Frequently Asked Questions

Product Review Sentiment Analysis System

Introduction

This document provides answers to the most common questions regarding the **Product Review Sentiment Analysis** system. It covers the technology stack, methodology, implementation details, and system requirements.

Q1: **What is the primary purpose of the Product Review Sentiment Analysis system?** The system is designed to automate the extraction and analysis of customer feedback from PDF documents, using Natural Language Processing (NLP) to classify sentiments and generate actionable business insights through interactive visualizations.

Q2: **Which programming language is the core of this project?** The primary programming language used is Python (v3.7+), selected for its robust ecosystem of data science and NLP libraries.

Q3: **What library is used for the web-based dashboard interface?** Streamlit is used for the frontend, allowing for rapid dashboard development and real-time user interactivity.

Q4: **How does the system extract text from PDF files?** The system uses the `PyPDF2` library to iterate through PDF pages and extract textual content via the `extract_text()` method.

Q5: **Which NLP library is used for sentiment classification?** The system utilizes `TextBlob`, which is built on NLTK and Pattern libraries, for sentiment polarity and subjectivity scoring.

Q6: **What is the range of the polarity score calculated by the system?** The polarity score ranges from a minimum of -1.0 to a maximum of $+1.0$.

Q7: **How is a "Positive" sentiment defined in terms of polarity?** A sentiment is classified as "Positive" if the polarity score is greater than 0.1.

Q8: **How is a "Negative" sentiment defined?** A sentiment is classified as "Negative" if the polarity score is less than -0.1.

Q9: **What constitutes a "Neutral" sentiment in this system?** Any polarity score falling within the buffer zone of $[-0.1, 0.1]$ is classified as "Neutral."

Q10: Why is a 0.1 threshold used for neutral classification instead of zero? The 0.1 threshold creates a "neutral buffer zone" to reduce noise, prevent misclassification of ambiguous feedback, and improve overall accuracy.

Q11: How does the system separate individual reviews from a continuous PDF text? The system uses Regular Expressions (`re`) with the pattern $r'\backslash n\backslash s*\backslash n'$ to split segments based on double newline paragraph breaks.

Q12: Is there a minimum character limit for a string to be considered a review? Yes, the system filters out segments with fewer than 20 characters to remove noise and artifacts.

Q13: Which library is used for creating the interactive charts? Plotly is used to render interactive Pie charts, Bar charts, and Histograms.

Q14: What data structure is used to manage the reviews internally? The system uses a `Pandas DataFrame` to organize data into columns such as `Review_ID`, `Review`, `Sentiment`, and `Polarity_Score`.

Q15: Can users export the analysis results? Yes, the system provides a CSV export functionality via the `st.download_button()` feature in Streamlit.

Q16: What does the Histogram visualization represent? The Histogram displays the frequency distribution of polarity scores across 30 bins, helping identify clustering patterns and sentiment intensity.

Q17: What are the key KPIs displayed on the dashboard? The dashboard displays the total count and percentage of Positive, Negative, and Neutral reviews using the `st.metric()` component.

Q18: What are the four layers of the system architecture? The architecture consists of the Input Layer (upload), Extraction & Processing Layer (regex), Analysis Layer (TextBlob), and Presentation Layer (Streamlit/Plotly).

Q19: How is the "Average Polarity Score" calculated? It is the arithmetic mean (μ) of all individual review polarity scores x_i calculated as $\mu = \frac{1}{n}\sum_{i=1}^{n} x_i$.

Q20: Can the system handle image-based or scanned PDFs? No, the current version requires text-based PDFs. Scanned images require OCR (Optical Character Recognition) which is proposed as a future enhancement.

Q21: What is the "Most Common Sentiment" metric? It represents the mode of the sentiment distribution, identifying the most frequently occurring category (Positive, Negative, or Neutral).

Q22: Does the system support multiple languages? Currently, the system is optimized for English only, though translation API integration is a proposed future enhancement.

Q23: What happens if a user navigates to the app but hasn't uploaded a file? The system displays a clear set of instructions and a feature list to guide the user on how to proceed.

Q24: What are the hardware requirements for this system? A minimum of $2GB$ RAM is required, though $4GB$ is recommended for processing large PDF documents.

Q25: What is the benefit of using a lexicon-based approach (TextBlob)? It provides high speed, requires no custom model training, and offers consistent, objective classification without the need for large labeled datasets.

Q26: How does the system handle very long reviews in the data table? Reviews are truncated to the first 200 characters for display in the table, but the full text is available in the individual review analysis section.

Q27: Is the implementation open source? Yes, the project is built entirely on open-source Python libraries like Pandas, Plotly, and TextBlob.

Q28: What is the command to run the application locally? The application is launched using the command: `streamlit run app.py`.

Q29: Can this system be used for hospitality industry feedback? Yes, it is designed for any industry that receives high volumes of feedback, including e-commerce, hospitality, and market research.

Q30: What is one major future enhancement planned for this project? Integrating transformer-based models like BERT or RoBERTa to better understand sarcasm and context-heavy feedback.