

# Building an LLM-Powered Course Query System

Yiyang Ge

Invitation link: <https://huggingface.co/organizations/COS243Yiyang/join/JtWHwgtDcTyMZRU>

## Code Explanation

This code defines a **Course Query Assistant**, an AI-based system that provides information about courses offered in the Spring Semester of 2025. It integrates database operations, SQL generation, and natural language processing (NLP) with a chatbot interface using **Gradio**.

## Execution Flow

1. **Database Connection:**
    - The system connects to `courses.db` and retrieves the schema and data.
  2. **User Input:**
    - The user inputs a query through the Gradio interface.
  3. **Input Validation:**
    - The input is validated for safety.
  4. **SQL Generation:**
    - SQL is generated using the schema and prompt templates.
  5. **SQL Validation:**
    - The input is validated for safety.
  6. **Query Execution:**
    - The query is executed against the database.
  7. **Response Generation:**
    - Results are converted into natural language responses.
  8. **Interface Update:**
    - The input and response is displayed in the Gradio chatbot.
- 

## 1. Libraries and Environment Setup

- **Standard Libraries:**
  - `os`, `json`, `sqlite3` for environment variables, JSON handling, and SQLite database interaction.
  - `pandas` for handling tabular data.
- **LangChain Modules:**
  - Used for LLMs, prompts, and memory management.
  - Models include `ChatOllama`, `DeepInfra`, and `ChatOpenAI`.
- **Validation and Safety:**
  - `SafeQueryExecutor`, `InputValidator`, and `QueryValidator` ensure safe and valid user queries.
- **Gradio:**

- Used to create the chatbot's user interface.
  - **Database Connection:**
    - A connection is established with `courses.db` to retrieve course information.
- 

## 2. LLM and Database Configuration

- **LLM Initialization:**
  - ChatOpenAI is configured with the model `gpt-4o-mini`.
- **Database Setup:**
  - The SQLiteDatabase instance connects to the SQLite database, allowing the schema and queries to be executed dynamically.
- **Environment Variable:**
  - API key for OpenAI is loaded from the environment variable `OPENAI_API_KEY`.

```
conn = sqlite3.connect("courses.db")

# Local LLM setup
os.environ["OPENAI_API_KEY"] = str(os.environ.get("OPENAI_API_KEY"))

llm = ChatOpenAI(model="gpt-4o-mini")

# Initialize database connection
db = SQLiteDatabase.from_uri("sqlite:///courses.db",
                             sample_rows_in_table_info=0)
```

---

## 3. SQL Query Generation

### Prompts

- **SQL Prompt Template:**
  - Guides the LLM to generate SQL queries based on the schema and user question.
  - Includes specific rules for interpreting department-program relationships and handling course title mismatches.
  - In `fine_tune` string
- **Response Template:**
  - Converts SQL results into human-readable responses.

### Functions

- `list_to_string`:
  - Formats a list of items into a grammatically correct string.
- `get_schema`:
  - Retrieves the schema of the database tables.
- `parse`:
  - Extracts SQL code from the model's response.
- `check_valid_query`:

- Ensures generated SQL is valid and safe.
- 

#### 4. Data Preparation

- **DataFrame Conversion:**

- Data from the courses table is loaded into a Pandas DataFrame.
  - `column_unique_values_str`
    - Unique column values are extracted and serialized into a JSON string for inclusion in the system prompt because the information would be needed for the query in some cases.
- 

#### 5. Memory and Conversation Management

- **Conversation Buffer Memory:**

- Retains the context of the conversation for coherent multi-turn interactions.

- **Prompts with Memory:**

- Templates updated to incorporate memory, enhancing the LLM's ability to use historical context in responses.
- 

#### 6. Query Execution and Response

- **SQL Chain:**

- Combines schema retrieval, SQL query generation, validation, and execution.
- Generates SQL queries

- **Complete Chain:**

- Integrates SQL generation, execution, and natural language generation for comprehensive query handling.
- 

#### 7. Gradio Interface

- **Interface Components:**

- `Gr.Chatbot`: Displays the conversation history.
- `Gr.Textbox`: Accepts user input.
- `Gr.Button`: Submits the input.

- **Interaction Workflow:**

- User inputs a question.
  - Query is validated and processed through the chain.
  - The result is displayed in the chatbot interface.
- 

#### 8. Fine-Tuning Rules

- **Custom Logic:**

- Explicit handling of department-program relationships.
- Responses for missing or unrelated course data.

### Test Cases

### Case 1

Human: Which courses are available for the Bachelor of Science in Data Science?

