



Stanford Course Fall'23: Creating Production Ready LLMs

arxiv-search

Sharat Sripada

Introduction - *arxiv search*

-
- Leverage pre-trained Large Language Models (LLM) for Information Retrieval (IR) tasks
 - Using techniques like *fine-tuning* - Low Ranking Adaptation/LoRA, Efficient fine tuning of quantized LLMs - qLoRA, Parameter efficient Fine Tuning/PEFT etc., *prompt engineering* and *Retrieval Augmented Generation (RAG)* fit custom data and build a personalized Question-Answer (QA)/Hybrid-search engine

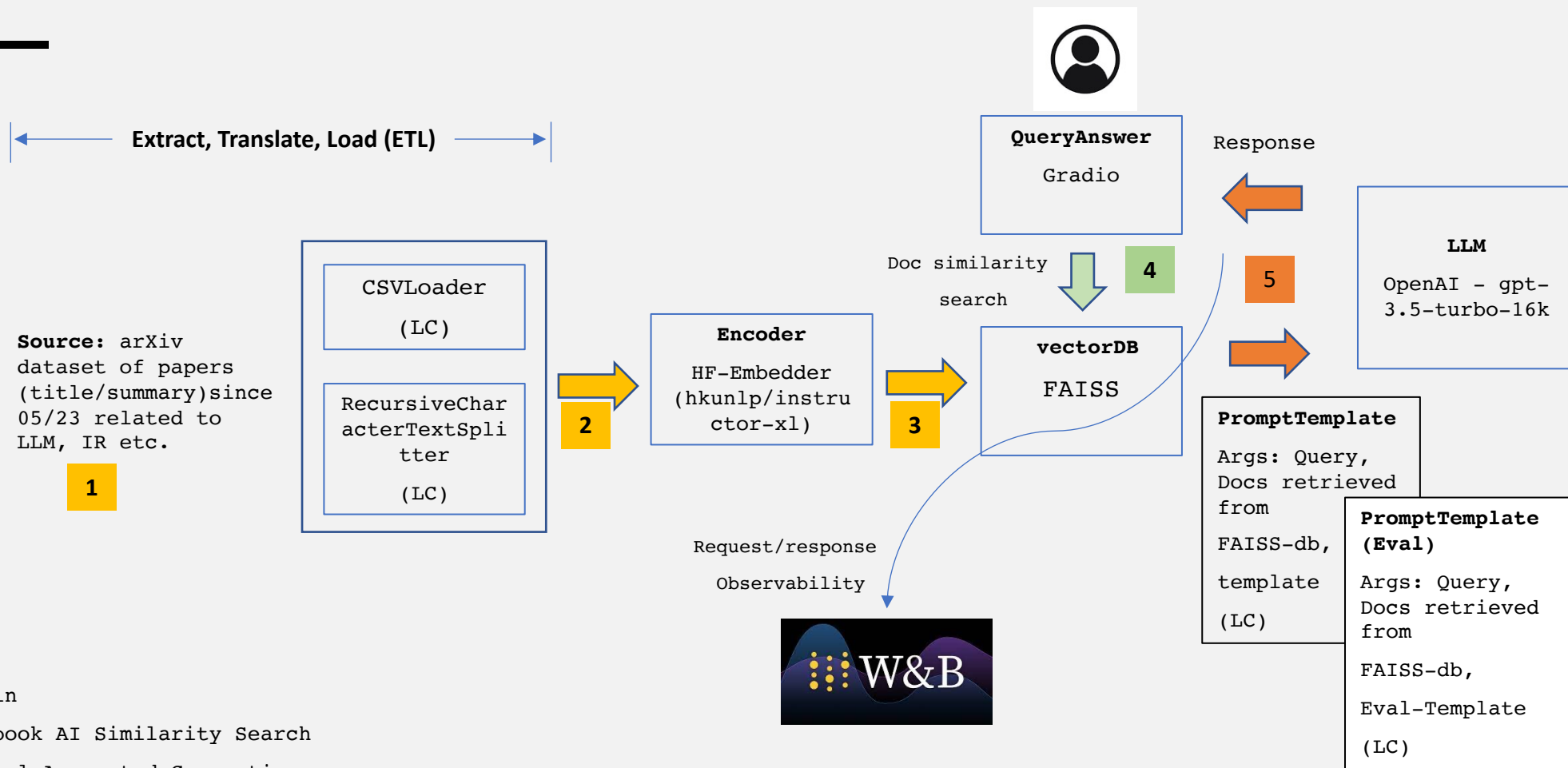
Problem Statement

The space of LLMs/IR is evolving at a rapid pace and research papers is one way to stay up-to-date, learn about problems being solved and innovate. Using papers from arxiv.org for past six-months and a pre-trained LLM (OpenAI - ChatGPT-3.5)* the objective is to build a QA model and seek answers to questions like:

