AI Chatbot Prototype with Emotion Detection

Muhammad Daniyal Qureshi
4excelerate.org
Excelerate
dannymail712@gmail.com

***Abstract*—This paper presents the design and implementation of an empathetic AI chatbot that detects and responds to user emotions using a pre-trained NLP model. The chatbot interface is built using Gradio and leverages the HuggingFace Transformers library for emotion classification.**

***Index Terms*—Empathetic AI, Chatbot, Emotion Detection, NLP, HuggingFace, Gradio**

I. Introduction

The goal of this project is to build a chatbot that simulates emotionally intelligent conversation by detecting user emotions and responding empathetically. Emotion-aware systems have growing importance in mental health, education, and customer support applications.

II. Objectives

- Detect emotions from natural language input
- Generate context-sensitive, empathetic replies
- Use open-source tools to ensure reproducibility
- Build an interactive and accessible web interface

III. Technologies Used

Component	Tool/Library
Language	Python 3.8+
Modeling Framework	HuggingFace Transformers
Interface	Gradio (Blocks API)
Emotion Model	distilbert-base-uncased-emotion

TABLE I
Technology Stack

IV. System Design

The system flow is as follows:

1) User inputs a message
2) The model detects the primary emotion
3) A predefined response is selected
4) Emotion label and reply are shown in chat

V. Emotion Classes

- Joy
- Sadness
- Anger
- Fear
- Surprise
- Love

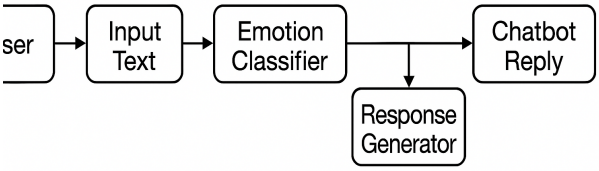

Fig. 1. System Architecture

VI. Implemented Features

- Emotion detection from raw text
- Rule-based empathetic responses
- Reset functionality
- Gradio-based interface
- Emotion label visualization

VII. Limitations and Future Work

Limitations:

- Only one emotion detected per input
- Cannot handle sarcasm or multiple tones
- Static (non-generative) replies

Future Improvements:

- Voice input/output integration
- Connect to live platforms (WhatsApp)
- Track long-term emotional trends
- Add LLM-based dynamic responses

VIII. Source Code

A. *chatbot.py*

```
from transformers import pipeline
import gradio as gr

emotion_classifier = pipeline("text-classification",
                              model="bhadresh-savani
    /distilbert-base-uncased-emotion")

def detect_emotion(text):
    result = emotion_classifier(text)[0]
    return result['label'], result['score']

```

```
def generate_response(user_input):
    emotion, _ = detect_emotion(user_input)
    responses = {
        "joy": "I'm glad to hear you're feeling happy!       ",
        "anger": "I can see you're upset. Let's talk about it.       ",
        "sadness": "I'm here for you.  I t s  okay to feel this way.       ",
        "fear": "That sounds scary. Want to share more?       ",
        "love": " T h a t s  heartwarming!       ",
        "surprise": "Wow!  T h a t s  unexpected. Tell me more!       "
    }
    reply = responses.get(emotion, " I m  listening ... Tell me more.")
    full_reply = f"{reply}\n\  n     Detected Emotion: *{emotion}*"
    return full_reply, emotion

def chatbot(user_input, history):
    if history is None:
        history = []
    reply, emotion = generate_response(user_input)
    history.append({"role": "user", "content": user_input})
    history.append({"role": "assistant", "content": reply})
    return history, history

def reset():
    return [], []

with gr.Blocks(title="Empathetic AI Chatbot") as demo:
    gr.Markdown("##        Empathetic AI Chatbot with Emotion Detection")
    chatbot_output = gr.Chatbot(label="Chat", type="messages")
    user_input = gr.Textbox(label="Your Message", placeholder="Type here...", lines=2)
    state = gr.State()
    with gr.Row():
        send_btn = gr.Button("Send")
        reset_btn = gr.Button("       Reset Chat")
    send_btn.click(fn=chatbot, inputs=[user_input, state], outputs=[chatbot_output, state])
    reset_btn.click(fn=reset, inputs=[], outputs=[chatbot_output, state])

demo.launch()
```

IX. Requirements File

requirements.txt:

```
transformers
torch
gradio
```

X. References

References

[1] Hugging Face Transformers. Available: https://huggingface.co/transformers/

[2] Gradio Documentation. Available: https://www.gradio.app/

[3] bhadresh-savani/distilbert-base-uncased-emotion. Available: https://huggingface.co/bhadresh-savani/distilbert-base-uncased-emotion