**Audio Based Stress Detection using DAIC-WOZ Dataset**

1. **Overview**

This machine learning model aims to detect stress levels based on input data. It utilizes various neural network architectures such as Deep Neural Networks (DNN), Convolutional Neural Networks (CNN), and Long Short-Term Memory networks (LSTM) for stress detection.

1. **Model Architecture**

The model consists of three different architectures:

* 1. Deep Neural Network (DNN)
     1. Layers:
        1. Input Layer (193 features)
        2. Dense Layer (64 units, ReLU activation)
        3. Dense Layer (32 units, ReLU activation)
        4. Output Layer (1 unit, Sigmoid activation)
  2. Convolutional Neural Network (CNN)
     1. Layers:
        1. Conv1D Layer (128 filters, kernel size 8, ReLU activation)
        2. MaxPooling1D Layer (pool size 2)
        3. Conv1D Layer (64 filters, kernel size 8, ReLU activation)
        4. MaxPooling1D Layer (pool size 2)
        5. Flatten Layer
        6. Output Layer (1 unit, Sigmoid activation)
  3. Long Short-Term Memory network (LSTM)
     1. Layers:
        1. LSTM Layer (128 units)
        2. Dense Layer (64 units, ReLU activation)
        3. Dense Layer (32 units, ReLU activation)
        4. Output Layer (1 unit, Sigmoid activation)

1. **Usage Instructions:**
   1. Prerequisites:
      1. Python (3.x recommended)
      2. Required libraries: TensorFlow, pandas, numpy, matplotlib, sklearn
   2. Steps:
      1. Set Up Environment:
         1. Ensure Python is installed. If not, download and install Python (3.x) from [Python's official website](<https://www.python.org/downloads/>).
      2. Install Required Libraries:
         1. Open the terminal or command prompt.
         2. Navigate to the project directory.
         3. Run `pip install -r requirements.txt` to install necessary libraries.
      3. Prepare the Data:
         1. Ensure the dataset is available or use your own dataset.
         2. Modify the file paths in the code (`data\_analysis.ipynb` and `stress\_detection.ipynb`) to point to your dataset if necessary.
      4. Run Data Analysis:
         1. Open `data\_analysis.ipynb` using Jupyter Notebook or any compatible IDE.
         2. Execute each cell sequentially to extract audio features from the test audio file.
      5. Run Stress Detection:
         1. Open `stress\_detection.ipynb` using Jupyter Notebook or any compatible IDE.
         2. Ensure that the extracted features from the previous step are accessible to `stress\_detection.ipynb`.
         3. Run each cell sequentially in `stress\_detection.ipynb` to perform data preprocessing, model training, and evaluation.
      6. Customization and Experimentation:
         1. Tweak hyperparameters, modify architectures, or adjust data preprocessing steps for experimentation and customization.
2. **Notes :**
   1. The provided code in the notebooks (`data\_analysis.ipynb` and `stress\_detection.ipynb`) demonstrates the entire workflow from data preprocessing to model evaluation.
   2. Customization and fine-tuning of hyperparameters may be required for specific use cases.