Speech Emotion Recognition

Tanuj Maturi   
Computer Science  
Univ of Missouri Kansas City[tm5cy@umsystem.edu](mailto:tm5cy@umsystem.edu)

Mamatha Gorantla  
Computer Science  
Univ of Missouri Kansas City[mgxh5@umsystem.edu](mailto:mgxh5@umsystem.edu)

Jithendra  
Computer Science  
Univ of Missouri Kansas City[jpgpy@umsystem.edu](mailto:jpgpy@umsystem.edu)

Harini  
Computer Science  
Univ of Missouri Kansas City[st3pc@umsystem.edu](mailto:st3pc@umsystem.edu)

# INTRODUCTION

Speech Emotion Recognition (SER) aims to automatically detect and categorize emotions expressed by voice signals. Our study examines the RAVDESS dataset using machine learning (ML) approaches such as K-Nearest Neighbors (KNN), Multilayer Perceptron (MLP), and deep learning methodologies such as Convolutional Neural Networks (CNN) and Gated Recurrent Units (GRU). This dataset contains a large collection of performed speech that depicts various emotional emotions. To achieve robust and exact emotion recognition, our work incorporates feature extraction methods such as MFCC (Mel-frequency cepstral coefficients) extraction and data augmentation approaches such as noise addition to improve model generalization.

Our main goal is to reliably categorize emotions including neutral, happy, sadness, angry, fear and surprise states using voice data. Our study aims to expand the field of emotion identification from speech by using a wide range of machine learning and deep learning methodologies. This has a wide range of applications, from affective computing and improved human-computer interaction to improving mental health monitoring and interventions.

# RELATED WORK

Speech emotion recognition (SER) has attracted a lot of attention lately because of its uses in affective computing and human-computer interaction. Using a variety of machine learning and deep learning approaches, the state of the art in SER is extracting discriminative features from audio signals and classifying them into distinct emotional categories. The usefulness of techniques like support vector machines (SVM), random forests, convolutional neural networks (CNNs), and recurrent neural networks (RNNs) for SER tasks has been investigated in earlier studies on tasks that are comparable to this one. CNN-based models, for example, have demonstrated potential in automatically learning hierarchical representations of audio characteristics, while RNN variations such as Long Short-Term Memory (LSTM) networks and Gated Recurrent Units (GRUs) are excellent at capturing temporal relationships in speech data. Furthermore, methods such as Multilayer Perceptron (MLPs) and k-Nearest Neighbors (KNN) have been investigated due to their ease of use and efficiency in classification tasks. Utilizing these methods with reference datasets such as RAVDESS (Ryerson Audio-Visual Database of Emotional Speech and Song) offers a useful starting point for investigating more complex SER models. Subsequent endeavors within this field may concentrate on refining model structures, characteristic depictions, and amalgamation tactics to augment SER efficacy in a variety of datasets and practical scenarios.

# METHODOLOGY

1. EDA

The RAVDESS dataset is a valuable resource for emotion recognition in speech and audio analysis. Created by Ryerson University researchers, it comprises 1440 audio files from 24 professional actors (12 male, 12 female) expressing eight emotions: neutral, calm, happy, sad, angry, fearful, disgust, and surprise. Each file's filename encodes details like emotion, intensity, statement type, repetition number, actor, and gender. Recordings were made in controlled conditions with actors instructed to convey specific emotions. This standardized and well-labeled dataset supports training and evaluating machine learning models for emotion recognition, benefitting research in affective computing and human-computer interaction.

We extracts metadata such as emotion labels, gender, and file paths from the audio filenames and creates a pandas Data Frame. It visualizes the distribution of emotions in the dataset using a bar plot.

A graph of emotions

Description automatically generated

Figure 1 Bar chart of count on diff emotion.

A close-up of a purple and pink image

Description automatically generated

Figure 2 spectrogram chart shows the frequency over time.

The first image is a bar plot that illustrates the evenly distributed emotion labels (happy, neutral, sad, angry, fear, surprise, and disgust) across the genders in the RAVDESS dataset. A mel spectrogram representation of an audio sample with a happy female performer can be seen in Image 2. Higher energy at particular frequencies is shown by brighter colors in the spectrogram, which shows the frequency content with time. The horizontal axis displays time, and the vertical axis depicts the mel frequency scale. This visual aid facilitates the analysis of the spectral properties linked to the happy emotional speech sample, providing insights into the relationship between various emotional expressions in the dataset and audio elements.

