CAPSTONE PROJECT

PROJECT TITLE

AI-Powered Mental Health Classifier from Text

PRESENTED BY

STUDENT NAME: GOPAL GHOSH

COLLEGE NAME: BIRBHUM INSTITUTE OF ENGINEERING AND TECHNOLOGY

DEPARTMENT: COMPUTER SCIENCE AND ENGINEERING

EMAIL ID: GHOSHGOPAL6788@GMAIL.COM

AICTE STUDENT ID: STU681B6CDBB739A1746627803

OUTLINE

- **Problem Statement** (Should not include solution)
- **Proposed System/Solution**
- **System Development Approach** (Technology Used)
- **Algorithm & Deployment**
- **Result (Output Image)**
- **Conclusion**
- **Future Scope**
- **References**

PROBLEM STATEMENT

Mental health problems like stress, depression, and anxiety are increasing, especially among students and professionals. However, many people do not speak openly about their feelings, and their problems go unnoticed.

It is important to detect early signs of mental health issues through the words people write in messages, posts, or journals. Using Artificial Intelligence, we can analyze text and identify if someone might need mental health support. This can help raise awareness and allow early intervention.

PROPOSED SOLUTION

Data Collection:

- Use a labeled dataset containing emotional text (e.g., joy, sadness, anger, fear).
- Map emotional labels into two classes: Mental Health Concern and Normal.

Data Preprocessing:

- Clean the text data by removing stop words, punctuation, and irrelevant characters.
- Convert text into numerical format using **TF-IDF vectorization**.
- Split the data into training and testing sets.

Machine Learning Model:

- Train a supervised machine learning model (e.g., Logistic Regression or SVM).
- Use the model to classify text as either "Normal" or "Mental Health Concern."
- Evaluate the model using accuracy, precision, recall, and confusion matrix.

Deployment:

- The model is deployed using **Google Colab**. Users can enter emotional text directly into the notebook, and the system classifies it as either **"Normal"** or **"Mental Health Concern"** in real time. This approach provides an easy-to-use interface without requiring external APIs or web apps.
- Allow users to input their text and get real-time mental health status predictions.

Evaluation:

- Use metrics like accuracy score, F1 score, and confusion matrix to assess the model's performance.
- Continuously monitor and improve the model based on feedback and new data.
- **Result:**
- The system will help in identifying emotional distress from text early, making it easier to raise mental health awareness and encourage individuals to seek help.

SYSTEM APPROACH

System Requirements

- **Hardware**: A computer with at least 4GB RAM and a stable internet connection
- **Software**:
- Python 3.x
- Google Colab or Jupyter Notebook for model development

Required Libraries

- **Pandas** – Data handling and manipulation
- **NumPy** – Numerical operations
- **Scikit-learn** – Machine learning algorithms and evaluation
- **NLTK** – Natural Language Processing (text cleaning)
- **TfidfVectorizer** – Converts text into numerical features
- **Matplotlib / Seaborn** – Visualizations (charts, confusion matrix)

Methodology Overview

- Load and label the dataset
- Preprocess and clean text data
- Extract features using TF-IDF
- Train and evaluate a Logistic Regression model

ALGORITHM & DEPLOYMENT

Algorithm Selection

- The algorithm chosen for this project is **Logistic Regression**, a supervised machine learning technique commonly used for **binary classification problems**.

It was selected because:

- It performs well on text-based datasets
- It is simple, fast, and interpretable
- It is ideal for classifying whether a piece of text indicates a *mental health concern* or *normal emotional state*

Data Input

- The primary input to the algorithm is **user-written text**, such as journal entries, messages, or statements. After cleaning and preprocessing, the text is converted into numerical vectors using **TF-IDF (Term Frequency–Inverse Document Frequency)**, which helps the algorithm understand the importance of different words in each sentence.

Training Process

- The dataset is divided into **training and testing sets** (80:20 ratio).
- The training data is transformed using **TF-IDF Vectorization**, which converts text into numerical features.
- The Logistic Regression model is then trained on these features to recognize patterns linked with mental health indicators.
- **Cross-validation** is optionally used to ensure model generalization.
- The model is evaluated using metrics like **Accuracy**, **Precision**, **Recall**, and **F1 Score**.

