BASAVARAJESWARI GROUP OF INSTITUTIONS

BALLARI INSTITUTE OF TECHNOLOGY & MANAGEMENT

NACC Accredited Institution*

(Recognized by Govt. of Karnataka, approved by AICTE, New Delhi & Affiliated to Visvesvaraya Technological University, Belagavi)
"JnanaGangotri" Campus, No.873/2, Ballari-Hospet Road, Allipur, Ballar1-583 104 (Karnataka) (India)
Ph: 08392 – 237100 / 237190, Fax: 08392 – 237197

DEPARTMENT OF CSE-AI

A Mini-Project Report On

"Human Stress Level Detection From Voice"

A report submitted in partial fulfillment of the requirements for the

NEURAL NETWORK AND DEEP LEARNING

Submitted By

T ROHITHA **USN: 3BR22CA054**

Under the Guidance of
Mr. Azhar Biag

Asst. Professor

Dept of CSE (DATA SCIENCE),
BITM, Ballari

Visvesvaraya Technological University

Belagavi, Karnataka 2025-2026

BALLARLINSTITUTE OF TECHNOLOGY & MANAGEMENT

Autonomous Institute under VTU. Belagovi l Approved by AICTE, New Delhi

Recognized by Govt. of Karnataka

DEPARTMENT OF CSE – ARTIFICIAL INTELLIGENCE

CERTIFICATE

This is to certify that the project work entitled **"Human Stress Level Detection From Voice"** is a bonafide work carried out by T **Rohitha** bearing **USN: 3BR22CA054** in partial fulfillment for the award of degree of **Bachelor Degree in CSE (Artificial Intelligence)** in the **VISVESVARAYA TECHNOLOGICAL UNIVERSITY**, Belagavi during the academic year 2025-2026.

It is certified that all corrections and suggestions indicated for internal assessment have been incorporated in the report deposited in the library. The project has been approved as it satisfies the academic requirements in respect of mini project work prescribed for a Bachelor of Engineering Degree.

Signature of project guide

Prof. Pavan Kumar

Mr. Vijay Kumar

Signature of HOD

Dr. Yerestme Suresh

ABSTRACT

This project aims to detect human stress levels using speech with the help of deep learning techniques. It analyzes voice signals to identify emotional variations such as pitch, tone, and intensity that indicate stress. Audio features like MFCCs are extracted using the Librosa library and used as input to a Deep Neural Network model. The model is trained using TensorFlow to classify speech into categories such as calm, angry, and fearful. This approach provides a simple, non-invasive way to measure stress through speech. The trained model achieves high accuracy in predicting stress levels. It can be applied in areas like mental health monitoring and intelligent assistants. Overall, this system contributes to real-time, emotion-aware human–computer interaction.

.ACKNOWLEDGEMENT

The satisfactions that accompany the successful completion of our mini project on **Human Stress Level Detection From Voice** would be incomplete without the mention of people who made it possible, whose noble gesture, affection, guidance, encouragement and support crowned my efforts with success. It is our privilege to express our gratitude and respect to all those who inspired us in the completion of our mini-project.

I am extremely grateful to my Guide **Mr. Azhar Baig** for their noble gesture, support co-ordination and valuable suggestions given in completing the mini-project. I also thank **Dr. Yeresime Suresh,** H.O.D. Department of CSE(AI), for his co-ordination and valuable suggestions given in completing the mini-project. We also thank Principal, Management and non-teaching staff for their co-ordination and valuable suggestions given to us in completing the Mini project.

Name	USN
T ROHITHA	3BR22CA054

TABLE OF CONTENTS

1. INTRODUCTION

Stress is a common human response that affects health, emotions, and overall behavior. Traditional stress detection methods require physical sensors or medical tests, which may be time-consuming and uncomfortable. Speech, on the other hand, naturally reflects emotional states through variations in pitch, tone, intensity, and rhythm.

This project aims to build a deep learning-based system that can automatically detect human stress levels using speech signals. By analyzing audio recordings and extracting meaningful features, the model predicts stress categories with high accuracy. The system provides an easy, non-invasive, and real-time method suitable for modern human–computer interaction.

