ECE324 Final Report: PENN

Gabriel Deza and Eric Keilty

Words: 1996

Penalty: 0%

Both Gabe and Eric give consent to Professor Rose to post the video of our final presentation to the course website.

Introduction

In the age of automation, more and more of our everyday human interactions are being replaced by interactions with computers and software. Self-checkouts in grocery stores, self-ordering kiosks in fast-food chains, and numerous smartphone applications have replaced many of these small social interactions. As a consequence, many industries are beginning to keeping data on these interactions in order to regulate customer satisfaction. For example, the emotional states of patients undergoing rehabilitation are monitored and their care adapted accordingly with the goal of creating a better overall experience of the patient and potentially lead to a faster recovery [1]. Software's assessment of human emotions are soon going to be the basis of important decisions that will impact the lives of many people.

Many methods have been developed to teach human emotions to computers. Sentiment Analysis is one such method, which is the process of analyzing text data and using artificial intelligence to classifying it into positive, negative, or neutral opinions [2]. However, in everyday conversations, the meaning of a sentence can change dramatically depending on how it's said. Simply using a speech-to-text software combined with sentiment analysis would lose the subtle nuance conveyed in speech. The goal of PENN (*Predicting Emotions using a Neural Network)* is to predict the mood/emotional state of a person based on a short audio file of them speaking. Since this boils down to complex classification problem with a large datasource, we believe that using a Neural Network is the best approach.

Illustration

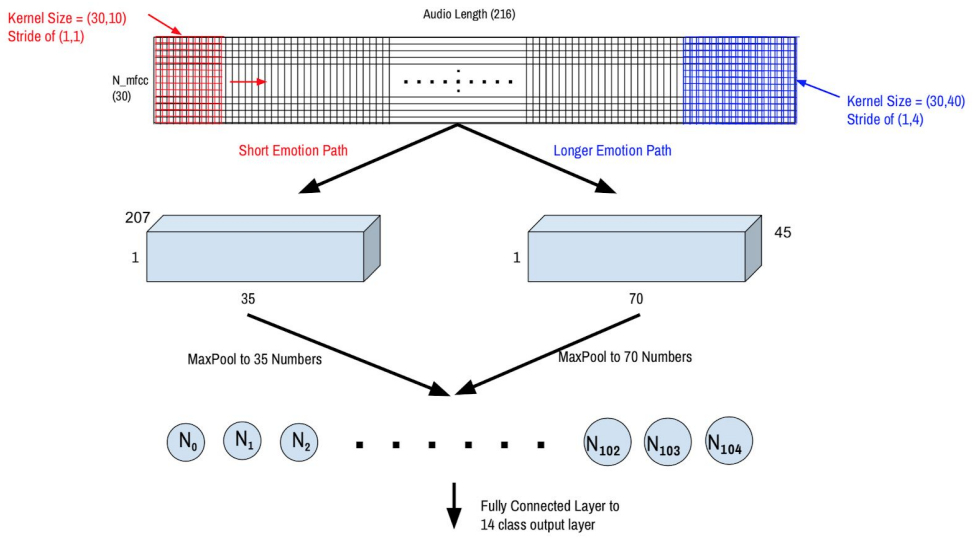


Figure 1: Detailed CNN Architecture

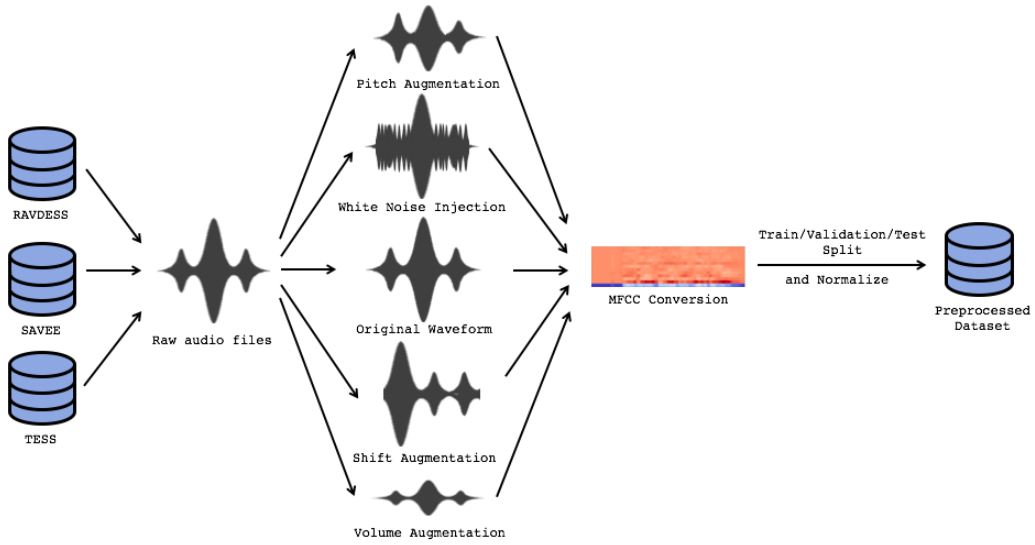


Figure 2: Preprocessing Illustration

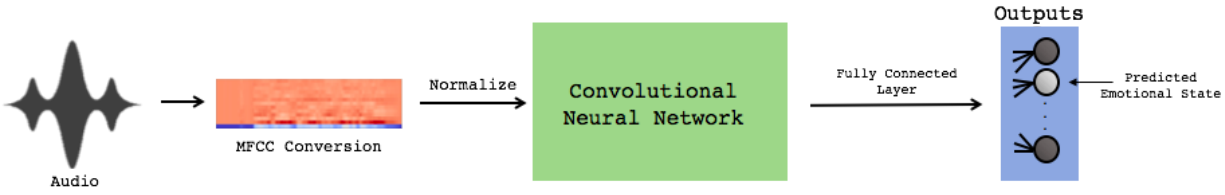


Figure 3: Using the Model Illustration

Background

In the field of Speech Emotion Recognition (SER), two different outlooks on this problem define the field's current efforts. Some experts disregard the domain of audio signals and rely on the power of machine learning. In this outlook, modern machine learning applications are the focus, and the audio/emotion nature of the problem is less relevant [3][4][5]. Other experts believe in a hard-coded heuristic approach that can be tailored to the domain of audio signals [6]. We implemented a mix both of these outlooks by using a machine learning model with an architecture designed to take advantage of how emotions are manifested audibly by humans.

