ChatDB: Interactive Database Query Assistant

**Team Details**

Group Category: Course Project ChatDB
Group Name: ChatDB 17

**Team Members Background and Skills**

**Dhyey Desai**:

I am a Master's student in Applied Data Science with a solid foundation in machine learning, data analytics, and deep learning. I have hands-on experience with Python, TensorFlow, and scalable infrastructure, along with strong proficiency in data management systems like SQL and Hadoop. My skills in natural language processing and advanced deep learning models have been essential in developing projects such as hate speech detection and medical image segmentation, and I am eager to apply these abilities to tackle complex challenges in dynamic environments.

**Prem Doshi**:

I am a graduate student specializing in Applied Data Science at the University of Southern California. I have a solid foundation in machine learning and data analysis, with extensive experience using Python, TensorFlow, and various data visualization tools. My expertise in developing advanced predictive models, integrating natural language processing techniques, and utilizing large language models (LLMs) will be crucial for enhancing data-driven insights and user experiences in innovative projects.

**Project Requirements**

1. Tech stack used for implementation:

    - Python: Primary programming language
    - mysql-connector-python: For MySQL database interactions
    - pymongo: For MongoDB database interactions
    - nltk: For natural language processing tasks
    - re: For regular expression operations in query matching

2. Plan to implement query syntax:

    We've implemented a keyword-based matching system using predefined query templates. The process involves:

    - Tokenizing and preprocessing the natural language input
    - Matching pre-processed tokens against predefined keywords
    - Selecting the appropriate query template based on matched keywords

- Extracting conditions from the input to populate the template
- Generating the final SQL or NoSQL query

3. Databases chosen:

    - MySQL: Representing SQL databases
    - MongoDB: Representing NoSQL databases This combination allows us to demonstrate the system's versatility across different database paradigms.

**Planned Implementation**

Our implementation strategy for ChatDB includes:

1. Database Connectivity:
    - Implemented connection methods for MySQL and MongoDB
    - Allows users to input their database credentials through the CLI
2. Query Pattern Matching:
    - Define a set of query patterns using regular expressions
    - Create templates for SQL and NoSQL queries corresponding to each pattern
3. Natural Language Processing:
    - Utilizing NLTK for tokenization, stop-words removal, and lemmatization
    - Matching pre-processed user input against predefined patterns
4. Query Execution:
    - Implement methods to execute queries on both SQL and NoSQL databases
    - Handle differences in query execution between database types
5. Result Presentation:
    - Create a function to display query results in a formatted table
6. User Interface:
    - Develop a menu-driven command-line interface for user interactions
    - Implemented basic error handling and user guidance features
7. Extensibility:
    - Design the system to allow easy addition of new query patterns and database types

**Deviation from initial proposal**: We simplified the NLP approach, focusing on keyword matching instead of more advanced NLP techniques to ensure robustness and ease of implementation.

**Status of the Project**

Completed:

- Database connections for MySQL and MongoDB
- Basic query pattern matching system
- Natural language query preprocessing and matching
- Query execution for both SQL and NoSQL databases
- Command-line user interface
- Result formatting and display in tabular form

Next steps:

- Implement more complex query patterns (e.g., joins, nested queries)
- Add data visualization capabilities for query results
- Add an expanded main menu with options for visualization, schema generation, and query suggestions
- Query suggestions based on the database schema
- Expand error handling and user guidance

**Challenges**

1. **Balancing flexibility and accuracy in natural language processing:**
    - Challenge: Interpreting diverse user inputs accurately while maintaining system flexibility
    - Solution: Focused on a keyword-based matching system with predefined patterns, allowing for easier expansion and maintenance
2. **Handling diverse query structures across SQL and NoSQL databases:**
    - Challenge: Creating a unified system that can generate appropriate queries for both database types
    - Solution: Implemented separate query templates for SQL and NoSQL, with a common matching system to select the appropriate template
3. **Ensuring robustness in query execution:**
    - Challenge: Handling potential errors in user input or database operations
    - Solution: Implemented basic error handling in query execution and user interface. Further improvements are planned

**Team Responsibilities**

**Dhyey Desai:**

- Database connection and interaction (MySQL and MongoDB)
- Query pattern implementation and natural language processing
- Query execution logic for both SQL and NoSQL databases

**Prem Doshi:**

- Command-line interface development
- User interaction flow and error handling
- Result formatting and presentation
- Documentation and testing

Both team members will collaborate on system architecture, query pattern design, and overall integration of components.

**Timeline**

| Week | Milestone | Responsible | Status |
|---|---|---|---|
| **1** | Project setup and architecture design | Both | Completed |
| **2** | Database connection implementation | Dhyey | Completed |
| **3** | Basic CLI structure and user input handling | Prem | Completed |
| **4** | Query pattern definition and matching system | Dhyey | Completed |
| **5** | Natural language processing implementation | Dhyey | Completed |
| **6** | Advanced query patterns implementation | Dhyey | In Progress |
| **7** | Data visualization integration | Prem | In Progress |
| **8** | Testing and bug fixing | Prem | Planned |
| **9** | Documentation and final touches | Both | Planned |
| **10** | Project presentation preparation | Prem | Planned |

This timeline is subject to adjustment as the project progresses.