- what is recent research in RAG
 - Explain PRAG
 - What is the title of the paper for PRAG
 - Explain KGR
 - What is the title of the paper for KGR
- are there any references to local language models
- what is local different from PEFT? (ask generic questions)

**trained on data till Jan'23*

Information Retrieval using RAG



LC - LangChain

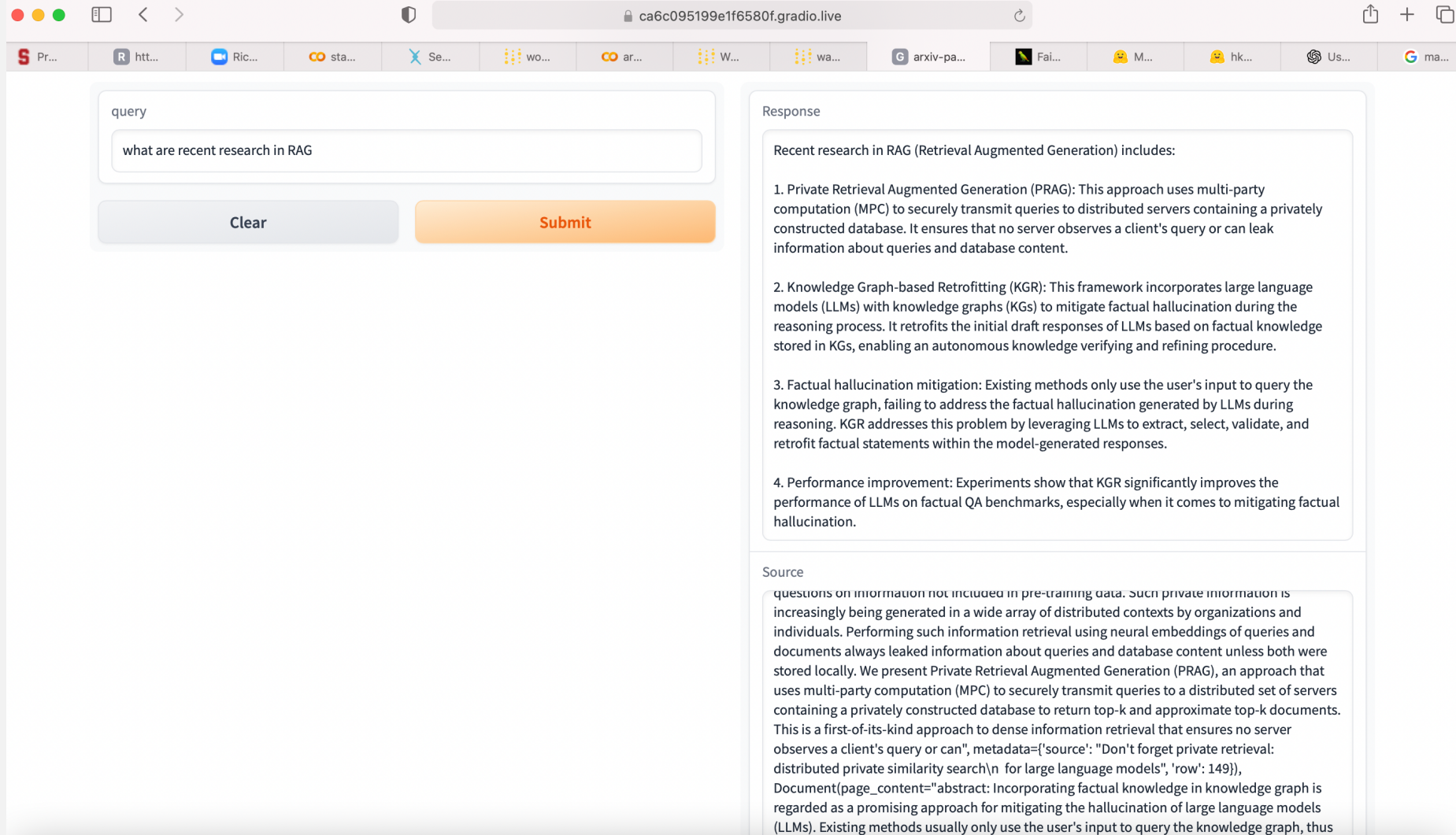
FAISS - Facebook AI Similarity Search

RAG - Retrieval Augmented Generation

W&B - Weights and Biases

Demo

arxiv-search - “What’s research community thinking about RAG”