1. DATA AUGMENTATION

To minimize overfitting and improve test set performance, a data augmentation phase is added, with the goal of increasing the amount of samples to allow for model generalization. This entails picking a raw audio sample and executing a transformation, with parameters determining the severity of the change. Four transformations are used: speed alteration (random speed increase or decrease with a ratio of 0.6 to 1.4), pitch adjustment (random shift up or down by -4 to +4 semitones), noise addition (gaussian noise with amplitudes ranging from 0.0005 to 0.002), and shift (shifting the audio sample by a random range of -5 to 5 milliseconds). Each augmenter applies to the whole dataset, resulting in a cardinality four times that of the original. Importantly, data augmentation is only applied to the training split, leaving the test split unaffected. This method helps to reduce overfitting while increasing model durability and performance.

A few sound waves in different directions

Description automatically generated with medium confidence

Figure 3 Different audio samples after Augmentation.

When enhancing data, it's important to maintain plausibility. Excessive data augmentation leads to ineffective generalization, whereas insufficient data augmentation repeats the training set. The augmentation ranges are empirically chosen after several trials.

1. FEATURE EXTRACTION

Mel-frequency cepstral coefficients (MFCCs) are a popular feature extraction technique used in various audio and speech processing tasks, including speech recognition, speaker identification, and emotion recognition. MFCCs are derived from the short-term power spectrum of an audio signal by applying the mel-frequency scale and decorrelating the mel spectrum. The mel-frequency scale is based on the human auditory system's response to different frequencies, which is more sensitive to lower frequencies than higher ones. The cepstral representation separates the source and filter components of the audio signal, making it effective for extracting relevant features. MFCCs capture the spectral envelope shape, which carries important information about the audio's timbre and formant structure, making them useful for tasks that involve analyzing and classifying different audio characteristics, such as speech emotions.

**A screenshot of a graph

Description automatically generated**

Figure 4 MFCC features (numbered 0 to 19), then 'labels' column.

The provided output appears to be a preview of the first few rows and columns of the 'Emotions' DataFrame created in the previous code snippet. Each row represents a feature vector extracted from an audio file, along with its corresponding emotion label. The columns represent the individual MFCC features (numbered 0 to 19), followed by the 'labels' column containing the emotion labels. The values in the feature columns are likely the mean MFCC values computed by the feat\_ext function. The emotion labels seem to be in the format 'gender\_emotion', such as 'male\_happy' or 'male\_neutral'. This DataFrame contains the extracted features and labels, which can be used for training a machine learning model for speech emotion recognition.

1. MODEL DEVELOPMENT

For the speech emotion recognition (SER) task on the RAVDESS dataset, we employed a combination of machine learning (ML) and deep learning (DL) models. We began with traditional ML techniques like k-Nearest Neighbors (KNN) and Multilayer Perceptrons (MLPs) for feature-based classification. Additionally, we utilized DL architectures such as Convolutional Neural Networks (CNNs) for automatic feature extraction from spectrograms, and Gated Recurrent Units (GRUs) for capturing temporal dependencies in speech sequences. These models were trained and evaluated on labeled audio data to classify emotions accurately.

Preprocessing steps for a machine learning model on emotions data. Feature (X) and label (Y) extraction from the dataset is the initial step. One-hot encoding of the labels follows. Training and testing sets make up the dataset. Features are normalized by applying standard scale. Consequently, the dimensions show that there were 1080 samples with the same features for testing and 3240 samples with 20 features for training, connected to 14 emotion classes. These preprocessing processes lay the basis for developing and accessing an accurate machine learning model for emotion classification.

## KNN

KNN (K-Nearest Neighbors) is a supervised learning algorithm designed for classification and regression. It works by locating the K nearest data points to a new data point in the feature space and then assigning the majority class to those neighbors (classification) or averaging their target values (regression). The technique computes distances between locations using metrics such as Euclidean distance and chooses the K nearest neighbors to generate predictions based on their labels or target values.

