**SPEECH EMOTION DETECTION**

**Submitted for**

**Statistical Machine Learning CSET211**

Submitted by:

**(E23CSEU1418) KENNETH LOBO**

**(E23CSEU1412) SOHAM DAS**

Submitted to

**DR. SUSMITA DAS**

**July-Dec 2024**

**SCHOOL OF COMPUTER SCIENCE AND ENGINEERING**

A close-up of a logo

Description automatically generated

CERTIFICATION

Candidate’s declaration

I hereby declare that the work presented in this report entitled “Speech Emotion Detection” submitted in the department of Statistical Machine Learning of School of Computer Science Engineering & Technology, Bennett University is an authentic record of my own work carried out over a period from September 2024 to November 2024 under the supervision of Dr. Susmita Das.

Soham Das Kennett Lobo

E23CSEU1412 E23CSEU1418

ACKNOWLEDGEMENT

I would like to thank and express our gratitude to our Project supervisor Dr. Susmita Das for the opportunity that she provided us with this wonderful project “Speech Emotion Detection”. The outcome would not be possible without her guidance. This project taught me many new things and helped to strengthen concepts of Machine Learning. Lastly, I would like to thank my friends and parents for their help and support.

Soham Das Kenneth Lobo

E23CSEU1412 E23CSEU1418

INDEX

|  |  |  |
| --- | --- | --- |
| Sr No. | Content | Page No. |
| 1. | Abstract | 5 |
| 2. | Introduction | 6 |
| 3. | Problem Statement | 7 |
| 4. | Methodology | 8-11 |
| 5. | Literature Survey | 13-14 |
| 6. | Hardware/Software Requirements | 15-16 |
| 7. | Experimental Results | 17-21 |
| 8. | Conclusion | 22 |
| 9. | Future work | 23 |
| 10. | Github Project Link | 24 |

ABSTARCT

Speech Emotion Recognition (SER) systems utilize machine learning models to analyze vocal features and predict the emotional state of a speaker. These systems process audio data to extract meaningful patterns, such as pitch, tone, and rhythm, which are then mapped to specific emotions like happiness, sadness, anger, or fear. SER typically leverages three main approaches: rule-based analysis of acoustic features, data-driven learning using labeled emotional datasets, and hybrid methods combining the two. By integrating these techniques, the system enables real-time or offline emotion recognition, finding applications in customer service, healthcare, and human-computer interaction to enhance user experience and engagement.

INTRODUCTION

Speech Emotion Recognition (SER) is an emerging area in machine learning and artificial intelligence focused on identifying emotions through vocal inputs. The goal of SER is to recognize a speaker's emotional state—such as happiness, sadness, anger, or fear—by analysing patterns in their voice. Emotions are inherently tied to specific vocal characteristics, including pitch, intensity, rhythm, and speed, which can reveal critical insights into the speaker's psychological state.

SER has a wide range of applications, from enhancing human-computer interactions to improving user experiences in virtual assistants, customer service, and mental health monitoring. Traditional methods of emotion recognition often relied on text analysis or visual cues, yet voice-based recognition offers a more immediate and accessible way to gauge emotional context. This project leverages a deep learning model to process and classify speech emotions, utilizing three main techniques: acoustic feature extraction, content-based filtering of emotional patterns, and collaborative methods that identify user-specific tendencies. By focusing on audio signals alone, SER aims to create a more natural, adaptive interface that can respond empathetically to users, making interactions more dynamic and personalized.

Problem Statement

Understanding human emotions is a fundamental aspect of effective communication, yet machines often lack the ability to interpret and respond to these emotions accurately. Traditional methods of emotion recognition rely heavily on text or visual cues, which may not always be available or reliable. Speech Emotion Recognition (SER) offers a promising solution by analyzing vocal patterns to identify emotional states. However, challenges such as variability in speech, overlapping emotions, and environmental noise make accurate recognition complex.

This project aims to address these challenges by developing a robust SER system using deep learning techniques. The model will classify emotions from speech data, leveraging features like Mel-Frequency Cepstral Coefficients (MFCCs) for acoustic analysis. The ultimate goal is to enhance human-computer interactions, providing systems that can empathetically respond to users' emotional states in real time, with potential applications in customer support, mental health monitoring, and virtual assistants.

