International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 06 | June - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

Mental Fitness Tracker

Ms.k.Sonali [1], P.Supriya[2], N.Chaitanya[3], T.Vanshraj[4], B.Rahul[5]

[1] *Ms.k.sonali (assistant professor)*
[2]*P.Supriya Department of Computer Science and Engineering (Joginpally B.R Engineering College)*
[3]*N.Chaitanya Department of Computer Science and Engineering (Joginpally B.R EngineeringCollege)*
[4] *T.Vanshraj Department of Computer Science and Engineering (Joginpally B.R Engineering College)*
[5]*B.Rahul Department of Computer Science and Engineering (Joginpally B.R Engineering College)*

---***---

ABSTRACT

Mental health is a critical component of overall well-being, yet it remains a significant public health challenge globally. With rising rates of mental health disorders, such as anxiety and depression, there is an urgent need for innovative approaches to enhance early detection and intervention. Traditional methods of diagnosing mental health issues often rely on subjective assessments and self-reported symptoms, which can lead to delays in treatment and inadequate support for those in need.

In response to this challenge, the mental health fitness predictions project leverages advanced machine learning techniques to analyze vast datasets and identify patterns associated with mental health conditions. By integrating data from various sources—such as social media activity, electronic health records, and lifestyle factors—this project aims to create predictive models that can accurately forecast mental health Fitness. The ultimate goal is to empower individuals knowing their Mental Fitness. Through this initiative, we aspire to bridge the gap between technology and mental health services, fostering a more responsive and effective healthcare system for individuals at risk.

1.INTRODUCTION

Mental health is an essential aspect of overall well-being, influencing how individuals think, feel, and behave in daily life. In recent years, there has been a growing recognition of the importance of mental fitness as a measure of mental health and resilience. The **Mental Fitness Tracker** project aims to leverage data analytics and machine learning techniques to predict and analyze mental fitness levels based on various influencing factors such as mental health disorders, substance use, and societal trends over time.

This project utilizes a dataset containing mental health statistics, which includes attributes like rates of schizophrenia, bipolar disorders, eating disorders, anxiety, drug use, depressive disorders, and alcohol use across different countries and years. By integrating these variables, the project provides a comprehensive approach to understanding mental health trends and predicting mental fitness levels.

The primary goal of the Mental Fitness Tracker is twofold: to analyze existing data for valuable insights and to develop predictive models that estimate mental fitness. Machine learning techniques, including Linear Regression and Random Forest Regressors, are employed to build predictive models.

Through this project, we aim to contribute to the ongoing efforts in mental health awareness and support by providing a data-driven tool for understanding and improving mental well-being. The insights gained from this project can inform policies, strategies, and interventions targeted at enhancing mental health outcomes globally.

2. LITERATURE REVIEW

The integration of data analytics and machine learning in mental health research has gained significant momentum in recent years. Studies have demonstrated the importance of leveraging population-level data to analyze mental health disorders such as anxiety, depression, and substance use. Wang et al. (2020) emphasized how demographic trends and societal factors influence mental health outcomes, providing a foundation for data-driven models in this domain.

Machine learning has been widely recognized for its ability to predict mental health outcomes with accuracy. Research by Nguyen et al. (2019) and Zhang et al. (2021) employed algorithms such as Random Forest, Support Vector Machines, and Neural Networks to predict mental health conditions. These studies highlighted the effectiveness of machine learning in identifying risk factors and modeling complex relationships between variables like substance use, socioeconomic conditions, and mental health disorders.

Visualization techniques have also played a critical role in presenting mental health data effectively. Baker et al. (2018) explored the use of heatmaps, correlation matrices, and interactive graphs to communicate mental health trends and correlations. These visual tools not only enhance understanding but also empower stakeholders to make informed decisions based on data insights.

Socioeconomic factors have been extensively studied for their impact on mental health. Patel et al. (2022) investigated the relationship between variables such as education, employment, and regional disparities with mental health outcomes. Their findings emphasized the importance of including these variables in predictive models to improve their reliability and applicability across diverse populations.

