SPEECH EMOTION RECOGNITION

A PREPRINT

Arzu Isik Topbas
arzuu.isik@gmail.com
Introduction to Artificial Intelligence

December 20, 2023

ABSTRACT

This research project undertakes a comprehensive analysis of speech emotion recognition. Harmonizing datasets involves negotiating disparate naming conventions and emotional expressions, establishing a standardized format for cohesive analysis. The distribution of emotions, excluding surprise, calm, and neutral, is well-balanced. The subsequent focus is on feature extraction through raw audio waveforms, frequency spectrum (FFT), short-time Fourier transform (STFT) spectrograms, and mel spectrograms. Three distinct convolutional neural network (CNN) models—Mel Spectrogram CNN, MFCCs CNN, and Mel Spectrogram CRNN—are developed and evaluated. Results indicate a 72% accuracy in classifying emotions, with Mel spectrogram and MFCC features displaying complementary strengths. The study concludes by suggesting avenues for improvement, emphasizing feature fusion, exploring specialized deep learning models, and addressing data imbalance. Future work involves real-life integration, applying sentiment analysis to predict stock market effects based on emotion-laden communications in S&P 500 earnings calls.

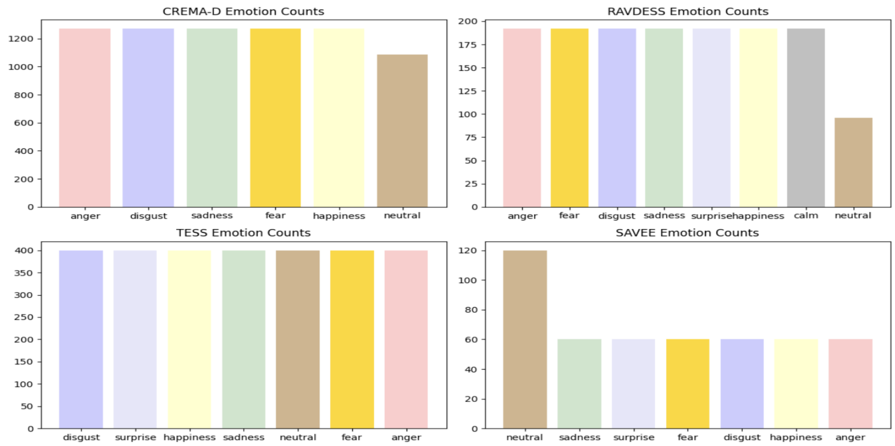
1 Introduction

Speech emotion recognition (SER) has emerged as a pivotal field with applications ranging from human-computer interaction to market prediction. This research bridges diverse emotional datasets necessitating harmonization for meaningful analysis. The project's core revolves around feature extraction methodologies, spanning raw audio waveforms, frequency spectrum analysis (FFT), short-time Fourier transform (STFT) spectrograms, and mel spectrograms. The subsequent development and evaluation of Mel Spectrogram CNN, MFCCs CNN, and Mel Spectrogram CRNN models contribute to the understanding of the strengths and limitations of different feature extraction methods. The study not only sheds light on the intricacies of emotion classification but also proposes avenues for enhancing accuracy. Looking forward, real-life integration aims at leveraging SER techniques to predict stock market effects based on emotional signals in S&P 500 earnings calls, providing investors with nuanced insights for more informed decision-making.

2 Methodology and Data

2.1 Data

This research project will utilize four datasets obtained from KAGGLE, namely RAVDESS, TESS, SAVEE, and CREMA-D. Each dataset features unique nomenclature for audio files and labels, and encompasses a diverse array of emotional states. Consequently, the harmonization of these datasets necessitates negotiations to establish a standardized format. Given the disparate naming conventions and the varied emotional expressions encapsulated within each dataset, the project will focus on aligning and unifying these elements to facilitate a cohesive and comprehensive analysis of speech emotion recognition utilizing artificial intelligence technologies.



As anticipated, the distribution of emotions in the dataset is well-balanced, except for surprise, calm, and neutral. Due to the similarity between the calm and neutral emotions, we will exclude them from further consideration.

2.2 Raw Audio Waveforms - Signal

A signal represents variations in a specific quantity over time. In the context of audio, this quantity is the air pressure. By sampling the air pressure at regular intervals over time, we create a waveform. The rate at which we sample this data can vary, but a common standard is 44.1kHz, corresponding to 44,100 samples per second. The resulting waveform captures the characteristics of the audio signal, and it can be further interpreted, modified, and analyzed using computer software. Appendix A.1

2.3 Frequency Spectrum (FFT)

The Fourier Transform is a mathematical tool essential for understanding audio signals. Audio signals consist of various single-frequency sound waves. The Fourier Transform breaks down a signal into its individual frequencies and amplitudes, transforming it from the time domain to the frequency domain, resulting in a spectrum. This is possible thanks to Fourier's theorem, which states that any signal can be expressed as a combination of sine and cosine waves. The fast Fourier transform (FFT) algorithm efficiently computes this transformation and is widely used in signal processing. Applying FFT to windowed segments of audio enables detailed analysis of frequency components in specific time frames, crucial for tasks like audio processing and analysis. Appendix A.2

