x)$.

Then, it conditions on both the retrieved z and the original input x to generate the output y — modeled as $p(y|z, x)$.

To obtain the overall likelihood of generating y , it treats z as a latent variable and marginalize over all possible documents z , yielding

$$p(y|x) = \sum_{z \in Z} p(y|z, x) p(z|x) .$$

Knowledge retriever

The retriever is defined using a dense inner product model

$$p(z|x) = \frac{\exp(f(x, z))}{\sum_{z'} \exp(f(x, z'))}$$

The relevance score $f(x, z)$ is defined as the inner product of the vector embeddings

Maximum Inner Product Search (MIPS)

Since the dataset for collecting candidate sentences, Z , is very large, computing

$$p(y|x) = \sum_{z \in Z} p(y|z, x) p(z|x) .$$

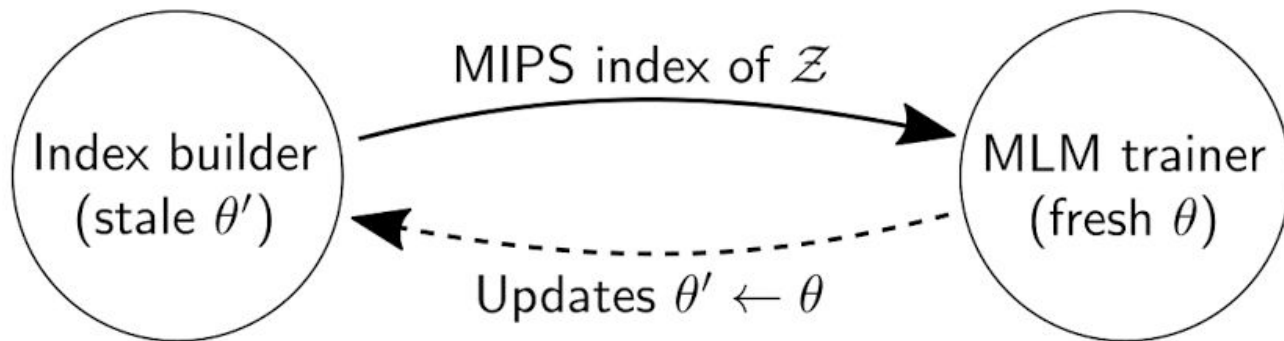
is very expensive.

Z is approximated to the k documents with the highest probability under $p(z|x)$.

MIPS precomputes the embedding of all the documents in Z indexing them

This indexing is refreshed at each n training step to make the embedding consistent with the update of the model parameters.

Maximum Inner Product Search (MIPS)



Knowledge-augmented encoder

x and **z** are joined into a single sequence that is feeded into a Transformer.

This allows to perform rich cross-attention between **x** and **z** before predicting **y**.

During pre-training it is used the same MLM objective used in BERT

For Open-QA fine-tuning the answer **y** can be found in a set of spans of text **S(z,x)**.

$$p(y|z, x) \propto \sum_{s \in S(z, y)} \exp\left(MLP\left[h_{start(s)}; h_{end(s)}\right]\right)$$

What does the retriever learn?

Since the knowledge retrieval of REALM is latent, it is not obvious how the training objective encourages meaningful retrievals

$$\begin{aligned}\nabla \log p(y | x) &= \sum_{z \in \mathcal{Z}} r(z) \nabla f(x, z) \\ r(z) &= \left[\frac{p(y | z, x)}{p(y | x)} - 1 \right] p(z | x).\end{aligned}$$

For each document \mathbf{z} , the gradient encourages the retriever to change the score $f(\mathbf{x}, \mathbf{z})$ by $r(\mathbf{z})$ — increasing if $r(\mathbf{z})$ is positive, and decreasing if negative.

Document \mathbf{z} receives a positive update whenever it performs better than expected.

Experimental Results

Datasets

REALM was evaluated on 3 datasets:

NaturalQuestions (Kwiatkowski et al., 2019) consists of naturally occurring Google queries and their answers.

WebQuestions (Berant et al., 2013) was collected from the Google Suggest API, using one seed question and expanding the set to related questions.

CuratedTrec dataset is a collection of question-answer pairs drawn from real user queries issued on sites such as MSNSearch and AskJeeves.