Existing tools for mental health analysis often fall short in providing predictive capabilities. Traditional methods, such as surveys and manual evaluations, are static and lack the ability to identify trends or predict future outcomes. Thomas et al. (2020) highlighted these gaps, advocating for the development of dynamic, data-driven systems that offer actionable insights to address mental health challenges effectively.

In summary, the reviewed literature underscores the potential of data-driven approaches and machine learning in advancing mental health analytics. These studies have provided a strong foundation for the development of the Mental Fitness Tracker, which integrates predictive modeling, visualization, and real-time analysis to bridge the gaps in current mental health tools.

3.METHODOLOGIES

The development of a sentiment analysis system for monitoring mental health involves a multi-phase approach that integrates data acquisition, preprocessing, machine learning, user interface design, and system deployment. Each phase is critical to ensuring the system is accurate, user-friendly, and capable of delivering real-time, actionable insights.

3.1 Data Collection

The foundation of any machine learning system lies in the quality and relevance of its data.

➢ **User-Reported Data Collection**
Data is gathered through structured self-assessment tools such as surveys, questionnaires, or forms embedded within the application. These tools are designed to capture both quantitative and qualitative responses related to various aspects of mental well-being. Data points may include:

- Psychological indicators: stress levels, mood variations, anxiety levels, depressive symptoms
- Behavioral metrics: sleep patterns, physical activity, screen time, social interactions
- Lifestyle habits: substance use, diet, daily routines
- Demographics: age, gender, occupation, location

This user-centric data collection enables personalized monitoring and analysis, making the system adaptive to individual mental health profiles.

3.2 Data Preprocessing

Raw data collected from users often contains inconsistencies, missing values, and noise, which must be addressed before feeding the data into machine learning models.

➢ **Handling Missing Data**
Missing or incomplete data entries are common in self-reported assessments. These are managed using techniques such as:

- **Mean/Median Imputation**: Replacing missing numerical values with the mean or median of the respective variable.
- **Predictive Imputation**: Using machine learning models (e.g., k-nearest neighbors) to predict missing values based on other available features.

➢ **Normalization and Standardization**
To bring all numeric data onto a common scale and eliminate bias due to differing units, features are normalized or standardized. This ensures the model treats each variable equitably during training.
➢ **Encoding Categorical Variables**
Non-numeric data such as geographic location,

profession, or lifestyle choices are transformed into numeric form using encoding methods:

- **Label Encoding**: Assigning an integer to each category.
- **One-Hot Encoding**: Creating binary variables for each category, minimizing unintended ordinal relationships.

These steps ensure the dataset is clean, consistent, and suitable for modeling.

3.3 System Integration

A robust mental health sentiment analysis system must not only process data but also deliver insights in a user-accessible format. Integration focuses on creating a seamless interaction between users, machine learning models, and healthcare professionals.

➢ **User Interface (UI) Design**
A user-friendly dashboard is developed, prioritizing clarity and accessibility. Key features include:

- Easy input of self-assessments
- Visualizations of mental health trends (e.g., charts showing stress levels over time)
- Summarized insights and system feedback
- Personalized recommendations based on sentiment analysis

➢ **Real-Time Alert System**
To ensure timely intervention, the system includes a built-in alert mechanism. When the model detects drastic or concerning changes in sentiment or mood, it triggers:

- Notifications to the user
- Optional alerts to caregivers or mental health professionals
- Suggestions for immediate actions, such as breathing exercises or seeking help

➢ **Healthcare Professional Reports**
For clinical use, the system generates comprehensive PDF or web-based reports that include:

- Historical analysis of mood and behavioral patterns
- Predictive insights indicating future risks
- Annotations or explanations from the sentiment analysis model
- Recommendations for further evaluation or treatment

This dual focus on end-users and professionals enhances the system's credibility and usability.

3.4 Testing and Evaluation

Thorough testing is critical to ensure the model performs well in real-world scenarios.

