MindGuard – AI-Powered Mental Health Chatbot

1. Description

MindGuard is an AI-powered mental health companion designed to provide empathetic, supportive, and safe conversations for users experiencing stress, anxiety, or emotional difficulties. It leverages a fine-tuned LLaMA 3.2B model hosted on Hugging Face, using parameter-efficient fine-tuning (PEFT) techniques to ensure optimized performance and resource use. MindGuard is accessible via a clean Gradio interface that simulates a chatbot experience.

2. Tech Stack Used

- LLaMA 3.2B by Unsloth (base model)
- PEFT model: serenity-AI_Therapist
- Hugging Face Transformers & Tokenizers
- Gradio for user interface
- PyTorch for model execution
- Python for scripting

3. How It Works

1. The user enters a message into the Gradio-based chat interface.
2. The system builds a conversational prompt by appending the new message to the existing chat history.
3. The tokenizer encodes this prompt and sends it to the LLaMA model (with the PEFT layer).
4. The model generates a context-aware, supportive response.
5. The chatbot interface displays the response and updates the chat history for future turns.

4. Use Cases with Examples

- **Mental Health Check-in**:
  Example: 'I feel overwhelmed with college work.' → MindGuard offers encouragement and grounding techniques.

- **Emotional Support**:
  Example: 'I had a bad day at work.' → MindGuard responds empathetically and asks follow-up questions.

- **Burnout Detection and Conversation**:
  Example: 'I can’t sleep or focus anymore.' → MindGuard provides potential signs of burnout and self-care advice.

- **General Conversations**:
  Example: 'What’s something I can do to feel better?' → MindGuard gives healthy coping strategies.

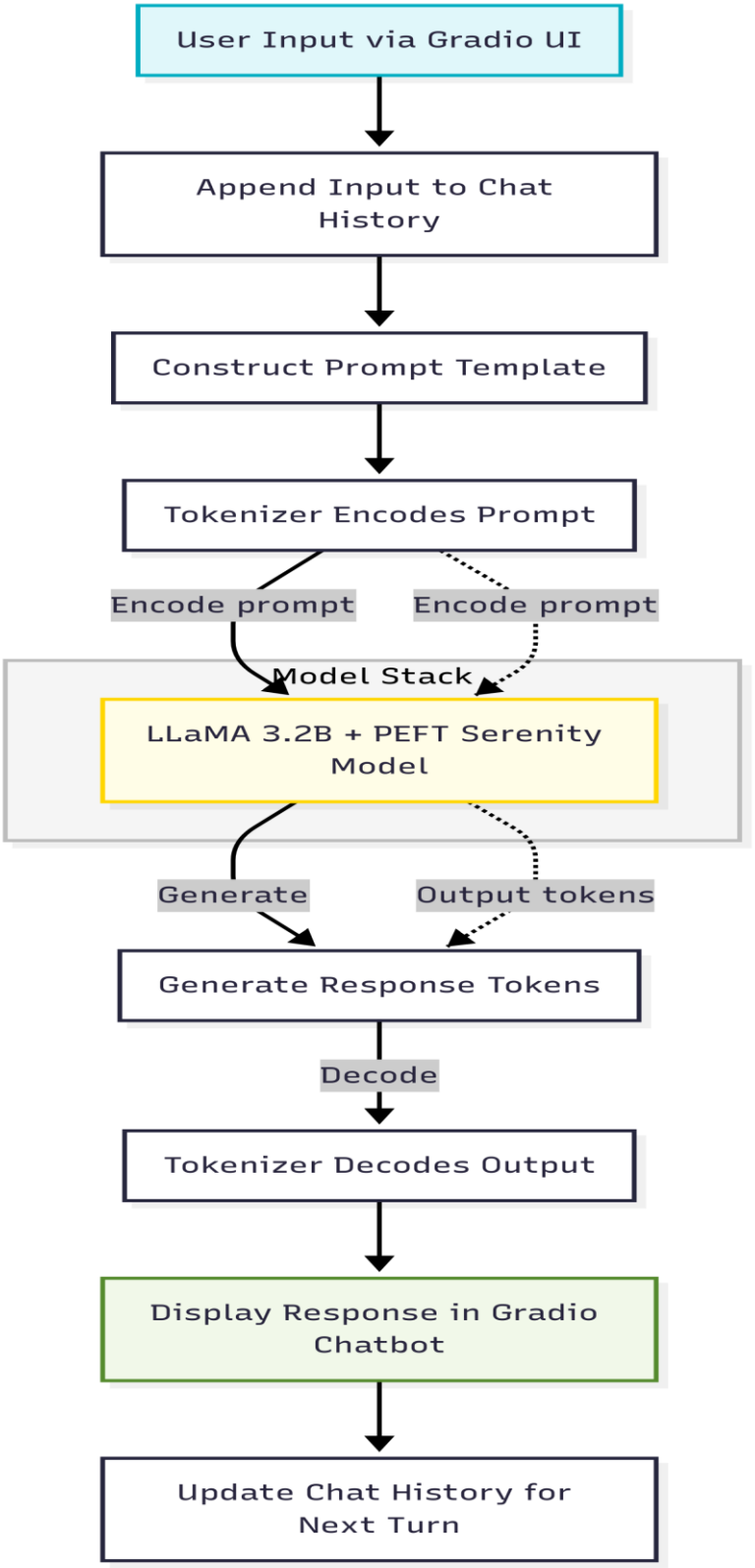
5. Future Scope

- Integration with voice input and TTS output for accessibility
- Incorporating context memory for longer conversations
- Real-time emotion detection and personalized responses
- Deployment as a mobile app
- Support for multilingual conversations
- Integration with professional help platforms or SOS directories

6. Workflow of MindGuard

1. **User Interaction:** Message is entered via the Gradio-based UI
2. **Session Handling:** Input is appended to the ongoing chat history
3. **Prompt Construction:** A formatted prompt is generated for the model
4. **Tokenization:** Prompt is encoded into model-compatible tokens
5. **Inference:** Tokens are processed by the LLaMA 3.2B model with Serenity PEFT
6. **Decoding:** Generated response tokens are converted back to human-readable text
7. **Response Delivery:** Output is shown in the chatbot interface
8. **History Update:** The conversation history is updated for contextual continuity

7. System Architecture



8. Screenshots