Feedback Assistant for Authors

1. Introduction

The aim was to implement an interactive feedback assistant designed to help authors understand their readers better. The assistant aggregates positive and negative feedback for books based on user (author) input and provides actionable suggestions for improvement on his work.

2. Approach and Methodology

A. **Data Preparation:**

- Merged the book reviews dataset with the books details to get the author against each book. Review details such as – title, rating and description which were mapped against each book and respective author were used

B. **Feedback Aggregation:**

- **Collating Reviews:** Aggregated reviews into a summary format containing the title, rating, and description.
- **Generating Feedback:** Used OpenAI's GPT-4o model to analyze and summarize the collated reviews into positive and negative feedback and provided recommendations for improvement areas of Improvement.

C. **Implementation:**

- **Input Handling:** The assistant takes user input for the author and book title(s), which can be a single title, multiple titles, or 'All'.
- **Data Processing:** Loaded and processed data from CSV files, merged datasets, and prepared review summaries with the input filters
- **API Integration:** Utilized OpenAI's API to analyse and generate feedback based on the collated summaries.
- **OpenAI Model used:** "gpt-4o-mini"
- **Prompt Used:**

```
Below are some collated reviews for a few books of {author_name}.
Reviews are in format "Title - Rating - Review"
Summarize the top 3-5 positive and top 3-5 negative takeaways from the reviews.
Also highlight the improvement areas for the author for his/her next work.
Remember to ignore if a comment is irrelavant to the book/work
```

- **Output Generation:** The assistant provides a summary of positive feedback, a summary of negative feedback, and actionable suggestions for improvement. It also takes care that irrelevant comments don't skew the output

3. Starter Code

//shared in a jupyter notebook - feedback_assistant.ipynb

4. Next Steps for MVP

A. Data Collection:

- **Expanding Dataset:** Collect additional reviews and data points to enhance the dataset. This will help in generating more accurate and comprehensive feedback.
- **Integrations:** Integrate additional data sources or review platforms

B. Enhance Feedback Analysis:

- **Fine-Tuning OpenAI Models:** Explore the possibility of fine-tuning the OpenAI models on domain-specific data to improve their performance on feedback summarization and analysis tasks.
- **Prompt Refining:** Improve summarization accuracy by refining prompt engineering and handling various feedback types more effectively

C. User Interface:

- Develop a user-friendly web/app interface to facilitate input and display results.

D. Testing and Validation:

- **Testing:** Conduct thorough testing to validate the feedback quality and ensure the assistant provides useful insights.
- **Author Feedback Mechanism:** Develop a feature that allows authors to provide feedback on incorrect or irrelevant reviews. This will help refine the review dataset and improve the accuracy of the feedback summaries

5. Conclusion

The feedback assistant successfully leverages OpenAI's GPT-4o-mini model to provide valuable insights for authors, summarizing reader feedback and offering suggestions for improvement. Further development will focus on enhancing feedback accuracy and user interaction.