

Assignment: AI Solution for Audio Audit of Customer Service Calls

Assignment Overview

You are tasked with developing an AI solution that can audit customer service calls by analyzing audio files against specific criteria related to customer service performance. This project will integrate speech recognition and LLM to assess the quality of interactions between agents and customers. The final program should provide insights into key elements of customer service and generate a detailed audit report.

Objectives

By the end of this assignment, you should have developed an AI-based solution that can:

1. **Transcribe audio files** into text using a speech-to-text model or API.
 2. **Analyze the transcription** using LLM prompts to check compliance with predefined audit criteria.
 3. **Generate a report** summarizing the findings, providing audit justifications, and ensuring that records are usable for future verification.
 4. This is an individual assignment
-

Deliverables

You are expected to submit the following components as part of the assignment:

1. **Audio Transcription Module:**
 - Develop a module that converts audio recordings into accurate text using a speech-to-text service (e.g Openai Whisper Speech to text).
2. **Audio Transcription Word Error Rate:**
 - Develop a function to calculate the Word Error Rate of the Transcription
3. **Audit Transcription Module:**
 - Implement an LLM prompt to analyze the transcription and evaluate it against the given customer service criteria.
4. **Reporting Module:**
 - Once you've completed the transcription and analysis, your final task is to generate an **audit report** using the software you developed. The report should include the following for each criteria:

Audit Criteria: List and explain the criteria used for the audit.

Audit Reason: Provide the audit reason for giving the conclusion to the criteria.

Result: Give the result in term of Pass, Fail or Not Applicable.

Use the provided python notebook template to fill in the deliveries.

Audio Transcription(Speech To Text)

You will be given the following audio files to perform the Speech To Text Transcription:

```
audio_file="Custom-Home-Builder.mp3"  
audio_file="Inbound-sales-audio-sample.mp3"  
audio_file = "Local-Plumber.mp3"  
audio_file = "Property-Management-Office.mp3"  
audio_file = "Real-State-Lead-Gen-1.mp3"  
audio_file = "Travel-Reservation.mp3"
```

Transcript Word Error Rate

You will be given the ground truth audio transcription files for each of the audio files.

Use the ground true audio transcription files and your Audio transcription file to calculate the Word Error Rate for each case.

To standardize the calculation of the Word Error Rate you need to use this python module.

<https://pypi.org/project/jiwer/>

You are only required to do the measure for the word error rate(WER)

Example:

```
from jiwer import wer

reference = "hello world"

hypothesis = "hello duck"

error = wer(reference, hypothesis)
```

In the calculation, propose solution to improve you calculation of the WER.

Audit Transcription

Audit Criteria for transcription

Your solution should evaluate the customer service agent interaction with the customer in the conversation based on the following criteria:

1. **Introduction:**
 - To check the customer service agent conducts introductory before conversation with the customer.
2. **Acquire Customer Information:**
 - It is important before we engage the customer, customer service agents need to acquire customer information.
3. **Politeness and Respect:**
 - To check customer service agent is polite and respectful when interacting with customers.
4. **Empathy and Understanding:**
 - To check customer service agent engagement skill during customer enquires or request.
5. **Gratitude:**
 - It is important that the customer service agent shows gratitude to the customer when the customer shows interest in the services/products mentioned.
6. **Provide Conclusion from the customer request:**
 - To check if the customer service agent summarizes the customer request. This act as the feedback to show the agent understands what was mentioned by the customer.
7. **Clarifying Questions:**

- When encounter situations where the customer did not share clear requirements, the customer service agent needs to take initiative to ask questions.
- 8. **Clarity of Language:**
 - Check agent's language clear, concise, and easy to understand.
- 9. **Relevance of Information:**
 - To check agent does provide relevant information to the customer's request.

Report Generation

Once you've completed the transcription and analysis, your final task is to generate an **audit report** using the software you developed. The report should include the following for each criteria:

1. **Audit Criteria:** List and explain the criteria used for the audit.
2. **Audit Reason:** Provide the audit reason for giving the conclusion to the criteria.
3. **Result:** Give the result in term of Pass, Fail or Not Applicable.