Furthermore, the research regarding SER is being implemented in industry very rapidly. For example, Uber is partnering with Attectiva in order to use emotion recognition software to improve customer experience in their vehicles [7].

Data and Data Processing

In this project there are two types of datasets. The first is the pre-training dataset, which is used to teach the model emotion classification. The second is the fine-tuning dataset, which is used to tailor the model to the microphone of the person using the software as different microphones process audio differently. The pre-training dataset is a combination of the dataset found in the RAVDESS, SAVEE, and TESS dataset [8][9][10]. The fine-tuning dataset was personally recorded by us on a MacBook. The data breakdown of each dataset is shown in Table 1.

Table 1: Data Source Breakdown

	RAVDESS		SAVEE		TESS		Personal	
	Male	Female	Male	Female	Male	Female	Male	Female
Angry	96	96	60	0	0	400	8	0
Disgust	96	96	60	0	0	400	8	0
Fear	96	96	60	0	0	400	0	0
Happy	96	96	60	0	0	400	8	0
Neutral	48	48	60	0	0	400	8	0
Sad	96	96	60	0	0	400	8	0
Surprised	96	96	60	0	0	400	8	0
Toal	1,248		420		2,800		48	

Each pre-training dataset uses a different file naming convention in order to provide relevant information about the audio file. These naming conventions are shown in Figure 5, 6 and 7 on the next page. The naming convention for the personally recorded dataset found in the GitHub repository is given below in Figure 4.