Guu, K., et al., Retrieval Augmented Language Model Pre-Training, ICML, 2020

Kwiatkowski, T., et al. Natural questions: a benchmark for question answering research, *TACL*, 2019

Berant, J., et al. Semantic parsing on freebase from question-answer pairs, *EMNLP*, 2013

Approaches compared

Retrieval-based Open-QA

Most existing Open-QA systems answer the input question by first retrieving potentially relevant documents from a knowledge corpus, and then using a **reading comprehension** system to extract an answer from the documents

Many approaches use non-learned heuristic retrieval such as sparse bag-of-words matching or entity linking on the question to select a small set of relevant documents. These documents are typically then re-ranked using a learned model, but coverage may be limited by the initial heuristic retrieval step.

Approaches compared

Generation-based Open-QA

An emerging alternative approach to Open-QA is to model it as a **sequence prediction task**: simply encode the question, and then decode the answer token-by-token based on the encoding.

GPT-2 hinted at the possibility of directly generating answers without using any given context via sequence-to-sequence.

T5 Raffel et al. (2019) and Roberts et al. (2020) showed that directly generating answers without explicit extraction from the given context is viable approach.

Guu, K., et al., Retrieval Augmented Language Model Pre-Training, ICML, 2020

Raffel, C., et al., Exploring the limits of transfer learning with a unified text-to-text transformer, *JMLR*, 2020

Roberts, A., et al., How Much Knowledge Can You Pack Into the Parameters of a Language Model?, EMNLP, 2020

Results

Table 1. Test results on Open-QA benchmarks. The number of train/test examples are shown in parentheses below each benchmark. Predictions are evaluated with exact match against any reference answer. Sparse retrieval denotes methods that use sparse features such as TF-IDF and BM25. Our model, REALM, outperforms all existing systems.

Name	Architectures	Pre-training	NQ (79k/4k)	WQ (3k/2k)	CT (1k /1k)	# params
BERT-Baseline (Lee et al., 2019)	Sparse Retr.+Transformer	BERT	26.5	17.7	21.3	110m
T5 (base) (Roberts et al., 2020)	Transformer Seq2Seq	T5 (Multitask)	27.0	29.1	-	223m
T5 (large) (Roberts et al., 2020)	Transformer Seq2Seq	T5 (Multitask)	29.8	32.2	-	738m
T5 (11b) (Roberts et al., 2020)	Transformer Seq2Seq	T5 (Multitask)	34.5	37.4	-	11318m
DrQA (Chen et al., 2017)	Sparse Retr.+DocReader	N/A	-	20.7	25.7	34m
HardEM (Min et al., 2019a)	Sparse Retr.+Transformer	BERT	28.1	-	-	110m
GraphRetriever (Min et al., 2019b)	GraphRetriever+Transformer	BERT	31.8	31.6	-	110m
PathRetriever (Asai et al., 2019)	PathRetriever+Transformer	MLM	32.6	-	-	110m
ORQA (Lee et al., 2019)	Dense Retr.+Transformer	ICT+BERT	33.3	36.4	30.1	330m
ORQA (more fine-tune epochs)	Dense Retr.+Transformer	ICT+BERT	34.8	35.4	28.7	330m
Ours (\mathcal{X} = Wikipedia, \mathcal{Z} = Wikipedia)	Dense Retr.+Transformer	REALM	39.2	40.2	46.8	330m
Ours (\mathcal{X} = CC-News, \mathcal{Z} = Wikipedia)	Dense Retr.+Transformer	REALM	40.4	40.7	42.9	330m

New datasets and models

KILT - Knowledge Intensive Language Tasks

KILT benchmark unifies 11 popular datasets for 5 knowledge intensive tasks

1. Fact-checking
2. Entity linking
3. Slot filling
4. Open domain QA
5. Dialog generation

All these datasets have been grounded in a single pre-processed Wikipedia dump.