Assignment Rubric

Your submission will be graded according to the following rubric:

1. Transcript Assessment: Word Error Rate (WER) - 15 Marks

Criteria	Excellent (>12.5-15.0)	Good (>10.0-12.5)	Satisfactory (>7.5-10.0)	Need Improvement (0.0-7.5)
Accuracy of Transcription	Very low Word Error Rate (WER) ($\leq 10\%$) with clear and accurate transcription.	Moderate WER (10-25%), mostly accurate transcription with few mistakes.	High WER (25-40%) with multiple errors but some comprehensibility.	Very high WER ($> 40\%$), transcription is inaccurate and difficult to understand.
WER Calculation Improvement methods	WER is calculated with good accuracy and good improvement method.	WER is calculated with some accuracy and with simple improvement method.	WER calculation is partially incorrect. No improvement method.	WER is not calculated or significantly incorrect. No improvement method.

2. Audit Result and Software Usability - 20 Marks

Criteria	Excellent (>17.5-20.0)	Good (>15.0-17.5)	Satisfactory (>10.0-15.0)	Need Improvement (0.0-10.0)
Criteria Justification	Fully meets the audit criteria, providing a clear and detailed justification for each audit result.	Meets most criteria, with clear but less detailed justifications.	Partially meets the audit criteria, some justifications unclear or incomplete.	Fails to meet the audit criteria or no clear justifications provided.
Audit Result Usability for Verification	Records from the audit are easily usable for verification, with clear structure and labeling.	Audit results are usable but could benefit from clearer structure or more detail.	Some audit results are usable, but overall lacks structure or clarity.	Audit results are poorly organized and not usable for verification.
Audit Process Clarity	The audit process is clearly explained with detailed steps.	The audit process is explained, but some steps may be unclear.	The audit process is only partially explained or missing important details.	The audit process is unclear or not explained at all.

3. Python Code: Organization and Documentation - 5 Marks

Criteria	Excellent (>4.5 – 5.0)	Good (>3.5-4.5)	Satisfactory (>2.5 – 3.5)	Need Improvement (0.0 – 2.5)
Code Organization	Code is well-organized, modular, and easy to follow. Follows best practices.	Code is organized but may have some areas that could be cleaner.	Code structure is somewhat confusing and could be significantly improved.	Code is disorganized and hard to understand.
Comments and Documentation	Code is fully commented, and documentation is comprehensive and clear.	Code is adequately commented but lacks detailed documentation.	Code has minimal comments or is poorly documented.	No comments or documentation are provided.

4. Python Code: Review - 5 Marks

This will be conducted after the code are submitted.

Criteria	Excellent (>4.5 – 5.0)	Good (>3.5-4.5)	Satisfactory (>2.5 – 3.5)	Need Improvement (0.0 – 2.5)
Clarity of code illustration	Code structure, logic, and key decisions are explained clearly, concisely, and logically.	Explanation is mostly clear with minor lapses in organization or detail.	Explanation covers basic points but lacks depth or has notable gaps in clarity.	Explanation is disorganized, confusing, or lacks a clear focus.
Understanding of Code	Demonstrates deep understanding of the code, including design choices, trade-offs, and performance implications.	Shows good understanding with minor gaps in trade-offs, optimizations, or alternative approaches.	Basic understanding; struggles to explain rationale for decisions or complex components.	Limited understanding of the code; cannot justify decisions or optimizations.

Additional Guidelines

- Your code should be written in **Python**, and you may use libraries to assist in building the solution.
- Make sure your code is well-structured and adheres to Python's best practices (e.g., following PEP 8 standards).
- Test your solution thoroughly with various audio samples to ensure it meets the requirements of the assignment.

Submission Format

- **Code:** Submit your Python code files with clear documentation.
- **Report:** Include the audit report generated by your software as part of the submission.

Use the provided python notebook template to fill in the deliveries.
Ensure that you print the result from the code for marking.

Deadline

Please submit your assignment by **9 Feb 2025 23:59hrs.**

Total Mark

Total marks: **45marks**