Skyler Estavillo, 4662-8928

CIS4930 Individual Coding Assignment
Spring 2023

1. Problem Statement

Speech Emotion Recognition (SER) is an interdisciplinary field at the intersection of psychology, linguistics, and computer science, which focuses on understanding and analyzing the emotional content of human speech. By recognizing the emotions conveyed by a speaker, SER systems can help in improving communication, enhancing user experience, and providing more personalized services across various domains, such as healthcare, customer service, and education.

The importance of understanding human emotions cannot be overstated, as it enables us to better comprehend individual needs and respond more effectively to various situations. Emotions play a crucial role in human communication, as they convey essential information about a person's mental state and intention, significantly affecting the interpretation of the message being conveyed.

In this project, we aim to develop a machine learning model that can accurately detect emotions in speech, regardless of the semantic content. By utilizing state-of-the-art techniques and leveraging the power of deep learning, we strive to create an efficient and reliable SER system. This system can then be integrated into various applications, ranging from virtual assistants and chatbots to mental health monitoring and support systems.

2. Data Preparation

The initial step was to separate testing and training files into their respective dataframes. I chose to do this first, as in previous projects I suffered with data leaks occurring after extracting features from my data. Now, I'm not sure if separation as the first step was truly necessary, but I do believe it kept my models safe from any potential leaks. 70% of the data in each 'emotion' folder was used for training, while the remaining 30% of files were reserved for testing.

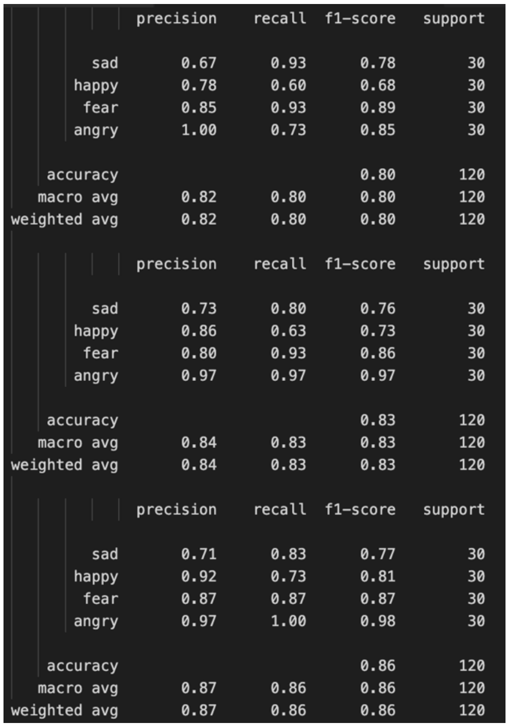
After this, it was important to select a few random files from my data directory and manually analyze some of the acoustic features. By observing amplitude over time, I could understand the overall structure and loudness variations of the audio signal. The frequency domain, obtained by applying the Fast Fourier Transform (FFT) to the time-domain signal, provided insight into the spectral content and distribution of energy across various frequencies in the signal. Finally, the Short-Time Fourier Transform (STFT) of the audio signal shows how the frequency content of the signal changes over time. Overall, I found the first two graphs the easiest to understand and draw patterns from before moving on to the next step.

Following the graphing step, the goal was to define a function which would extract the features of interest from the audio. These features included loudness, MFCC values, zero crossing rate data, chroma values, and Mel-spectrogram data. These features then *had to be scaled using the MinMaxScaler library before being re-inserted into the dataframe. The last job of this function was to return these values as a NumPy array. In order to map emotions to the files, I used the file path to assign an integer to each emotion. In other words, if the file was located in the 'sad' data folder, it would be assigned an integer of 0. 'Happy' files were assigned a 1, 'fear' files were assigned a 2, and 'anger' files were assigned a 3. This would make it easy to build classification models down the line, though I would use emotion mapping again to make my confusion matrices and classification reports easier to read.*

3. Model Development

- Model Training
    - *The training was pretty straightforward, as much of the setup was similar to Assignments 1 and 2. The models selected were Support Vector Machine, Gaussian Naïve Bayes Classifier, and Random Forest Classifier.*

      SVM is effective for high-dimensional datasets such as the ones encountered in speech emotion recognition tasks. The Naïve Bayes classifier assumes that features are conditionally independent to simplify the learning process. While this is not always true, this works well in many cases, particularly those will a limited amount of training data. Finally, the random forest classifier does a great job of handling high-dimensional data and preventing overfitting. Logistic regression was not used because this classifier performs best in contexts where there is a linear relationship between features and the target variable. However, in speech emotion recognition, the relationship is often non-linear and complex.
- Model Evaluation