2.4 Short-Time Fourier Transform (STFT) Spectrogram

While the FFT excels at analyzing static signal frequencies, it struggles with dynamic signals like music or speech. To address this, we employ the short-time Fourier transform (STFT), computing FFTs on overlapping windows to create a spectrogram. Visualized as stacked FFTs, the spectrogram represents a signal's amplitude variations over time and frequencies. Behind the scenes, the y-axis is on a log scale, and color is in decibels, aligning with human perception's sensitivity to specific frequency and amplitude ranges. The spectrogram offers a concise visual insight into dynamic frequency changes in audio signals.Appendix A.3

2.5 Mel Spectrogram

2.5.1 The Mel Scale

Research indicates that human perception of frequencies is not linear. We're more adept at discerning differences in lower frequencies than in higher ones. To address this, Stevens, Volkmann, and Newmann proposed the mel scale in 1937, aiming to create a pitch unit where equal pitch intervals sound equally distant to listeners. Consequently, we apply a mathematical operation to convert frequencies to the mel scale.Appendix A.4

2.5.2 The Mel Spectrogram

A mel spectrogram transforms frequencies into the mel scale. Surprisingly, this seemingly complex concept translates into just a few lines of code, making the implementation of a mel spectrogram remarkably straightforward.

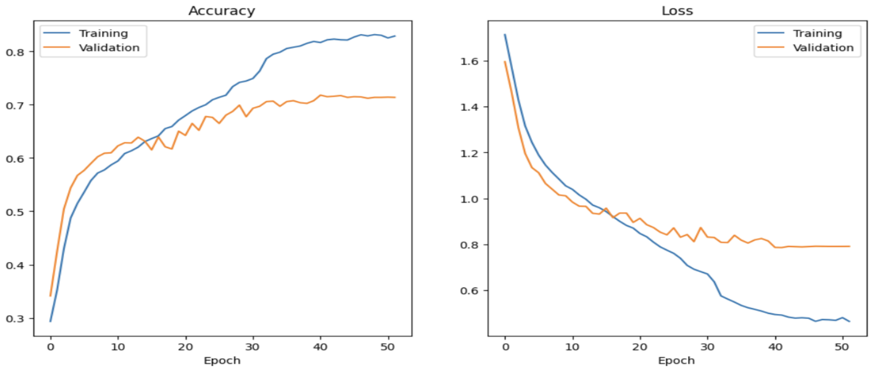
3 Modeling

3.1 Mel Spectrogram CNN Model

The initial step involves encoding emotion labels using the LabelEncoder and transforming audio files into mel spectrograms. The dataset is split into training and testing sets, with additional standardization for data preprocessing.

The Mel Spectrogram CNN Model employs a classical convolutional neural network structure. It consists of multiple convolutional layers, each followed by max-pooling, and a global average pooling layer to capture essential features. Dropout is incorporated for regularization, and the model is finalized with a dense layer using softmax activation for multi-class classification.

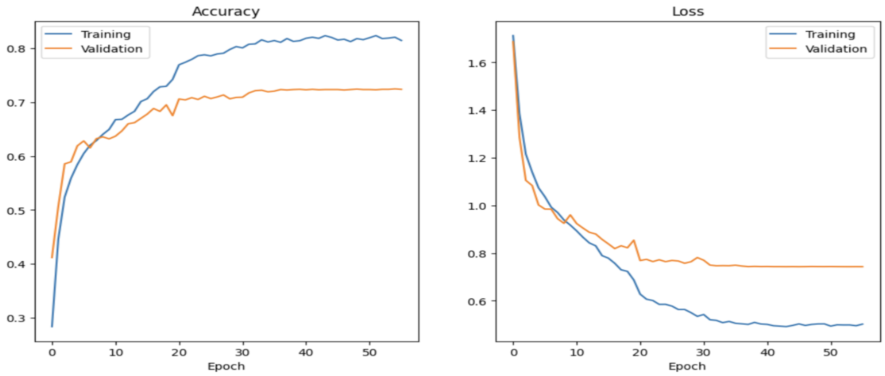
Training spans 100 epochs with early stopping and learning rate scheduling to enhance convergence. The model demonstrates a respectable 72% accuracy on the test set. The class-wise metrics, including precision, recall, and F1-score, provide insights into the model's performance across different emotion categories.



3.2 MFCCs CNN Model

Similar to the Mel Spectrogram CNN Model, this model extracts Mel-Frequency Cepstral Coefficients (MFCCs) from audio files. Data preprocessing involves standardization and reshaping to prepare the input for the subsequent CNN layers.

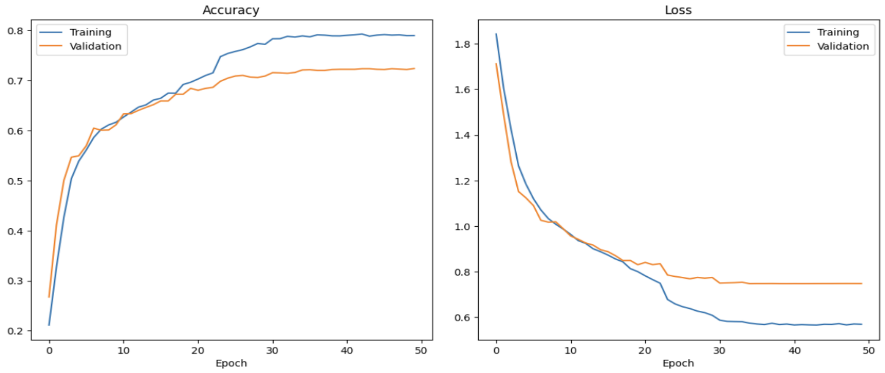
The MFCCs CNN Model shares similarities with the Mel Spectrogram CNN Model but introduces batch normalization after the first convolutional layer. This additional layer aids in maintaining stable activations throughout training, potentially contributing to the model's robustness. The training setup mirrors the Mel Spectrogram CNN Model, with comparable results. The model achieves a 72% accuracy on the test set, and class-wise metrics reveal its performance on individual emotion classes.



