Project Title	**Deployment GUVI GPT Model using Hugging Face**
Skills take away From This Project	**Deep Learning,Transformers,Hugging face models,LLM, Streamlit or Gradio**
Domain	**AIOPS**

Problem Statement:

The task is to deploy a fine-tuned GPT model, trained specifically on GUVI's company data, using Hugging Face services. Students are required to create a scalable and secure web application using Streamlit or Gradio, making the model accessible to users over the internet. The deployment should leverage Hugging Face spaces resources and any database to store the username and login time.

Objective:
To deploy a pre-trained or Fine tuned GPT model using HUGGING FACE SPACES, making it accessible through a web application built with Streamlit or Gradio.

Business Use Cases:

1. **Customer Support Automation**:
    - **Scenario**: Integrate the fine-tuned GPT model with GUVI's customer support system to automate responses to frequently asked questions, reducing the workload on support staff and improving response times.
    - **Application**: The model can handle initial customer inquiries, provide information on courses, pricing, and enrollment procedures, and escalate complex issues to human agents when necessary.
2. **Content Generation for Marketing**:
    - **Scenario**: Use the model to generate marketing content, such as blog posts, social media updates, and email newsletters, tailored specifically to GUVI's audience.
    - **Application**: The marketing team can input topics or keywords into the web application, and the model will generate relevant, high-quality content that can be edited and published.
3. **Educational Assistance for Students**:
    - **Scenario**: Implement the model as a virtual teaching assistant within GUVI's educational platform to help students with their queries and provide explanations on various topics.
    - **Application**: Students can interact with the virtual assistant through the web application to get immediate answers to their questions, clarifications on course material, and personalized study recommendations.

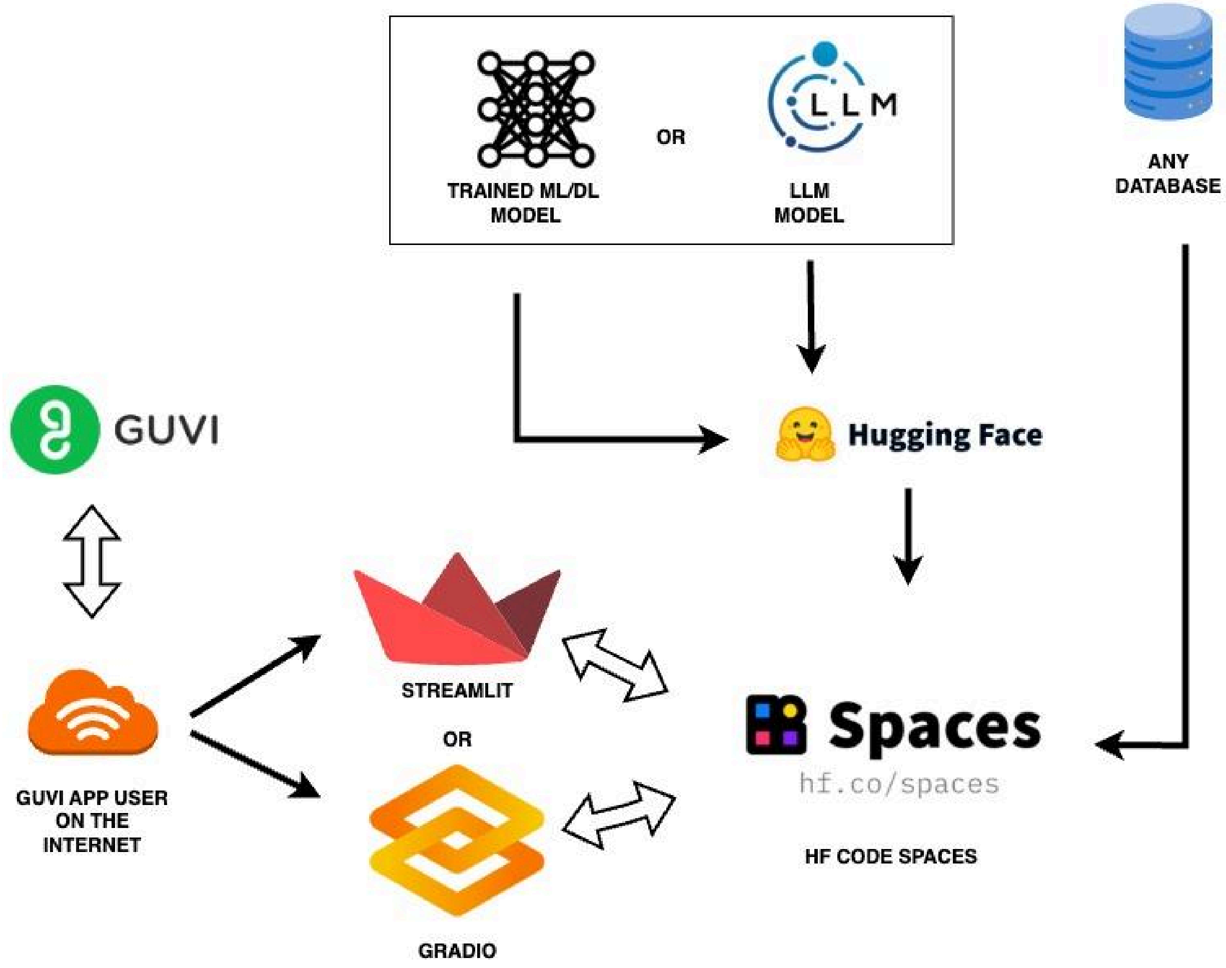
4. **Internal Knowledge Base**:

