Project Code	**LLM-SQL**
Type	DATA601 (Academic research project)
Title	Develop a LLM Agent to Query SQL Databases
Department	Maths and Stats
Academic supervisor	Phil Davies (Philip.Davies@canterbury.ac.nz)
Location	University of Canterbury
Students	Group

Project Summary

Project Overview

The goal of this project is to develop an advanced AI-powered query system that leverages the capabilities of large language models (LLMs) to interact with SQL databases. Using LangChain, for example, a framework for developing applications powered by language models, your team will create an intelligent agent capable of understanding natural language queries, translating them into SQL commands, executing these commands against a database, and returning the results to the user in a human-readable format.

Objectives

- **Familiarization with LangChain:** Understand the architecture and capabilities of LangChain and how it can be used to build applications with language models.
- **Database Design and Management:** Design and implement a SQL database suitable for storing and retrieving data relevant to the project's focus area.
- **Natural Language Processing:** Develop the LLM agent to accurately interpret and translate natural language queries into SQL commands.
- **Query Execution and Result Formatting:** Ensure the agent can execute SQL queries and format the results in a user-friendly manner.
- **User Interface Development:** Create a simple, intuitive interface for users to interact with the LLM agent.

Deliverables

- **Project Plan:** A detailed project plan outlining the steps, timeline, and resources required.
- **Database Schema:** A well-designed schema for the SQL database, including tables, relationships, and sample data.
- **LLM Agent:** A functional LLM agent using LangChain capable of understanding and processing natural language queries.
- **Interface:** A user-friendly interface for interacting with the LLM agent.
- **Documentation:** Comprehensive documentation covering the development process, system architecture, usage instructions, and future work recommendations (project report).
- **Presentation:** A final presentation demonstrating the project, key findings, challenges encountered, and solutions implemented.

Skills and knowledge required

- Proficiency in Python programming
- Understanding of SQL and database management
- Knowledge of natural language processing (NLP) techniques

- Familiarity with machine learning concepts and frameworks
- Experience with front-end/user-interface development (optional but beneficial)

Tools and technology

- LangChain (or equivalent): Framework for developing applications with language models
- SQL Database: MySQL, PostgreSQL, or SQLite
- Python Libraries: SQLAlchemy, Pandas, NLTK, or SpaCy
- Front-End Framework: Shiny, Streamlit, Flask or Django for creating the user interface (optional)

Assessment criteria

- **Technical Implementation:** Accuracy and efficiency of the LLM agent in processing queries and interacting with the database.
- **Innovation:** Creativity in problem-solving and the application of advanced NLP techniques.
- **User Experience:** Usability and design of the user interface.
- **Documentation and Presentation:** Clarity, thoroughness, and professionalism in documentation and presentation.

Guidelines for writing the report and presentation may be found on the DATA60x LEARN page. Be sure to document all the effort you have made.

Project Timeline

The following is a suggested project timeline.

- Weeks 1-2: Project planning, research, and familiarization with LangChain and relevant tools.
- Weeks 3-4: Database design and initial setup.
- Weeks 5-7: Development of the LLM agent.
- Weeks 8-9: Interface development and integration.
- Weeks 10-11: Testing, debugging, and optimization.
- Week 12: Final documentation, presentation preparation, and project submission.

By the end of this project, your team will have gained practical experience in applying language models to real-world problems, enhancing your skills in natural language processing, database management, and application development.

Learning Outcomes

Technical Skills:

- Weeks 1-2: Project planning, research, and familiarization with LangChain and relevant tools.
- Master data pre-processing techniques for text data, including cleaning, tokenization, and labeling for NLP tasks.
- Develop proficiency in LLM prompt engineering and fine-tuning for tasks like ticket classification, sentiment analysis, and response generation.
- Gain experience in designing and coordinating multi-agent systems using frameworks like LangChain or AutoGen.
- Learn to implement and evaluate NLP models using metrics such as classification accuracy, F1-score, and BLEU/ROUGE for response quality.
- Build practical skills in creating user-friendly interfaces with tools like Streamlit, Flask, or Shiny for data-driven applications.

Soft Skills:

- Enhance teamwork and collaboration through role-based contributions and integrated system development.

- Strengthen project management abilities by adhering to milestones, managing timelines, and coordinating tasks.
- Improve communication skills through clear documentation, presentations, and discussions of technical and ethical considerations.

Portfolio Impact:

- Create a showcase project demonstrating end-to-end data science skills, from data pre-processing to agent-based automation and UI deployment, with strong industry relevance.

Documentation and Iterative Improvements

Students should document challenges, solutions, and iterative refinements to demonstrate the development process clearly, enhancing learning outcomes and project transparency.

Summary

This project balances technical complexity with practical utility, showcasing the power of LLMs and agent-based workflows in data science.