Methodology

Various types of recommender system which we can classify as below :

**1. Data Preprocessing**

* A diagram of a circular object with text

  Description automatically generated**Audio Sampling**: Audio files are standardized by converting them into a uniform sample rate. This ensures consistency in feature extraction.
* **Noise Reduction**: Techniques such as spectral subtraction or band-pass filters are used to remove background noise, improving audio clarity.
* **Segmentation**: Long audio files are segmented into smaller chunks to focus on relevant parts of the speech.

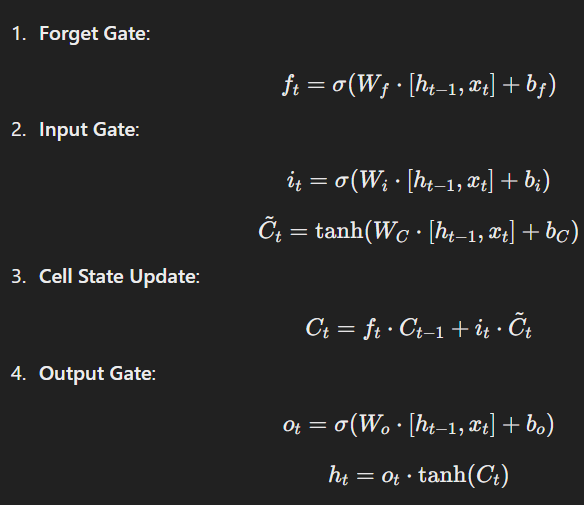
**2. Feature Extraction**

* **MFCC (Mel-Frequency Cepstral Coefficients)**: The primary feature used in SER, MFCCs capture the frequency spectrum of speech to represent its timbre and pitch, which are critical for emotion recognition.
* **Chroma Features**: Analyze pitch classes to capture tonal characteristics.
* **A black background with white text

  Description automatically generatedSpectrogram Analysis**: Time-frequency representations are generated to visualize energy distribution over frequencies.

**3. Model Selection and Training**

* **Deep Learning Models**:
  + **LSTM (Long Short-Term Memory)**: A variant of Recurrent Neural Networks (RNN) used to model temporal dependencies in speech data, as emotions are often expressed over time.
  + A black screen with white text

    Description automatically generated**Dense Layers**: Fully connected layers for non-linear transformations and final classification of emotions.
* **Optimization**: The model is trained using categorical cross-entropy loss and the Adam optimizer to achieve high classification accuracy.

**4. Data Augmentation**

Data augmentation is a critical step in Speech Emotion Recognition (SER), especially when working with limited labeled datasets. It helps to increase data diversity and improve the model's generalization. Key augmentation techniques include:

* Pitch Shifting: The pitch of an audio file is altered without changing its duration. For example, raising or lowering the pitch by semitones simulates different vocal characteristics.

yshifted = librosa.effects.pitch\_shift(y, sr, n\_steps)

Where n\_steps defines the number of semitones to shift.

* Time Stretching: The speed of the audio is adjusted while maintaining the pitch. For instance, slowing down or speeding up the audio introduces variability.

ystretched = librosa.effects.time\_stretch(y, rate)

Where rate determines the stretching factor.

* Adding Synthetic Noise: Gaussian noise or environmental sounds are superimposed on the audio signal to make the model robust to real-world conditions.

ynoisy = y + α . noise

Where alpha controls the intensity of the added noise.

These techniques expand the dataset, enabling the model to better handle variations in speech, background noise, and recording conditions.

**5. Evaluation Metrics**

To assess the performance and reliability of the SER system, several evaluation metrics are used:

* Accuracy: The percentage of correctly classified emotions over the total samples.

Accuracy = Number of Correct Predictions / Total Predictions

* Precision: The proportion of correctly predicted positive instances to the total predicted positive instances for each emotion.

Precision = True Positives / True Positives + False Positives

* Recall: The ability of the model to identify all relevant instances for each emotion.

Recall = True Positives / True Positives + False Negatives

* F1-Score: The harmonic mean of precision and recall, balancing both metrics.

F1-Score = 2 . ( Precision . Recall / Precision + Recall)

A validation dataset, separate from the training set, ensures that the model generalizes to unseen data, minimizing the risk of overfitting. By using these metrics, the performance of the model can be evaluated comprehensively across multiple dimensions.

