Intelli-Research Assistant

Identifying Research Gaps in Multidisciplinary Knowledge Graphs + RAG

Jiuyuan Xie, Ipsita Praharaj, Sahithya Senthilkumar

Carnegie Mellon University, Heinz College

ABSTRACT

- The proliferation of academic **literature necessitates advanced tools for efficient retrieval** and comprehension of scholarly articles.
- Our project introduces a cutting-edge **retrieval-augmented generation framework that utilizes a custom-built Neo4j knowledge graph** to streamline the discovery of research papers.
- We harness the capabilities of finely tuned AI models—GPT-3.5, Code Llama, and Llama—to accurately **translate textual queries into Cypher queries**, facilitating the precise extraction of academic documents
- Currently, our system excels in the **summarization of complex research findings, distilling multiple documents into concise, digestible summaries** based on metadata parameters such as authorship, abstracts, citation frequency, and thematic elements. This summarization process is crucial for researchers who need to quickly assimilate the essence of extensive literature.

MOTIVATION

Challenges with Traditional Retrieval systems using LLM

Reduced Embedding Search Space

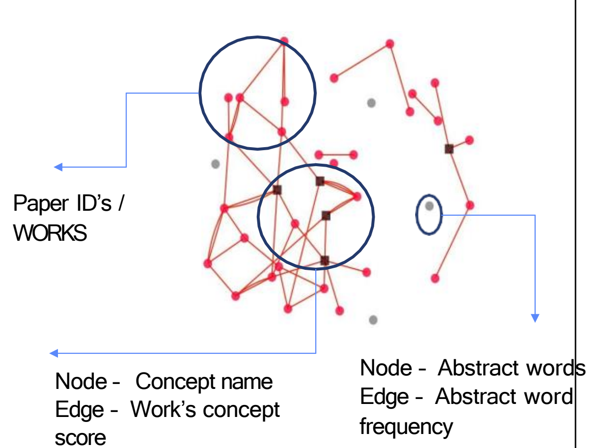
- Graph search reduces search space, improving recommender efficiency & accuracy

Backward Data compatibility:

- E-commerce, entertainment, media, and social platforms utilize Graph DB. Handling complex, ever-changing data, enabling real-time insight extraction

LLM limitations due to Cold Starts:

- LLM often fail to account for new data and context added to dataset due to past training cut off

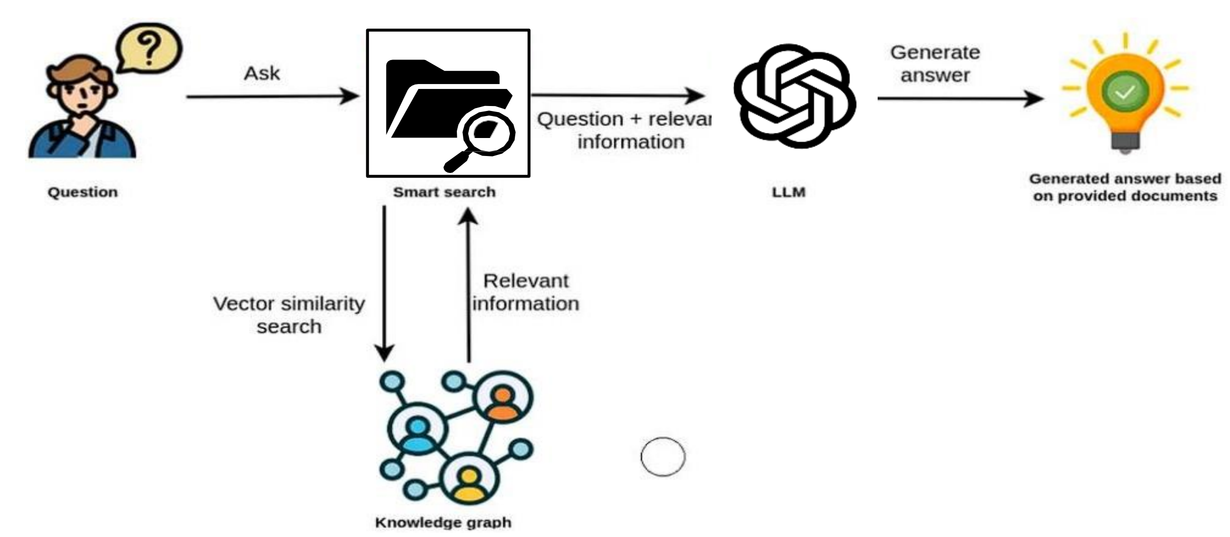


Need for Graph Database

Why LLM?