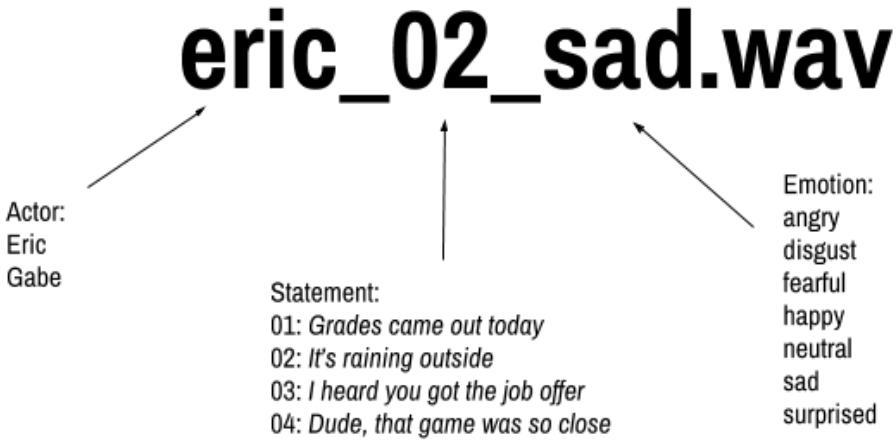


Figure 4: File Naming Convention for the Personally collected dataset

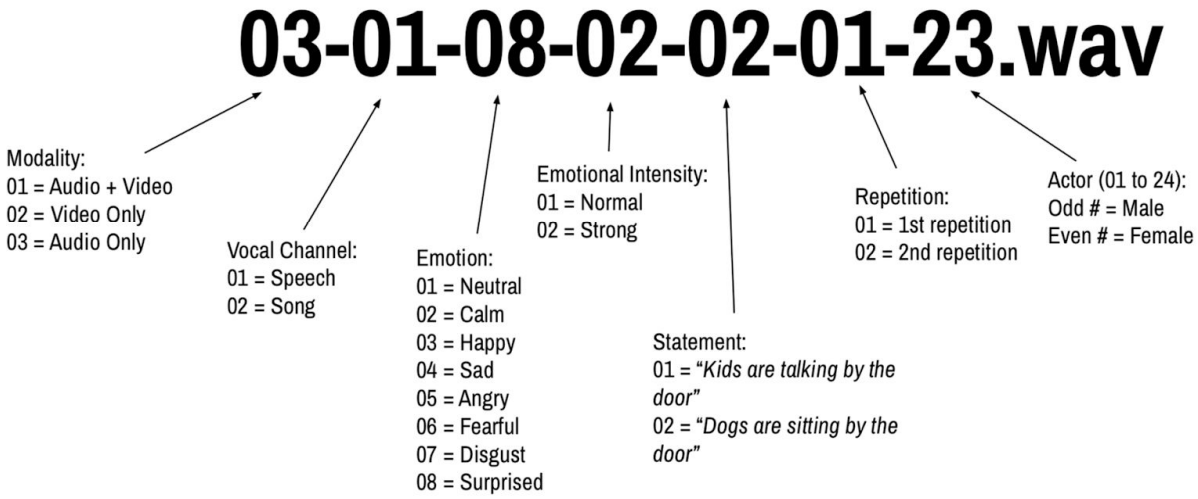


Figure 5: File Naming Convention for RAVDESS

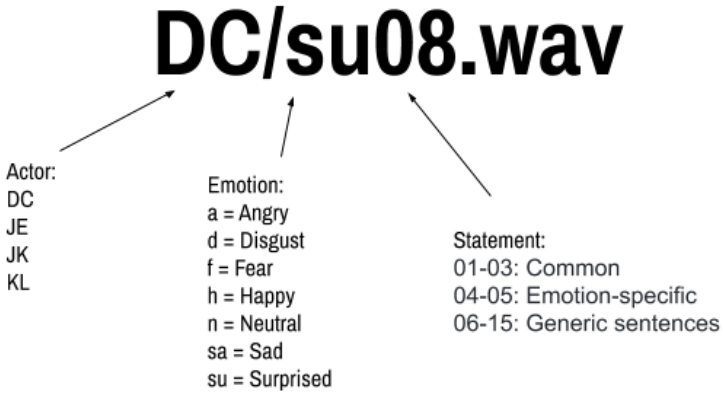


Figure 6: File Naming Convention for SAVEE

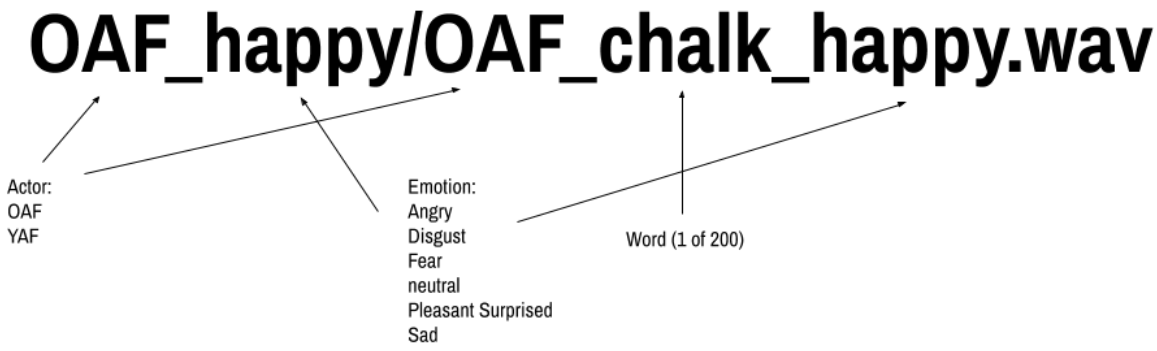


Figure 7: File Naming Convention for TESS

Preprocessing has 6 stages: rearranging files, extracting data and labels, data augmentation, Mel Frequency Cepstral Coefficient (MFCC) conversion, training/validation/test split, and normalization. This process is shown in Figure 8.

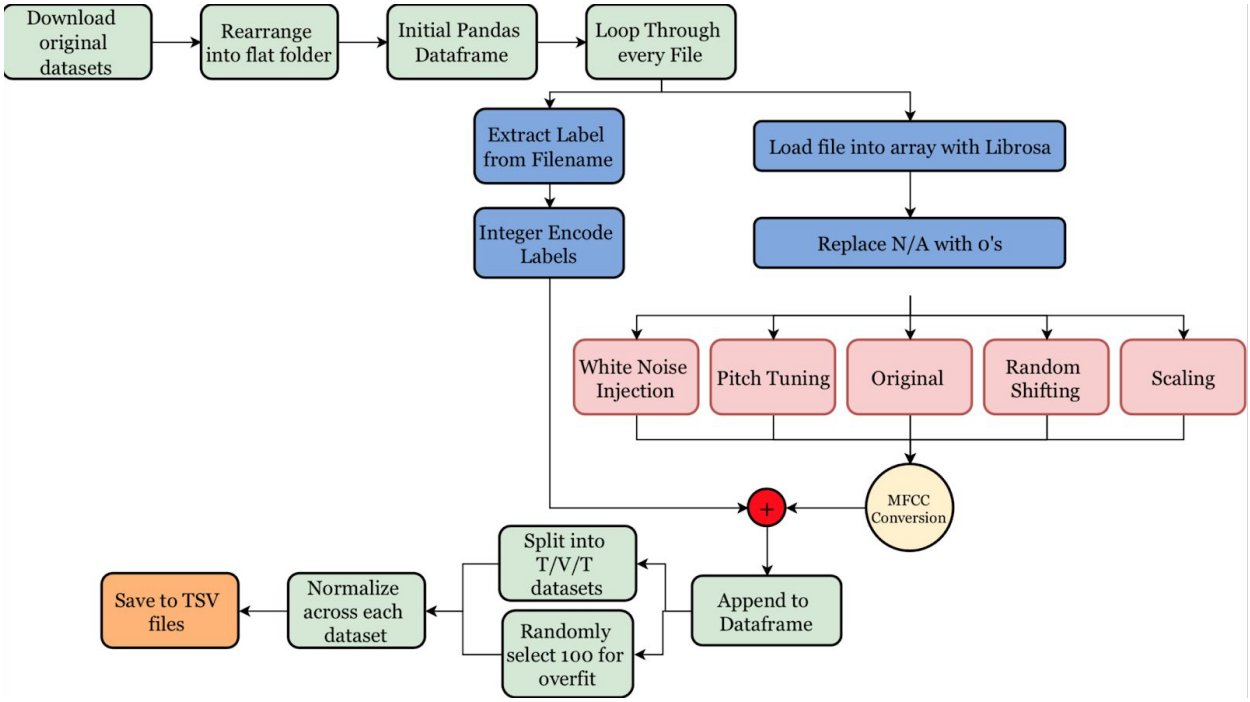


Figure 8: Preprocessing Plan

Rearranging files consists of taking the files from the downloaded datasets and flattening them into a single directory. This makes the files much easier to iterate over. Extracting data and labels consists of looping through each file, loading the raw audio data, and extracting relevant labels such as emotion, actor, and audio length. Four different types of data augmentation are then applied to a subset of the data. The raw audio data is then converted to its MFCC representation. This is now the full dataset, which is then split into the training, validation, and test set. The test set is obtained by removing all data recorded by certain actors. The remaining data is split between the training and validation set ensuring equal class distribution. Additionally, 100 random samples from the original full dataset are saved separately as the overfit dataset. Finally, all data is normalized using the mean and standard deviation of the training data.