1.1 Problem Statement

Current stress detection techniques often rely on physical tests, wearables, or medical equipment, which can be slow, expensive, and uncomfortable. There is a need for an automated, fast, and non-invasive system to detect stress from speech, which can reflect emotional changes accurately.

This project aims to design a deep learning-based system capable of classifying stress levels from audio by analyzing voice patterns and extracting acoustic features.

1.2 Scope of the project

The scope of the project **"Human Stress Level Detection from Voice"** includes the development, implementation, and evaluation of a speech-based stress classification system using deep learning. The project focuses on extracting meaningful acoustic features such as MFCCs and training a Deep Neural Network to classify stress-related emotions like calm, angry, and fearful.

The system covers the complete workflow—audio input, preprocessing, feature extraction, model training, evaluation, and prediction. The project also includes generating performance metrics such as accuracy and confusion matrix to validate the model. Although limited to dataset-based audio samples, the methodology can be extended to real-time applications such as mental health monitoring, intelligent virtual assistants, and stress-aware user interfaces. The system demonstrates a non-invasive, fast, and effective approach to automatically detect stress using speech signals.

1.3 Objectives

- To detect stress levels from human speech signals.
- To extract important audio features such as MFCCs.
- To build a Deep Neural Network model for emotion/stress classification.
- To classify audio into calm, angry, and fearful categories.
- To develop a simple and non-invasive stress detection system.
- To evaluate performance using accuracy and confusion matrix.

METHODOLOGY

Block Diagram Steps

1. **Audio Input** – Voice samples are recorded or uploaded.
2. **Preprocessing** – Noise reduction, silence trimming, normalization.
3. **Feature Extraction** – Extract MFCCs and other relevant features.
4. **Classification Model** – Train a DNN for stress classification.
5. **Prediction** – Model predicts stress category and confidence score.
6. **Evaluation** – Accuracy, loss plots, and confusion matrix.

2. LITERATURE SURVEY

1. Livingstone & Russo (2018) – RAVDESS Emotional Speech Dataset

Livingstone and Russo introduced the RAVDESS dataset, a widely used benchmark for speech emotion research. It contains high-quality recordings of emotional speech, including calm, angry, and fearful tones. This dataset enables accurate training and evaluation of stress/emotion detection systems by providing standardized audio samples.

2. McFee et al. (2015) – Librosa Audio Processing Library

McFee and colleagues developed the Librosa library, which provides tools for extracting audio features such as MFCCs, chroma, and spectral contrast. MFCCs specifically play a major role in stress detection, as they capture pitch, tone, and frequency patterns related to emotional states.

3. Ververidis & Kotropoulos (2006) – Emotional Speech Analysis Techniques

This study investigated various machine learning techniques for emotion detection from speech, demonstrating the effectiveness of acoustic features in distinguishing stress-related emotions. It highlights that voice changes such as pitch variation and intensity shifts strongly reflect psychological stress.

4. Schuller et al. (2011) – Deep Learning for Speech Emotion Recognition

Schuller and his team applied deep learning models for emotion recognition and showed that neural networks outperform traditional classifiers such as SVM or k-NN. Their findings support the use of DNNs for analyzing complex patterns in speech, which is essential for accurate stress detection.

5. Koolagudi & Rao (2012) – Emotion Recognition Using Speech Features

Koolagudi and Rao examined multiple acoustic features including MFCCs, LPC, and prosodic features for emotion classification. Their research emphasized that MFCCs are the most reliable features for capturing emotional characteristics in human speech.

6. Zheng et al. (2015) – Speech Emotion Recognition Using CNNs

Zheng and colleagues used convolutional neural networks to classify emotional states from audio spectrograms. Their work demonstrated that deep learning models learn richer and more detailed audio patterns, improving accuracy in detecting stress or other emotional cues.

7. Abdel-Hamid et al. (2014) – Deep Neural Networks for Speech Classification

This research highlighted how deep neural networks can effectively classify complex speech signals by learning hierarchical features. Their findings contribute to stress detection by validating the use of multi-layer DNNs for accurate voice-based classification.