3.3 Mel Spectrogram CRNN Model

The Mel Spectrogram CRNN Model blends convolutional and recurrent layers. Convolutional layers are followed by bidirectional LSTM layers, adding a temporal aspect to feature extraction. The global average pooling, reshaping, dropout, and dense softmax layer complete the architecture.

Training parameters remain consistent with the previous models. The Mel Spectrogram CRNN Model achieves a 72% accuracy on the test set, showcasing the effectiveness of combining both convolutional and recurrent layers for audio emotion recognition.



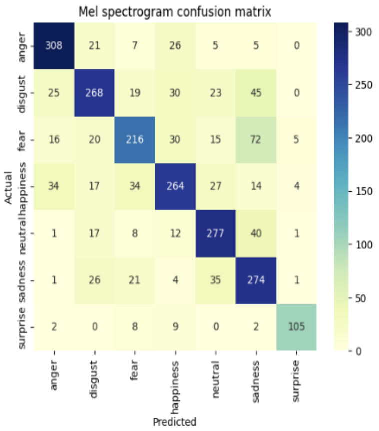
4 Conclusion

Analyzing the results, we see that both Mel spectrogram and MFCC features achieve similar overall accuracy of 72% in classifying emotions from speech. However, their strengths and weaknesses differ. Mel spectrogram excels in identifying surprise (0.87 F1-score) and neutral emotions (0.75 F1-score), likely due to its effective representation of high-frequency components and spectral variations. MFCC, on the other hand, shines in recognizing anger (0.82 F1-score) and disgust (0.69 F1-score), potentially benefiting from its focus on perceptually relevant features.

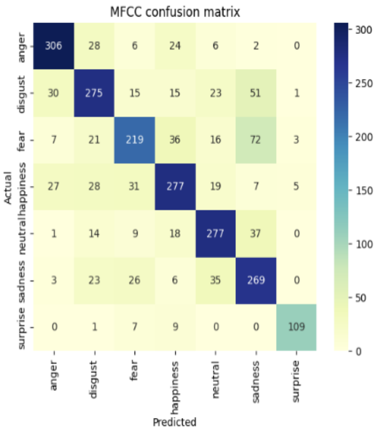
However, both features struggle with fear (0.63 F1-score for Mel spectrogram, 0.64 for MFCC) and sadness (0.67 F1-score for both). This suggests that capturing the subtle acoustic cues associated with these emotions requires further exploration. Interestingly, CRNN performs comparably in terms of overall accuracy but exhibits slightly lower scores for anger and fear.

Overall, these findings highlight the strengths and limitations of different feature extraction methods in emotion recognition. While Mel spectrogram and MFCC demonstrate promising results, potential avenues for improvement include:

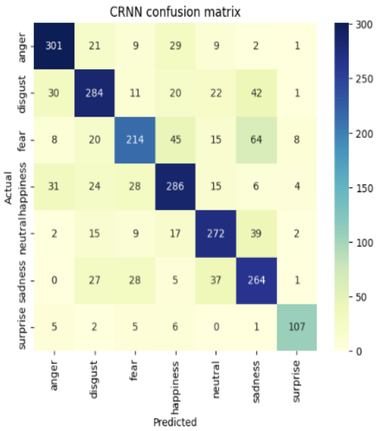
Feature fusion: Combining Mel spectrogram and MFCC features might leverage their complementary strengths. Deep learning models: Exploring CNNs or LSTMs specifically designed for audio tasks could potentially improve accuracy and address nuanced emotions. Data balancing: Over-sampling minority classes like fear and sadness could enhance model performance on these emotions.


Mel spectrogram confusion matrix
Actual
Predicted
300
250
200
150
100
50
0

	anger	disgust	fear	happiness	neutral	sadness	surprise
anger	308	21	7	26	5	5	0
disgust	25	268	19	30	23	45	0
fear	16	20	216	30	15	72	5
happiness	34	17	34	264	27	14	4
neutral	1	17	8	12	277	40	1
sadness	1	26	21	4	35	274	1
surprise	2	0	8	9	0	2	105


MFCC confusion matrix
Actual
Predicted
300
250
200
150
100
50
0

	anger	disgust	fear	happiness	neutral	sadness	surprise
anger	306	28	6	24	6	2	0
disgust	30	275	15	15	23	51	1
fear	7	21	219	36	16	72	3
happiness	27	28	31	277	19	7	5
neutral	1	14	9	18	277	37	0
sadness	3	23	26	6	35	269	0
surprise	0	1	7	9	0	0	109


CRNN confusion matrix
Actual
Predicted
300
250
200
150
100
50
0

	anger	disgust	fear	happiness	neutral	sadness	surprise
anger	301	21	9	29	9	2	1
disgust	30	284	11	20	22	42	1
fear	8	20	214	45	15	64	8
happiness	31	24	28	286	15	6	4
neutral	2	15	9	17	272	39	2
sadness	0	27	28	5	37	264	1
surprise	5	2	5	6	0	1	107