The resulting data distribution across each pre-training dataset is shown in Table 2 and the percentage of total data in each split is given in Table 3. It is important to note that creating this split took 17 hours on a MacBook Pro.

Table 2: Data Split Class Distribution

	RAVDESS						SAVEE						TESS						Total					
	Train		Valid		Test		Train		Valid		Test		Train		Valid		Test		Train		Valid		Test	
	M	F	M	F	M	F	M	F	M	F	M	F	M	F	M	F	M	F	M	F	M	F	M	F
Angry	288	288	72	72	40	120	180	0	45	0	75	0	0	320	0	80	0	0	468	608	117	152	115	120
Disgust	288	288	72	72	40	120	180	0	45	0	75	0	0	320	0	80	0	0	468	608	117	152	115	120
Fear	288	288	72	72	40	120	180	0	45	0	75	0	0	320	0	80	0	0	468	608	117	152	115	120
Happy	288	288	72	72	40	120	180	0	45	0	75	0	0	320	0	80	0	0	468	608	117	152	115	120
Neutral	288	288	72	72	60	180	180	0	45	0	150	0	0	320	0	80	0	0	468	608	117	152	210	120
Sad	288	288	72	72	40	120	180	0	45	0	75	0	0	320	0	80	0	0	468	608	117	152	115	120
Surprised	288	288	72	72	40	120	180	0	45	0	75	0	0	320	0	80	0	0	468	608	117	152	115	120
Total	4032		1008		1200		1260		315		600		2240		560		0		7532		1883		1800	

Table 3: Data Split Percentages

	Training Data		Validation Data		Test Data		Total	
	Male	Female	Male	Female	Male	Female	Male	Female
RAVDESS	2016	2016	504	504	300	900	2820	3420
SAVEE	1260	0	315	0	600	0	2174	0
TESS	0	2,240	0	560	0	0	0	2800
Total	3276	4256	819	1064	900	900	4994	6220
Percent	43.5 %	56.5 %	43.5 %	56.5 %	50.0%	50.0%	44.5 %	55.5 %
	67.76 %		16.05 %		16.19%		100%	

Architecture

The architecture of the final model is a Convolutional Neural Network (CNN). Since the raw audio data was converted to it's MFCC representation, which can be interpreted as a 2D array with dimensions of number of MFCC bands by audio length, this representation easily be processed by a CNN similar to that of an image. This is shown below in Figure 8.

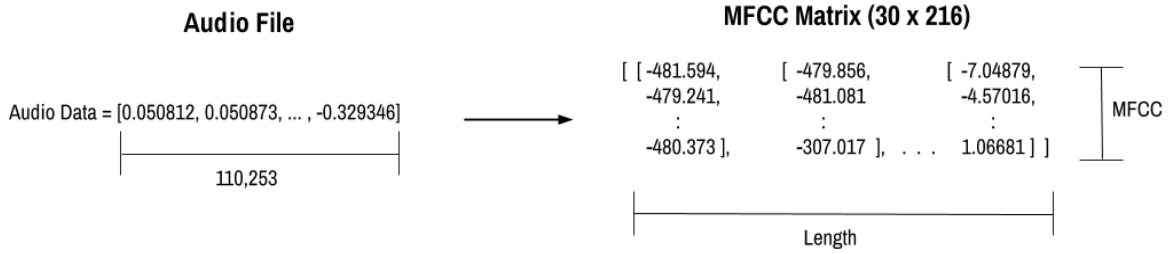


Figure 8: Representation of audio file before and after MFCC conversion

It was found that certain emotions can manifest both in short bursts and longer periods. To ensure both are captured, the CNN consists of two independent convolution branches, one with a small and the other a

larger kernel. Each branch has a single convolution layer, which is passed into a max pooling layer and dropout layer. Finally, these branches are concatenated together and passed into a fully connected layer with a softmax activation function. This model is shown in Figure 2 in the Illustration section.

Baseline Model

In the initial proposal, the baseline model was going use the library *my-voice-analysis* in order to extract audio features from each audio file. However, since the audio files are short, the values of the features extracted did not differ enough for the model to be predictive. Instead, the baseline model is a Logistical Regression, which was created using a simple one-layer MLP. The input is the flattened data and the output layer has one node for each class. The goal of the baseline is to see the performance of a model that does not take the structure of the data into account.

Quantitative Results

The total quantitative results from training, validation, and test across all model are shown below in Table 4. These are used to determine how well each model performs on each dataset.

Table 4: Loss and Accuracy across each dataset and across each model

	Training		Validation		Test		
	Loss	**Accuracy**	**Loss**	**Accuracy**	**Loss**	**Accuracy**	**Top 2 Accuracy**
Baseline (MLP)	0.0189	100.00%	0.9699	78.07%	3.342	29.89%	86.09%
RNN	0.0600	98.70%	0.7409	81.20%	3.0444	35.67%	90.81%
CNN	0.1181	99.80%	0.4001	89.16%	1.6756	43.22%	94.49%

The corresponding loss and accuracy plots for the baseline and CNN are given in Figures 9 and 10 below. The purpose of these plots is to see if the model is overfitting or underfitting. The RNN was omitted because it was not the best performing model.

