Open-Source LLMs for Technical Q&A: Lessons from StackExchange

Zeerak Babar, Nafiz Imtiaz Khan, Muhammad Hassnain, Vladimir Filkov
Department of Computer Science
University of California, Davis
Davis, CA, US



SEET2025

decallab

UCDAVIS COMPUTER SCIENCE

Motivation

- Developers are using AI tools more than ever to get quick, helpful answers.
- Big-name AI tools (like ChatGPT, Gemini, Claude-3) work well but are expensive and not always easy to access.
- Free, open-source AI models can run locally and are getting close to matching paid tools in quality.
- We still don’t know how well these open-source models work for answering real developer questions.
- Adding extra context with Retrieval-Augmented Generation (RAG) could make AI answers more accurate and useful.
- Combining RAG with fine-tuning might unlock even better, more developer-friendly results.

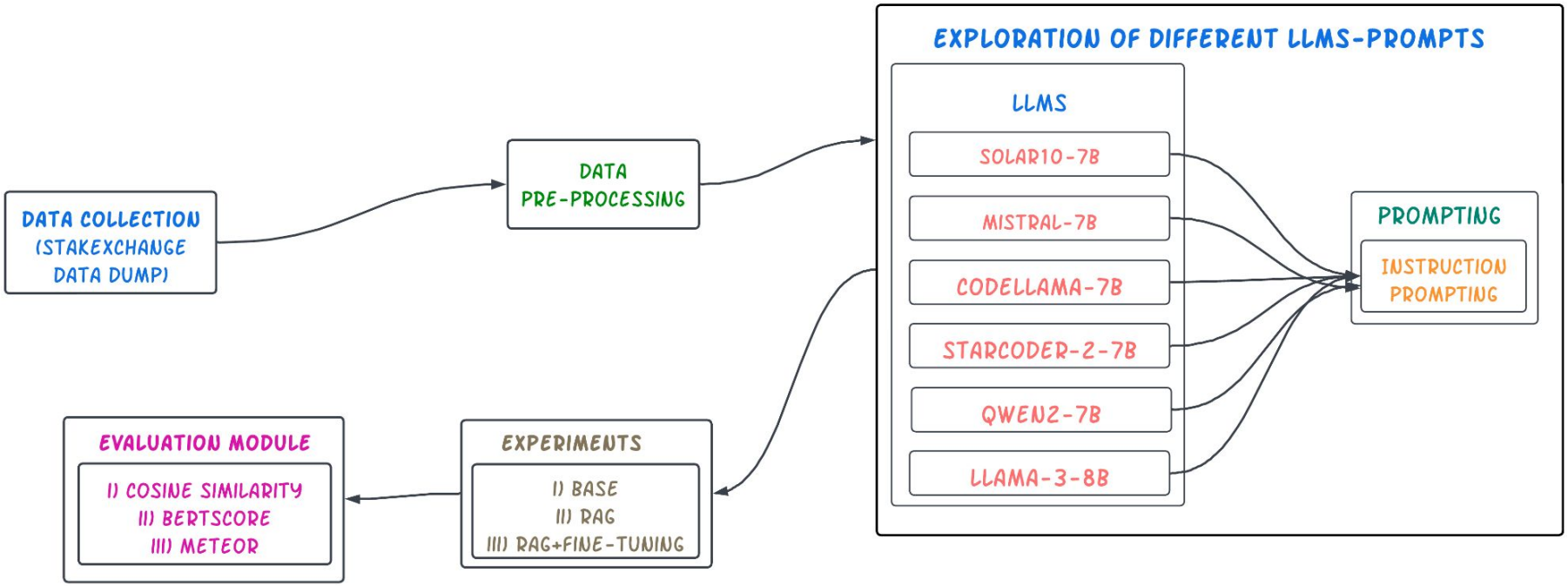


Research Questions

- **RQ1:** Can local LLMs effectively answer technical questions in a base setting?
- **RQ2:** To what extent does the RAG framework enhance LLM response quality by incorporating additional contextual information?
- **RQ3:** Can fine-tuning within the RAG framework further improve model performance?



Methodology



Methodology

Data Collection

- Used the publicly available **StackExchange Data Dump** instead of direct scraping.
- Dataset spans **May 2014 – Dec 2023** and contains 73,560 posts, 81,283 comments, from 54,037 authors.
- Randomly selected **41,000 posts,** and split into:
  - 1,000 for testing.
  - 20,000 for RAG database.
  - 20,000 for fine-tuning.

Data Synthesis

- Selected posts with an **Accepted Answer** for reliable comparison.
- Removed HTML tags to keep clean text.