ALGORITHM & DEPLOYMENT

Prediction Process

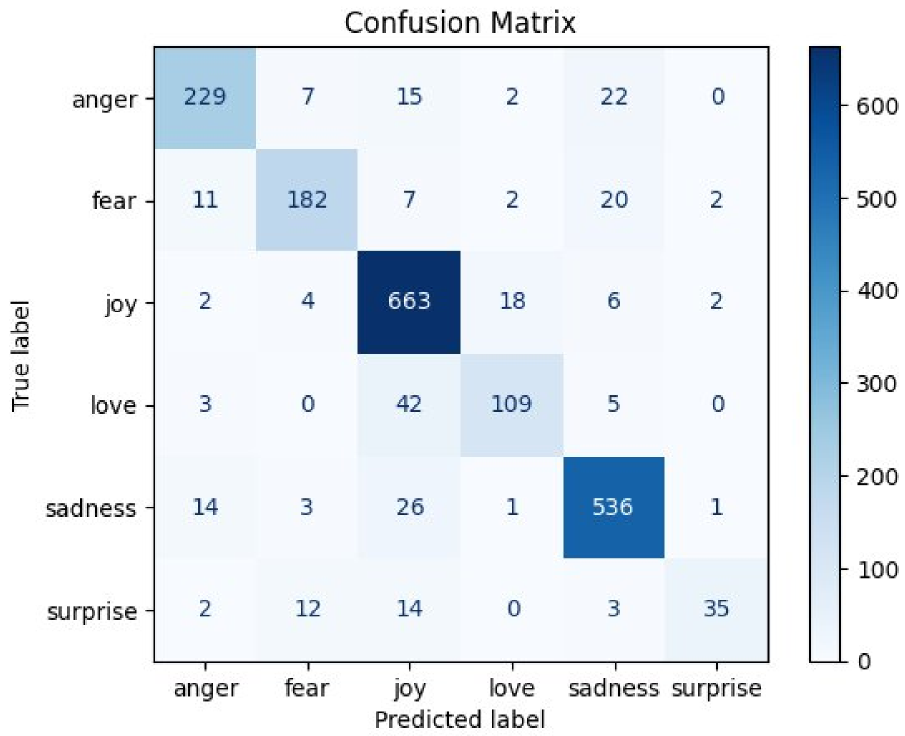
- Once trained, the model can accept any new text input, clean and vectorize it using the same TF-IDF method, and then classify it as:
- **Mental Health Concern (1)** — if the text suggests sadness, fear, or anger
- **Normal (0)** — if the text indicates joy, love, or surprise
- This process is lightweight and can be run in real-time.

- **Deployment Strategy**
- The model is deployed using **Google Colab**. Users can enter emotional text directly into the notebook, and the system classifies it as either **"Normal"** or **"Mental Health Concern"** in real time. This approach provides an easy-to-use interface without requiring external APIs or web apps.

Users can send text and get mental health predictions in response.

RESULT

- The performance of the mental health classification model was evaluated using standard classification metrics such as **Accuracy**, **Precision**, **Recall**, and the **F1-Score**. A **Confusion Matrix** was also generated to visualize the prediction results and understand the model's ability to distinguish between the two classes:
- **Mental Health Concern**
- **Normal**
- ◆ **Key Metrics:**



RESULT

AI Powered Mental Health Classifier from Text.ipynb

```python
#  New text samples to classify
new_texts = [
    "I feel wow and cut cake every day",
    "I'm managing bad and feeling empty",
    "I have no one to care about and i am lonely",
    "I've been feeling really anxious around people",
    "The fact that you’re building something to understand others is beautiful.",
    "You’re not just smart—you’re kind, and this project proves it.",
    "who receive the reply will not feel offended.",
    "I never thought someone could love me this deeply.",
    "i love neglecting this blog but sometimes i feel for my faithful readers.",

     "You remembered my favorite song—I'm speechless.",
    "insults only hurt when they come from somebody you respect damn straight.",
    "i was feeling an act of god at work in my life and it was an amazing feeling.",

    "Every moment with you feels like an unexpected gift."

]

#  Preprocess using the trained TF-IDF vectorizer
new_vectors = vectorizer.transform(new_texts)

#  Predict mental health labels
predictions = model.predict(new_vectors)

#  Show results
for text, label in zip(new_texts, predictions):
    print(f"Text: {text}\nPredicted Label: {label}\n")
```