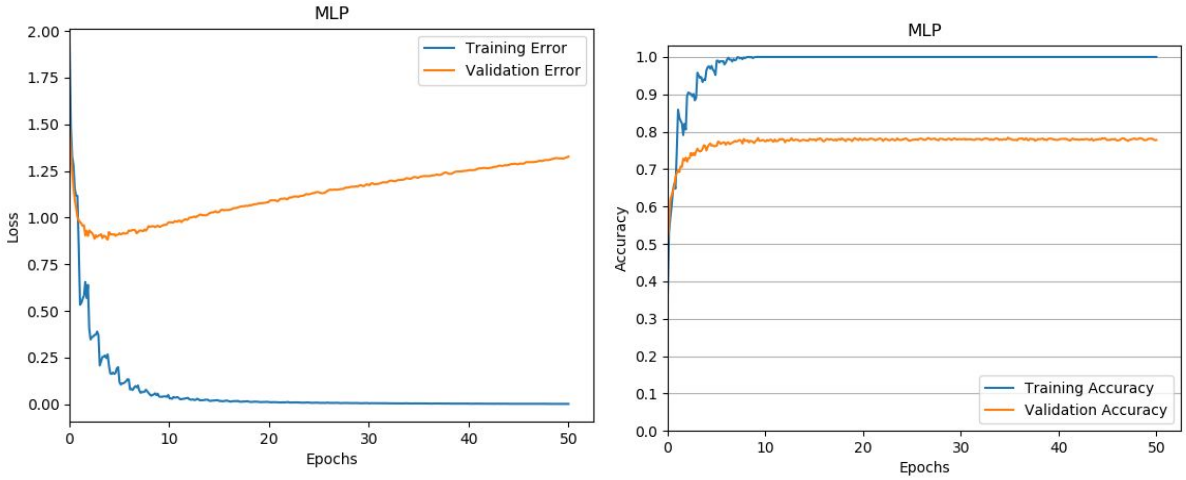


Figure 9: Loss and Accuracy for the Baseline (MLP)

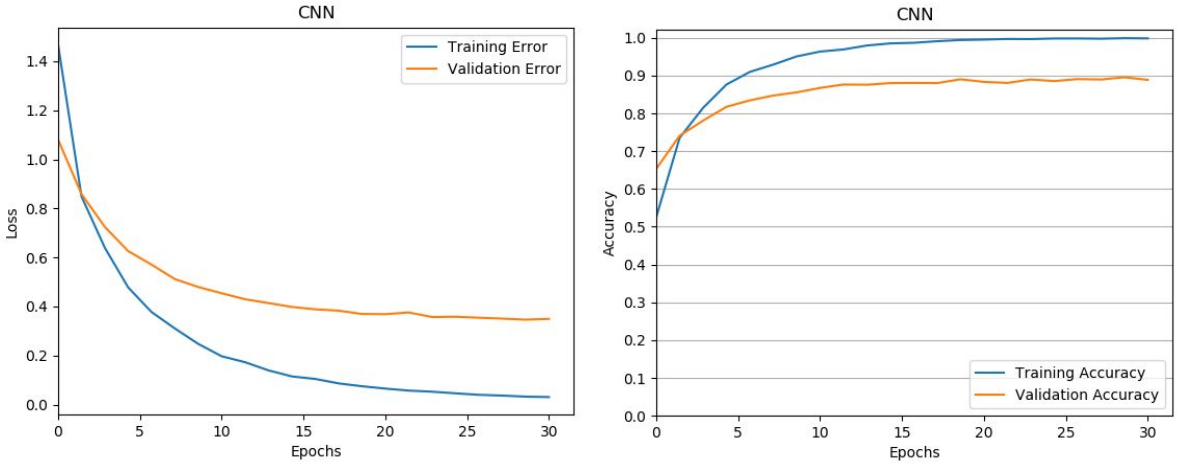


Figure 10: Loss and Accuracy for the CNN

The confusion matrices of the CNN for the validation and test dataset are given in Figures 11 and 12. These are used to see where the CNN excels and where it has difficulty.