5 Real Life Integration (Future Work)

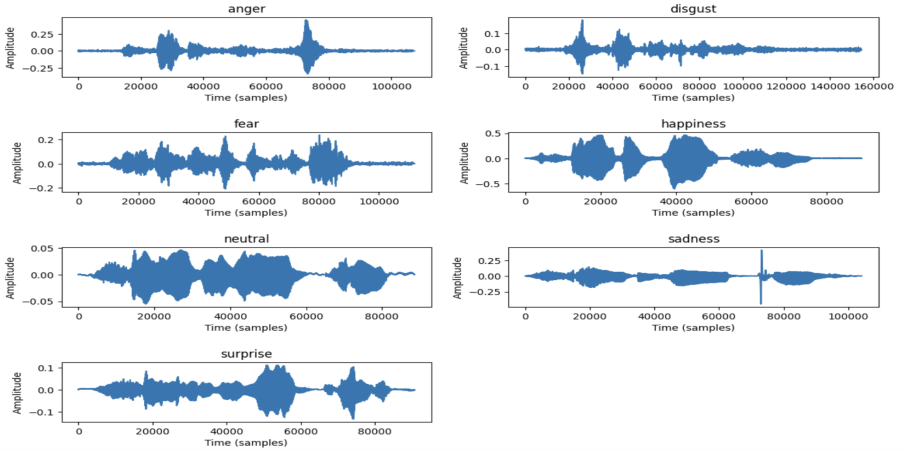
The application of sentiment analysis utilizing Speech Emotion Recognition (SER) on next quarterly earnings calls from S&P 500 businesses could be the real-world integration and future development. The main goal is to use SER approaches to identify and measure the emotional expressions that speakers are making throughout these calls. Crucially, the study takes into consideration the speech's content, making sure that the emotional impact is separated from and assessed separately from the informational content offered.

The research attempts to forecast potential stock market consequences by using the SER-based analysis of emotional signals, such as tone, attitude, and expression, in the upcoming earnings calls. By using this study to find patterns and trends in emotional signals that may influence market behavior, investors might gain predictive insights. Because the incorporation of SER into the sentiment research process indicates a more advanced understanding of emotional dynamics, investors may now proactively consider these emotional subtleties in addition to conventional financial measurements.

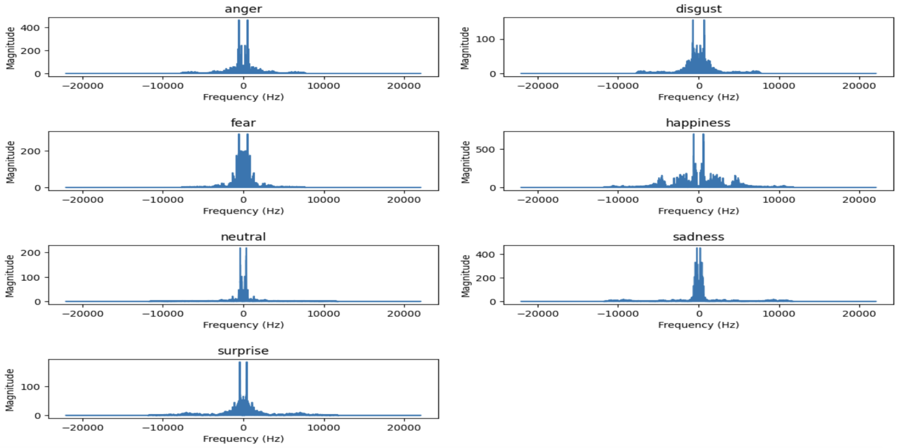
The ultimate goal is enhancing investors' decision-making capacity by offering a predictive, code-driven understanding of emotional nuances in the communications of S&P 500 corporations during their upcoming earnings calls. Thanks to the helpful technique known as SER-based sentiment analysis, investors may be able to make better decisions based on both the real financial data and the expected emotional impact concealed in the speeches.