ca6c095199e1f6580f.gradio.live

query

what are recent research in RAG

Clear Submit

Response

Recent research in RAG (Retrieval Augmented Generation) includes:

1. Private Retrieval Augmented Generation (PRAG): This approach uses multi-party computation (MPC) to securely transmit queries to distributed servers containing a privately constructed database. It ensures that no server observes a client's query or can leak information about queries and database content.
2. Knowledge Graph-based Retrofitting (KGR): This framework incorporates large language models (LLMs) with knowledge graphs (KGs) to mitigate factual hallucination during the reasoning process. It retrofits the initial draft responses of LLMs based on factual knowledge stored in KGs, enabling an autonomous knowledge verifying and refining procedure.
3. Factual hallucination mitigation: Existing methods only use the user's input to query the knowledge graph, failing to address the factual hallucination generated by LLMs during reasoning. KGR addresses this problem by leveraging LLMs to extract, select, validate, and retrofit factual statements within the model-generated responses.
4. Performance improvement: Experiments show that KGR significantly improves the performance of LLMs on factual QA benchmarks, especially when it comes to mitigating factual hallucination.

Source

questions on information not included in pre-training data. Such private information is increasingly being generated in a wide array of distributed contexts by organizations and individuals. Performing such information retrieval using neural embeddings of queries and documents always leaked information about queries and database content unless both were stored locally. We present Private Retrieval Augmented Generation (PRAG), an approach that uses multi-party computation (MPC) to securely transmit queries to a distributed set of servers containing a privately constructed database to return top-k and approximate top-k documents. This is a first-of-its-kind approach to dense information retrieval that ensures no server observes a client's query or can", metadata={'source': "Don't forget private retrieval: distributed private similarity search\n for large language models", 'row': 149}}, Document(page_content="abstract: Incorporating factual knowledge in knowledge graph is regarded as a promising approach for mitigating the hallucination of large language models (LLMs). Existing methods usually only use the user's input to query the knowledge graph, thus

arxiv-search - “PRAG, ok. Ask more about PRAG before I dive in”

ca6c095199e1f6580f.gradio.live

Pr...

htt...

Ric...

sta...

Se...

wo...

af...

W...

wa...

arxiv-pa...

Fai...

M...

hk...

Us...

ma...

query

explain PRAG

Clear

Submit

Response

PRAG stands for Private Retrieval Augmented Generation.

Key points from the articles:

- PRAG is an approach that uses multi-party computation (MPC) to securely transmit queries to a distributed set of servers containing a privately constructed database.
- The goal of PRAG is to enable large language models (LLMs) to answer questions on information not included in their pre-training data.
- PRAG ensures that no server observes a client's query or can access the training data.
- PRAG is a first-of-its-kind approach to dense information retrieval that enhances the capabilities of LLMs.

Based on the provided information, PRAG is a method that combines secure transmission of queries and a privately constructed database to enable large language models to retrieve information not included in their pre-training data. It ensures privacy and enhances the information retrieval capabilities of these models.

Source

(LLMs) allow them to answer a range of queries based on existing learned knowledge, information retrieval to augment generation is an important tool to allow LLMs to answer questions on information not included in pre-training data. Such private information is increasingly being generated in a wide array of distributed contexts by organizations and individuals. Performing such information retrieval using neural embeddings of queries and documents always leaked information about queries and database content unless both were stored locally. We present Private Retrieval Augmented Generation (PRAG), an approach that uses multi-party computation (MPC) to securely transmit queries to a distributed set of servers containing a privately constructed database to return top-k and approximate top-k documents. This is a first-of-its-kind approach to dense information retrieval that ensures no server observes a client's query or can", metadata={'source': "Don't forget private retrieval: distributed private similarity search\n for large language models", 'row': 149}}, Document(page_content='training in-batch negative samples to 80,000, and for each query, we extracted five hard negatives. Concurrently, we incorporated a progressive learning mechanism, enabling the model to dynamically modulate its attention to the samples throughout the entire training process. Additionally, PEG is trained on more than 100 million

arxiv-search - “I like PRAG..give me the reference title/paper”

ca6c095199e1f6580f.gradio.live

S Pr... R htt... Ric... sta... Se... wo... ar... W... wa... arxiv-pa... Fai... M... hk... Us... ma...