RAG - Retrieval Augmented Generation

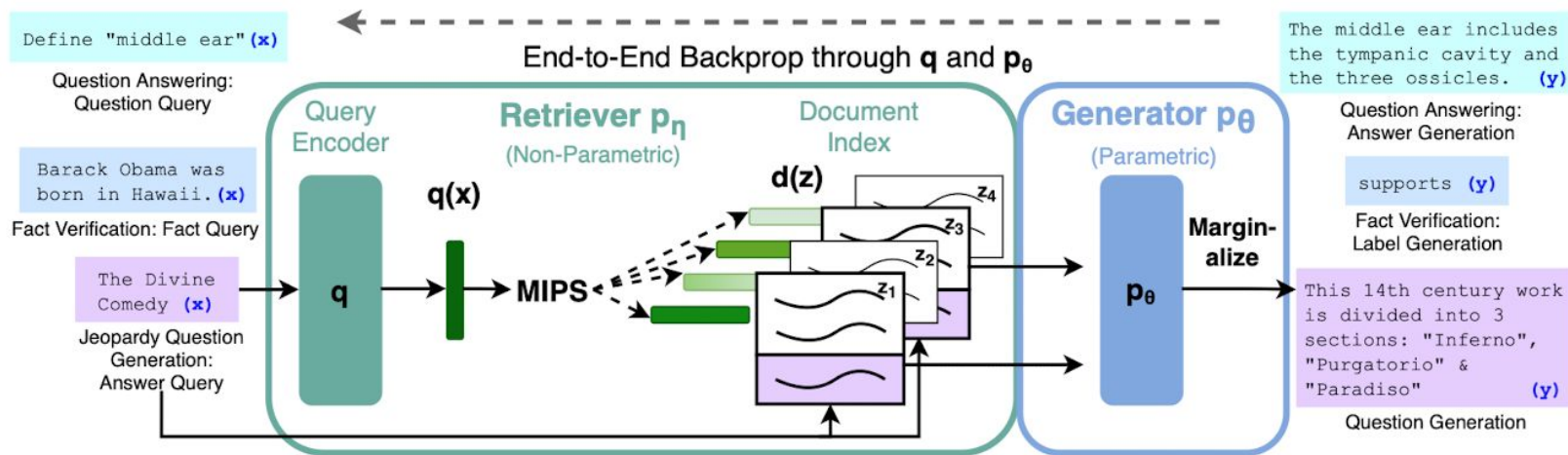
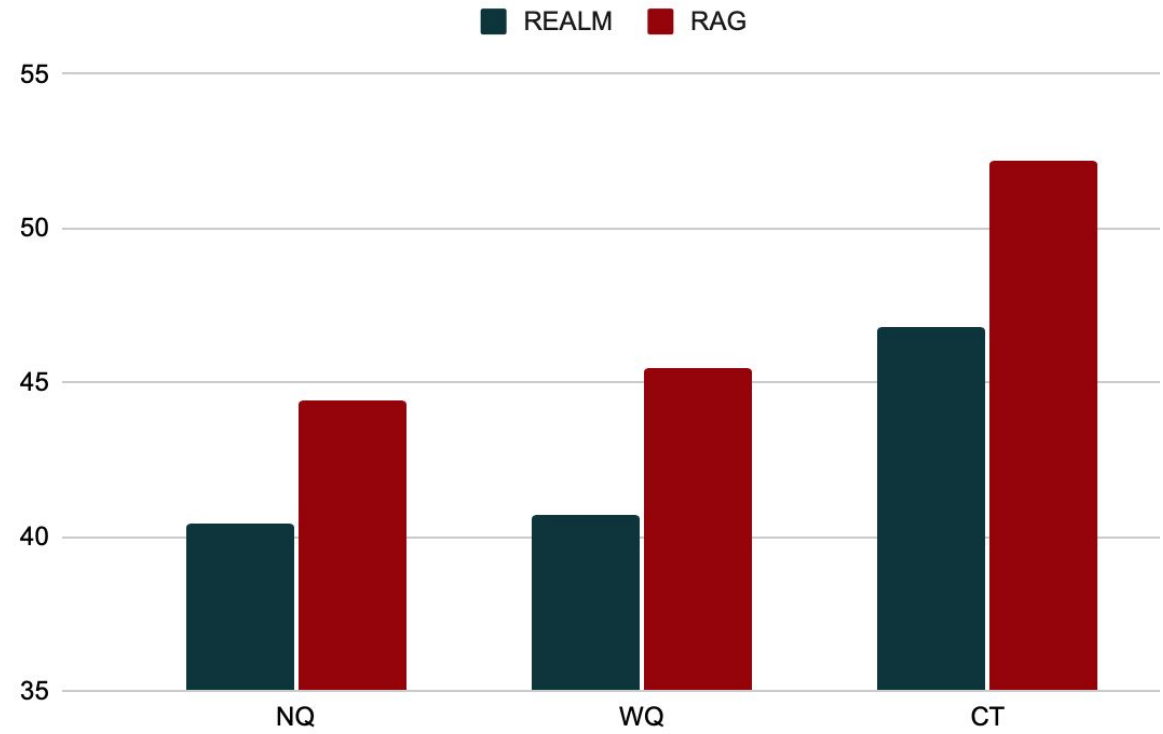











Figure 1: Overview of our approach. We combine a pre-trained retriever (*Query Encoder* + *Document Index*) with a pre-trained seq2seq model (*Generator*) and fine-tune end-to-end. For query x , we use Maximum Inner Product Search (MIPS) to find the top-K documents z_i . For final prediction y , we treat z as a latent variable and marginalize over seq2seq predictions given different documents.