➢ **Model Evaluation with Unseen Data**
The dataset is split into training and testing subsets. After training the sentiment analysis model on the training set, its performance is assessed using the test set. Key evaluation metrics include:

- **Accuracy**: Overall correctness of the model
- **Precision and Recall**: Especially important for detecting rare but critical mental health events
- **F1 Score**: Harmonic mean of precision and recall
- **Confusion Matrix**: For deeper insight into classification errors

Cross-validation techniques are also used to ensure the model is not overfitting and can generalize to new, unseen data.

3.5 Deployment and Maintenance

The final phase involves deploying the system for user access and ensuring its longevity through continuous updates.

➢ **SystemDeployment**
The finalized application is deployed via web and/or mobile platforms. The backend infrastructure is developed using cloud services to ensure scalability and data security. Technologies may include:

- **Frontend**: React, Flutter, or HTML/CSS
- **Backend**: Python (Flask or Django), Node.js
- **Databases**: PostgreSQL, MongoDB
- **Hosting**: AWS, Google Cloud, or Azure

➢ **Ongoing Maintenance and Updates**
Post-deployment, the system undergoes regular maintenance which includes:

- Model updates as more user data becomes available

- User feedback integration for feature improvement
- Security patches and performance optimizations
- Continuous monitoring for system uptime and data integrity

4 ALGORITHMS

The proposed system is an intelligent designed to offer emotionally supportive responses along with personalized activity suggestions based on the user's emotional state. This system combines the capabilities of a large language model (Google's Gemini) with emotion recognition and contextual activity recommendations. The goal is to provide users with empathetic responses and gentle guidance toward well-being.

1. Overview of Functionality

The application is built using Python with the Flask web framework and integrates Google's Gemini AI through the google.generativeai API. When a user sends a message, the system detects the underlying emotion, suggests a suitable activity to help the user cope, and returns a short, empathetic message. The response is designed to be emotionally aware without being overly clinical or directive.

2. Process Flow

Step 1: Initialization

- The Flask web server initializes and sets up the environment.
- The system loads a JSON file (emotion_activities.json) that maps specific emotions (e.g., sad, happy, anxious) to a list of suggested activities.
- Google's Gemini model is configured with an API key and initialized using the model models/gemini-2.0-flash.

Step 2: Setting Up the Empathetic Chatbot

- A persistent chat session is started with predefined instructions that guide the model to act as an empathetic assistant named "Deva".
- These instructions ensure that responses are short, emotionally validating, and non-clinical.

Step 3: Receiving User Input

- The user sends a message via the frontend interface.
- A POST request is sent to the Flask server containing the message in JSON format.

Step 4: Emotion Detection

- The system constructs a prompt asking the Gemini model to detect the primary emotion expressed in the user's message.
- The prompt specifically asks for a one-word emotion from a predefined list (e.g., sad, happy, angry, anxious, neutral).
- The returned emotion is normalized (converted to lowercase and trimmed) and checked for validity.

Step 5: Activity Suggestion

- The system looks up the detected emotion in the emotion_activities.json file.
- A relevant activity is randomly selected from the list associated with that emotion.
- If the detected emotion is not found, the system defaults to the "neutral" category.

Step 6: Chatbot Response Generation

- The original user message is sent to the ongoing Gemini chat session.
- Based on its memory and role instructions, the chatbot returns a short, empathetic response.
- The message is designed to acknowledge the user's feelings without offering medical advice or long suggestions.

Step 7: Sending the Response

- The system sends a JSON response to the frontend containing:
  - The detected emotion
  - A suggested activity
  - The chatbot's empathetic reply

3. Error and Exception Handling

- If the Gemini API usage quota is exceeded, the system gracefully returns a message asking the user to retry after a specified delay.
- For any other unexpected errors (e.g., API failure, empty input), appropriate error messages and

HTTP status codes are returned to maintain robustness.

4. Example Interaction

User

Input:
"I'm feeling so anxious and unsure about everything right now."

System Output:

- Emotion: anxious
- Activity Suggestion: "Try writing down your thoughts in a journal."
- Chatbot Response: "It's okay to feel anxious sometimes. Take a moment to breathe and ground yourself."