A Appendix

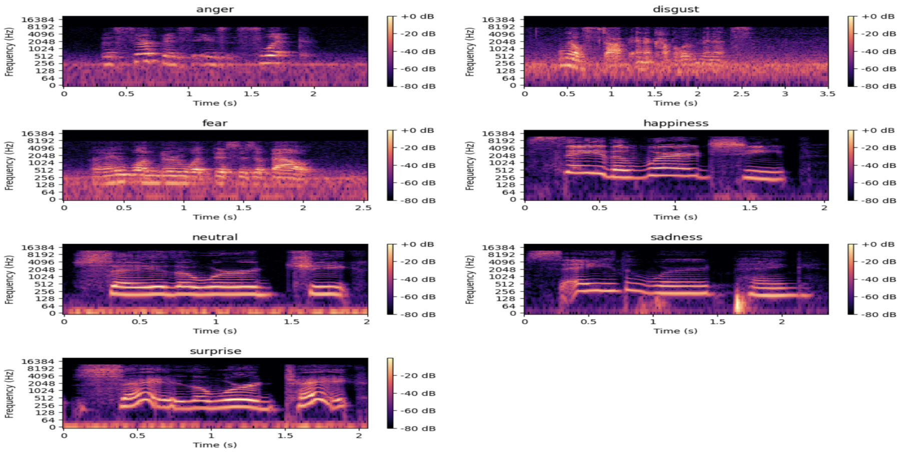
A.1 Raw Audio Waveforms - Signal



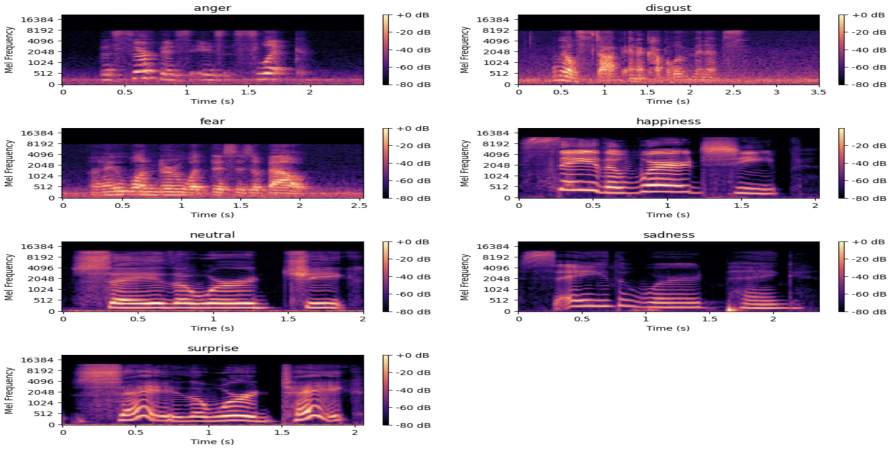
A.2 Frequency Spectrum (FFT)



A.3 Short-Time Fourier Transform (STFT) Spectrogram

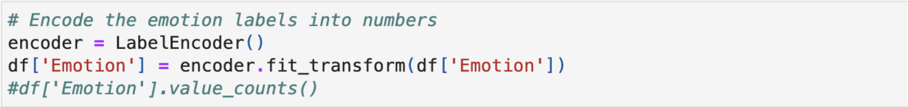


A.4 Mel Spectrogram

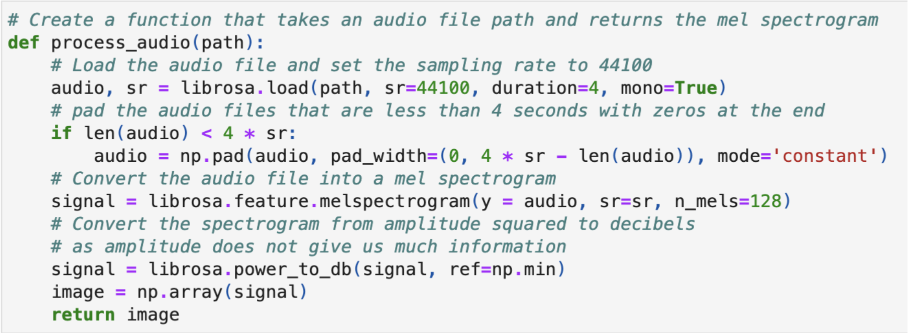


A.5 Python Code

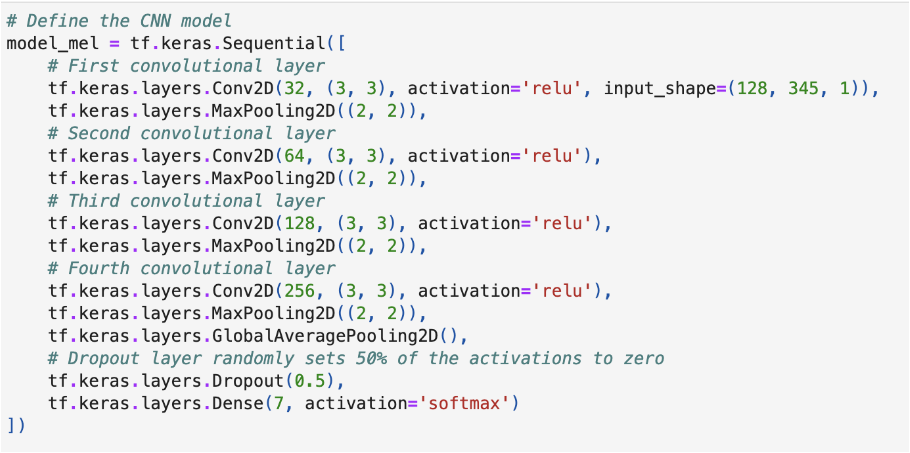
A.5.1 Mel Spectrogram CNN Model



```
# Encode the emotion labels into numbers
encoder = LabelEncoder()
df['Emotion'] = encoder.fit_transform(df['Emotion'])
#df['Emotion'].value_counts()
```