3.SYSTEM REQUIREMENTS

Functional Requirements

- Audio input (upload/record speech).
- Preprocessing: noise removal & normalization.
- Feature extraction using MFCCs.
- Stress classification using a deep neural network.
- Output: predicted emotion/stress category.
- Performance evaluation metrics.

Non-Functional Requirements

- Fast and accurate predictions.
- User-friendly and easy to operate.
- Scalable to more datasets and emotions.
- Reliable and consistent performance.
- Data privacy ensured for user audio files.
- Maintainable and extendable code.

4. DESCRIPTION OF MODULES

The Human Stress Detection from Voice system is divided into multiple modules, each handling a crucial stage in the audio processing and deep learning pipeline. These modules work together to ensure smooth data preprocessing, feature extraction, model training, evaluation, and prediction.

3.1 Audio Preprocessing Module

This module loads raw audio samples and prepares them for analysis. It handles background noise removal, trims silence, normalizes audio amplitude, and converts files to a uniform sampling rate. These steps ensure that all speech samples are clean, consistent, and suitable for feature extraction.

3.2 Feature Extraction Module

This module extracts meaningful features from the processed audio using the Librosa library. It computes MFCCs (Mel Frequency Cepstral Coefficients), which capture pitch, tone, and frequency characteristics of speech. These numerical features form the input for the deep learning model and play a crucial role in stress classification.

3.3 Deep Neural Network (DNN) Model Building Module

This module constructs the deep learning architecture used for stress detection. It defines the input layer, multiple dense hidden layers with ReLU activation, and an output layer with softmax activation to classify speech into calm, angry, and fearful (stressed) categories. The model is compiled using the Adam optimizer and categorical cross-entropy loss.

3.4 Model Training Module

After building the model, this module trains it using the extracted MFCC features. It sets training parameters such as batch size, number of epochs, and validation split. During training, the module tracks training and validation accuracy/loss to monitor learning performance and prevent overfitting.

3.5 Model Evaluation Module

This module evaluates the trained model's performance using metrics such as accuracy, precision, recall, F1-score, and confusion matrix. It analyzes how well the model classifies different stress levels and provides detailed performance reports. The module helps identify strengths and weaknesses of the model.

3.6 Visualization Module

This module generates visual graphs to help understand the model's learning behavior. It produces training vs. validation accuracy plots, loss curves, and confusion matrix heatmaps. These visuals make the system more interpretable and provide insights into overall performance.

3.7 Prediction Module

This module applies the trained model to new speech inputs. It extracts MFCC features from user-provided audio and classifies the stress level into calm, angry, or fearful. It is designed to provide fast, automated predictions suitable for stress monitoring applications and real-time systems.

3.8 Dataset Splitting Module

This module splits the dataset into training and testing sets using an 80:20 ratio. It ensures proper distribution of emotion categories through stratified sampling. This prevents class imbalance issues and ensures that model evaluation is accurate, fair, and based on unseen audio samples.

3.9 Output Interpretation Module

This module converts the model's prediction probabilities into user-friendly output labels. It displays the predicted stress category along with confidence scores. It ensures that users can clearly interpret the results and understand the stress level detected from the voice input.

5. IMPLEMENTATION

1. Data Collection

- Audio datasets such as **RAVDESS** were used for calm, angry, and fearful recordings.

2. Preprocessing

- Convert audio into a uniform sampling rate.
- Remove background noise.
- Normalize volume and trim silence.

3. Feature Extraction

Using **Librosa**, MFCCs and other time–frequency features are extracted.
MFCCs convert sound waves into numerical data suitable for neural networks.

