arXivBot

Amr Sherif, amrshrif@stanford.edu

Department of Computer Science, Stanford University

CS224N Natural Language Processing with Deep Learning

Introduction & Motivation

- Given the accelerating pace of research and the monthly addition of 20,000 new papers on arXiv (9,000 in Computer Science alone), keeping up with the latest research has become increasingly challenging.
- There is a critical need for automated tools that can efficiently manage, retrieve, and summarize relevant information.
- Previous retrieval systems have faced significant challenges in handling multi-hop queries, which require reasoning across multiple documents, and have struggled to provide high-quality, contextually accurate results.

Project Goal & Proposed Solution

- The goal is to develop an advanced arXiv assistant that can, efficiently retrieve and select relevant research papers based on criteria specified by the user, such as submission date, domain, category, and topic.
- Answer specific user questions regarding the contents and findings of these papers.
- Provide summaries and highlight key points from the paper topics, aiding in quick comprehension of the latest research advances in a certain domain.

- The task involves question-answering retrieval and textual data extraction from a Knowledge Base (K), To build a model (M) that can map a user query (Q) to an answer (A), where A consists of nodes in K that satisfy the query Q.
- The input to the model is a query, and the output is a set of predicted answers, which should be accurate and relevant.

$$f : Q \times K \rightarrow A$$

Dataset

- A large-scale synthetic domain-specific dataset for instruction tuning, comprising 168,000 examples, was created through an automated process involving the parsing, splitting, and semantic chunking of arXiv full-text PDFs, followed by the generation of questions and answers pairs employing a high performance LLM.

Technical Approach

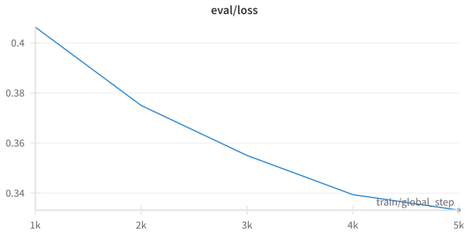
- QLoRA Quantization: The base model is a block-wise k-bit quantized Mistral-7B-Instruct-v0.2, which Reduces memory usage while preserving full 16-bit fine-tuning task performance.
- Parameter Efficient Fine-Tuning (PEFT) with LoRA: Minimizes the number of trainable parameters and GPU memory requirements, making fine-tuning more efficient.

LoRA augments a linear projection through an additional factorized projection.

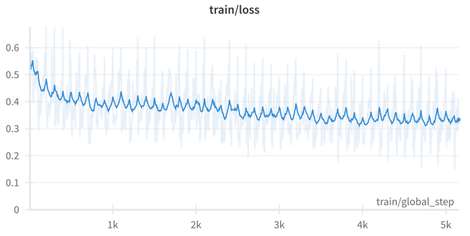
Given a projection $XW = Y$ with $X \in \mathbb{R}^{b \times h}, W \in \mathbb{R}^{h \times o}$:

$$Y = XW + sXL_1L_2$$

where $L_1 \in \mathbb{R}^{h \times r}$ and $L_2 \in \mathbb{R}^{r \times o}$, and s is a scalar.



Eval Loss shows a consistent decrease across all eval batches



Train loss was decreasing as expected in initial training steps then, started to fluctuate. Most probably due to dropout which is deactivated during Eval. Training was done on only 1 epoch.

Results

Training Loss	Epoch	Step	Validation Loss	Input Tokens Seen
0.6005	0.1938	1000	0.4064	1827684
0.5877	0.3877	2000	0.3750	3600506
0.4922	0.5815	3000	0.3551	5407592
0.498	0.7753	4000	0.3394	7199648
0.5224	0.9692	5000	0.3332	8960242

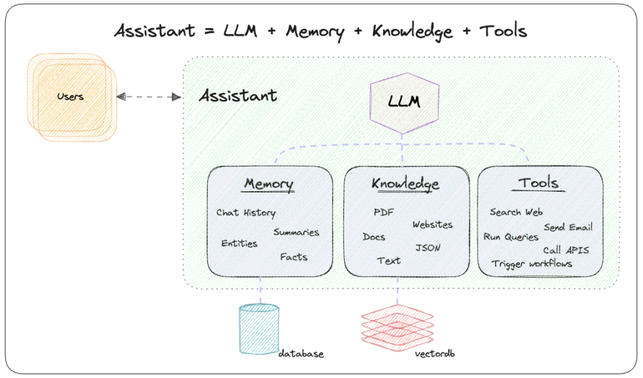
Table 2: Fine-tuning Results

Model	Blue	Rouge	Rouge L	Pass@1	Pass@10
. Vanilla Mistral	31.68	60.31	46.33	0.33	0.41
LoRA finetuned	36.77	63.83	48.17	0.31	0.38

Table 3: Fine-tuning Evaluation Results

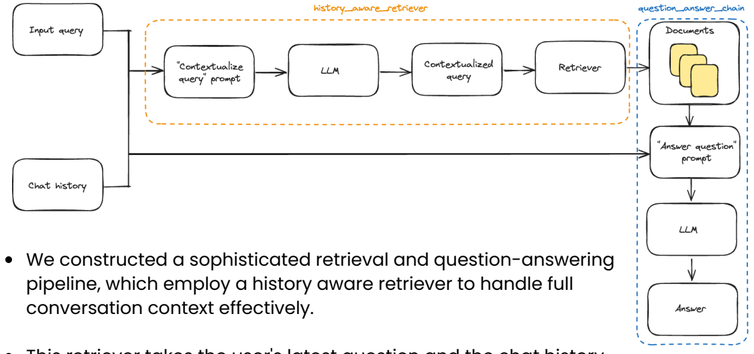
Modular Retrieval Augmented Generation

- We employed a Modular Retrieval Augmented Generation (RAG) Process: Combining retrieval and generation tasks to provide comprehensive and contextually relevant answers.



- In-Context Learning: Allows the model to adapt to new queries in subsequent conversation turns based on the context provided in the conversation history.

- Function Calling: Enhances the chatbot's ability to execute specific tasks based on user inputs, in our case the capability to call arXiv search API to fetch relevant papers in case no relevant information available in the knowledge base.



- We constructed a sophisticated retrieval and question-answering pipeline, which employ a history aware retriever to handle full conversation context effectively.

- This retriever takes the user's latest question and the chat history, then reformulating the query to ensure it incorporates the necessary context.

References

https://www.langchain.com

https://www.phidata.com