Assignment: Tools and Platforms for Generative AI and Prompt Engineering

Objective:

The objective of this assignment is to explore and experiment with various generative AI tools and platforms such as OpenAI Playground, Hugging Face, LangChain, and custom APIs. Students will gain hands-on experience building and automating prompt workflows, integrating generative AI models into larger systems, and fine-tuning outputs using APIs.

Scenario:

You are tasked with building and experimenting with generative AI tools to create real-time applications that leverage advanced prompt engineering. This project involves working with different platforms and APIs to fine-tune your prompts, automate workflows, and integrate generative AI models into functioning systems like chatbots and content generation tools.

Requirements:

1. **Exploring and Experimenting with OpenAI Playground:**
   - **Scenario**: You need to design a prompt in OpenAI Playground that generates detailed content for a marketing campaign. The goal is to create a prompt that generates product descriptions, taglines, and promotional text for a new product launch.
   - **Task**: Using OpenAI Playground, experiment with different prompt designs and parameters (e.g., temperature, max tokens) to generate engaging and relevant content for a product.
     - **Example Prompt**: "Generate a catchy tagline, product description, and promotional content for a new eco-friendly coffee mug designed for environmentally-conscious consumers. Focus on sustainability, design, and user benefits."
   - **Deliverables**: Submit the generated content, along with an explanation (150-200 words) of how you used the Playground's features to improve the quality of the outputs.
2. **Using Hugging Face for Fine-Tuning Models:**
   - **Scenario**: You are working on a project to develop a chatbot that can answer questions about computer science topics. You need to fine-tune a model using Hugging Face's Transformers library.

- **Task**: Explore Hugging Face and experiment with fine-tuning a pre-trained model for the task of answering technical questions on computer science concepts. Start with a general language model and fine-tune it with specific data (e.g., questions and answers related to algorithms).
    - **Example Prompt**: "Answer the following question about sorting algorithms: What is the difference between bubble sort and merge sort?"
- **Deliverables**: Submit your fine-tuned model results, including the code used for fine-tuning (if applicable) and a brief explanation (150-200 words) of how fine-tuning the model impacted the chatbot's responses.

3. **Exploring LangChain for Automating Prompt Workflows:**
    - **Scenario**: You want to automate a workflow that integrates a generative AI model into a multi-step process, such as extracting key data from documents and generating summaries.
    - **Task**: Using LangChain, design a multi-step workflow where an AI model reads a document, extracts key pieces of information, and generates a summary of the document. Use LangChain's tools to automate the integration of these tasks.
        - **Example Workflow**: "Create a prompt that reads a technical paper on machine learning, extracts key concepts, and then generates a concise summary of the paper's findings."
    - **Deliverables**: Submit a description of the workflow you built, including the code and a brief explanation (150-200 words) of how LangChain facilitated the automation and integration of tasks.
4. **Building a Real-Time Application Using Generative AI Tools:**
    - **Scenario**: You are tasked with building a real-time chatbot for customer service that generates responses based on user inquiries about product availability and shipping details. The chatbot should provide answers dynamically and automatically using generative AI.
    - **Task**: Use platforms like OpenAI, Hugging Face, or custom APIs to build the backend for this chatbot. Focus on integrating real-time data, such as current product stock levels or shipping times, and generating responses based on these inputs.

        - **Example Prompt**: “What is the current status of the ‘Eco-Friendly Coffee Mug’ product in stock, and when will it be shipped to the customer?”
    - **Deliverables**: Submit a working prototype of the chatbot or a detailed demonstration of the chatbot’s backend integration. Include any code used for generating responses and a 200-300 word explanation of how you integrated real-time data into the generative AI model.

5. **Interactive Demo: Using Generative AI Tools for Content Generation:**
    - **Scenario**: You are tasked with designing a content generation tool that creates blog posts on various topics. This tool should be able to generate full-length articles from a single user-provided title or subject.
    - **Task**: Build a demo of a content generation platform using generative AI tools. The system should take a topic (e.g., "AI in Healthcare") and generate a coherent, structured blog post that includes an introduction, body, and conclusion.
        - **Example Prompt**: “Generate a 1000-word blog post on the topic ‘The Impact of AI on Healthcare,’ including an introduction, overview of AI applications in healthcare, challenges, and future outlook.”
    - **Deliverables**: Submit a working demo or video of your content generation tool in action, along with a 150-200 word explanation of how you structured the prompt and ensured the output met the content requirements.

6. **Integrating Custom APIs for Fine-Tuning Outputs:**
    - **Scenario**: You need to create a system that automatically adjusts the tone of generative AI outputs based on user preferences (e.g., formal vs. casual tone). This system should use an API to adjust the tone based on user inputs.
    - **Task**: Design a custom API that can modify the tone of responses generated by a model. The API should take in a prompt, generate a response, and then modify the tone based on user-specified parameters.
        - **Example Prompt**: “Write an email response to a customer asking about product shipping details, with a formal tone.”
        - **API Usage**: Adjust the tone of the generated response to be more casual or friendly based on the API input.

    - **Deliverables**: Submit the API code or a working prototype, and a 150-200 word explanation of how you integrated custom APIs to fine-tune generative outputs.

Deliverables:

- **Individual Submission**:
    - Generated content from OpenAI Playground with an explanation of the parameters used.
    - Results from Hugging Face fine-tuning, including any code and fine-tuning strategies.
    - A description of the LangChain workflow, with code and an explanation.
    - A working prototype or detailed demonstration of the real-time chatbot, including backend integration of generative AI tools.
    - A video or demo of the content generation tool in action, along with an explanation of how the prompt was structured.
    - Code for the custom API for tone adjustment, with an explanation of how it fine-tunes the outputs.

Knowledge Graph Metadata

Main Topic Details

- generative AI models

  Related Terms: experience building automating, impacted chatbot responses, integrate generative, sorting algorithms difference, data questions answers, bubble, tone description langchain, tuning outputs using, prompt current, reads document extracts

- different prompt designs

  Related Terms: design prompt openai, develop chatbot answer, tools platforms generative, content marketing campaign, tagline product, topic impact ai, tokens generate, playground features improve, structured blog post, experiment various

- tone adjustment

  Related Terms: api tone adjustment, integrated custom apis, tune model using, model api prompt, generate response modify, ai outputs, create automatically adjusts, transformers library integrating, deliverables submit api, face fine

Concept Details

- Generative Ai Tools

- Example Prompt

· Generative Ai Models

· Real Time Data

· Fine Tune

· The Code

· A Brief Explanation

· Generative Ai Tools

· Task

· An Explanation

· Example Prompt

· Custom Apis

· A Prompt

· An Introduction

· The Tone

· A Model

Entity Details

· LangChain

· Deliverables

· Playground

· OpenAI Playground

· OpenAI

· Hugging Face

· Healthcare

· API