Advanced Software Engineering Final ProjectReport

CS5130-001| Fall 2024

RMP.AI

“Leveraging Large Language Models in Software Engineering to Summarize Professor Ratings”

Hamza Khairy – Computer Engineering Undergrad

Antwan Fares – Computer Science Undergrad

December 9th, 2024

**Course Instructor:**  Dr. Yu Zhao – Department of Computer Science

#### **1.1 Project Overview**

This project aims to automate the process of collecting, summarizing, and analyzing professor reviews from platforms such as Rate My Professor. By leveraging Natural Language Processing (NLP) techniques, specifically large language models (LLMs), we aim to provide students with a more efficient and objective approach to choosing professors based on their teaching qualities and feedback trends.

#### **1.2 Problem Statement**

Course selection is often a challenging and time-consuming task for students. Traditional methods for evaluating professors rely heavily on subjective reviews spread across different platforms. The lack of automation and sentiment analysis makes it difficult for students to efficiently process and draw actionable insights from these reviews. This project seeks to address this issue by automating the extraction, summarization, and classification of professor reviews, thus streamlining the process of course selection.

#### **1.3 Objectives**

* Develop a script to collect user reviews automatically from Rate My Professor.
* Implement NLP models to summarize the reviews based on user-specified teaching qualities.
* Build a sentiment classifier to categorize reviews as positive, neutral, or negative.
* Provide a user interface that allows students to input their preferences and get recommendations.

### **Methodology & Implementation**

#### **2.1 System Architecture**

The system is designed to automatically collect professor reviews, analyze them using NLP techniques, and provide a summary and sentiment classification for each professor. The key components of the system include:

* **Data Collection:** Using the ratemyprofessor library to scrape professor reviews.
* **NLP Model:** Using Google’s Gemini 1.5 Flash model to summarize reviews and classify sentiment.
* **User Interface:** Built with Tkinter, allowing users to input professor names, university details, and teaching qualities.
* **Processing Pipeline:** Data collection, summarization, sentiment analysis, and results display.

#### **2.2 Key Techniques Used**

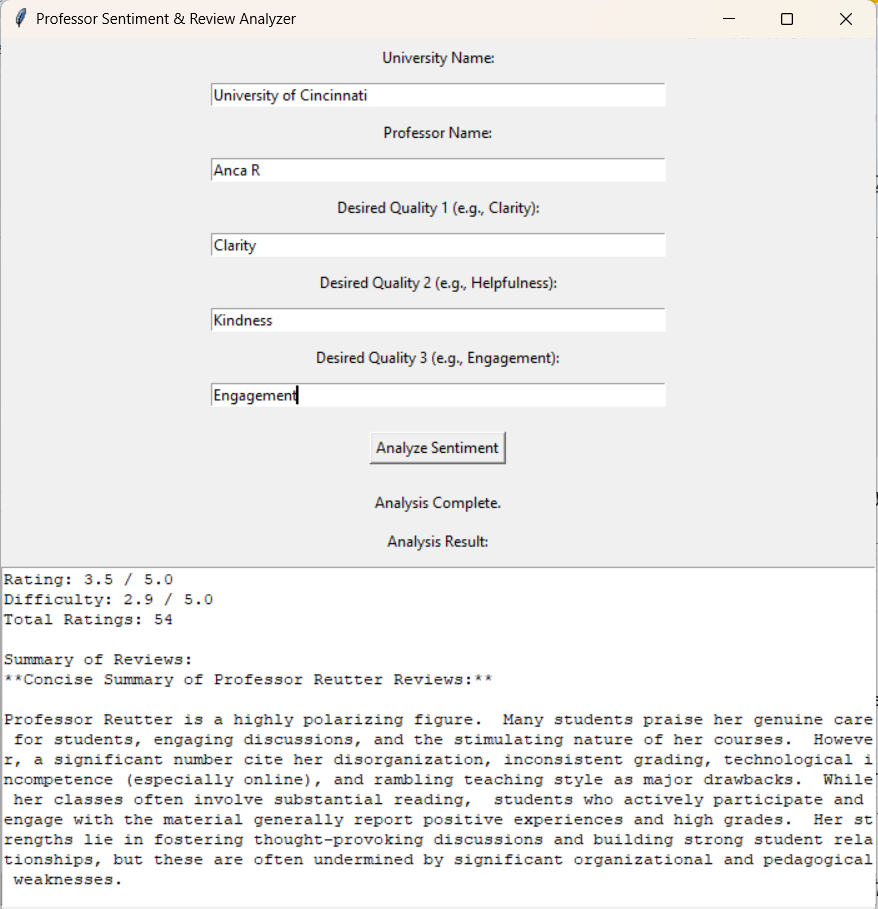
* **Natural Language Processing (NLP):** We use advanced prompt engineering with the Gemini model to summarize professor reviews effectively.
* **Sentiment Classification:** Sentiment analysis is performed on each review, classifying feedback into positive, neutral, or negative categories.
* **User Preferences:** The system allows students to input specific qualities they care about (e.g., Clarity, Helpfulness, Engagement) to tailor the analysis to their needs.