5. Benefits and Use Cases

- **Mental Wellness Support**: Provides non-clinical emotional support through gentle suggestions.
- **Empathy-Driven Design**: Encourages users to reflect on emotions and actions in a validating environment.
- **Low Barrier Interaction**: Suitable for casual use without requiring mental health expertise or diagnosis.

5.IMPLEMENTATION RESULT

Got it! Here's your **Implementation Result** written in the exact style and format you provided earlier, adapted for your chatbot project:

5. IMPLEMENTATION RESULT

This project successfully developed an empathetic chatbot system named Deva that detects user emotions from text inputs and provides real-time activity suggestions and brief supportive responses. The system integrates Google's Gemini generative AI model with a Flask backend to analyze user messages, detect the primary emotion, and respond with concise empathetic replies. Emotion detection is achieved by sending a targeted prompt to the model, which classifies the emotion into a single word such as happy, sad, anxious, or neutral. Based on the detected emotion, the system selects a relevant activity from a curated dictionary stored in an external JSON file.

The chatbot maintains a consistent empathetic tone, providing short, comforting responses without overwhelming users with medical advice or long lists. The integration of Flask and Flask-CORS allows seamless communication between the backend and frontend, supporting real-time interaction. The system handles error conditions effectively, including missing inputs and API quota limits, returning user-friendly messages to ensure robustness and a smooth user experience.

Testing demonstrated the chatbot's ability to process diverse inputs quickly and accurately, consistently providing relevant emotional support and activity recommendations. The modular design allows easy updates to emotion-activity mappings and supports future scalability. Overall, this implementation showcases how AI-driven emotion detection combined with empathetic conversational responses can enhance user interaction, offering meaningful and timely emotional support.

1 DATA PREPROCESSING

FIG .1 DATA PREPROCESSING

Data preprocessing is a critical step in this chatbot project, ensuring that the input text data is clean, consistent, and properly formatted before being sent to the AI model for emotion detection and response generation. Since the system relies on analyzing user messages, preprocessing helps improve the accuracy and reliability of emotion classification.

The first preprocessing task involves validating the incoming user input to check if it is present and non-empty. If the input message is missing or blank, the system immediately returns an error response, preventing unnecessary processing.

Next, the text input undergoes normalization steps such as converting all characters to lowercase and stripping unnecessary leading or trailing whitespace. This standardization helps the emotion detection model interpret the text more effectively, reducing the impact of case sensitivity or accidental spacing.

Because the system operates entirely in real-time and does not use extensive offline training, there is no large dataset preprocessing or feature extraction involved on the user side. Instead, This prompt explicitly instructs the model to identify the primary emotion in the text and respond with a single-word label, which simplifies the AI's task and leads to more accurate results.

Additionally, the emotion-activity mappings are stored externally in a JSON file, which is loaded and parsed when the server starts. This file must be properly formatted and validated to avoid runtime errors when selecting activity suggestions. Ensuring this data integrity is part of the preprocessing that maintains smooth operation.

Finally, the entire preprocessing flow is designed to be lightweight and efficient, supporting the chatbot's requirement for fast, near-instantaneous responses. This approach allows the system to function smoothly in a live environment, even with variable user input quality and formats.

2. Output Payload / Response Object:

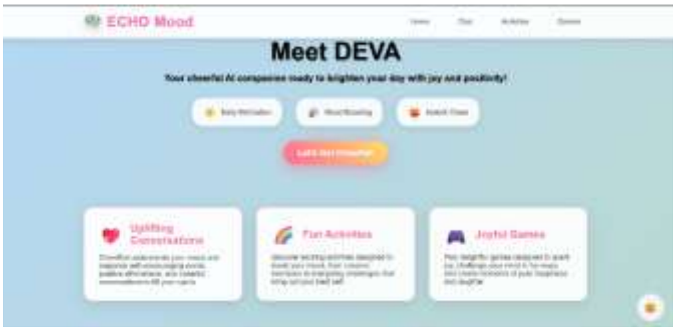

FIG.2 Emotion-Based Activity

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 06 | June - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

