SymTrain Simulation Intelligence Assistant

Industry Partner: SymTrain
Tech Stack: Python, Streamlit, Docker, Optional Vision Detection
Team Size: Students may work in teams of up to 4 students.
Submit your work to GitHub Classroom: https://classroom.github.com/a/nc2fjNwA

1. Project Overview

The primary goal of this project is to build a fully automated customer assistance pipeline. Given a customer request, your system should generate the steps needed to help the customer complete a task, such as updating a payment method, filing an insurance claim, or booking a flight. Your workflow will involve extracting information from training simulations, categorizing requests, designing a few-shot learning pipeline, and presenting the result through an interactive Streamlit app.

You will work with a real dataset provided by SymTrain. Inside the zip folder you will find simulations from multiple companies. Each company has its own subfolder and each subfolder contains:
1. A JSON conversation file representing the customer and agent dialogue.
2. A set of web UI image slices used in the simulation.

2. Where to Find Transcript Data

All transcript information is stored in the JSON file under audioContentItems. Each item contains sequenceNumber, actor, and fileTranscript.

You may merge transcript lines into one continuous string such as:
Trainee: How may I help you? SYM: I need to update my payment method. Trainee: Here is what you need to do…

3. Where to Find Image File IDs for Bonus

visualContentItems contains fileId values and hotspots describing buttons, text fields, and highlight regions. Use these only for the bonus.

4. Tasks

Task 1: Load the dataset (5 points)
Load each JSON file and extract audioContentItems.

Task 2: Merge dialogue text (5 points)
Merge transcript lines into one continuous string while preserving speaker roles.

Task 3: Extract call reasons and steps (20 points)
Identify the reason and extract rewritten steps from agent turns. Attempt using a transformer and compare with GPT.

Task 4: Categorize all simulations (10 points)
Create meaningful categories and assign each simulation. Attempt categorization using a transformer and compare with GPT.

Task 5: Generate steps for test data using GPT few shot learning (25 points)
Few shot examples must come from the same category. Identify category, retrieve examples, and generate steps. Attempt transformer comparison.

Test inputs for evaluation:
test_1: Hi, I ordered a shirt last week and paid with my American Express card. I need to update the payment method because there is an issue with that card. Can you help me?
test_2: Hi, I need to update the payment method for one of my recent orders. Can you help me with that?
test_3: Hi, I am Sam. I was in a car accident this morning and need to file an insurance claim. Can you help me?
test_4: Hi, can you help me file a claim?
test_5: Hi, I recently ordered a book online. Can you give me an update on the order status?
test_6: Hi, I have been waiting for two weeks for the book I ordered. What is going on with it? Can you give me an update?

LLM output must be JSON formatted as follows:

```
{
  "category": "",
  "reason": "",
  "steps": []
}
```

Task 6: Streamlit application (10 points)
Build an app that accepts user input, predicts category, and generates reason and steps.

Task 7: Packaging and Dockerization (10 points)
Provide a structured Python package and a working Dockerfile.

Task 8: Slides and recorded presentation (5 points)
Slides and presentation must include a walkthrough of your design, Streamlit screenshots, real-time demo of all six test inputs, and a few GPT vs transformer examples.

5. Bonus Task: Vision based step to image mapping (up to 10 points)

Use fileId values to identify images. For each generated step, find the most relevant image and highlight the correct UI element. Include results in slides.

6. Submission Requirements

Only one student per team needs to submit to GitHub Classroom.

The submitted repository must include:
1. Full code repository
2. Dockerfile
3. Slides
4. Recorded presentation
5. Bonus work if completed

Deadline: December 12