REALM vs RAG



Ready for real-world applications?

 All  News  Images  Videos  Shopping  More Settings Tools



About 489,000,000 results (0.76 seconds)

33,000 years ago

The "Clovis first theory" refers to the 1950s hypothesis that the Clovis culture represents the earliest **human** presence in **the Americas**, beginning about 13,000 years ago; evidence of pre-Clovis cultures has accumulated since 2000, pushing back the possible date of the first peopling of **the Americas** to 33,000 years ago.

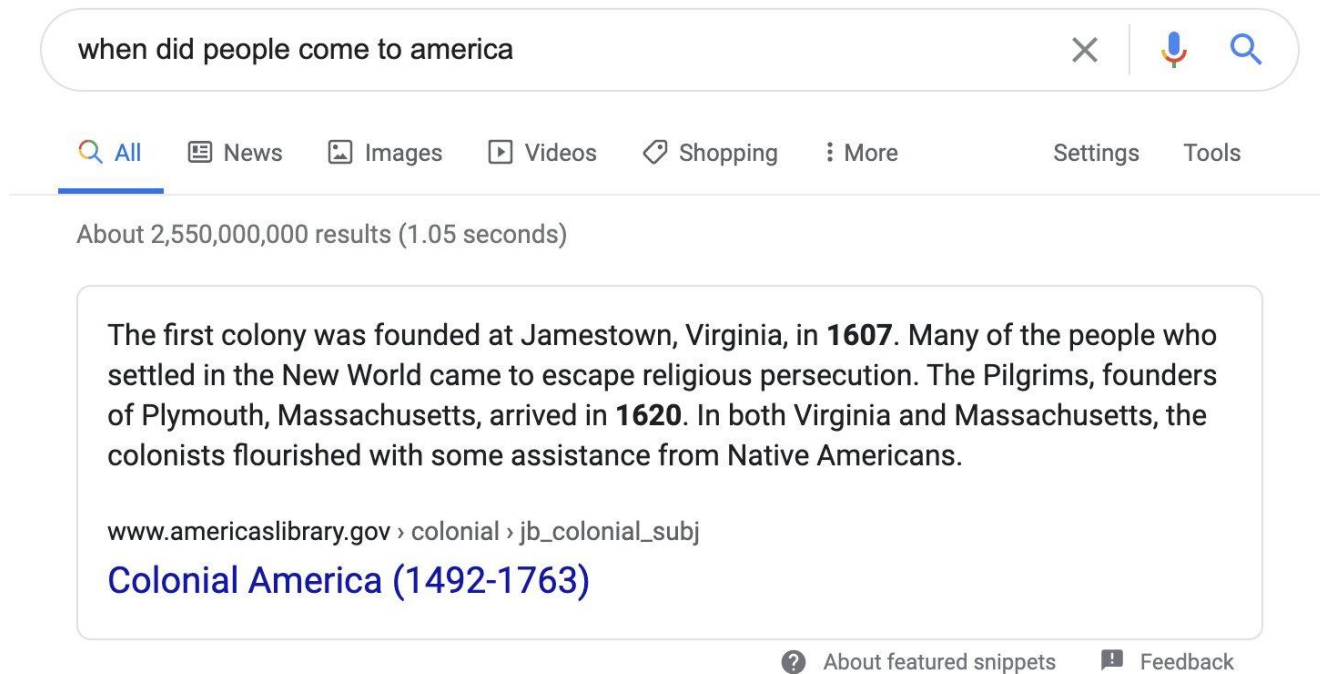
en.wikipedia.org › wiki › Settlement_of_the_Americas

[Settlement of the Americas - Wikipedia](#)

 About featured snippets  Feedback



Ready for real-world applications?



Credit to:

Hank Green  @hankgreen · 25 gen
Hey @Google, this is...real weird.

Thank you for your attention!



SAPIENZA
UNIVERSITÀ DI ROMA



Consolidator Grant
MOUSSE No. 726487