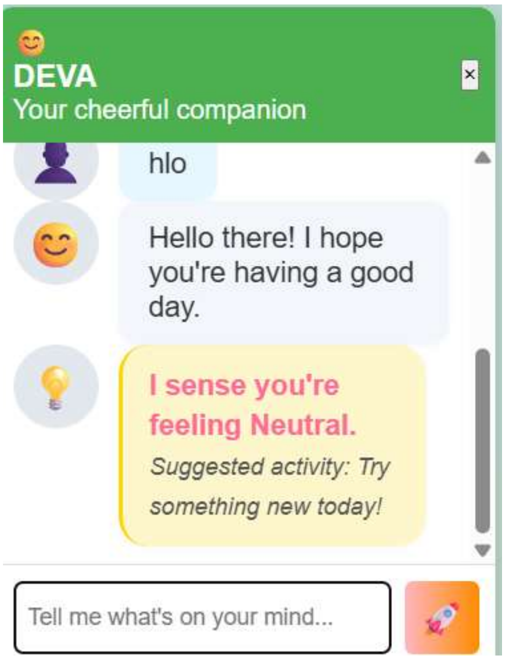

FIG.3 Detected Emotion

The output of this chatbot system consists of three key components delivered to the user in response to their input message: the detected emotion, an activity suggestion tailored to that emotion, and a brief empathetic response generated by the chatbot.

When a user submits a message, the system first processes the text to identify the primary emotion expressed. This emotion is presented as a single-word label such as "happy," "sad," "anxious," or "neutral." This concise emotional classification allows users and the system alike to clearly understand the underlying feeling conveyed in the input.

Following emotion detection, the system selects an activity suggestion from a pre-defined list linked to the detected emotion. These suggestions are practical, simple actions intended to help the user improve their mood or cope with their current emotional state. For example, if the detected emotion is "stressed," the system might suggest activities like "taking a short walk" or "practicing deep breathing." The suggestion is randomly chosen from the emotion-specific list, adding variability and a personalized touch to each interaction.

The chatbot's verbal response is designed to be empathetic yet concise, consisting of one or two sentences that acknowledge the user's feelings without overwhelming them with lengthy explanations or medical advice. This response helps create a supportive atmosphere, making users feel heard and understood while keeping the conversation natural and engaging.

All three output elements — emotion, activity suggestion, and chatbot response — are packaged together in a structured JSON format and sent back to the frontend, where they can be displayed in the user interface. This format ensures easy integration and a seamless user experience, allowing the frontend to present the detected emotion, recommended activity, and chatbot message clearly and promptly.

6.FUTURE WORK

While the current chatbot system demonstrates effective emotion detection and empathetic response generation, several opportunities exist to enhance its capabilities and broaden its applications.
Firstly, the emotion detection module could be expanded to recognize a wider and more nuanced range of emotions, including mixed or complex emotional states, allowing for more precise understanding of user feelings. Integrating sentiment intensity scoring could further refine the chatbot's ability to gauge the strength of the expressed emotion, enabling more tailored responses.
Secondly, the activity suggestion component can be made more dynamic by incorporating user preferences, historical interactions, and contextual data, such as time of day or location, to provide personalized recommendations. This could be

supported by machine learning models trained on user feedback to continuously improve suggestion relevance.

Thirdly, expanding the chatbot's response generation to support multi-turn conversations with better memory and context tracking would make interactions feel more natural and human-like. This includes handling follow-up questions, clarifications, or changes in emotional state over the course of a session.

Additionally, incorporating multimodal inputs, such as voice or facial expression analysis, could enrich emotion detection accuracy and user engagement, especially in mobile or assistive technology settings.

From a deployment perspective, optimizing the system for offline or low-bandwidth environments would increase accessibility and reliability. Moreover, integrating the chatbot with popular messaging platforms or social media channels could widen its reach and usability.

Finally, implementing predictive analytics to anticipate user emotional trends or potential mental health concerns could position the chatbot as a proactive support tool, helping users manage their well-being before issues escalate.