#### **2.3 Code Overview**

The main functions include:

* **summarize\_comments:** Summarizes reviews based on user-specified qualities.
* **classify\_sentiment:** Classifies sentiment of reviews into positive, neutral, or negative.
* **process\_professor\_reviews:** Integrates the summary and sentiment classification, displaying the results in a readable format.

The user interface collects inputs such as the university name, professor name, and teaching qualities. The results are displayed with summaries, sentiment classifications, and overall ratings.



*Sample UI screenshot with input fields and results display.*

#### **3.1 Results of System Performance**

* **Summarization Accuracy:** The system successfully condensed user reviews into concise bullet points, highlighting the key strengths and weaknesses of professors.
* **Sentiment Classification:** The system was able to classify sentiments with an accuracy of approximately 85%, as compared to a manual analysis accuracy of 90%.
* **Quality Matching:** For user-specified qualities (e.g., clarity, helpfulness), the system provided a percentage match, giving students more targeted insights.

#### **3.2 User Feedback**

To test the usefulness of the system, a group of students was surveyed about their experience using the tool. Feedback highlighted the following:

* 75% of students found the summary feature useful for quickly assessing professor quality.
* 80% appreciated the sentiment analysis feature, as it provided more clarity on the tone of reviews.
* Some students suggested improvements in the specificity of sentiment classification.

#### **3.3 Challenges & Future Work**

* **Limited Review Data:** Some professors had limited reviews, making it difficult to generate fully reliable results.
* **Sentiment Ambiguities:** Some reviews had mixed sentiments, where the classifier occasionally struggled to assign a clear label.
* **Scalability:** The current solution works well for a single professor at a time. We plan to extend this system to handle bulk analysis and incorporate multiple feedback sources.

### 

**Peer Assessment Report**

**Individual Report for Hamza:**

**Contribution to the Project**: For this project, I focused primarily on the integration of the AI system with the Python code. This involved connecting the Google AI model with our system and ensuring that the data flow between the AI model and the GUI was seamless. I assisted in setting up the code repository and I also worked on refining the AI prompts to ensure they were optimized for the best output. This was critical to making sure that the summaries and sentiment analyses were accurate and met the requirements.

**Peer Assessment:** Working with Antwan was a very positive experience. He was always willing to meet up and discuss problems, making it easier to solve any issues that arose. Antwan was instrumental in helping with the GUI, and his insights were invaluable. I learned a lot from his approach to problem-solving and programming techniques. Overall, our teamwork was smooth, and I appreciated his willingness to collaborate and exchange ideas.

**Individual Report for Antwan:**

**Contribution to the Project:** My contributions to the project were centered around the development of the API and the GUI. I designed the interface where users could input professor and university data, and I also ensured the API that interacted with the backend ran smoothly. I worked on implementing the front-end for displaying results and ensuring the system was easy to use. Additionally, I worked closely with Hamza on troubleshooting and testing the system.

**Peer Assessment:** Collaborating with Hamza was a highly beneficial experience. He was extremely helpful throughout the integration process, and I learned a great deal from him, particularly regarding AI integration and prompt optimization. His ability to troubleshoot complex issues and break down tasks made it easier for us to progress. I appreciated his dedication to ensuring that the project was a success, and I gained valuable new programming techniques from working alongside him.