**Mental Health Analysis and Prediction Using Social Media**

**By**  
**Kondappa**

**Sujith**

**Adithya**

**10/09/2024**

**1. Introduction**

Nowadays, mental health has become one of the most popular slogans for those who believe that all their problems can be explained by the peculiarities of their nervous system. However, this is indeed very true considering the increased use of social media platforms where people are widely encouraged to speak out in emotions, anger, tendencies, fear among other issues. With the aid of the **Natural Language Processing (NLP)** and **Machine Learning (ML)** based Mental Health Analysis and Prediction System, the aim is to provide activities to forbear such dispositions where the fear stratifies the wearer to keep off certain goals. In the analysis, the system processes comments from social media and estimates the risk of a user having any one of the mentioned mental health conditions:

* Stress
* Depression
* Bipolar Disorder
* Personality Disorder
* Anxiety

It’s the purpose of the research to construct a system which could take input from users in the form of text and output whether the user is in good or bad mental health by judging the sentiment and context of the text. It delivers predictions in the instant and can alternatively be adjusted to observe social networking sites to forestall early cases of mental dejection.

**2. Project Overview**

Project Objectives:

* **Data Collection**: Gather social media data related to mental health.
* **Data Preprocessing**: Clean and preprocess text data for sentiment and condition analysis.
* **NLP & Sentiment Analysis**: Use NLP to detect sentiment in user posts.
* **Machine Learning Modeling**: Build and train models to classify text into mental health conditions.
* **Web Interface Development**: Create a web interface that allows users to submit their input and view the analysis results.

**3. Technologies and Tools Used**

The project was undertaken using the following technologies:

* **Python**: For developing and supporting machine learning models and backend work.
* **Flask**: A micro web framework used for developing the web interface.
* **Scikit-learn**: A popular machine learning library used for building and training the machine learning models.
* **The Natural Language Toolkit (NLTK):** A library designed for various text preprocessing processes including stopword integration.
* **Term Frequency-Inverse Document Frequency (TF-IDF):** An external library that serves as a feature extraction technique for encoding textual data from which a machine learning model can comprehend.
* **Seaborn and Matplotlib:** These are charting libraries applied in conjunction with visual representation of the confusion matrix and other statistics.
* **GitHub**: A platform for repository management and development in a collaborative environment.
* **Visual Studio Code (VSCode):** An IDE that provides development and debugging capabilities for coding and maintaining the project.

**4. System Architecture**

The system architecture involves several layers that could be described to play different roles in the process:

1. **Data Preprocessing Layer**:
   * This layer handles the cleansing and preprocessing of the raw data. It removes unnecessary components from the text, such as stopwords, URLs, and special characters. The resulting clean text is then converted into numerical features using the **TF-IDF vectorizer**.
2. **Model Training Layer**:
   * In this layer, the machine learning model Logistic Regression is trained on the preprocessed data. To partition the data into the train and valid parts fairly, so that 80% of the observations are used for training and the rest 20% is used to test the model. Once these features are computed, this custom layer goes on forward to learn the data, and predict values according to the state learnt.
3. **Web Interface Layer**:
   * This layer is built using Flask and provides a web-based user interface. Users can enter comments, which are passed to the backend for prediction. The result is returned and displayed on the webpage in real-time.
4. **Prediction and Output Layer**:
   * After receiving user input, the system preprocesses the input text and runs it through the trained model. The predicted mental health condition is then displayed to the user on the web interface, along with a confidence score or accuracy of the model.

**5. User Guide**

The next section will show step by step how the project can be configured and executed on a local machine

**Step 1: Clone from the GitHub Repository**

To begin, clone the project repository from GitHub using the following steps:

* Open **VSCode** or any preferred Integrated Development Environment (IDE).
* Open the terminal and run the following command:

**git clone https://github.com/username/mental-health-prediction.git**

* Navigate to the project directory:

**cd mental-health-prediction**

**Step 2: Set Up Virtual Environment**

To avoid dependency conflicts, create a virtual environment:

* Create the virtual environment:

**python -m venv venv**

* Activate the virtual environment:
  + On **Windows**:

venv\Scripts\activate

* + On **macOS/Linux**:

source venv/bin/activate

**Step 3: Install Project Dependencies**

The project relies on several Python libraries, which are listed in the requirements.txt file. Install these dependencies by running:

**pip install -r requirements.txt**

**Step 4: Run the Flask Application**

Once all dependencies are installed, you can run the Flask application to start the web interface:

**python app.py**

Now, your application will be present inside your browser at the following address - **http://192.168.86.115:8080/**

**6. Key Features**

The following key features are being provided by the system:

* **Prediction of a user’s mental health in real-time:** A web-based user will provide a particular text on which the system will identify which of the given fare disorders namely Stress and Depression or Anxiety, Bipolar Disorder, Personality Disorder is dominant.
* **User interface module:** A user interface module is to provide a friendly interface and interaction with the system, which is why it will be Flask based.
* **Model Evaluation:** The confusion matrix is created to visualize model performance and simultaneously show the capability of the model in classifying different states of mental health.
* **Ease in Setup:** This project is easy to set up on the local machine, as how to clone the repository and set up an environment for running the app has been given in a very good way.
* **High Accuracy**: This balanced model of machine learning therefore keeps a good balance between accuracy and recall in predicting the status of mental health conditions.