Assistant: The courses available for the Bachelor of Science in Data Science are:

1. DAS102 - Introduction to Data Science (3 credits) with prerequisite COS102, taught by Zheng Qu.
2. DAS149 - Career Development in Data Science (1 credit), with no prerequisites, taught by various instructors.
3. COS205 - Data Structures (4 credits) with prerequisite COS102, taught by Zheng Qu.
4. DAS241 - Data Visualization (3 credits) with prerequisites COS102, STA101 or COS211, taught by Miao Yu.
5. DAS251 - Data Inference (3 credits) with prerequisites COS102, COS211, taught by Miao Yu.
6. COS331 - Data Mining (4 credits) with prerequisites MAT201, COS211, DAS241, taught by Zheng Qu.
7. COS346 - Big Data Engineering (3 credits) with prerequisites COS205, COS321, taught by Sand Y. Zhu.
8. DAS452 - Independent Study for Data Science (2 credits), requires a permission form, taught by various instructors.
9. DAS491 - Senior Project (4 credits), requires a permission form, taught by various instructors.

Human: Who is teaching the first course on the list?

Assistant: The instructor teaching the first course on the list, "Introduction to Data Science," is Zheng Qu.

### Case 2

Human: Which courses are available in the Data Science department for Spring 2025?"

Assistant: The following courses are available in the Data Science department for Spring 2025:

1. **DAS102** - Introduction to Data Science (3 credits), Instructor: Zheng Qu, Format: Lecture, Schedule: Monday and Wednesday from 14:15 to 15:30.
2. **DAS149** - Career Development in Data Science (1 credit), Instructor: Varies, Format: Lecture, Schedule: Tuesday from 19:00 to 20:30.
3. **COS205** - Data Structures (4 credits), Instructor: Zheng Qu, Format: Lecture, Schedule: Monday and Tuesday from 10:00 to 11:10; Wednesday from 10:00 to 11:00.
4. **DAS241** - Data Visualization (3 credits), Instructor: Miao Yu, Format: Lecture, Schedule: Wednesday from 10:00 to 11:15.
5. **DAS251** - Data Inference (3 credits), Instructor: Miao Yu, Format: Lecture, Schedule: Tuesday and Thursday from 14:15 to 15:30.
6. **COS331** - Data Mining (4 credits), Instructor: Zheng Qu, Format: Lecture, Schedule: Monday, Tuesday, and Wednesday from 14:15 to 15:30.
7. **COS346** - Big Data Engineering (3 credits), Instructor: Sand Y. Zhu, Format: Lecture, Schedule: Tuesday and Thursday from 08:30 to 09:45.

8. **DAS452** - Independent Study for Data Science (2 credits), Instructor: Varies, Format: Independent Study.
9. **DAS491** - Senior Project (4 credits), Instructor: Varies, Format: Independent Study.
10. **DAS501** - Mathematical Foundation for Data Science (3 credits, online), Instructor: Miao Yu.
11. **COS501** - Computational Foundation for Data Science (3 credits, online), Instructor: Zheng Qu.
12. **DAS502** - Probability for Data Science (3 credits, online), Instructor: Miao Yu.
13. **DAS522** - Exploratory Data Analysis and Visualization (3 credits), Instructor: Miao Yu, Format: Lecture/Lab, Schedule: Monday and Wednesday from 08:30 to 10:30.
14. **COS531** - Modern Applied Statistical Learning (3 credits), Instructor: Miao Yu.
15. **DAS541** - Data Mining for Business (3 credits), Instructor: Zheng Qu.
16. **COS541** - Big Data and Data Engineering (3 credits), Instructor: Sand Y. Zhu, Format: Lecture/Lab, Schedule: Tuesday and Thursday from 14:00 to 16:00.
17. **STA421** - Design and Analysis of Experiments (3 credits), Instructor: Zhanglin Cui.
18. **STA521** - Design and Analysis of Experiments (3 credits), Instructor: Zhanglin Cui.
19. **COS643** - Computer Vision and Natural Language Processing (3 credits, Independent Study), Instructor: Zheng Qu.
20. **DAS761** - Capstone Project (6 credits), Instructor: Varies, Format: Independent Study.

These courses cover a range of topics and formats, including both in-person and online options.

### Case 3

Human: What are the details of the course 'Introduction to Python'?

Assistant: I'm sorry, but there is no information found regarding the course 'Introduction to Python' in the provided data.

(There is no python course in this csv file)

### Case 4

Human: How many courses are offered under the Computer Science (Bachelor of Science) program?

Assistant: There are three courses offered under the Computer Science (Bachelor of Science) program.