Methodology

Model Selection

- Chose **six open-source LLMs** (7B–10.7B parameters) for code + natural language tasks:
  - Mistral-7B
  - Solar10.7B
  - CodeLlama-7B
  - Qwen2-7B
  - StarCoder2-7B
  - LLaMA-3 8B

Prompt Design

- Used a **vanilla prompt** asking the model to answer technical questions using provided context when available.
- Same template used for all experiments to ensure consistency

Methodology

Prompt Template

```
You are a helpful assistant with expertise in coding and technical topics.

Use the following context to answer the question as accurately as possible,
focusing on technical details and code examples if relevant.

Context: <<CONTEXT>>

Question: <<QUESTION>>

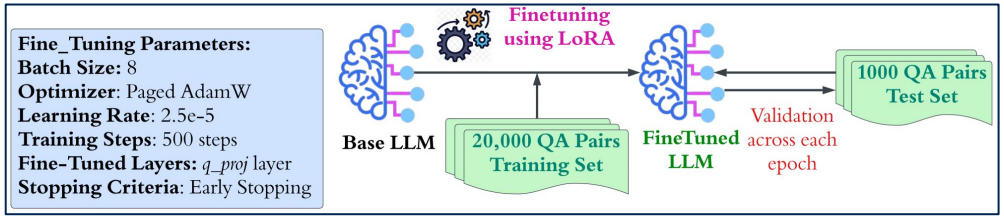
Provide a helpful and accurate answer, focusing on coding and technical topics.
```

Methodology

Experimental Design

- **Base setting:** Only the query is given to the LLM.
- **RAG setting:** Retrieves top-5 semantically similar Q&A pairs from vector DB; irrelevant context filtered out with embedding distance.
- **RAG + Fine-tuning:** Fine-tuned the best-performing model with LoRA on 20,000 Q&A pairs.

Methodology



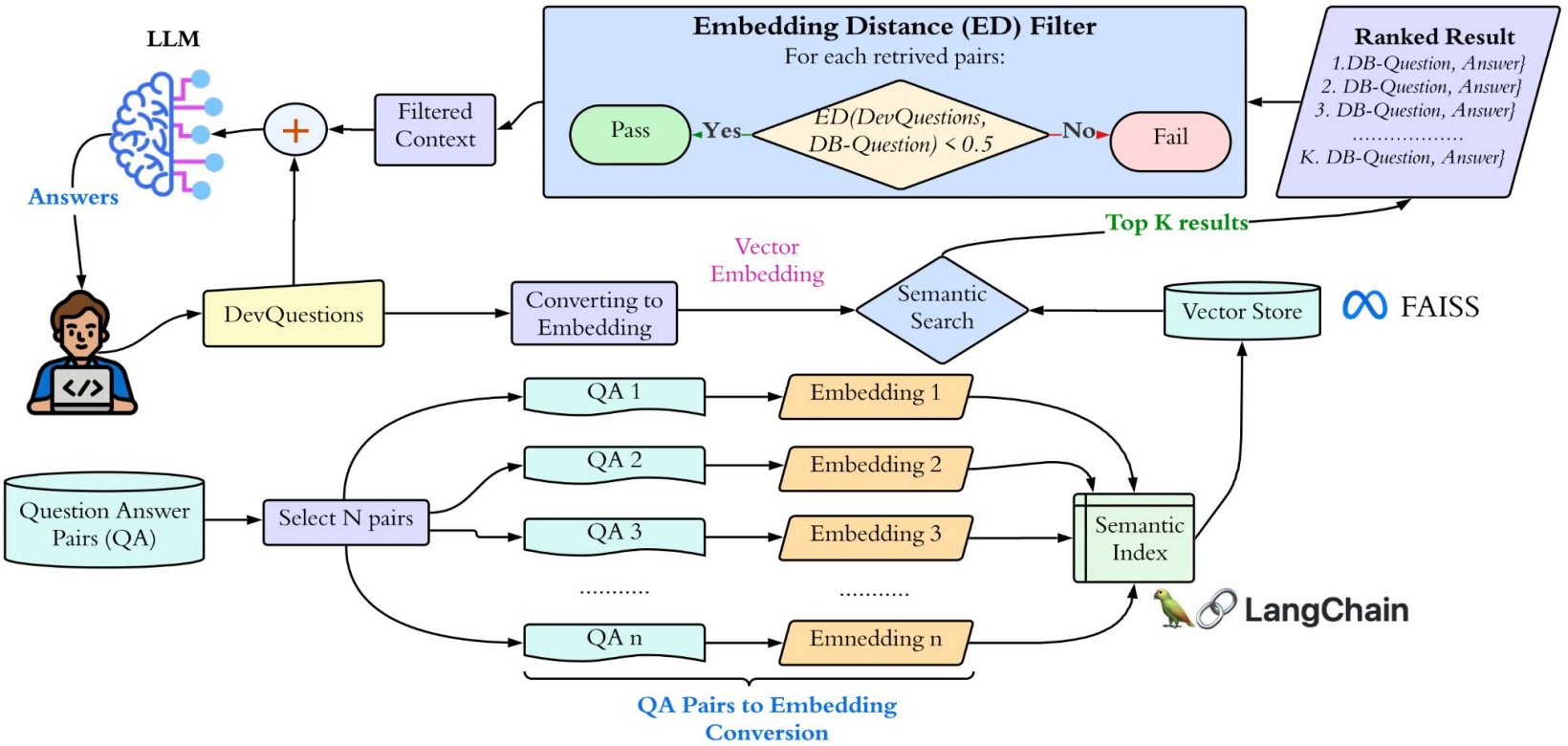
Fine-Tuning Setup

- LoRA fine-tuning on q_proj layer.
- Batch size = 8, learning rate = 2.5e-5, 500 steps, early stopping.
- Used 4-bit quantization + mixed precision to save memory.