- •Semantic Concept Deciphering
- •English-to-Cypher Conversion
- •Creative Search Ideas suggestion/ Summarization after KG space is defined

PROPOSED METHODS



Efficient and Accurate retrieval via custom knowledge graphs + LLM

Custom-Created Knowledge Graphs:

- Developed using a Semantic Scholar APIs and a subset of Microsoft Academic Graph.
- Optimized for efficient query processing and data for Academic papers

Cypher + Retrieval Augmented Generation (RAG):

Human text to cypher generation using LLM

Ingest knowledge graph query results into language models for accuracy and speedy retrieval

Use of Various Language Models:

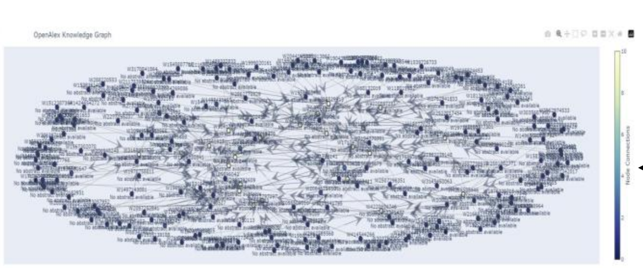
- Employs multiple Large Language Models (LLMs) for varied data handling.
- Each LLM specializes in creating complex cyphers for effective data retrieval.

Fine-Tuning with Qlora on Llama, CodeLlama, PALM, GPT 3.5

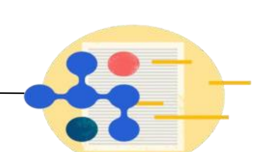
- Applies Qlora, an advanced fine-tuning method, for Llama Code Llama.
- Improves understanding of complex queries and contexts.
- Black Box GPT 3.5 Fine-Tuning: Uses OpenAI's fine-tuning API

Knowledge Graph Dataset Creation

Retrieve embeddings of abstracts from OpenAI and feed it to the custom KG(Knowledge graphs)



A subset of 5000 papers taken from Microsoft Academic Graph which contained computer science papers from top conferences. It has five intrinsic entities(Paper, Author, Affiliation, Venue, Domain)



Used Semantic Scholar API to retrieve abstract data for each paper and populated our graph with it.

Nodes	Count
Papers	5047
Conferences	20
Authors	8680
Affiliation	692
Domain	1923

Relationships	Size
Author_in_affiliation	14488
author_write_paper	28192
paper_cite_paper	17166
paper_in_domain	41274
paper_in_venue	10094

paper

<id>	263
name	"learning to reduce the semantic gap in web image retrieval and annotation"
embed	[-0.0065175248309969 9, 0.011905429884791374, -0.012292547151446342, -0.014634286984801292, -0.011118504218757153, 0.02815164625644684, -0.006092330906... Show all
id	"79867488"
abstract	"We study in this paper the problem of bridging the semantic gap between low-level image features a

Evaluation Dataset Creation

- Custom data - Created a dataset using **Playground GPT - 4**
- X= QUESTION STRING in English with **database schema** | Y = RUNNING CYPHER QUERY
- As a sanity check, the query is executed on the database to ensure it is a valid query.
- Used **one shot** and **few shot** prompting techniques to generate a set of **50 easy, medium and hard** cypher queries = **150 datapoints**
- For the queries that were generated incorrectly, two techniques were used to obtain correct queries.
  - ➢ Queries which had **near regex match** to the syntax was **corrected manually**
  - ➢ Queries that varied a lot was corrected by **chain of thought prompting**.
    - ➢ By feeding back to GPT 4

input_text (string · lengths 111–232)	output_text (string · lengths 32–356)
Generate cypher query for the question: Identify the collaboration network of 'Sebastian Thrun' within…	MATCH (a:author {name: 'Sebastian Thrun'})-[:author_write_paper]->(p:paper) MATCH (p)<-…
Generate cypher query for the question: Which paper in 'Parallel Computing' has authors from the most…	MATCH (p:paper)<-[:author_write_paper]-(a:author)-[:author_in_affiliation]->(aff:affiliation), (p)-…
Generate cypher query for the question: List all the papers from 'University of Oxford' related to…	MATCH (aff:affiliation {name: 'University of Oxford'})<-[:author_in_affiliation]-(a:author)-…
Generate cypher query for the question: What is the network of co-authorship like for 'Daphne Koller' in…	MATCH (a:author {name: 'Daphne Koller'})-[:author_write_paper]->(p:paper)-[:paper_in_venue]->…
Generate cypher query for the question: How many papers from 'Stanford University' are related to…	MATCH (aff:affiliation {name: 'Stanford University'})<-[:author_in_affiliation]-(a:author)-…
Generate cypher query for the question: Which authors are affiliated with more than one affiliation? Only…	MATCH (a:author)-[:author_in_affiliation]->(aff:affiliation) WITH a, COUNT(DISTINCT aff) AS…