KNN can be helpful for emotion recognition by classifying new audio or speech data based on its similarity to labeled training examples with known emotions. The algorithm identifies the nearest neighbors and assigns the most common emotion label from those neighbors. The KNeighborsClassifier from scikit-learn is initialized with n\_neighbors=4, setting the number of nearest neighbors to consider for classification to 4. The classifier is trained on the x\_train and y\_train data from the RAVDESS dataset, which contains features and corresponding emotion labels. Predictions are made on the x\_test data using clf1.predict(x\_test). The accuracy scores are calculated for both the training set (0.587) and test set (0.421) using clf1.score(). The relatively low scores suggest that KNN with 4 neighbors may not be the ideal model for emotion recognition on this dataset. Adjusting the number of neighbors or trying different algorithms could potentially improve performance.

## MLP

A Multi-Layer Perceptron classifier is a type of artificial neural network used to perform classification tasks. It is made up of several layers of interconnected nodes or neurons, such as an input layer, one or more hidden layers, and an output layer. The MLP learns to translate input features to output classes by altering the weights between neurons while training on labeled data. During prediction, it computes activations via the layers to determine the most probable class.

MLP Classifier can effectively model complex relationships between audio features and emotion labels through its multi-layer neural network architecture, potentially improving emotion recognition accuracy. The MLP Classifier from scikit-learn is initialized with various hyperparameters like alpha, batch\_size, hidden\_layer\_sizes, and max\_iter. It is trained on the x\_train and y\_train data. The accuracy scores are then calculated on the training set (0.933) and test set (0.627). The high training accuracy but lower test accuracy indicates potential overfitting, where the model has learned the training data too well but struggles to generalize to unseen data. Techniques like regularization, early stopping, or increasing training data size could help improve the model's generalization performance.

## GRU

The GRU is a type of recurrent neural network architecture designed to handle sequential input. It comprises of two gates: reset and update. The reset gate specifies how much of the prior state should be ignored while computing the new state, hence capturing short-term dependence. The update gate determines how much of the prior state should be maintained and how much of the new candidate state should be introduced, which helps capture long-term dependencies. The GRU combines the cell and hidden states into a single state, making it more efficient than LSTMs. GRUs can successfully simulate long-term and short-term patterns in sequential data by selectively updating their states based on the input and prior states, making them suitable for applications such as language modeling, machine translation, and time series forecasting.

GRU design is critical for accurately capturing temporal relationships in speech signals. The model consists of many GRU layers, each with 50 units, which allow for detailed pattern detection in sequential data. By integrating return sequences, the GRU layers preserve temporal information over numerous time steps, allowing the model to detect tiny differences in speech that indicate distinct emotional states. Dropout layers with a rate of 0.3 are interleaved to reduce overfitting and improve the model's generalizability. The last dense layer with 14 units is the output layer, which maps the retrieved characteristics to different emotion classes. The model attained an accuracy of around 23.70% on the test data, demonstrating its ability to reliably categorize emotions from voice data.

## CNN

CNNs are a class of deep learning architecture that has demonstrated remarkable efficacy across a range of computer vision applications, including object identification, picture segmentation, and image recognition. CNNs are ideally suited for processing data having a grid-like structure, such as pictures or time-series data, like audio signals, because of their ability to automatically and adaptively learn spatial hierarchies of characteristics from their input data. CNNs are able to efficiently capture the local patterns in these types of data by taking use of the substantial spatially local correlation that exists in them. This enables effective feature extraction and reduces computing cost.

Our CNN architecture is specifically designed to recognize speech emotions. The max-pooling and batch normalization layers come after the 1D convolutional layer, which has a lot of filters (2048) and a small kernel size (5). To capture characteristics at various scales, this pattern is repeated with decreasing filter sizes (1024, 512). From the input audio data, low-level and high-level features are extracted by the convolutional layers. LSTM (Long Short-Term Memory) layers are included in the design after the convolutional layers. These layers are ideal for capturing temporal dependencies in sequential input, such as voice. The last layers are dense (completely connected) layers with dropout regularization. They use a SoftMax activation function to aggregate the learnt data. On the test set, the model's accuracy was 86.20%.

# RESULTS AND DISSCUSSIONS

The CNN model outperformed the other models, including MLP (Training set score: 0.933, Test set score: 0.627), KNN (Training set score: 0.587, Test set score: 0.421), and GRU (Accuracy on test data: 23.70%). CNNs are especially useful for applications requiring image or spatial data because they can automatically learn and extract significant features via convolutional and pooling layers, making them ideal for capturing local patterns and spatial correlations in input data. The MLP overfitted, resulting in a considerable reduction in performance from training (93.3%) to test set (62.7%). The KNN and GRU models performed badly, possibly owing to underfitting, incorrect model selection, or improper hyperparameters. Underperformance might be caused by overfitting, underfitting, poor model selection, wrong hyperparameter tuning, data quality difficulties, or a suboptimal model architecture.