Evaluation Metrics

- **Cosine Similarity** – lexical + semantic similarity.
- **BERTScore** – semantic alignment with human-written answers.
- **METEOR** – fluency, word order, and synonym handling.

Architectural Setup for RAG Pipeline



Study Findings

Models	Base Setting			RAG Architecture			RAG+Fine-Tuned Model		
	Cosine Similarity	BERT	METEOR	Cosine Similarity	BERT	METEOR	Cosine Similarity	BERT	METEOR
CodeLlama-7B	0.54	0.54	0.22	0.54	0.55	0.25	X	X	X
Solar-10.7B	0.56	0.54	0.23	0.64	0.59	0.31	0.65	0.63	0.61
Mistral-7B	0.32	0.41	0.11	0.57	0.55	0.26	X	X	X
StarCoder2-7B	0.50	0.52	0.20	0.58	0.56	0.28	X	X	X
Qwen2-7B	0.53	0.53	0.22	0.60	0.57	0.29	X	X	X
LLaMA 3 8B	0.55	0.53	0.23	0.61	0.58	0.30	X	X	X

RQ1 – Can local LLMs effectively answer technical questions in a base setting?

- **Solar-10.7B** performed best (Cosine 0.56, BERT 0.54).
- LLaMA-3 8B and CodeLlama-7B followed closely behind.
- Mistral-7B underperformed (Cosine 0.32, BERT 0.41).

Study Findings

Models	Base Setting			RAG Architecture			RAG+Fine-Tuned Model		
	Cosine Similarity	BERT	METEOR	Cosine Similarity	BERT	METEOR	Cosine Similarity	BERT	METEOR
CodeLlama-7B	0.54	0.54	0.22	0.54	0.55	0.25	X	X	X
Solar-10.7B	0.56	0.54	0.23	0.64	0.59	0.31	0.65	0.63	0.61
Mistral-7B	0.32	0.41	0.11	0.57	0.55	0.26	X	X	X
StarCoder2-7B	0.50	0.52	0.20	0.58	0.56	0.28	X	X	X
Qwen2-7B	0.53	0.53	0.22	0.60	0.57	0.29	X	X	X
LLaMA 3 8B	0.55	0.53	0.23	0.61	0.58	0.30	X	X	X

RQ2 – Does RAG improve LLM responses by adding context?

- **Performance improved for all models** with RAG.
- Solar-10.7B had the largest gain (Cosine ↑ to 0.64, BERT ↑ to 0.59, METEOR ↑ to 0.31).
- RAG helps with **semantic accuracy** and **linguistic fluency** by providing relevant information.

Study Findings

Models	Base Setting			RAG Architecture			RAG+Fine-Tuned Model		
	Cosine Similarity	BERT	METEOR	Cosine Similarity	BERT	METEOR	Cosine Similarity	BERT	METEOR
CodeLlama-7B	0.54	0.54	0.22	0.54	0.55	0.25	X	X	X
Solar-10.7B	0.56	0.54	0.23	0.64	0.59	0.31	0.65	0.63	0.61
Mistral-7B	0.32	0.41	0.11	0.57	0.55	0.26	X	X	X
StarCoder2-7B	0.50	0.52	0.20	0.58	0.56	0.28	X	X	X
Qwen2-7B	0.53	0.53	0.22	0.60	0.57	0.29	X	X	X
LLaMA 3 8B	0.55	0.53	0.23	0.61	0.58	0.30	X	X	X

RQ3 – Does fine-tuning within RAG further improve performance?

- Fine-tuning **Solar-10.7B** within RAG gave **extra gains** in BERTScore and METEOR.
- Cosine Similarity remained similar to RAG-only results.
- Fine-tuning sharpened **semantic richness** and **fluency** of responses.
- Integration of RAG + fine-tuning leads to **more accurate and contextually appropriate answers**.

Limitations & Future Work

- **Small dataset size** for some experiments — larger datasets could improve reliability.
- **Limited model variety** — only six LLMs in the 7B–10.7B parameter range were tested.
- **Basic prompting approach** — did not explore advanced prompt engineering methods like Chain-of-Thought, Tree-of-Thought, or n-shot prompting.
- **Scope limited to StackExchange technical sites** — results may not generalize to other domains or question types.

Conclusion

- Locally deployed, fine-tuned **open-source LLMs** can be viable alternatives to proprietary models for software engineering tasks.
- **Solar-10.7B** performed best, especially with **RAG + fine-tuning**.
- RAG integration improves **semantic accuracy** and **fluency**.
- Fine-tuning within RAG adds further gains, producing **more accurate and context-aware responses**.
- Open-source LLMs with RAG and fine-tuning offer a **cost-effective, accessible solution** for AI-powered developer assistance.

Resources and Contact

GitHub:
https://github.com/zeerakb1/llm-developers-evaluation

Papers:
http://www.cs.ucdavis.edu/~filkov/

Email:
nikhan@ucdavis.edu