RESULTS

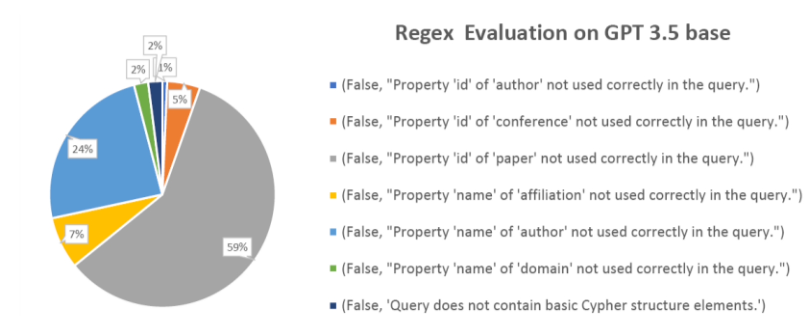
#1 FINE TUNING EXPERIEMENTS

#1 LLM models (baseline vs Fine-tuned) compared for executable query generation

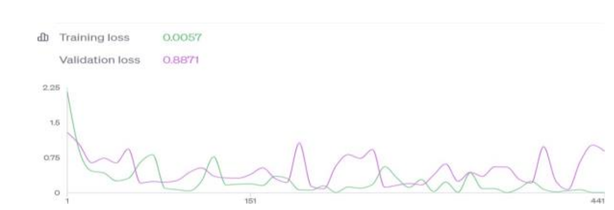
	Palm2	Palm2 Fine_tuned	Llama2	Llama2 LoRA_tuned	GPT 3.5	GPT 3.5 Fine_tuned
Train Runnable scripts	0.276074	0.533742	0.00	0.503067	0.570552	0.901840
Test Runnable scripts	0.464286	0.714286	0.00	0.892857	0.821429	0.964286
All Runnable	0.301370	0.570776	0.00	0.570776	0.621005	0.908676

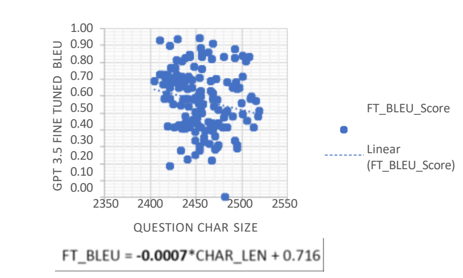
#2 LLM models (baseline vs Fine-tuned) compared for semantically correct query generation

Language translation evaluation scores	Palm2	Palm2 Fine_tuned	Llama2	Llama2 LoRA_tuned	GPT 3.5	GPT 3.5 Fine_tuned
BLEU	0.060464	0.321889	0.0000310	0.479566	0.154573	0.786719
ROUGE@1	0.501176	0.733045	0.036400	0.640206	0.595740	0.907770
ROUGE@2	0.218997	0.497367	0.007779	0.549149	0.315950	0.905520



Fine-tuned GPT





#2 PROMPT EVALUATION EXPERIMENTS

Finding Summarizing Best Prompts	Coherence score GPT 3.5	GPT Notes
Generate paragraph of study	5	Better coherence and detail
Generate paragraph of study (include study ID)	4.5	More comprehensive, includes study ID
How many studies in the database	4	Concise and direct answer
Write summary of study in database	3.5	More detailed and informative
Write summary of study, include all information	3	Comprehensive coverage of study details

GPT 3.5 Fine Tuned Summary – Good for Cypher

```
EXACT ENGLISH QUESTION BELOW------ How many studies in the database EXACT QUERY BELOW-----
- MATCH (p:paper) RETURN COUNT(p) AS paperCount EXACT QUERIED ANSWER BELOW------ [5047]
FINAL ANSWER QUERY BELOW------ MATCH (p:paper) RETURN COUNT(p) AS numPapers numPapers 5047
```

GPT 3.5 Base Summary

```
EXACT ENGLISH QUESTION BELOW------
How many studies in the database
EXACT QUERY BELOW------ MATCH (p:paper) RETURN count(p) AS studyCount
EXACT QUERIED ANSWER BELOW------ [5047]
FINAL ANSWER QUERY BELOW------ The cypher query returned a result of
5047 studies in the database. The schema provided includes information
about nodes such as paper, author, conference, affiliation, and domain,
as well as relationships between these nodes.
```

CONCLUSIONS

1. **GPT 3.5 fine tuned** has the **best performance** in terms of run rate **(~2X score increase)**
   1. Larger model size correlates positively with higher BLEU and ROUGE scores
   2. *Interesting – LLama2 starts with almost 0 to 0.64 BLEU Score **(~6X score increase)***
2. Generated Queries have slight negative (FT_BLEU = **-0.0007***CHAR_LEN + 0.716)
3. Regex Evaluation of Generated Queries trends observed –
   1. More similar **node/relationship names(author, paper) – High Syntax Error rate**
4. Summarization is more consistent with *Query first – LLM Later* approach