# 🧠 MoodMate - Emotion Detection Journal (Hackathon Project)

This notebook allows users to input journal entries, detect emotions using a fine-tuned BERT model, and get personalized suggestions. It also logs entries to a CSV file.

```
# Step 1: Install dependencies
!pip install -q transformers gradio pandas
```

```
# Step 2: Import Libraries
from transformers import pipeline
import gradio as gr
import pandas as pd
from datetime import datetime
```

```
# Step 3: Load Emotion Classification Model
model_name = "nateraw/bert-base-uncased-emotion"
emotion_classifier = pipeline("text-classification", model=model_name, tokenizer=model_name)
```

```
/usr/local/lib/python3.11/dist-packages/huggingface_hub/utils/_auth.py:94: UserWarning:
The secret `HF_TOKEN` does not exist in your Colab secrets.
To authenticate with the Hugging Face Hub, create a token in your settings tab (https://huggingface.co/settings/tokens), set it as secret in you
You will be able to reuse this secret in all of your notebooks.
Please note that authentication is recommended but still optional to access public models or datasets.
  warnings.warn(
config.json: 100% 768/768 [00:00<00:00, 10.0kB/s]
pytorch_model.bin: 100% 438M/438M [00:06<00:00, 105MB/s]
model.safetensors: 100% 438M/438M [00:12<00:00, 60.0MB/s]
tokenizer_config.json: 100% 252/252 [00:00<00:00, 3.17kB/s]
vocab.txt: 232k/? [00:00<00:00, 1.20MB/s]
special_tokens_map.json: 100% 112/112 [00:00<00:00, 633B/s]
Device set to use cpu
```

```
# Step 4: Define CSV Logging Function
def log_to_csv(text, emotion, score):
    log = {
        "timestamp": datetime.now().strftime("%Y-%m-%d %H:%M:%S"),
        "entry": text,
        "emotion": emotion,
        "confidence": round(score * 100, 2)
    }
    df = pd.DataFrame([log])
    df.to_csv("mood_log.csv", mode='a', header=not pd.io.common.file_exists("mood_log.csv"), index=False)
```

```
# Step 5: Define Emotion Analysis Function
def analyze_emotion(text):
    if not text.strip():
        return "Please write something.", None, None

    result = emotion_classifier(text)[0]
    label = result["label"]
    score = result["score"]

    suggestions = {
        "joy": "Keep spreading joy! Write about what made you happy.",
        "sadness": "It’s okay to feel sad. Try expressing what’s bothering you.",
        "anger": "Take a deep breath. Write out your frustrations.",
        "fear": "You are safe. Consider talking to someone you trust.",
        "love": "That's beautiful. Reach out to someone you care about.",
        "surprise": "Interesting! Was it a good surprise or not? Reflect on it."
    }
    suggestion = suggestions.get(label, "Keep expressing your emotions.")

    log_to_csv(text, label, score)

    return f"Emotion: {label.title()} ❤️", f"Confidence: {round(score * 100, 2)}%", suggestion
```

```
# Step 6: Gradio UI
iface = gr.Interface(
    fn=analyze_emotion,
    inputs=gr.Textbox(lines=7, placeholder="Write your journal entry here...", label="Journal Entry"),
    outputs=[
        gr.Textbox(label="Detected Emotion"),
        gr.Textbox(label="Confidence"),
        gr.Textbox(label="Wellness Suggestion")
    ],
    title="🧠 MoodMate - Emotion-Aware Journal",
    description="Track your mood using NLP and get personalized wellness tips."
```

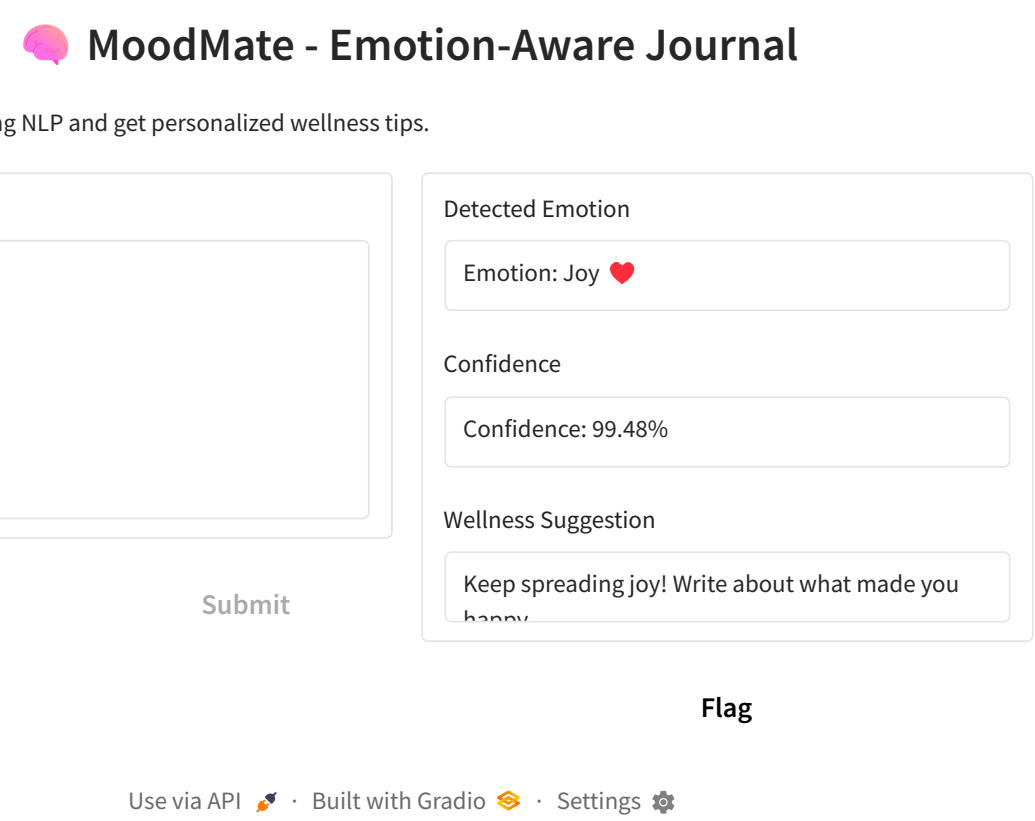
```
)

# Step 7: Launch UI
iface.launch()
```

```
It looks like you are running Gradio on a hosted a Jupyter notebook. For the Gradio app to work, sharing must be enabled. Automatically setting

Colab notebook detected. To show errors in colab notebook, set debug=True in launch()
* Running on public URL: https://93e128fe4818e10d48.gradio.live

This share link expires in 1 week. For free permanent hosting and GPU upgrades, run `gradio deploy` from the terminal in the working directory t
```




```
Start coding or generate with AI.
```