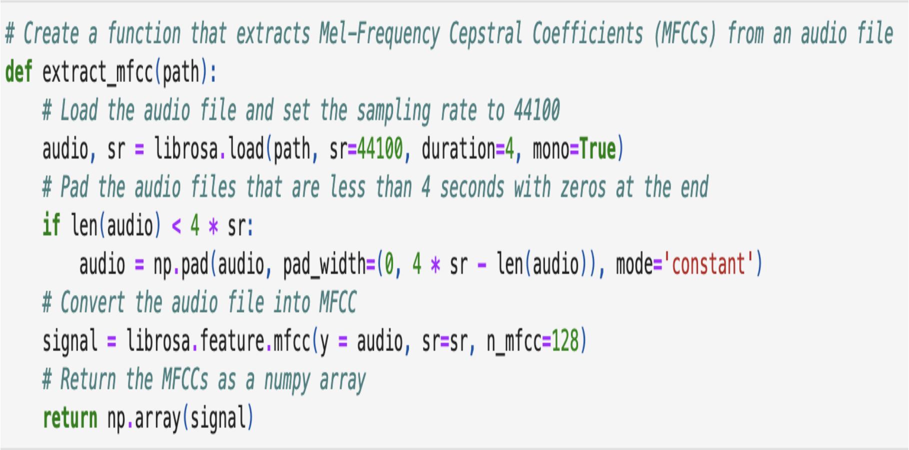


```
# Create a function that takes an audio file path and returns the mel spectrogram
def process_audio(path):
    # Load the audio file and set the sampling rate to 44100
    audio, sr = librosa.load(path, sr=44100, duration=4, mono=True)
    # pad the audio files that are less than 4 seconds with zeros at the end
    if len(audio) < 4 * sr:
        audio = np.pad(audio, pad_width=(0, 4 * sr - len(audio)), mode='constant')
    # Convert the audio file into a mel spectrogram
    signal = librosa.feature.melspectrogram(y = audio, sr=sr, n_mels=128)
    # Convert the spectrogram from amplitude squared to decibels
    # as amplitude does not give us much information
    signal = librosa.power_to_db(signal, ref=np.min)
    image = np.array(signal)
    return image
```

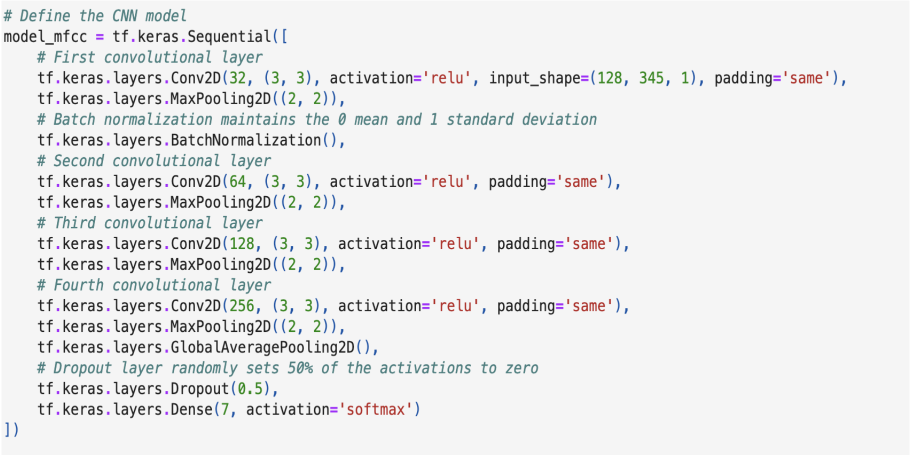


```
# Define the CNN model
model_mel = tf.keras.Sequential([
    # First convolutional layer
    tf.keras.layers.Conv2D(32, (3, 3), activation='relu', input_shape=(128, 345, 1)),
    tf.keras.layers.MaxPooling2D((2, 2)),
    # Second convolutional layer
    tf.keras.layers.Conv2D(64, (3, 3), activation='relu'),
    tf.keras.layers.MaxPooling2D((2, 2)),
    # Third convolutional layer
    tf.keras.layers.Conv2D(128, (3, 3), activation='relu'),
    tf.keras.layers.MaxPooling2D((2, 2)),
    # Fourth convolutional layer
    tf.keras.layers.Conv2D(256, (3, 3), activation='relu'),
    tf.keras.layers.MaxPooling2D((2, 2)),
    tf.keras.layers.GlobalAveragePooling2D(),
    # Dropout layer randomly sets 50% of the activations to zero
    tf.keras.layers.Dropout(0.5),
    tf.keras.layers.Dense(7, activation='softmax')
])
```

A.5.2 MFCCs CNN Model

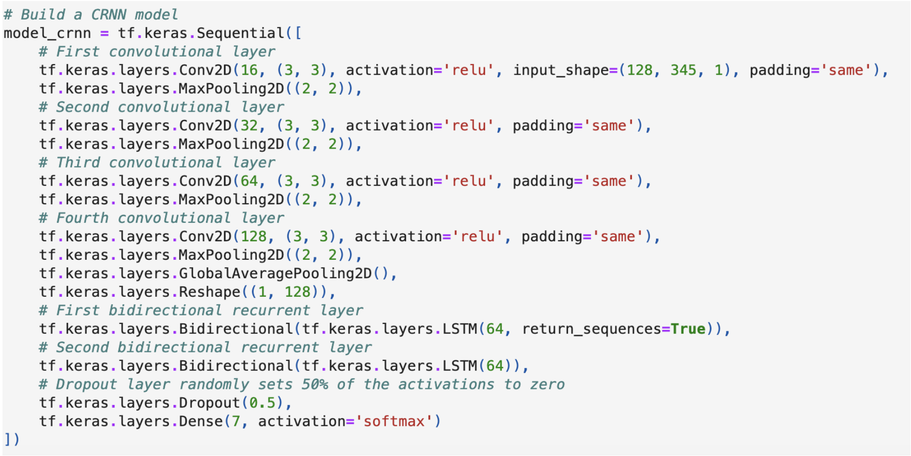


```python
# Create a function that extracts Mel-Frequency Cepstral Coefficients (MFCCs) from an audio file
def extract_mfcc(path):
    # Load the audio file and set the sampling rate to 44100
    audio, sr = librosa.load(path, sr=44100, duration=4, mono=True)
    # Pad the audio files that are less than 4 seconds with zeros at the end
    if len(audio) < 4 * sr:
        audio = np.pad(audio, pad_width=(0, 4 * sr - len(audio)), mode='constant')
    # Convert the audio file into MFCC
    signal = librosa.feature.mfcc(y = audio, sr=sr, n_mfcc=128)
    # Return the MFCCs as a numpy array
    return np.array(signal)
```