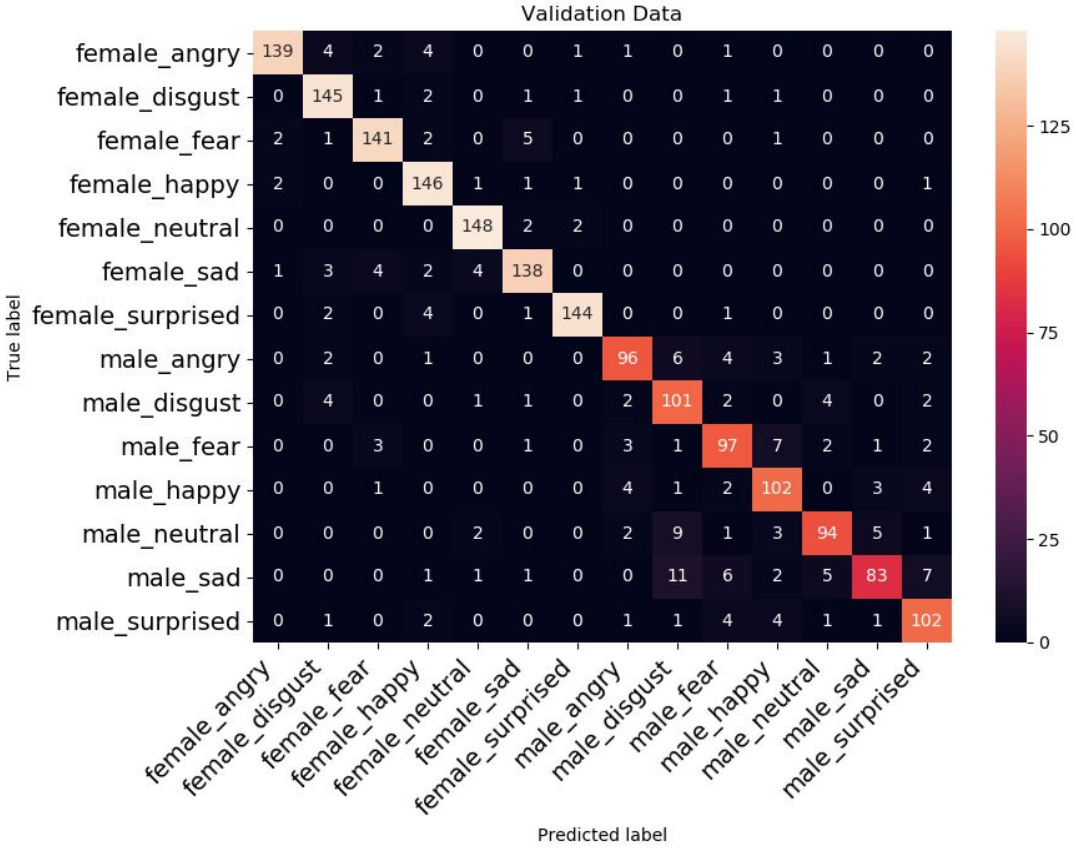


Figure 11: Confusion Matrix of the Validation Dataset for the CNN

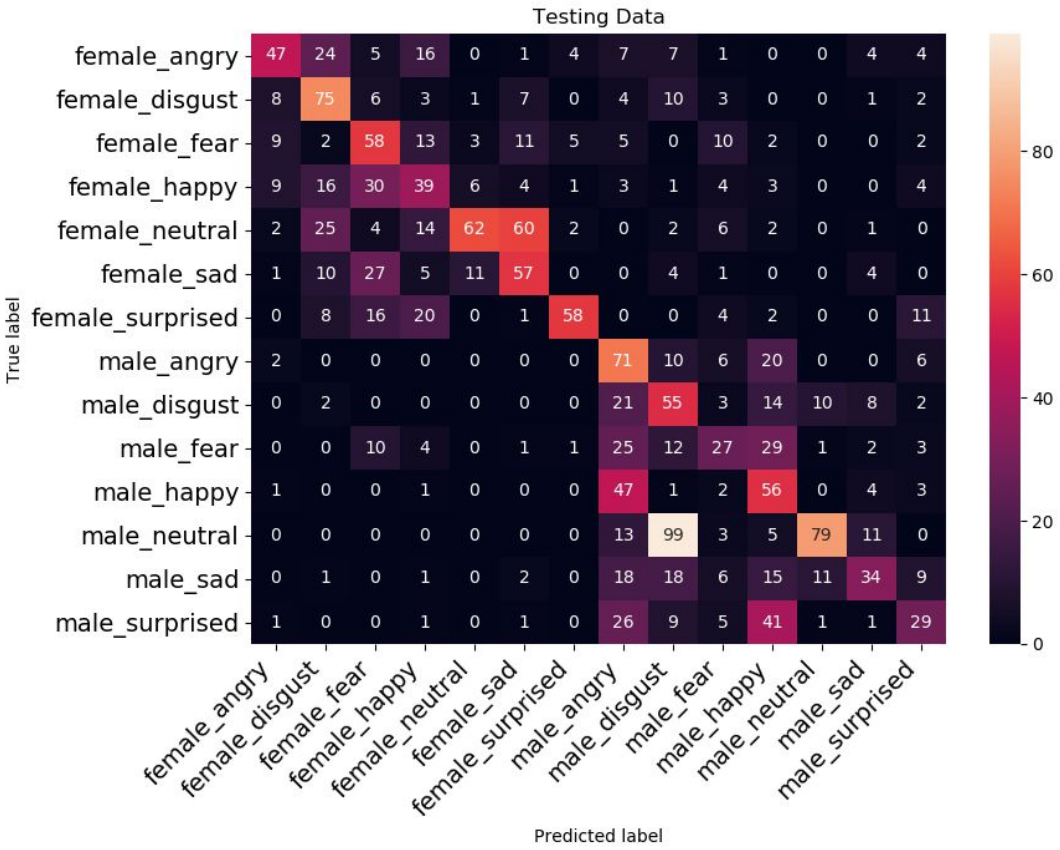


Figure 12: Confusion Matrix of the Test Dataset for the CNN

Finally, the confusion matrix of the CNN after fine-tuning is given in Figure 13. This is used to determine if transfer learning was effectively implemented.