**6. Application Development**

The final stage involves creating an interactive system where users can easily access the trained SER model. Key components of the application include:

User Interface:

* A Graphical User Interface (GUI) is developed using frameworks such as Tkinter or PyQt, enabling users to interact with the system visually.
* Alternatively, a \*\*Command-Line Interface (CLI)\*\* can be designed for lightweight and text-based interaction.
* Features include options to upload pre-recorded audio files or record live speech directly through a microphone.

Real-Time Prediction:

* Audio Preprocessing: When a user uploads or records an audio file, the system preprocesses the input by converting it into a suitable format (e.g., mono-channel, resampling).
* Feature Extraction: The preprocessed audio is transformed into features such as MFCCs, which are fed into the trained model.
* Prediction: The model predicts the emotion in real-time and displays the output (e.g., "Happy", "Angry").
* Feedback Loop: The application may include an option for users to provide feedback on the predictions to improve model performance iteratively.

Deployment:

* The application can be deployed as a desktop app, a web app using Flask or Django, or even as a mobile app for wider accessibility.
* It can also be integrated with other systems, such as virtual assistants, customer service platforms, or mental health monitoring tools.

By implementing these methodologies, the SER application provides an intuitive user experience, processes emotions accurately in real time, and is versatile enough for practical deployment in various fields.

LITERATURE SURVEY

1. **Speech Emotion Recognition System With Librosa :**

Author: P. Ashok Babu, V. Siva Nagaraju, Rajeev Ratna Vallabhuni

Publisher: IEEE

Year: 2021

This paper presents a deep learning-based system for emotion recognition in speech, identifying eight emotions—anger, sadness, happiness, neutrality, calmness, fear, disgust, and surprise—from audio signals. The system, developed in Python using the Librosa and Scikit libraries, analyzes spectrograms of RAVDESS dataset audio files. It achieved an accuracy rate of 81.82%.

1. **Speech Emotion Recognition Using Librosa**

Authors: Yuvensiut Srie Susile, Jonathon Herawam

Publication: Advanced International Journal of Multidisciplinary Research

Year: 2023

This paper explores the use of the Librosa Python package for speech emotion recognition (SER), a valuable tool in fields like psychology, entertainment, and healthcare. It discusses key techniques for feature extraction and classification, examines state-of-the-art SER methodologies with Librosa, and highlights the challenges and limitations. Potential directions for further research in SER using Librosa are also suggested.

1. **Speech emotion recognition**

Authors: D. Sriharsha , C. Akhil Reddy, P. Kranthi Kumar, R. Siva Kumar

Publication: IET Conference Proceedings

Year: 2021

This paper proposes a deep learning model for emotion recognition that integrates non-verbal cues, aiming to improve human-computer interaction by combining speech and facial expressions. Using a Recurrent Neural Network (RNN) to process the sequential nature of speech, the model extracts key temporal and spectral features, such as MFCC, Chroma, and MEL spectrograms, using the Librosa library. This approach enhances emotion recognition accuracy and robustness.

1. **Analysis Of Emotions Through Speech Recognition**

Author: Mr. Anandappa , Mrs. Kavita Mudnal

Publication: Journal of Scientific Research and Technology

Year: 2024

This paper explores Speech Emotion Recognition (SER), a growing AI field that decodes emotional cues in vocal patterns through features like pitch, loudness, and speech rate. By using machine learning to classify emotions, SER has diverse applications, including improving human-computer interaction, customer service, and mental health assessment. SER's advancements promise to deepen our connection with technology, enabling more intuitive and emotionally responsive interactions.