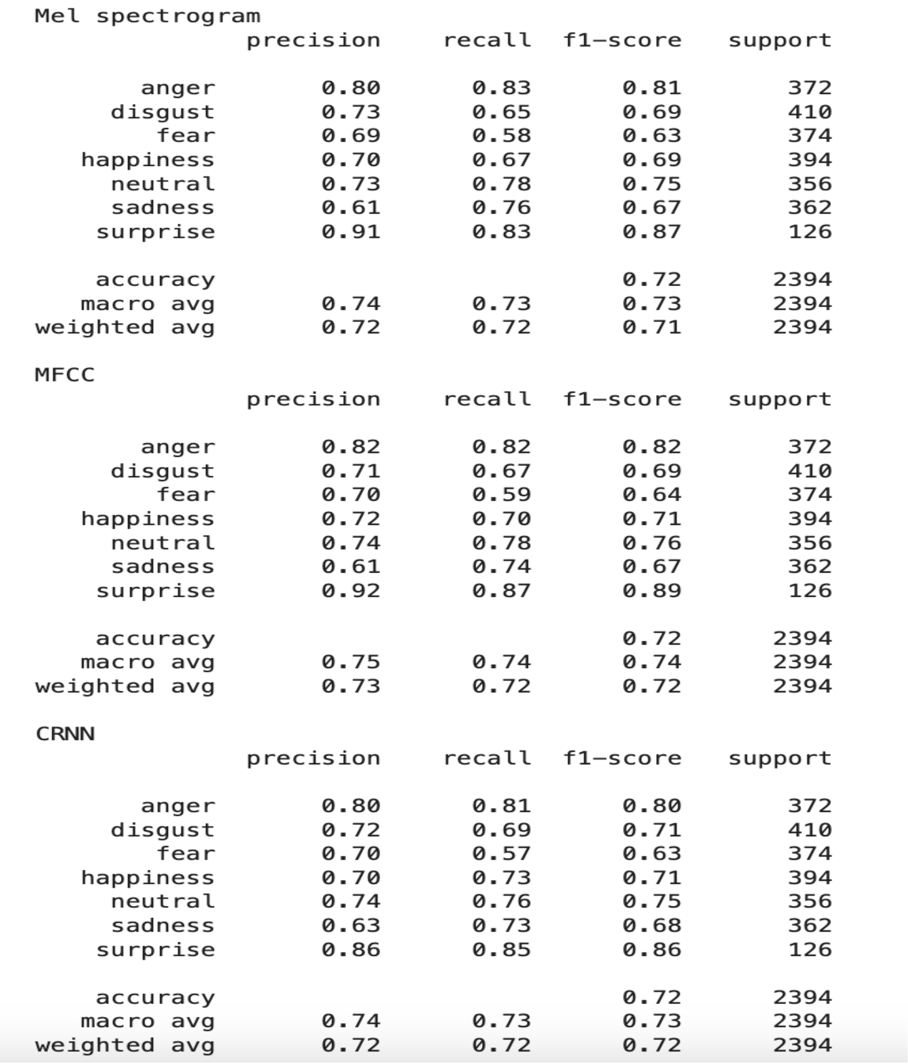
```python
# Define the CNN model
model_mfcc = tf.keras.Sequential([
    # First convolutional layer
    tf.keras.layers.Conv2D(32, (3, 3), activation='relu', input_shape=(128, 345, 1), padding='same'),
    tf.keras.layers.MaxPooling2D((2, 2)),
    # Batch normalization maintains the 0 mean and 1 standard deviation
    tf.keras.layers.BatchNormalization(),
    # Second convolutional layer
    tf.keras.layers.Conv2D(64, (3, 3), activation='relu', padding='same'),
    tf.keras.layers.MaxPooling2D((2, 2)),
    # Third convolutional layer
    tf.keras.layers.Conv2D(128, (3, 3), activation='relu', padding='same'),
    tf.keras.layers.MaxPooling2D((2, 2)),
    # Fourth convolutional layer
    tf.keras.layers.Conv2D(256, (3, 3), activation='relu', padding='same'),
    tf.keras.layers.MaxPooling2D((2, 2)),
    tf.keras.layers.GlobalAveragePooling2D(),
    # Dropout layer randomly sets 50% of the activations to zero
    tf.keras.layers.Dropout(0.5),
    tf.keras.layers.Dense(7, activation='softmax')
])
```