**7. Challenges Faced**

**1. Noisy Text Data**

Social media data is often noisy, containing irrelevant information such as URLs, special characters, and stopwords. This noise can affect the model’s ability to learn useful patterns from the data.

**Solution**: We implemented an extensive text preprocessing pipeline that included stopword removal, lowercasing, and the removal of URLs and special characters to clean the text.

1. **Initial Model Performance**

During the initial phases, the model’s accuracy was suboptimal due to improper feature extraction and hyperparameter tuning.

**Solution**: We refined the preprocessing pipeline, employed **TF-IDF vectorization**, and performed hyperparameter tuning on the Logistic Regression model, which significantly improved model performance.

1. **Deployment Compatibility**

Deploying the project in such a way that it can be easily cloned and set up by other users posed some challenges, especially in ensuring all dependencies were correctly installed.

**Solution**: We included a **requirements.txt** file and a detailed setup guide in the project’s repository to ensure users can easily set up and run the project on their local machines.

**8. Future Improvements**

**1. Integrating Advanced Models**

In the future, more advanced NLP models, such as **BERT**, are planned to be integrated. BERT should do a better job in capturing the context of the language and the subtlety of certain texts, increasing prediction accuracy.

**2. Real-time Data Collection**

We aim to enhance the system by integrating APIs like **Tweepy** (for Twitter) and **PRAW** (for Reddit), allowing real-time data collection from social media platforms. This will enable continuous monitoring of mental health-related discussions online.

**3. User Feedback Mechanism**

This model will be enhanced through continuous work with real data, where the feedback mechanism will involve user confirmation of the system's predictions or refuting them. Personalized feedback would help the model adapt to user-specific behaviors.

**4. Ethical Considerations**

As the system handles sensitive data, we plan to implement additional layers of privacy and ethical guidelines to ensure the anonymity and security of user data. We will also explore bias mitigation techniques to avoid skewed predictions that may negatively impact users.

**9. Screenshots**

Here are a few screenshots of the application in action:

**Web Interface:**  
This screenshot shows the homepage of the web application where users can input comments.

**A screenshot of a test

Description automatically generated**

**Prediction Results:**  
The system predicts the mental health condition based on the text input and displays the result.

**A screenshot of a test

Description automatically generated**

**A screenshot of a test

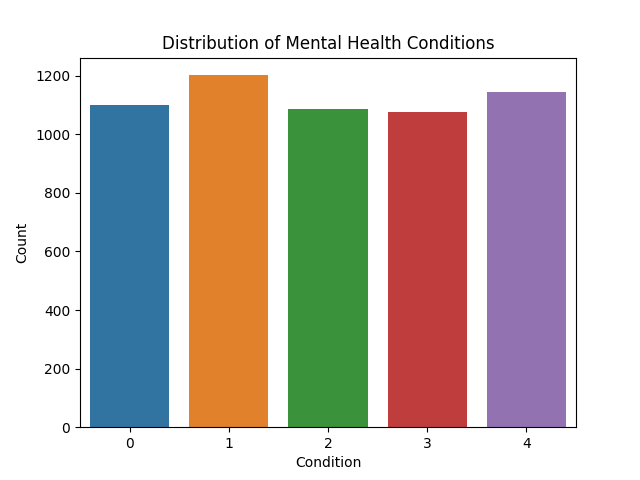
Description automatically generated**

**Confusion Matrix:**

This confusion matrix visualizes the model’s performance on the test data.

**A screen shot of a computer

Description automatically generated**

****

**A graph with blue bars

Description automatically generated**

**10. Conclusion**

The proposed Mental Health Analysis and Prediction system will predict the condition of a person's mental health through his or her comments on social media. This could be implemented in real time to spot potential mental issues using Natural Language Processing, machine learning, and an effective web-based interface. While the system works well, there is ample scope for its improvement, such as the incorporation of advanced models and collecting data in real time.

This project highlights the potential of AI and machine learning in mental health monitoring, offering critical insights that could assist in early detection and intervention.