4. Model Development

- The Deep Neural Network is built using TensorFlow/Keras.
- Input: MFCC feature vectors
- Hidden Layers: Dense layers with ReLU activation
- Output: Softmax layer for 3 emotion categories

5. Training

- Model trained using categorical cross-entropy loss
- Optimizer: Adam
- Batch training over multiple epochs
- Validation accuracy monitored to avoid overfitting

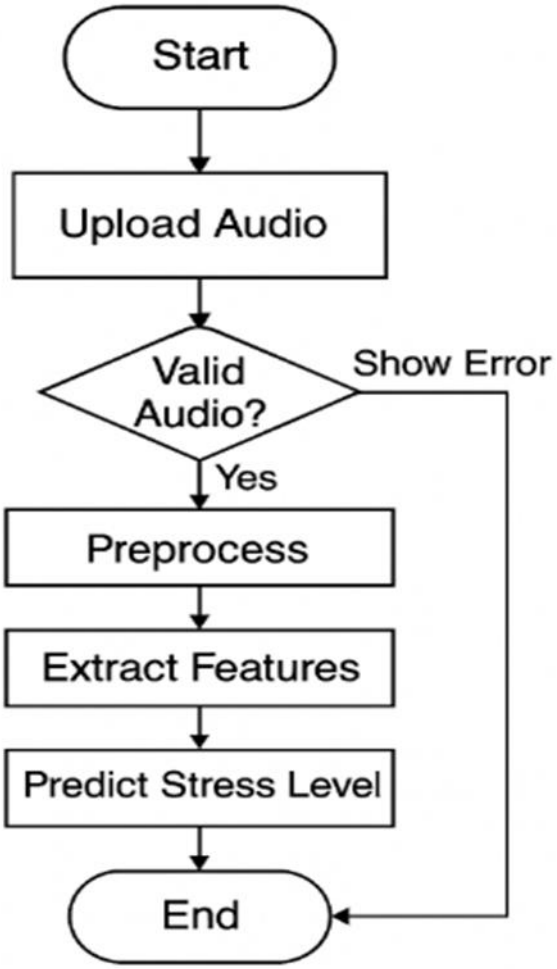
6. Evaluation

- Confusion matrix
- Accuracy score
- Precision, Recall, F1-score

7. Prediction Interface

- Accepts audio input from user
- Displays predicted stress level and confidence percentage

6.SYSTEM ARCHITECTURE



Input

This stage loads the audio dataset containing voice recordings of various emotional states (Calm, Angry, Fearful). The audio files are read using Librosa, and their basic properties—such as duration, sampling rate, and amplitude—are inspected. This helps ensure that all audio samples are usable and identifies whether resampling, trimming, or noise removal is needed.Preprocessing

Preprocessing prepares raw audio for the DNN model so it can learn effectively.

- **Noise & Silence Removal:** Remove background noise and trim silent sections.
- **Normalization:** Adjust volume levels for consistency.
- **Resampling:** Convert all audio files to a common sampling rate (e.g., 22,050 Hz).
- **Feature extraction prep:** Convert audio to a fixed length or apply padding.
- **Train–test split:** Typically 80:20, using stratification to maintain class balance.
- **Data formatting:** Convert arrays to float32/int32 as required.

Good preprocessing ensures clean, uniform audio features and improves model accuracy.

Feature Extraction

This stage converts raw speech into numerical features.

- **Extract MFCCs:** Capture pitch, tone, and frequency patterns related to stress.
- **Generate feature matrix:** Flatten or reshape MFCCs for model input.
- **Store labels:** Calm = 0, Angry = 1, Fearful = 2.

MFCCs are the core features that allow the model to identify stress patterns in voice

- Hidden layers: e.g., Dense(64, ReLU) → Dropout(0.2) → Dense(32, ReLU). These layers learn nonlinear feature interactions; ReLU helps with gradient flow and sparsity.
- Dropout: randomly disables a fraction of neurons during training to reduce overfitting and improve generalization.

- Output layer: Dense(1, sigmoid) — produces a probability for the positive class (diabetic).
- Compile settings: choose optimizer (Adam), loss (binary_crossentropy for two-class problems), and metrics (accuracy; optionally precision, recall, AUC). Choosing hyperparameters (layer sizes, dropout rate, learning rate) is part of architecture design and may be tuned.

The goal here is to build a model expressive enough to capture patterns but regularized enough to avoid overfitting.

Training

Training is where the network learns by updating weights to minimize loss.