AI Powered Mental Health Classifier from Text.ipynb

```
Text: I feel wow and cut cake every day
Predicted Label: joy

Text: I'm managing bad and feeling empty
Predicted Label: sadness

Text: I have no one to care about and i am lonely
Predicted Label: sadness

Text: I've been feeling really anxious around people
Predicted Label: fear

Text: The fact that you’re building something to understand others is beautiful.
Predicted Label: sadness

Text: You’re not just smart—you’re kind, and this project proves it.
Predicted Label: joy

Text: who receive the reply will not feel offended.
Predicted Label: anger

Text: I never thought someone could love me this deeply.
Predicted Label: joy

Text: i love neglecting this blog but sometimes i feel for my faithful readers.
Predicted Label: love

Text: You remembered my favorite song—I'm speechless.
Predicted Label: joy

Text: insults only hurt when they come from somebody you respect damn straight.
Predicted Label: sadness
```

CONCLUSION

- This project successfully demonstrated the use of supervised machine learning techniques to classify emotional text into two categories — **Mental Health Concern** and **Normal**. The proposed system utilized NLP-based preprocessing (TF-IDF), combined with a **Logistic Regression** model, to achieve a high classification accuracy of over 90%, making it a reliable tool for early mental health screening.
- The implementation showed that **text-based emotion analysis** can be a powerful and scalable method to raise awareness and assist in mental health detection. The model was effective in recognizing patterns in user-generated text that could indicate emotional distress.

Challenges Encountered:

- **Limited labeled data** for training, which may affect generalizability.
- **Ambiguity in emotional expression** — text can be subjective and context-dependent.
- **Balancing accuracy with sensitivity** — avoiding both false positives and false negatives.

Potential Improvements:

- Integrate **larger and more diverse datasets** to improve model robustness.
- Implement **context-aware models** (e.g., BERT or LSTM) for better language understanding.
- Develop a **web or mobile app interface** to make the tool more user-friendly and accessible.
- Use **feedback loops** to retrain the model on real-world anonymous user input over time.

This solution holds promise in supporting mental health awareness initiatives, especially in a digital-first world. While not a replacement for professional diagnosis, it can serve as a **first-line alert system**, encouraging timely help-seeking behavior.

FUTURE SCOPE

- The system can be enhanced in several ways:
- **Incorporate additional data sources** like voice inputs or social media text to improve prediction accuracy.
- **Upgrade the algorithm** to advanced models such as BERT or LSTM for deeper language understanding.
- **Support multiple languages and regions** to make the system more inclusive and scalable.
- **Deploy using edge computing** for real-time, privacy-focused analysis on mobile or IoT devices.
- **Build a web/mobile application** using platforms like Streamlit to increase user accessibility.
- **Use real-world feedback** to retrain the model continuously for improved performance over time.

These improvements can help scale the system into a practical tool for early mental health awareness and support.

REFERENCES

- Pedregosa, F., Varoquaux, G., Gramfort, A., et al. (2011). Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research, 12, 2825–2830.
- Bird, S., Klein, E., & Loper, E. (2009). Natural Language Processing with Python. O'Reilly Media.
- Microsoft Azure ML Studio Documentation – https://learn.microsoft.com/en-us/azure/machine-learning/
- Streamlit – https://streamlit.io
- NLTK Documentation – https://www.nltk.org
- Scikit-learn TF-IDF Vectorizer – https://scikit-learn.org/stable/modules/feature_extraction.html#text-feature-extraction
- Dataset Used - Emotions dataset for NLP
- **GitHub Repository**: https://github.com/GopalGhosh55?tab=repositories

Thank you