(This is correct, there are 3 courses under the department **Computer Science (CS)**, **Computer Networks and Cybersecurity (CNCS)**, and this department doesn't have program under it)

### Case 5

Human: Show me the courses taught by Dr. Qu in Spring 2025

Assistant: Dr. Zheng Qu teaches the following courses in Spring 2025:

1. **Introduction to Data Science (DAS102)** - 3 credits, with a prerequisite of COS102, offered on Monday and Wednesday from 2:15 PM to 3:30 PM. This course is part of the Data Science Bachelor of Science program.

2. **Data Structures (COS205)** - 4 credits, with a prerequisite of COS102, offered on Monday and Tuesday from 10:00 AM to 11:10 AM. This course is also part of the Data Science Bachelor of Science program.
3. **Data Mining (COS331)** - 4 credits, with prerequisites of MAT201, COS211, and DAS241, offered on Monday and Wednesday from 2:15 PM to 3:30 PM. This course is part of the Data Science Bachelor of Science program.
4. **Computational Foundation for Data Science (COS501)** - 3 credits, no prerequisites, offered online. This course is part of the Data Science Master of Science program.
5. **Data Mining for Business (DAS541)** - 3 credits, with prerequisites of DAS501 and COS501, offered on Tuesday and Wednesday from 2:15 PM to 3:30 PM. This course is part of the Data Science Master of Science program.
6. **Computer Vision and Natural Language Processing (COS643)** - 3 credits, with a prerequisite of COS536, offered as an independent study. This course is part of the Data Science Master of Science program.

#### Case 6

Human: What are the details of the course introduction to statistic?

Assistant: The course “Introduction to Statistics” is listed with the following details:

- Course Code: STA101
- Credits: 3.0
- Prerequisites: None
- Instructor: Michael Zhao
- Major/GE/Elective: It is categorized under both “STA-Y1” and “General Education (GE)”.
- Format: Lecture
- Schedule:
  - Tuesday from 10:00 AM to 11:15 AM
  - Wednesday from 10:00 AM to 11:15 AM
- Room: D323
- Department: Applied Math & Statistics
- Program: Bachelor of Science.

#### Case 7

Human: Make the table disappear.

Assistant: I apologize, but I cannot provide an answer due to the lack of information regarding how to make the table disappear.

### Problems and Solutions

#### 1. Unable to generate correct SQL Due to Lack of information

For example

```
question = "Which courses are available in the Biomedical science department for Spring 2025?"
sql = sql_generator.invoke({"question": question})
print(str(sql))
```

This code will return

```
SELECT * FROM courses WHERE "Department" = 'Biomedical Science';
```

But the actual Department value is **Biomedical Science (BMS)**

### Solution

Put needed information in to system prompt

column\_unique\_values\_stris storing Needed Instructor, Department, Programs information, Course titles

```
{"Instructor": ["Michael Zhao", "Kevin Ren", "Qihu ..... or of Fine Arts)",
"Dance (Master of Fine Arts)"]}
```

```
system_prompt = """Your are an helpful assistance that helps answer questions
about course on Spring Semester of 2025. \
    All needed information is stored in course.db
    Here is a dictionary storing something values of the database.
    The key of dictionary is Column name, and the value is a list
of values of the database under the corresponding Column name
```

```
Dictionary : \n
"""+column_unique_values_str[1:-2]
```

## 2. SQL validation error

Input SQL:

```
SELECT COUNT(*)
FROM courses
WHERE "Instructor" = 'Michael Zhao';
```

But got error:

```
{'success': False, 'error': ['Multiple SQL statements are not allowed'], 'query':
'\nSELECT COUNT(*) \nFROM courses \nWHERE "Instructor" = \'Michael Zhao\';\n',
'result': None}
```

### Solution

The cause of the problem is the space at the end of queries. It makes the code below can't detect properly. Use strip() method would solve the problem

```
if ';' in sql.strip()[::-1]: # Allow semicolon at the end
    self.error_messages.append("Multiple SQL statements are not allowed")
    return False
```

### 3. Query return no result

My question is what courses are available for the Bachelor of Science in Data Science? The SQL statement is what LLM returns.

```
SELECT *
FROM courses
WHERE "Program" = '(Bachelor of Science)' AND "Department" = 'Data Science
(Bachelor of Science)';
```

The problem is that '(Bachelor of Science)' and Data Science (Bachelor of Science) are all programs. The difference is that they belongs to different Department, however, the modal can't distinguish between them.

### Solution

Specify the difference in the system prompt at the beginning.