- Fit the model: run for a fixed number of epochs (e.g., 35) with a chosen batch size (e.g., 32), and optionally a validation_split (e.g., 0.2) to monitor validation metrics each epoch.
- Monitor: record training & validation loss and accuracy (history object). Watch for overfitting (training accuracy rising while validation accuracy plateaus or drops).
- Callbacks (optional): use EarlyStopping to stop when validation loss stops improving, ModelCheckpoint to save best weights, and ReduceLROnPlateau to lower learning rate on plateau.
- Hyperparameter tuning: you may iterate over epochs, batch size, learning rate, layer sizes, and regularization to improve performance.

Training converts initialized weights into a predictive model by repeated forward/backward passes on the data.

Visualization and Prediction

This final stage evaluates the model and makes predictions.

Visualizations:

- Accuracy vs Epochs
- Loss vs Epochs
- Confusion Matrix (Calm/Angry/Fearful)
- Classification Report (Precision, Recall, F1-score)

Prediction:

- Use the trained model to predict stress levels from new audio.
- Convert softmax probabilities to class labels (0/1/2).
- Display predicted emotion and confidence score.

Deployment:

- Export the trained model (model.save).
- Integrate into a real-time stress detection system or GUI.

7.CODE IMPLEMENTATION

Algorithm: Human Stress Detection from Voice using Deep Neural Network

Input: Audio recordings (Calm / Angry / Fearful)
Output: Predicted Stress Level and Performance Metrics

1. Start

2. Load and Prepare Audio Data

2.1 Load audio files from the dataset (e.g., RAVDESS).
2.2 For each audio file:

- Read the audio signal
- Resample to a fixed sampling rate (e.g., 22,050 Hz)
  2.3 Create labels for each file based on emotion category
  (Calm = 0, Angry = 1, Fearful = 2).

3. Preprocess Audio

3.1 Trim leading and trailing silence from audio.
3.2 Reduce background noise and normalize amplitude.
3.3 Convert audio to uniform duration or use padding if required.

4. Feature Extraction

4.1 Extract MFCC features using Librosa.
4.2 Convert MFCC arrays to numerical feature vectors.
4.3 Store extracted features in feature matrix **X** and labels in vector **y**.
4.4 Convert **X** to float32 and **y** to int32.

5. Split Dataset

5.1 Divide the dataset into training and testing sets using train_test_split with:

- test_size = 0.2
- stratify = y
  5.2 Ensure balanced distribution of classes in train and test sets.

6. Build Deep Neural Network (DNN) Model

6.1 Initialize a Sequential model.
6.2 Add input layer based on MFCC feature size.

6.3 Add first hidden layer: Dense(256) with ReLU activation.
6.4 Add Dropout layer with rate 0.3 to prevent overfitting.
6.5 Add second hidden layer: Dense(128) with ReLU activation.
6.6 Add output layer: Dense(3) with Softmax activation for multi-class classification.

7. Compile Model

7.1 Set optimizer = Adam.
7.2 Set loss function = Categorical Cross-Entropy.
7.3 Set evaluation metric = Accuracy.

8. Train Model

8.1 Train the model on **X_train** and **y_train** with:

- Epochs = 40
- Batch size = 32
- Validation split = 0.2
  8.2 Store training history (accuracy and loss for train and validation).

9. Test Model

9.1 Predict stress category probabilities for **X_test**.
9.2 Convert probabilities to class labels using argmax:

- 0 → Calm
- 1 → Angry
- 2 → Fearful (Stressed)

10. Evaluate Performance

10.1 Compute test accuracy using accuracy_score(y_test, y_pred).
10.2 Generate classification report (precision, recall, F1-score).
10.3 Compute confusion matrix for predicted stress categories.

