goML

GoML - Technical Evaluation

Time Duration: 24 Hours

Introduction:

You are tasked with developing a web application for a Generative AI-based Quality Management System (QMS) specifically designed for airline travel call centers. This system will process call recordings to analyze call center operations, leveraging the Lyzr AI API for its generative AI capabilities, focusing on transcribing audio files and extracting meaningful insights from these transcriptions. Your application will calculate various Key Performance Indicators (KPIs) to evaluate the quality of calls and the overall performance of agents.

(Take Logical Assumptions whenever needed)

Objectives

- Develop a web application that allows users to upload a zip file containing MP3 audio files of call recordings.
- Extract and transcribe the audio files using the Lyzr AI API.
- Analyze the transcriptions to calculate and display relevant KPIs, thereby assessing call quality and agent performance.

System Requirements

- Basic understanding of Python, web development, and REST APIs.
- Familiarity with Flask or Streamlit for web development.
- Experience with deploying applications on platforms like Vercel.

Step-by-Step Implementation Guide

1. Project Setup: Initialize a Git repository, set up a virtual environment, and install necessary libraries.

2. Building the UI: Create a user-friendly interface with a file upload option for the zip file containing MP3 files.
3. Uploading and Processing: Implement functionality for users to upload the zip file and extract MP3 files efficiently.
4. Integrating Lyzr AI API: Integrate the API to transcribe audio files and ensure accurate transcription.
5. Problem Bucketing/Classification : Each call needs to be catered to specfic problem, that you have to analyse using LLMs Problem Category : Cancellation Flight, Reschedule Flight, Flight Refund Related, Flight Delay Complaint, Staff Behavior Complaint, Miscellaneous

6. Analyzing Transcriptions: Extract insights from transcriptions, such as sentiment, keywords, or specific phrases indicating customer satisfaction or dissatisfaction.
7. Calculating KPIs: Code functions to calculate KPIs like customer satisfaction scores, resolution times, and agent performance metrics.
8. Displaying Results: Design a dashboard to clearly present the calculated KPIs through tables, charts, or graphs.
9. Testing and Deployment: Conduct thorough testing and deploy the application to Vercel.
10. Documentation and Reporting: Maintain detailed documentation and provide a final report summarizing the project.

Key Performance Indicators (KPIs)

Call-Level KPIs:

- Average Handle Time (AHT): Measures the total average duration an agent spends on a call, including talk time, hold time, and after-call work. It’s crucial for assessing individual agent efficiency and ensuring quick resolution times.
- First Call Resolution (FCR): Indicates the percentage of calls resolved on the first interaction, showcasing the agent's ability to solve issues efficiently without follow-up.
- Customer Satisfaction Score (CSAT): A metric derived from customer feedback post-call, reflecting the customer's satisfaction with the service received.
- Call Abandonment Rate: The rate at which customers disconnect the call before reaching an agent, highlighting potential issues in call management or long wait times.

- Call Resolution Rate: The percentage of calls that result in the caller's issue being resolved, indicating the effectiveness of the agents' problem-solving skills.
- Call Transfer Rate: Measures how often calls are transferred to another agent or department, which can indicate the need for better first-contact resolution or agent training.
- Average Speed of Answer (ASA): The average time it takes for a call to be answered by an agent, reflecting the call center's efficiency and customer wait times.
- Call Rating: A metric derived from post-call surveys where customers rate the quality of the call, indicating the perceived effectiveness of the agent and the satisfaction with the call resolution.
- Silence Ratio: The proportion of call time without conversation, indicating possible gaps in communication or the need for more efficient problem-solving.
- Overcall Rate: The percentage of calls that exceed a predefined optimal duration, pointing to potential inefficiencies in handling customer queries.
- Error Rate: The frequency of errors made by agents during calls, such as providing incorrect information or failing to follow proper protocols.

.

Enterprise-Level KPIs:

- Service Level: The percentage of calls answered within a predetermined time frame, reflecting the overall efficiency and responsiveness of the call center.
- Problem Bucketing at which problem time how much time is getting spent
- Customer Effort Score (CES): Assesses the ease with which customers can get their issues resolved, indicating the overall effectiveness of the service channels.
- Contact Quality: A holistic measure of the effectiveness and efficiency of customer interactions, often assessed through quality assurance evaluations.
- Agent Satisfaction Score: Measures the job satisfaction levels among agents, which can impact overall performance and customer satisfaction.
- Customer Retention Rate: The percentage of customers who remain with the company after interacting with the call center, indicative of overall service effectiveness and customer loyalty.
- Compliance Adherence Rate: Measures how well the call center adheres to industry regulations and company policies, critical for maintaining legal and ethical standards.

Output Acceptance Criteria & Submission Instructions:

Candidates must submit their project deliverables as follows:

- GitHub Repository Link: Submit the complete codebase as a public GitHub repository. Ensure the repository contains:
- Well-commented source code.
- A detailed README file outlining the project setup, dependencies, and execution instructions.
- Working Application Link: Provide a URL where the working application is hosted. Platforms like Streamlit, Heroku, or Replit are recommended for easy deployment and accessibility.
- Documentation (Word Docx): Submit a detailed report or article explaining the methodology, technologies used, key findings, and any challenges overcome during the project. The document should be in Word (.docx) format.

Strict Compliance Notice:

- Originality: Each submission must be the original work of the participant. Immediate disqualification will occur if a candidate is found using replicated or AI-generated code.
- Plagiarism Check: All code will be subjected to similarity and plagiarism checks. Submissions found to be overly similar to existing solutions or flagged by code-plagiarism checkers will result in immediate disqualification from further round

Evaluation Criteria (no max points limit)

- Code Submission (5 Points): Basic requirement for a passing submission.
- Code + Working Link (10 Points): A functional web app hosted on platforms like Streamlit or Replit or Vercel
- Code + Working Link + Unique Article (12 Points): Above deliverables plus a unique article explaining the project.
- Extra Points (3 Points): Awarded for accuracy, code cleanliness, and overall quality of the submission.
- Accurate KPI Calculation: 1 point will be awarded for each accurately calculated KPI. With 14 KPIs specified for both call-level and enterprise-level, this can

significantly impact the total score, encouraging thoroughness and precision in the application's analytical capabilities