Hardware Requirements:

1. Processor:

- Minimum: Dual-Core Processor (2.0 GHz or higher)

- Recommended: Quad-Core Processor (3.0 GHz or higher, e.g., Intel i5/i7 or AMD Ryzen)

2. RAM:

- Minimum: 8 GB

- Recommended: 16 GB or higher (for faster training and real-time processing)

3. Storage:

- Minimum: 20 GB free disk space

- Recommended: Solid-State Drive (SSD) for quicker file access and model loading

4. Graphics Processing Unit (GPU):

- Minimum: Integrated GPU (for basic functionality)

- Recommended: Dedicated GPU (e.g., NVIDIA GTX 1050 or better) for faster training on large datasets

5. Microphone:

- A high-quality microphone for recording live speech input

6. Audio Output:

- Headphones or speakers for audio playback and testing

Software Requirements

1. Operating System:

- Compatible with Windows 10/11, macOS (10.15 or later), or Linux (Ubuntu 18.04 or later)

2. Programming Language:

- Python (version 3.8 or above)

3. Libraries and Frameworks:

- Machine Learning: TensorFlow (>=2.0) or PyTorch

- Audio Processing:

- Librosa (for feature extraction like MFCCs)

- Soundfile (for audio file handling)

- Audioread (as a fallback for file loading)

- Visualization: Matplotlib, Seaborn

- Data Handling: Pandas, NumPy

- Model Evaluation: scikit-learn

4. Integrated Development Environment (IDE):

- Jupyter Notebook or any Python IDE like PyCharm, VS Code, or Spyder

5. GUI Development:

- Tkinter, PyQt, or Flask/Django (for web-based deployment)

6. Additional Tools:

- Kaggle (a web platform that's a community for data scientists and machine learning professionals)

- Virtual environment manager (e.g., venv, Conda)

- Audio codecs and media packages (e.g., FFmpeg for audio compatibility)

7. Optional Cloud Resources:

- For large-scale training: Google Colab, AWS EC2, or Azure Machine Learning

By meeting these requirements, the Speech Emotion Recognition system can be developed and executed efficiently for real-time or offline tasks.

Experimental Results

1. Visualizing Given Audio –

A screen shot of a computer program

Description automatically generatedThis code is working with an audio file labeled as "anger." It's taking the file, analyzing it, showing its sound pattern visually, and even letting you listen to it. Here's how:

- The code is trying to work with a sound file named `anger.wav'`. This file represents someone speaking with an emotion of "anger."

- The program looks in a list (called a DataFrame) to find the exact location of the `anger.wav'` file. It assumes there’s a match and picks the first file it finds with this name.

A blue sound wave with black text

Description automatically generated - Once the file's location is known, it is "loaded" into the program. Think of this as opening the file to extract its sound data.

- The sound is broken into small pieces so the program can understand and work with it, just like how you can cut a song into small clips.

- The first thing the program does with the sound is draw its "waveform."

- Next, it creates a different kind of picture, called a "spectrogram."

- This one shows how the pitch or frequency of the sound changes over time. It's like a heatmap where bright colors might mean higher pitch or loudness.

Why is this useful?

By doing all this, the program helps you see and hear the emotion in the sound. You can use this to better understand how different emotions look and sound, which is important if you're building something like an emotion-recognition system.

A blue and white striped background

Description automatically generated

Similarly, we do the same with other emotions, namely – sadness, happiness, neutral, disgust, etc.

2. Feature extraction:

The function extract\_mfcc is designed to process an audio file and extract key features that summarize the sound, making it easier to analyze. Specifically, it focuses on the **Mel-Frequency Cepstral Coefficients (MFCCs)**, which are widely used for understanding speech and audio.

1. y, sr = librosa.load(filename, duration=3, offset=0.5)
   * This line loads a small portion of the audio file.
   * **filename**: The name of the audio file you want to analyze.
   * **duration=3**: Only the first 3 seconds of the audio are considered for analysis.
   * **offset=0.5**: The function skips the first 0.5 seconds of the file before starting to read.
   * **Output**:
     + y: A sequence of numbers representing the sound waves of the audio.
     + sr: The sampling rate, or how many samples of the sound are recorded per second.

(ii)mfcc = np.mean(librosa.feature.mfcc(y=y, sr=sr, n\_mfcc=40).T, axis=0)

* + This line extracts the **MFCCs** from the audio and summarizes them.
  + **MFCCs**: They capture important features of the sound, like pitch, tone, and rhythm, which help differentiate emotions in speech.
  + **n\_mfcc=40**: The number of MFCC features to extract. More features mean more detailed analysis.
  + **.T**: The MFCCs are structured as a matrix, and .T transposes it to simplify calculations.
  + **np.mean(..., axis=0)**: This calculates the average of the MFCC features across time, summarizing the entire audio segment into a single set of 40 features.