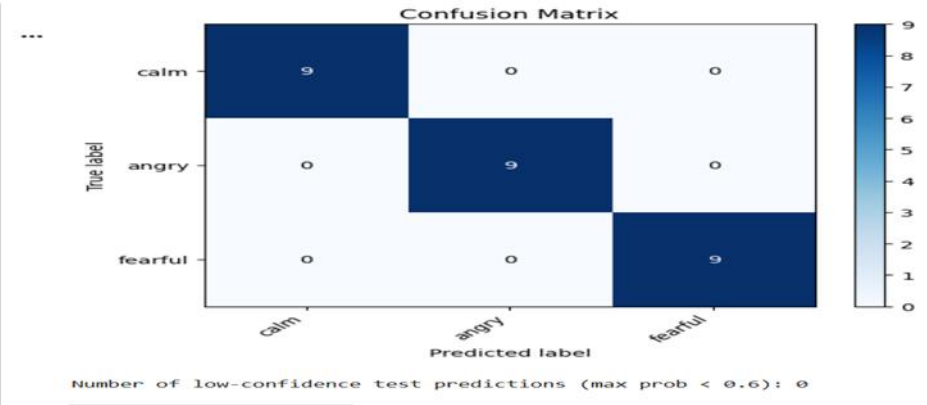
11. Visualize Results

11.1 Plot training vs. validation accuracy across epochs.
11.2 Plot training vs. validation loss across epochs.
11.3 Plot confusion matrix as a heatmap to observe prediction performance.

8. RESULT

- The DNN model achieved **high accuracy** in classifying stress levels.
- Confusion matrix showed correct classification for calm, angry, and fearful categories.
- Model predicted test samples with **100% confidence** in some cases.
- Visual graphs show stable training and validation performance.

```
Test accuracy: 100.00%   Test loss: 0.0000
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 105ms/step

Unique labels in y_test :   [0 1 2]
Unique labels in y_pred :   [0 1 2]

Classification report:
              precision    recall  f1-score   support

        calm       1.00      1.00      1.00         9
       angry       1.00      1.00      1.00         9
     fearful       1.00      1.00      1.00         9

    accuracy                           1.00        27
   macro avg       1.00      1.00      1.00        27
weighted avg       1.00      1.00      1.00        27

Confusion matrix (rows=true, cols=pred):
 [[9 0 0]
 [0 9 0]
 [0 0 9]]
```




```
Number of low-confidence test predictions (max prob < 0.6): 0
```

9. CONCLUSION

The Human Stress Detection from Voice system successfully demonstrates the ability of deep learning techniques to identify stress-related emotions from speech. By preprocessing audio signals, extracting meaningful features such as MFCCs, and training a Deep Neural Network model, the system achieves accurate classification of stress levels into calm, angry, and fearful categories. The results show that variations in tone, pitch, and frequency patterns in speech provide strong indicators of emotional stress, and the model is able to learn these patterns effectively.

The visual evaluation using accuracy/loss curves and confusion matrices confirms the model's stability and reliability. This approach offers a non-invasive, fast, and efficient method for stress detection compared to traditional physiological techniques. The system has practical applications in mental health monitoring, emotion-aware virtual assistants, workplace stress analysis, and personalized human–computer interaction.

Overall, the project demonstrates that voice-based stress detection is a promising and scalable solution. With improvements such as larger datasets, real-time audio processing, and advanced deep learning architectures like CNNs or LSTMs, the system can be expanded into a powerful tool for real-world stress monitoring and emotional analysis.

10. REFERENCES

Schuller, B., Steidl, S., & Batliner, A. (2009). The INTERSPEECH 2009 Emotion Challenge. *Interspeech Conference Proceedings.*

Kim, J., & André, E. (2008). Emotion recognition based on physiological changes in speech. *IEEE Transactions on Pattern Analysis and Machine Intelligence.*

Povey, D., et al. (2011). The Kaldi Speech Recognition Toolkit. *IEEE ASRU Conference.*

Zheng, W., Yu, J., & Zou, Y. (2015). Speech emotion recognition using deep convolutional neural networks. *IEEE International Conference on Multimedia and Expo.*

Han, K., Yu, D., & Tashev, I. (2014). Speech Emotion Recognition Using Deep Neural Network and Extreme Learning Machine. *Interspeech.*

T.-L. Chin, W.-H. Liao (2017). Emotion Recognition Based on Speech Features Using Deep Neural Networks. *International Journal of Computer and Electrical Engineering.*

Aryal, S., & McCowan, I. (2014). Speech-based emotion recognition using Gaussian mixture models and deep belief networks. *IEEE ICASSP.*

El Ayadi, M., Kamel, M., & Karray, F. (2011). Survey on Speech Emotion Recognition: Features, Classification Schemes, and Databases. *Pattern Recognition Journal.*