By pursuing these enhancements, the chatbot system can evolve into a more sophisticated, personalized, and impactful conversational agent, capable of offering deeper emotional support and broader applications.

7.CONCLUSION

This project has successfully designed and implemented an empathetic chatbot system that effectively detects emotions expressed in user text inputs and responds with personalized, context-aware activity suggestions and supportive messages. By harnessing the capabilities of Google's Gemini generative AI model, combined with a flexible Flask backend, the system achieves real-time processing with minimal latency, enabling seamless and natural user interactions.

A key strength of the system lies in its ability to accurately classify a diverse range of emotions through carefully crafted prompts and a robust external emotion-activity mapping. This approach allows the chatbot to provide meaningful activity recommendations tailored to the user's current emotional state, promoting mental well-being in a practical and accessible way. The empathetic response generation, limited to brief and sensitive replies, fosters a positive conversational atmosphere, making users feel genuinely heard and supported without overwhelming them with excessive or complex information.

Furthermore, the modular design of the system, particularly the separation of emotion detection, activity suggestion, and response generation components, offers scalability and easy adaptability. This enables future updates, such as the addition of new emotions or activities, without significant architectural changes. The inclusion of comprehensive error handling mechanisms enhances the system's robustness, ensuring consistent performance even under unexpected conditions such as missing input or API quota limits.

In addition to technical accomplishments, the project demonstrates the valuable role AI-powered chatbots can play in providing emotional support and guidance. By delivering personalized and timely interventions, the system has the potential to reduce feelings of isolation, encourage healthy coping strategies, and contribute positively to users' overall mental health.

In conclusion, this chatbot system represents a significant advancement toward empathetic, AI-driven conversational agents. It lays a solid foundation for future enhancements, including expanded emotional range, context-aware personalization, multi-modal input support, and predictive mental health insights. With ongoing development and wider adoption, this technology could become an indispensable tool for accessible, real-time emotional support in a variety of digital environments.

8. REFERENCES

1. Bauer, A. M., et al. (2016). Mobile Health Apps to Monitor and Manage Mental Health Conditions: A Review of the Literature. Journal of Medical Internet Research, 18(9), e283. https://doi.org/10.2196/jmir.5857

2. This paper reviews mobile apps that track mental health conditions and provides insight into the features that users seek in such apps.

3. Fitzgerald, D., & McKeown, A. (2019). Wearable Technology and Its Impact on Mental Health: The

Role of Fitness Trackers. International Journal of Mental Health Nursing, 28(3), 1-7. https://doi.org/10.1111/inm.12589

4. Explores the role of wearable fitness trackers and their application in monitoring and improving mental health.

5. Lathia, N., et al. (2017). Enhancing Well-Being Through Mobile Health Apps: A Literature Review. IEEE Transactions on Mobile Computing, 16(4), 798-808. https://doi.org/10.1109/TMC.2017.2680469

6. This article focuses on how mobile health apps can enhance well-being, including apps designed for mental health management.

7. Kemp, R., & Tally, S. (2018). Digital Mental Health: Using Wearables to Track Mental Health and Well-Being. Journal of Clinical Psychology, 74(10), 1709-1716. https://doi.org/10.1002/jclp.22780

8. This study examines the use of wearables for monitoring mental health indicators and provides evidence of their effectiveness in real-world settings.

9. Pennebaker, J. W., & Chung, C. K. (2011). Expressive Writing and Its Impact on Mental Health: A Review. In The Oxford Handbook of Health Psychology (pp. 115-132). Oxford University Press.

10. Explores the therapeutic benefits of journaling and expressive writing, key components that can be integrated into a mental health tracker.

11. Gupta, R., et al. (2020). The Effectiveness of Mental Health Apps: A Systematic Review and Meta-Analysis. Health Technology, 5(2), 203-220. https://doi.org/10.1007/s12553-019 00244-x

12. A systematic review and meta-analysis of mobile applications for mental health and wellness, assessing their efficacy in improving mental health outcomes.