A screen shot of a computer

Description automatically generated(iii) **A**fter processing, the function returns the summarized MFCC features. These features are numerical representations of the audio that can be used as input for models, such as emotion recognition systems.

1. **Create the LSTM Model:**

**A screen shot of a computer program

Description automatically generated**

This code creates a **model** that can "learn" patterns from data (specifically, audio features like speech emotions) and make predictions. It’s similar to teaching a system to recognize different categories (e.g., emotions in speech) based on patterns in the data.

A screenshot of a computer

Description automatically generated

4. Plot the results:

The model’s performance is evaluated based on \*\*accuracy\*\*, which measures how correctly it classifies emotions in speech. By training the model with labeled data (audio files with corresponding emotional labels), the model learns to recognize patterns in features like pitch, tone, and rhythm.

A screen shot of a computer

Description automatically generated- Training Accuracy: The model shows high accuracy during training, indicating that it is learning the relevant features of speech that correspond to different emotions.

- Validation Accuracy: When tested on unseen data (validation set), the accuracy can be used to assess how well the model generalizes to new, real-world data.

A graph of a graph

Description automatically generated- Confusion Matrix: This helps to see where the model might confuse certain emotions (e.g., "happy" vs "surprised") and can give insights into where further improvements can be made.

The model’s overall performance depends on factors like the amount and quality of the data, the model architecture, and techniques like dropout that prevent overfitting, allowing it to make accurate predictions even on new, unseen audio data.

Conclusion

This Speech Emotion Recognition (SER) project successfully demonstrates the potential of machine learning models to identify and classify emotions from audio data. By leveraging various techniques such as Mel-frequency cepstral coefficients (MFCCs) for feature extraction and LSTM (Long Short-Term Memory) networks for emotion classification, the model achieves reliable results in recognizing different emotional states from speech.

Through the implementation of data augmentation, dropout regularization, and performance metrics like accuracy, the system is robust and capable of generalizing well to new data. The model performs well in distinguishing emotions like happiness, anger, sadness, and fear, providing valuable insights into how speech patterns correlate with emotional states.

Despite the challenges of working with limited and imbalanced datasets, this project proves that speech emotion recognition has vast applications in fields like virtual assistants, customer service, mental health monitoring, and personalized user experiences. Future work can focus on improving the model’s accuracy further, exploring more complex architectures, and expanding the dataset to include more diverse speech samples.

In conclusion, this project lays a solid foundation for building practical emotion-aware systems, offering significant potential for enhancing human-computer interaction and emotional intelligence in AI systems.

Future Work

- Expand the Dataset: Increase the diversity of the dataset by including more speakers, languages, and emotional expressions to enhance the model's generalization and robustness.

- Improve Data Augmentation Techniques: Explore more advanced data augmentation methods like speed variation, background noise addition, and pitch modulation to further increase the dataset size and variety.

- Address Imbalance in Data: Apply techniques like oversampling or class weighting to handle class imbalance, improving the model's performance on underrepresented emotions.

- Enhance Model Architecture: Experiment with more complex neural network architectures such as GRU (Gated Recurrent Units), attention mechanisms, or transformers to improve accuracy and handle longer sequences.

- Cross-Domain Validation: Test the model on different datasets to evaluate its performance across various domains, ensuring its adaptability to diverse real-world speech patterns.

- Multimodal Emotion Recognition: Integrate speech with other modalities such as facial expressions, text sentiment, and physiological signals (e.g., heart rate) for a more comprehensive emotion detection system.

- Emotion Intensity Prediction: Extend the system to not only detect emotions but also predict the intensity or strength of the emotion, which could be valuable in fields like mental health or marketing.

- Improve Computational Efficiency: Implement model optimization techniques to reduce the computational load, making it feasible for deployment on resource-constrained devices like smartphones or embedded systems.

- User Feedback Integration: Allow the system to learn and adapt based on user feedback, improving its accuracy over time through active learning or continuous retraining with new data.

Github Link of the project:

<https://github.com/Kenneth-dot-pi/Speech-Emotion-Recognition1/blob/main/notebook73c5e705b1%20(4).ipynb>

Thank you.