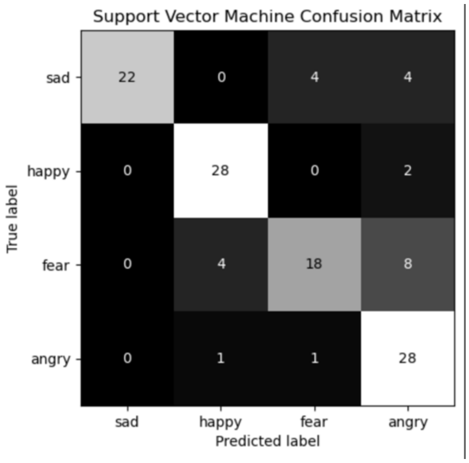


```
              precision    recall  f1-score   support

         sad       0.67      0.93      0.78        30
       happy       0.78      0.60      0.68        30
        fear       0.85      0.93      0.89        30
       angry       1.00      0.73      0.85        30

    accuracy                           0.80       120
   macro avg       0.82      0.80      0.80       120
weighted avg       0.82      0.80      0.80       120
              precision    recall  f1-score   support

         sad       0.73      0.80      0.76        30
       happy       0.86      0.63      0.73        30
        fear       0.80      0.93      0.86        30
       angry       0.97      0.97      0.97        30

    accuracy                           0.83       120
   macro avg       0.84      0.83      0.83       120
weighted avg       0.84      0.83      0.83       120
              precision    recall  f1-score   support

         sad       0.71      0.83      0.77        30
       happy       0.92      0.73      0.81        30
        fear       0.87      0.87      0.87        30
       angry       0.97      1.00      0.98        30

    accuracy                           0.86       120
   macro avg       0.87      0.86      0.86       120
weighted avg       0.87      0.86      0.86       120
```

The image above displays the classification reports for Support Vector Machine, Naïve Bayes, and the Random Forest classifier respectively.

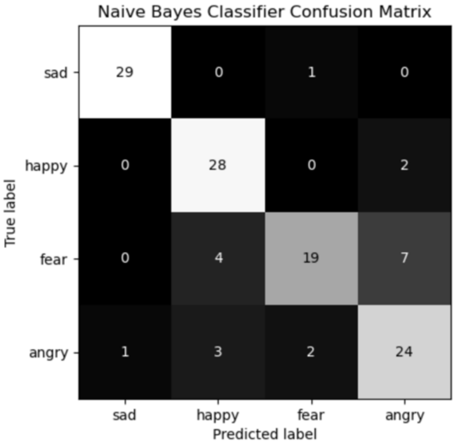
In determining the utility of a model, there are a couple metrics to consider. Accuracy describes the total number of accurate classifications divided by the total number of classifications made. It is a useful metric, but may be less valuable when the cost of a false negative is high. Precision, on the other hand, is useful when the cost of a false positive is high, such as in an email spam detection system. Recall, though calculated differently than accuracy, is also an important metric to consider when the cost of a false negative is high. Finally, the F1 score provides a balance between precision and recall.

In this case, it was important that happiness was distinguished from sadness, and that both of these emotions were distinguished from fear and anger. Fear and anger were the emotions that caused the most confusion for our classifiers, as they likely had many feature patterns in common.



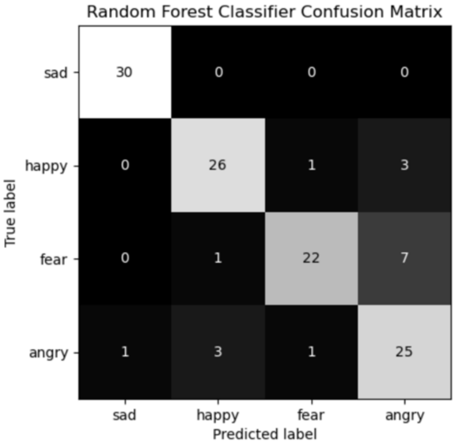
Support Vector Machine Confusion Matrix

True label \ Predicted label	sad	happy	fear	angry
sad	22	0	4	4
happy	0	28	0	2
fear	0	4	18	8
angry	0	1	1	28



Naive Bayes Classifier Confusion Matrix

True label \ Predicted label	sad	happy	fear	angry
sad	29	0	1	0
happy	0	28	0	2
fear	0	4	19	7
angry	1	3	2	24



Random Forest Classifier Confusion Matrix

True label \ Predicted label	sad	happy	fear	angry
sad	30	0	0	0
happy	0	26	1	3
fear	0	1	22	7
angry	1	3	1	25

Happiness and sadness were distinctly separated, though sadness was sometimes confused for fear or anger in the SVM model.

4. Discussion

Overall, the model seems to perform reasonably well. The accuracy, precision, recall, and F1-scores for each class (sad, happy, fear, and angry) are fairly high, indicating that the models are able to distinguish between the emotions with a good degree of accuracy. In conclusion, this suggests that the model, while certainly not perfect, address the problem well.

The main challenge I met during the development process was mapping the emotions to the tuples of data obtained from analyzing each audio files. Additionally, I was originally not sure how to map the emotions to both the classification reports and confusion matrices in order to make the output easier to interpret. The solution for this ended up being relatively straightforward and can be found in my code.

This assignment was extremely interesting. Firstly, it provided valuable knowledge that will be applied in my group's final project. Secondly, the process of analyzing speech and trying to identify patterns and make predictions has high utility. I also believe that there is a lot of potential to monetize many of the ideas presented in this assignment.

5. Appendix

https://github.com/cyrus-estavillo/SpeechEmotionRecognition