**MoodifyMe**

**MoodifyMe Project Proposal**

**Team Members**

* CHAMSEDDINE Lea
* EDDE Chiara
* HANNA Marielouise

**GitHub Repository**: <https://github.com/marielouisehanna/Machine_learning>

**1. Introduction**

**1.1 Problem Statement**

Understanding and managing our emotions is crucial for mental well-being, yet many people struggle with tracking their moods and identifying the factors that influence them. MoodifyMe aims to address this by providing a journaling app that leverages machine learning to analyze journal entries and predict emotional states. Based on the analysis, the app will recommend personalized content, such as music and movies, to improve or maintain the user's mood.

**1.2 Key Features**

* Mood Prediction: Using Natural Language Processing (NLP), the app will analyze journal entries and predict the user's mood.
* Personalized Recommendations: The app will suggest music, movies tailored to the user's mood.

**2. Technical Components**

**2.1 Mood Prediction Pipeline**

**Data Collection:** <https://www.kaggle.com/datasets/debarshichanda/goemotions>,

* Main Dataset: We will use the GoEmotions dataset, which contains over 58,000 labeled Reddit comments categorized into 27 distinct emotions, including "neutral". The dataset was created by Google Research and poses several challenges:
  + High Number of Emotions: The model must classify entries into 28 categories, including 27 emotions and a neutral category.
  + Class Imbalance: About 30% of samples are labeled as "neutral".
  + Multi-label Classification: Each journal entry can be associated with up to five different emotions.

**GoEmotions Taxonomy:**

* Admiration, Amusement, Anger, Annoyance, Approval, Caring, Confusion, Curiosity, Desire, Disappointment, Disapproval, Disgust, Embarrassment, Excitement, Fear, Gratitude, Grief, Joy, Love, Nervousness, Optimism, Pride, Realization, Relief, Remorse, Sadness, Surprise, Neutral.

**Text Preprocessing**

* Basic NLP Steps:
  + Tokenization: Break text into words.
  + Stopword Removal: Filter out common, non-essential words (e.g., "the", "a").
  + Text Cleaning: Remove irrelevant characters and symbols.
* Handling Emojis and Slang: Emojis and slang will be translated to their corresponding emotional meanings (e.g., "😂" → "happy").

**Model:**

* BERT Model: We will utilize BERT because it is a powerful transformer-based model that excels at understanding the context of words within sentences. This contextual understanding makes it ideal for analyzing complex emotions in text, like the journal entries in our project. BERT's ability to capture relationships between words, even over long distances, allows it to accurately predict mood based on Indirect emotional expressions
  + Performance Evaluation: The model's effectiveness will be assessed using metrics such as accuracy and precision.

**Feature Extraction & Model Setup :**

* BERT Fine-Tuning: We will adjust hyperparameters to optimize the model's performance.
* Data Cleansing: We will standardize text, translating emojis and handling contractions, special characters, and acronyms.

**· Handling the "Neutral" Class :**

We recognize that the "neutral" class in the dataset may act as noise, as it represents ambiguous or non-emotional content. Therefore, we will adjust the model to better handle this noise, potentially by:

* + Filtering Neutral Entries: Neutral entries will be excluded from mood predictions to reduce model confusion.
  + Reclassification or Custom Weighting: We may reclassify neutral entries into a more relevant emotional category or apply custom weighting to mitigate their impact on the overall prediction.

**2.2 Application Design**

Database Design

* Songs Database: Collection of songs categorized by mood (e.g., happy, calm, sad), keywords, artist and genre.
* Movies Database: Collection of movies categorized by mood, keywords, genre and director.

Recommendation System

* Content Matching: The app will recommend music, movies based on the predicted mood. After predicting the user's mood, the app will randomly select content from the relevant category to keep the recommendations dynamic and engaging.

**3. Planned Improvements**

* Database Expansion: We will continuously add new songs and movies, with an emphasis on cultural relevance and diversity.
* Personalization: Over time, the app will adapt to users’ preferences, offering increasingly tailored recommendations. Features like a like/dislike system will help the app learn from user interactions.
* Model Optimization: We will explore other models, such as BiLSTM or attention-based models, to further improve mood prediction accuracy.

**4. Conclusion**

MoodifyMe will help users better understand and track their moods through journal entries. By leveraging BERT, a state-of-the-art deep learning model, the app will predict moods and offer personalized content recommendations. The app’s goal is to help users feel better, providing them with emotional support through tailored music, movies, and more.