query

What is the title of the paper for PRAG

ClearSubmit

Response

The title of the paper for PRAG is "Private Retrieval Augmented Generation (PRAG): An Approach for Secure Information Retrieval in Language Models using Multi-Party Computation (MPC)".

Source

questions on information not included in pre-training data. Such private information is increasingly being generated in a wide array of distributed contexts by organizations and individuals. Performing such information retrieval using neural embeddings of queries and documents always leaked information about queries and database content unless both were stored locally. We present Private Retrieval Augmented Generation (PRAG), an approach that uses multi-party computation (MPC) to securely transmit queries to a distributed set of servers containing a privately constructed database to return top-k and approximate top-k documents. This is a first-of-its-kind approach to dense information retrieval that ensures no server observes a client's query or can", metadata={'source': "Don't forget private retrieval: distributed private similarity search\n for large language models", 'row': 149}}, Document(page_content='title: Towards Better Parameter-Efficient Fine-Tuning for Large Language\n Models: A Position Paper\nabstract: This paper delves into the pressing need in Parameter-Efficient Fine-Tuning (PEFT) for Large Language Models (LLMs). While LLMs possess remarkable capabilities, their extensive parameter requirements and associated computational demands hinder their practicality and scalability for real-world applications. Our position paper highlights current states and the necessity of further studying into the topic, and recognizes significant challenges and open issues that must be addressed to fully harness the powerful abilities of LLMs. These challenges encompass novel efficient PEFT architectures, PEFT for different learning settings, PEFT combined with model compression techniques, and the exploration of PEFT for multi-modal LLMs. By presenting this position paper, we aim to stimulate further research and foster discussions surrounding more efficient and accessible PEFT for LLMs.', metadata={'source': 'Towards Better Parameter-Efficient Fine-Tuning for Large Language\n Models: A Position Paper', 'row': 164}}]

“Monitor infrastructure/GPU, query-response time..”

Overview

Workspace

Runs

Jobs

Automat.

Sweeps

Reports

Artifacts

Runs (1)

Search runs

Name (1 visualized)

wandering-durian-1

sharatsv-stanford > Projects > arxiv-paper-search

Search panels with regex

Create report

Tables 1

Add panel

All Traces

	Success	Timestamp	Input	Output	Chain	Error	Model ID
8	True	2023-11-29 13:14:14	0.question: The title of the paper for PRAG is "Private Retrieval"	0.text: No, the response is not similar to the source given. The	LLMChain(Ch	—	df168d
7	True	2023-11-29 13:14:10	0.question: What is the title of the paper for PRAG	0.text: The title of the paper for PRAG is "Private Retrieval"	LLMChain(Ch	—	1402bb
6	True	2023-11-29 13:11:38	0.question: PRAG stands for Private Retrieval Augmented Generation	0.text: Yes, the response is similar to the source given. Both mention	LLMChain(Ch	—	a4146e
5	True	2023-11-29 13:11:20	0.question: explain PRAG 0.depth: abstract: While the flexible	0.text: PRAG stands for Private Retrieval Augmented Generation	LLMChain(Ch	—	cb8136

Trace Timeline (#1)

Model Architecture (ID: 9eb38398c61371ce)

1: LLMChain 23935ms

2: ChatOpenAI 23933ms

LLMChain 23935ms

Result Set 1

Inputs

Future work

- Host LLM (example: LLaMA-cpp) locally on a small form-factor like Raspberry-Pi/Rock-Pi
- Experiment with Fine-tuning using LoRA/qLoRA versus RAG
- Improve query-response time (to < 1sec)
- Explore fetching summary of PDF/paper to QA model like:
 - what are recent research in RAG? (works)
 - what is the title of the paper? (works)
 - *fetch the corresponding paper and summarize it (TBD)*
- Feedback from Cohort