1. Accuracy of all models

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Models | KNN | MLP | GRU | CNN |
| Accuracy | 42.1 | 62.7 | 23.7 | 86.2 |

In the below image, On training data, the training loss graph exhibits a smooth fall that stabilizes around 0.0636, demonstrating effective error reduction. On the other hand, the testing loss curve varies about 0.0631, which may indicate overfitting or difficulties extrapolating to unobserved validation data, which might affect the predictions of emotional states.

A screenshot of a graph

Description automatically generated

In the below image, A steady improvement in training accuracy to about 21% indicates that the training data were well learned. The testing accuracy, which peaks at about 23.9% and sporadically falls below 22%, suggests that there may be issues with generalizing to fresh validation data and may necessitate more tweaking. A graph of blue and orange lines

Description automatically generated

In the below image, Over 150 epochs, the training loss gradually drops, suggesting that the training data is well learned, and it eventually reaches a very low value. On the other hand, the testing loss shows more peaks and troughs in its oscillations, which may indicate overfitting or difficulties when extrapolating to data that has not yet been observed.

A graph of a graph of a graph

Description automatically generated with medium confidence

In the below image, The training accuracy increases significantly, hitting a stunning 98.15%, demonstrating the model's potent learning potential. The testing accuracy of 86.2%, albeit marginally lower than expected, indicates strong generalization to new data. On the other hand, the difference in accuracy between training and testing stages suggests that overfitting may have occurred, or that more regularization may be required to improve generalization performance.

A graph of a graph

Description automatically generated with medium confidence

##### FUTURE SCOPE

Future work can involve exploring more advanced neural network architectures, such as attention mechanisms or transformer models, to better capture long-range dependencies in speech data. Additionally, incorporating multimodal information, such as facial expressions or body language, could provide a more comprehensive understanding of emotional states. Ensemble methods and transfer learning techniques could also be investigated to leverage the strengths of multiple models and pre-trained representations.

##### CONCLUSION

Through this project, we explored various machine learning and deep learning models for speech emotion recognition on the RAVDESS dataset. The CNN model achieved the highest accuracy of 86.2%, outperforming traditional approaches like KNN and MLP. Data augmentation and feature extraction techniques like MFCC were employed to enhance model performance. Despite promising results, there is still room for improvement in capturing the nuances of emotional expressions in speech.

##### REFERENCES

1. Subbarao, M. & Terlapu, Sudheer. (2021). Speech Emotion Recognition Using K-Nearest Neighbor Classifiers. 10.1007/978-981-16-3342-3\_10.
2. M. Venkata Subbarao, Sudheer Kumar Terlapu, Nandigam Geethika, & Kudupudi Durga Harika. (2021). Speech Emotion Recognition Using K-Nearest Neighbor Classifiers. Advances in Intelligent Systems and Computing, 123–131. https://doi.org/10.1007/978-981-16-3342-3\_10
3. Aouani, H., & Ayed, Y. B. (2020). Speech Emotion Recognition with deep learning. Procedia Computer Science, 176, 251–260. https://doi.org/10.1016/j.procs.2020.08.027
4. R. Anusha, P. Subhashini, D. Jyothi, P. Harshitha, J. Sushma and N. Mukesh, "Speech Emotion Recognition using Machine Learning," 2021 5th International Conference on Trends in Electronics and Informatics (ICOEI), Tirunelveli, India, 2021, pp. 1608-1612, doi: 10.1109/ICOEI51242.2021.9453028.
5. K. S. Chintalapudi, I. A. K. Patan, H. V. Sontineni, V. S. K. Muvvala, S. V. Gangashetty and A. K. Dubey, "Speech Emotion Recognition Using Deep Learning," 2023 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, 2023, pp. 1-5, doi: 10.1109/ICCCI56745.2023.10128612.
6. Zhao, Y., Shu, X. Speech emotion analysis using convolutional neural network (CNN) and gamma classifier-based error correcting output codes (ECOC). Sci Rep 13, 20398 (2023). https://doi.org/10.1038/s41598-023-47118-4