A.5.3 Mel Spectrogram CRNN Model



```python
# Build a CRNN model
model_crnn = tf.keras.Sequential([
    # First convolutional layer
    tf.keras.layers.Conv2D(16, (3, 3), activation='relu', input_shape=(128, 345, 1), padding='same'),
    tf.keras.layers.MaxPooling2D((2, 2)),
    # Second convolutional layer
    tf.keras.layers.Conv2D(32, (3, 3), activation='relu', padding='same'),
    tf.keras.layers.MaxPooling2D((2, 2)),
    # Third convolutional layer
    tf.keras.layers.Conv2D(64, (3, 3), activation='relu', padding='same'),
    tf.keras.layers.MaxPooling2D((2, 2)),
    # Fourth convolutional layer
    tf.keras.layers.Conv2D(128, (3, 3), activation='relu', padding='same'),
    tf.keras.layers.MaxPooling2D((2, 2)),
    tf.keras.layers.GlobalAveragePooling2D(),
    tf.keras.layers.Reshape((1, 128)),
    # First bidirectional recurrent layer
    tf.keras.layers.Bidirectional(tf.keras.layers.LSTM(64, return_sequences=True)),
    # Second bidirectional recurrent layer
    tf.keras.layers.Bidirectional(tf.keras.layers.LSTM(64)),
    # Dropout layer randomly sets 50% of the activations to zero
    tf.keras.layers.Dropout(0.5),
    tf.keras.layers.Dense(7, activation='softmax')
])
```

A.6 Final result



```
Mel spectrogram
              precision    recall  f1-score   support

       anger       0.80      0.83      0.81       372
     disgust       0.73      0.65      0.69       410
        fear       0.69      0.58      0.63       374
   happiness       0.70      0.67      0.69       394
     neutral       0.73      0.78      0.75       356
     sadness       0.61      0.76      0.67       362
    surprise       0.91      0.83      0.87       126

    accuracy                           0.72      2394
   macro avg       0.74      0.73      0.73      2394
weighted avg       0.72      0.72      0.71      2394

MFCC
              precision    recall  f1-score   support

       anger       0.82      0.82      0.82       372
     disgust       0.71      0.67      0.69       410
        fear       0.70      0.59      0.64       374
   happiness       0.72      0.70      0.71       394
     neutral       0.74      0.78      0.76       356
     sadness       0.61      0.74      0.67       362
    surprise       0.92      0.87      0.89       126

    accuracy                           0.72      2394
   macro avg       0.75      0.74      0.74      2394
weighted avg       0.73      0.72      0.72      2394

CRNN
              precision    recall  f1-score   support

       anger       0.80      0.81      0.80       372
     disgust       0.72      0.69      0.71       410
        fear       0.70      0.57      0.63       374
   happiness       0.70      0.73      0.71       394
     neutral       0.74      0.76      0.75       356
     sadness       0.63      0.73      0.68       362
    surprise       0.86      0.85      0.86       126

    accuracy                           0.72      2394
   macro avg       0.74      0.73      0.73      2394
weighted avg       0.72      0.72      0.72      2394
```

References

[1] Juslin, K. N., & Scherer, K. R. (2005). Vocal expressions of emotion in Japanese and Swedish. Speech Communication, 47(3-4), 303-335. https://zenodo.org/records/1188976

[2] Livingstone, S. R., & Russo, F. A. (2018). The Toronto emotional speech synthesis database. In Proceedings of the 10th International Conference on Speech Synthesis (ICSS) (pp. 321-325). https://tspace.library.utoronto.ca/handle/1807/24487

[3] Koelstra, S., & Pantic, M. (2009). A DE-stressed emotional speech database for affective computing research. In Proceedings of the 2nd International Conference on Affective Computing and Intelligent Interaction (pp. 100-105). https://www.tensorflow.org/datasets/catalog/savee

[4] Kim, H. S., Scherer, K. R., Oh, Y. J., & Park, H. S. (2013). Emotion recognition from speech using Gaussian mixture models: Investigation of parameter settings and feature selection. Information Sciences, 253, 11-22. https://github.com/topics/crema-d

[5] Rabiner, L., & Schafer, R. W. (2007). Digital processing of speech signals. Pearson Prentice Hall.

[6] Allen, J. B., & Oppenheim, D. R. (1979). Discrete-time signal processing. John Wiley & Sons.

[7] Griffin, D. W., & Lim, J. S. (1984). Generalized speech spectrum estimation. IEEE Journal on Selected Areas in Communications, 2(3), 258-267.

[8] Stevens, S. S., Volkmann, J., & Newman, E. B. (1937). A scale for the measurement of the psychological magnitude pitch. The Journal of the Acoustical Society of America, 8(3), 18-38.

[9] LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (2013). Handbuch der Informatik. Springer Berlin Heidelberg.

[10] Li, T., & Deng, L. (2017). Recurrent convolutional neural networks for end-to-end speech emotion recognition. In 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 5296-5300). IEEE.

[11] Murty, P., & Rani, K. (2011). Mel-frequency cepstral coefficients based emotion recognition in speech using support vector machines. International Journal of Computer Applications, 39(11), 32-37.

[12] Xue, J., & Li, J. (2016). Bidirectional LSTM recurrent neural network for speech emotion recognition. In INTERSPEECH (pp. 2224-2228).

[13] Speech Emotion Recognition (SER) on Papers With Code: https://paperswithcode.com/task/speech-emotion-recognition

[14] An ongoing review of speech emotion recognition by Mohammad, R. & Salamon, G. (2015): https://www.sciencedirect.com/science/article/pii/S0925231223000103

[15] Speech Emotion Recognition Tutorial by PyTorch: https://dagshub.com/Vargha-Kh/Speech-Emotion-Classification-with-PyTorch