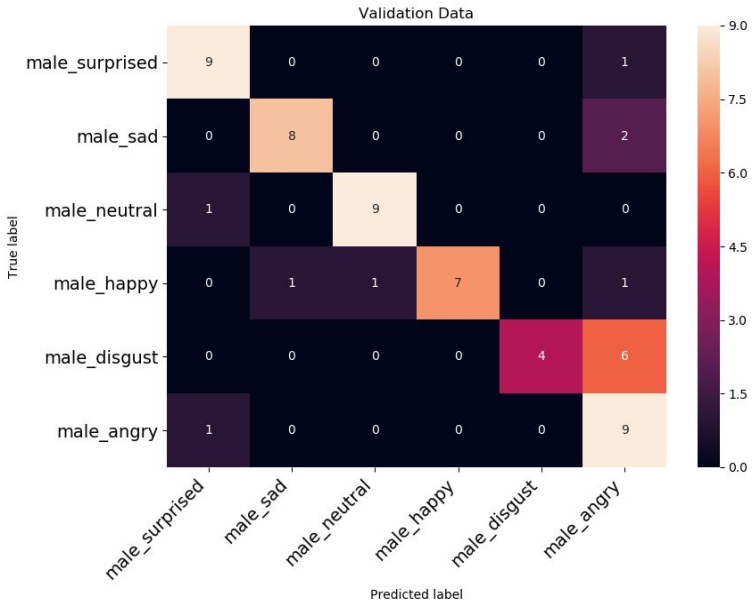
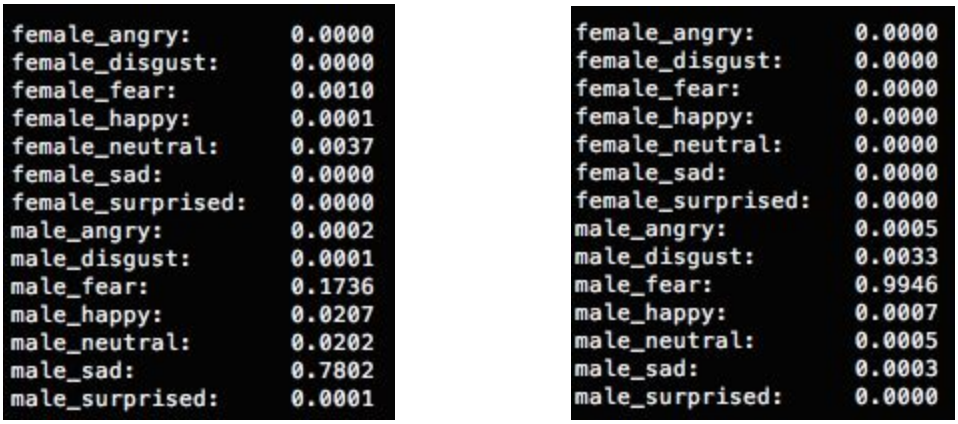


Figure 13: Confusion Matrix of the CNN after Fine-Tuning

Qualitative Results

The following are outputs of the pre-trained model from the RAVDESS dataset (Figure 14). The file names of the audio clips are “03-01-06-01-01-01-07.wav” (left) and “03-01-06-01-01-01-01.wav” (right). The correct label for both inputs is male fear.

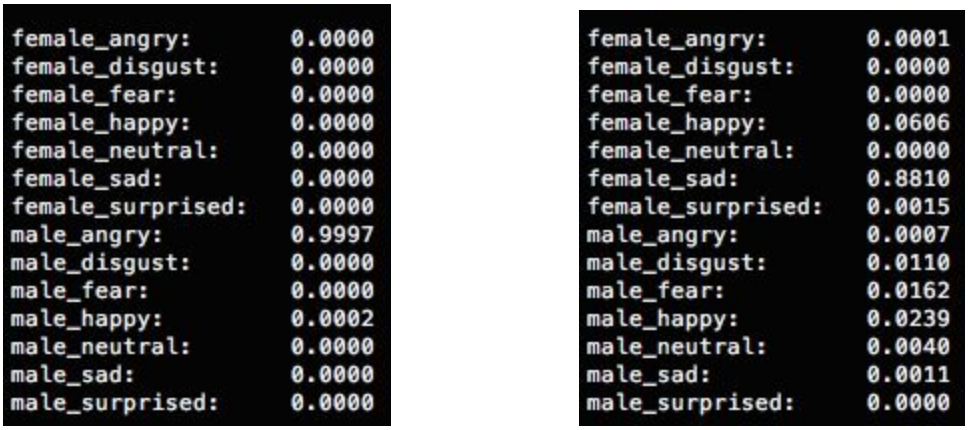


female_angry:	0.0000
female_disgust:	0.0000
female_fear:	0.0010
female_happy:	0.0001
female_neutral:	0.0037
female_sad:	0.0000
female_surprised:	0.0000
male_angry:	0.0002
male_disgust:	0.0001
male_fear:	0.1736
male_happy:	0.0207
male_neutral:	0.0202
male_sad:	0.7802
male_surprised:	0.0001

female_angry:	0.0000
female_disgust:	0.0000
female_fear:	0.0000
female_happy:	0.0000
female_neutral:	0.0000
female_sad:	0.0000
female_surprised:	0.0000
male_angry:	0.0005
male_disgust:	0.0033
male_fear:	0.9946
male_happy:	0.0007
male_neutral:	0.0005
male_sad:	0.0003
male_surprised:	0.0000

Figure 14: Pre-Trained Model Outputs

The following are outputs of the model from the fine-tuned model (Figure 15). These were obtained by live recording audio using the demo.py script. The left image is the result of Eric (male) saying the phrase “I hate you” in a loud voice. The right image is the result of my roommate (female) saying the phrase “my dog died” in a sad voice.



female_angry:	0.0000
female_disgust:	0.0000
female_fear:	0.0000
female_happy:	0.0000
female_neutral:	0.0000
female_sad:	0.0000
female_surprised:	0.0000
male_angry:	0.9997
male_disgust:	0.0000
male_fear:	0.0000
male_happy:	0.0002
male_neutral:	0.0000
male_sad:	0.0000
male_surprised:	0.0000

female_angry:	0.0001
female_disgust:	0.0000
female_fear:	0.0000
female_happy:	0.0606
female_neutral:	0.0000
female_sad:	0.8810
female_surprised:	0.0015
male_angry:	0.0007
male_disgust:	0.0110
male_fear:	0.0162
male_happy:	0.0239
male_neutral:	0.0040
male_sad:	0.0011
male_surprised:	0.0000

Figure 15: Fine-Tuned Model Outputs

Discussion and Learning

Beginning with the quantitative measures, the final loss and accuracy values in Table 4 clearly shows that the CNN is the best performing model, as it has the lowest loss and highest accuracy in the validation and test data. The loss and accuracy plots further support this claim. The MLP plots (Figure 9) are clearly overfitting and show that the dataset is simply being memorized. By contrast, the CNN model plots (Figure 10) show that the validation loss plateaus at slightly higher value than the training data, but does not diverge. This indicates the model is overfitting, but not to the extent of the baseline. The confusion matrices give further insight.

