AI Memory Diary Generator
Submitted to YBI Foundation

Submitted by: **Abhishek Ashok Khadapkar**
Course: AI & Generative AI Training Program

Contents

1.   Introduction

The project **AI Memory Diary Generator** is an interactive application that uses Artificial Intelligence and Generative AI to analyze a user's image and mood description to create a personalized diary entry. It combines facial emotion recognition, sentiment analysis, and GPT-2 text generation.

2.   Objectives

- Detect emotion from a user-uploaded face image

- Analyze mood from text input

- Generate a diary entry using GPT-2

- Provide a simple web interface using Gradio

3.   Tools and Libraries Used

- **Google Colab** – Cloud environment

- **Gradio** – For creating UI

- **DeepFace** – Emotion detection

- **Transformers (GPT-2)** – Text generation

- **TextBlob** – Sentiment analysis

- **Python** – Programming language

4.   Project Workflow

1. User uploads image via Colab

2. User writes a mood description

3. Image is analyzed for emotion (DeepFace)

4. Text is analyzed for sentiment (TextBlob)

5. GPT-2 generates the diary entry

6. Output is shown using Gradio

5. Code Explanation

5.1. Install Required Libraries

```
!pip install gradio deepface transformers textblob
```

5.2. Import Libraries

```
import gradio as gr
from deepface import DeepFace
from transformers import pipeline
from textblob import TextBlob
from google.colab import files
```

5.3. Load GPT-2 Model

```
diary_generator = pipeline("text-generation", model="gpt2")
```

5.4. Detect Emotion from Image

```
def detect_emotion_from_image(image_path):
    try:
        result = DeepFace.analyze(
            img_path=image_path, actions=['emotion'],
                ↪ enforce_detection=False)
        emotion = result[0]['dominant_emotion']
        return emotion.capitalize()
    except Exception:
        return "Could not detect"
```

5.5. Detect Sentiment from Text

```
def detect_emotion_from_text(text):
    blob = TextBlob(text)
    polarity = blob.sentiment.polarity
    if polarity > 0.2:
        return "Positive"
    elif polarity < -0.2:
        return "Negative"
    else:
        return "Neutral"
```

5.6. Upload Image via Colab

```
uploaded_file = files.upload()
image_path = list(uploaded_file.keys())[0]
```

5.7. Generate Diary Entry

```
def generate_diary(text_input):
    image_emotion = detect_emotion_from_image(image_path)
    text_emotion = detect_emotion_from_text(text_input)

    prompt = (f"My day was {image_emotion.lower()} and I felt "
             f"{text_emotion.lower()}. {text_input}. Here's my "
                 ↪ diary:")

    generated = diary_generator(prompt, max_length=100,
        ↪ num_return_sequences=1)
    diary = generated[0]['generated_text']

    result_text = (f"Emotion from Image: {image_emotion}\n"
                   f"Emotion from Text: {text_emotion}\n\n"
                   f"Diary Entry:\n{diary}")

    return image_path, result_text
```

5.8. Launch Gradio UI

```
iface = gr.Interface(
    fn=generate_diary,
    inputs=gr.Textbox(lines=4, placeholder="How are you feeling 
        ↪ today?", label="Describe Your Mood"),
    outputs=[
        gr.Image(label="Uploaded Image"),
        gr.Textbox(label="AI Generated Diary Entry")
    ],
    title="AI Memory Diary Generator",
    description="Upload an image, describe your mood, and generate a
        ↪  diary entry."
)
iface.launch()
```

6.   Output
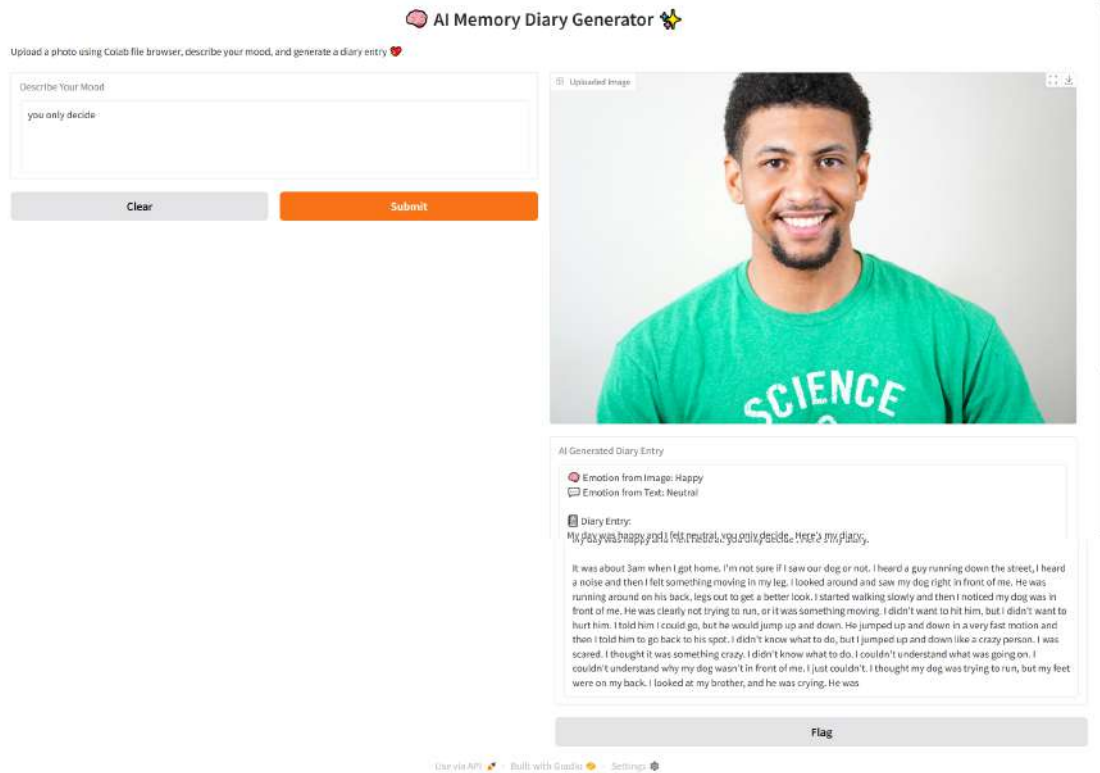

Figure 1: User uploaded image and mood description

Figure 2: Generated diary entry based on inputs

7. Project Notebook Link

The complete implementation and code execution of this project can be accessed through the following Google Colab link:

- **Click here to open the Colab Notebook**

8. Conclusion

This project uses AI to turn mood and face input into a creative diary entry. It shows how emotion detection and text generation can come together for mental wellness and journaling support.

9. Future Scope

- Add voice input instead of typing

- Export diary as PDF

- Track user mood over time

- Recommend music or quotes

10.   References

1. DeepFace GitHub

2. Transformers Documentation

3. Gradio Official

4. TextBlob Docs