- **Scenario**: Develop an internal knowledge base tool for GUVI employees, enabling them to quickly access company-related information and resources.
- **Application**: Employees can use the web application to query the fine-tuned GPT model for information on company policies, procedures, and other internal documents, improving efficiency and knowledge sharing within the organization.

5. **Training and Onboarding**:

- **Scenario**: Assist in the training and onboarding process of new employees by providing instant access to training materials and answering common questions about the company.
- **Application**: New hires can interact with the web application to learn about GUVI's mission, values, and operations, making the onboarding process smoother and more engaging.

Approach:



1. **Data Preparation**:
    - Store the app.py file and any additional necessary files in an Amazon S3 bucket.
2. **Infrastructure Setup**:
    - Launch an Amazon EC2 instance with appropriate IAM roles and security groups.
    - Ensure the EC2 instance has internet access via an Internet Gateway.
3. **Environment Configuration**:
    - Install required packages on the EC2 instance (e.g., Streamlit, Boto3, transformers, torch).
    - Download the app.py file from S3.
4. **Application Deployment**:
    - Run the Streamlined application on the EC2 instance.
    - Optionally use ngrok for temporary public access during testing.
5. **Security Configuration**:
    - Configure a security group to allow inbound traffic on the port used by the Streamlit app (default: 8501).

General Guidelines

- **Small Scale (Tens of Thousands of Tokens)**:
    - Fine-tuning with around 10,000 to 50,000 tokens can be sufficient for simple tasks or when only minor adjustments to the model's behavior are needed.
    - Suitable for highly specialized tasks with very specific types of content.
- **Medium Scale (Hundreds of Thousands of Tokens)**:
    - Fine-tuning with around 100,000 to 500,000 tokens can provide more substantial adjustments and is often sufficient for many practical applications.
    - Useful for moderately complex content where the model needs to learn specific patterns and vocabulary.
- **Large Scale (Millions of Tokens)**:
    - Fine-tuning with 1 million or more tokens can yield significant improvements, especially for complex or diverse content.
    - Necessary for highly complex tasks or when the model needs to generate very accurate and contextually rich text.

Example for GUVI’s Company Data

Given that GUVI’s company data might include a variety of text types (such as FAQs, blog posts, course descriptions, and internal documents), a medium-scale dataset might be appropriate:

- **Medium Scale**: Aim for around 100,000 to 500,000 tokens. This could translate to approximately 50,000 to 200,000 words, depending on the length and complexity of the text.
    - Collect a diverse set of documents covering different aspects of GUVI’s operations and services.
    - Ensure the data includes examples of typical user queries and responses, marketing content, educational materials, and internal communications.

Steps to Fine-Tune the Model

1. **Data Collection or Extraction**:
    - Gather text data from various sources within GUVI, such as website content, user queries,social media, blog posts, and training materials.
2. **Data Preparation**:
    - Clean and preprocess the text data, ensuring it is in a format suitable for training (e.g., removing special characters, normalizing text).
3. **Tokenization**:
    - Use the GPT-2 tokenizer to convert the text data into tokens. Ensure the data is tokenized consistently to match the pre-trained model’s requirements.
4. **Fine-Tuning**:
    - Use the Hugging Face Transformers library or similar tools to fine-tune the GPT-2 model on the prepared dataset.
    - Monitor the training process to prevent overfitting and ensure the model generalizes well to new data

Results:

- **Functional Web Application**: A fully functional web application that users can access to interact with the pre-trained GPT model.
- **Scalable Deployment**: A scalable deployment framework using Hugging Face services services.
- **Documentation**: Comprehensive documentation outlining setup, deployment, and usage instructions.

Project Evaluation metrics:

- **Functionality**: The application should correctly load the pre-trained model and generate coherent text responses based on user input.
- **Performance**: The application should respond to user queries within an acceptable time frame.
- **Scalability**: The setup should be able to handle multiple users concurrently.
- **Security**: Proper security measures should be in place, including the use of security groups and IAM roles.
- **Usability**: The web interface should be user-friendly and intuitive.

Technical Tags:

List any technical tags or keywords relevant to the project.