The confusion matrix of the validation data shows strong gender and emotion classification. However, the confusion matrix of the test data shows only strong gender classification with moderate emotion classification. No matter how the architecture was adjusted, model couldn't do better than about 45% testing accuracy. A 45% testing accuracy with 14 classes is still reasonably good, since random guessing would yield about 7%, but the goal was to obtain a much higher accuracy. Upon further inspection of the data, it was found that many of the audio samples were even difficult for humans to distinguish. This is why Table 4 includes the top 2 accuracy. The CNN achieves a top 2 accuracy of 95%, which proves the model is generalizing as opposed to randomly guessing. It is simply the case that the difference between some emotions are fuzzy and the quality of the actors in the datasets are not necessarily great.

The qualitative measures give some examples of inputs and associated outputs of the model. Figure 14 shows examples of the pre-trained CNN giving a correct and an incorrect prediction. Generally these are the two types of outcomes of the model. When the model is correct it has a high degree of confidence, and when it is incorrect it has a low degree of confidence. This is why the top 2 accuracy is so high relative to the top 1 accuracy. This further demonstrates the model has in fact generalized outside the training data.

Finally, Figure 13 and 15 show the results of the CNN after fine-tuning with inputs being audio recorded on our laptops. These results clearly show that transfer learning was successfully implemented.

There are two important take-aways from this project. First, the importance of the dataset. Initially, we were using only the RAVDESS with no augmentation and the model did not learn properly. Adding two additional datasets and augmentation we achieve very good results. The second key take-way is that reality is often not cleanly divided into categories. As a result, some of the data being used on the model could have been interpreted as many different emotions even though the label only contained one.

Some further improvements of the project are to include a wider set of emotions and look into more modern architecture tailored for emotion recognition. Furthermore, a multi-label classifier could be implemented as people often express multiple emotions at any given time.

Ethical Framework

The ethical implications of this project depend on the motives of those using it. Using the example given in the Background section, Uber is partnering with Attectiva to use emotion recognition software to improve customer experience in their vehicles [7]. In this scenario, better emotion recognition can increase beneficence and justice, since the service will be better tailored to the needs of the customer and Uber can ensure the same level of customer satisfaction. On the other hand, it can lead to a decrease in autonomy, since the customer is unwillingly giving emotional information to Uber. This example could apply to most companies in the service industry. In this scenario it leads to a lot of good, since what benefits the company also benefits the customer. However, this is not the case. Consider how an advertising company would use this product. They would be incentivized to exploit a person's current emotional state and advertise products the person might not otherwise buy. This could be seen as a decrease in beneficence and justice and a major decrease in autonomy.

Lastly, knowing that some device might be listening and recording your emotional state may cause paranoia and stress resulting in people expressing themselves differently. This can be seen as a decrease in autonomy as people will become much more calculated with the way they speak around others in fear of seeming a certain emotion and it's connotation.

References

[1] M. Egger, M. Ley, and S. Hanke, “Emotion Recognition from Physiological Signal Analysis: A Review,” Electronic Notes in Theoretical Computer Science, vol. 343, pp. 35–55, 2019.

[2] “Sentiment Analysis: The Only Guide You'll Ever Need,” MonkeyLearn, 09-Oct-2019. [Online]. Available: https://monkeylearn.com/sentiment-analysis/. [Accessed: 24-Oct-2019].

[3] Z. Huang, M. Dong, Q. Mao, and Y. Zhan, “Speech emotion recognition using cnn,” in Proceedings of the 22nd ACM international conference on Multimedia, pp. 801–804, ACM, 2014.

[4] S. Yoon, S. Byun, and K. Jung, “Multimodal speech emo- tion recognition using audio and text,” in 2018 IEEE Spoken Language Technology Workshop (SLT), pp. 112–118, IEEE, 2018.

[5] K. Han, D. Yu, and I. Tashev, “Speech emotion recognition using deep neural network and extreme learning machine,” in Fifteenth annual conference of the international speech communication association, 2014.

[6] L. Chen, S. Gunduz, and M. Ozsu, “Mixed Type Audio Classification with Support Vector Machine,” *2006 IEEE International Conference on Multimedia and Expo*, 2006.

[7] D. Coldewey, “Affectiva and Uber want to brighten your day with machine learning and emotional intelligence,” TechCrunch, 13-Sep-2016. [Online]. Available: https://techcrunch.com/2016/09/13/affectiva-and-uber-want-to-brighten-your-day-with-machine-learning-and-emotional-intelligence/ . [Accessed: 03-Dec-2019].

[8] Livingstone SR, Russo FA (2018) The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS): A dynamic, multimodal set of facial and vocal expressions in North American English. PLoS ONE 13(5): e0196391. https://doi.org/10.1371/journal.pone.0196391 .

[9] Surrey Audio-Visual Expressed Emotion (SAVEE) Database. [Online]. Available: http://kahlan.eps.surrey.ac.uk/savee/References.html . [Accessed: 3-Dec-2019].

[10] TSpace. [Online]. Available: https://tspace.library.utoronto.ca/handle/1807/24487 . [Accessed: 3-Dec-2019].