Data Set:

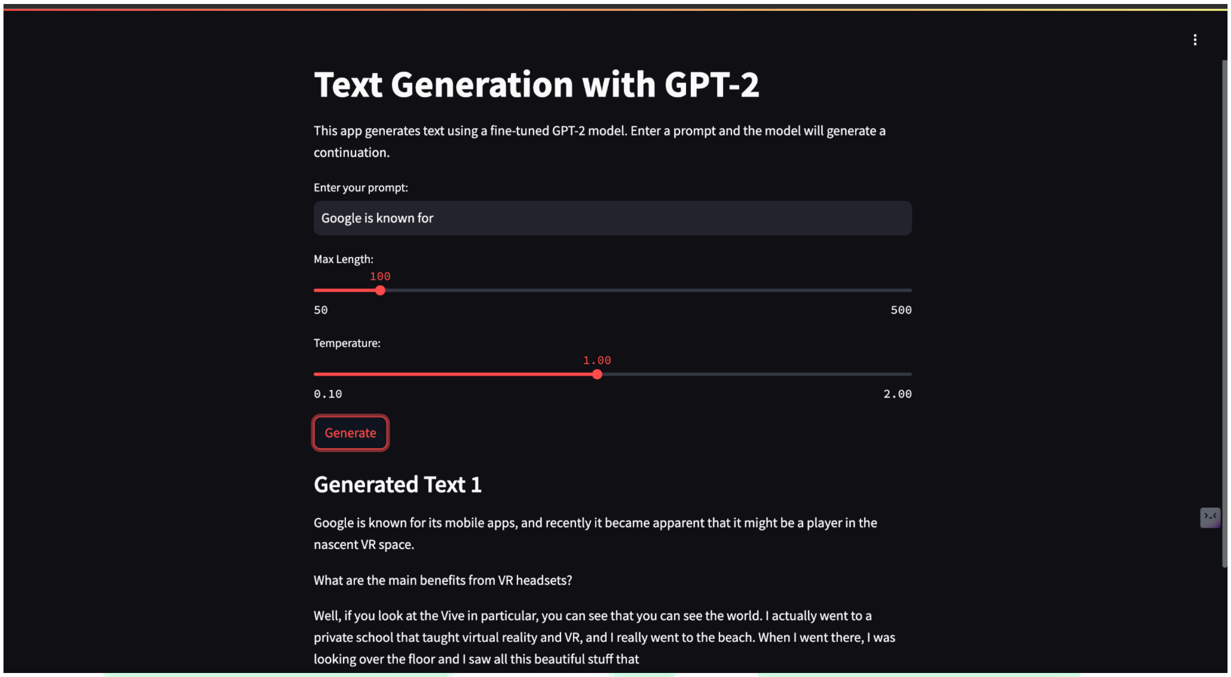
Data Collection or Extraction:

Gather text data from various sources within GUVI, such as website content, user queries,social media, blog posts, and training materials.

Project Deliverables:

- **Source Code**: Complete source code for the Streamlit application.
- **Documentation**: Detailed documentation including setup instructions, usage guide, and explanation of the architecture.
- **Deployment Scripts**: Scripts used for setting up the environment and deploying the application on Hugging Face services.
- **Project Report**: A report summarizing the project, approach taken, and results achieved.(optional)

Expected app simulation (can vary depending upon the developer's creativity)



Text Generation with GPT-2

This app generates text using a fine-tuned GPT-2 model. Enter a prompt and the model will generate a continuation.

Enter your prompt:

Google is known for

Max Length:

100

50 500

Temperature:

1.00

0.10 2.00

Generated Text 1

Google is known for its mobile apps, and recently it became apparent that it might be a player in the nascent VR space.

What are the main benefits from VR headsets?

Well, if you look at the Vive in particular, you can see that you can see the world. I actually went to a private school that taught virtual reality and VR, and I really went to the beach. When I went there, I was looking over the floor and I saw all this beautiful stuff that

Project Guidelines:

- **Coding Standards**: Follow PEP 8 coding standards for Python.
- **Version Control**: Use Git for version control. Regular commits with clear messages are expected.
- **Documentation**: Ensure all code is well-documented. Include comments and docstrings where necessary.
- **Testing**: Write and include unit tests to verify the functionality of individual components.
- **Resource Management**: Monitor and manage Hugging Face services resources to avoid unnecessary costs. Ensure proper shutdown of resources when not in use.

Timeline:

14 days from the date of document issuance.

PROJECT DOUBT CLARIFICATION SESSION (PROJECT AND CLASS DOUBTS)

About Session: The Project Doubt Clarification Session is a helpful resource for resolving questions and concerns about projects and class topics. It provides support in understanding project requirements, addressing code issues, and clarifying class concepts. The session aims to enhance comprehension and provide guidance to overcome challenges effectively.
Note: Book the slot at least before 12:00 Pm on the same day

Timing: Tuesday, Thursday, Saturday (5:00PM to 7:00PM)

Booking link :https://forms.gle/XC553oSbMJ2Gcfug9

LIVE EVALUATION SESSION (CAPSTONE AND FINAL PROJECT)

About Session: The Live Evaluation Session for Capstone and Final Projects allows participants to showcase their projects and receive real-time feedback for improvement. It assesses project quality and provides an opportunity for discussion and evaluation.
Note: This form will Open on Saturday and Sunday Only on Every Week

Timing: Monday-Saturday (11:30PM to 12:30PM)

Booking link : https://forms.gle/1m2Gsro41fLtZurRA