Emotionally Intelligent AI - Creating a Companion that Cares

December 10, 2020

Caleb Dame, Cory Hunter, and Daniel Mortenson

Classifying the emotions of a speaker is an mostly unexplored application of machine learning with potential to augment the capabilities of medical virtual assistants. In this paper, we discuss which models work for emotional classification of speech audio, along with the implications and feasibility of scaling the algorithm to work with the general populace. We show that predicting emotion using a Random Forest Classifier is a tenable solution. The ethics and hypothetical uses and misuses of this algorithm are set out.

1 Introduction

Processing human speech is one of the most prevalent topics in machine learning. Most current research involving language processing is aimed at translating audio into words, and then extracting meaning from a list of words. In this paper, we will explore the feasibility of predicting emotion directly from audio, without regard to the words or language being used.

1.1 Motivation

At the time of writing, every commonly used voice assistant follows this translation pattern: (1. audio 2. words 3. meaning). This excludes additional meaning hidden within the audio: the emotion of the speaker. Although the "7-38-55 rule" (used by many to describe the percentage of communication happening through words, tone of voice, and body language respectively) is completely false (Yaffe, Philip 2011), there is no doubt that the tone of a speaker's voice does affect the overall meaning of their speech. A personal assistant capable of recognizing emotion could have far-reaching implications for health care. Emotion state affects mental and physical health, especially for vulnerable populations including the elderly and ill. Virtual assistants equipped with voice emotion detection could free up time for healthcare professionals and provide an additional medical metric to measure the well-being of a patient.

1.2 Previous Research

Speech Emotion Recognition (SER), is an emerging area of study in the machine learning community. Landmark studies have shown that Deep Stride Convolutional Neural Networks (DSCNNs) are effective at learning and predicting emotion from audio (Mustaqeem and Kwon 2019). Since audio and emotion are both complicated and diverse, all recent research in SER has dealt with deep learning methods, which tend to be computationally expensive, cryptic and require massive ammounts of data. To best align with the purpose of the class and our current skillset, we have decided to use more simple machine learning methods, which train faster, are more interpretable, and require much less data.

1.3 Research Questions

In this paper, we use a labeled dataset to determine the feasibility of predicting emotion from voice audio. In 2011, Kate Dupuis and M. Kathleen Pichora-Fuller of the University of Toronto conducted a study analyzing the usage of prosody (speech rhythms) between older and younger people in different emotional states, comprising 2,800 audio samples. Fortunately, this dataset matches the needs of this paper. Using just the last word of each sentence in this labeled dataset, we will answer the following questions:

1. Using non-deep Machine Learning Methods, how well can we predict the emotions of a speaker, provided a personal, labeled dataset?
2. Which algorithm gives the most accurate results for predicting emotion?
3. Given labeled samples of multiple people and multiple emotions, how well can a machine learning algorithm predict a speaker's identity?

We understand that much more research will be required to develop a generalizable audio-emotion algorithm that works across all demographics, languages, etc. The purpose of this paper is to prove the feasibility of such an algorithm by developing an algorithm that makes accurate predictions on the small dataset we have available.

2 Data

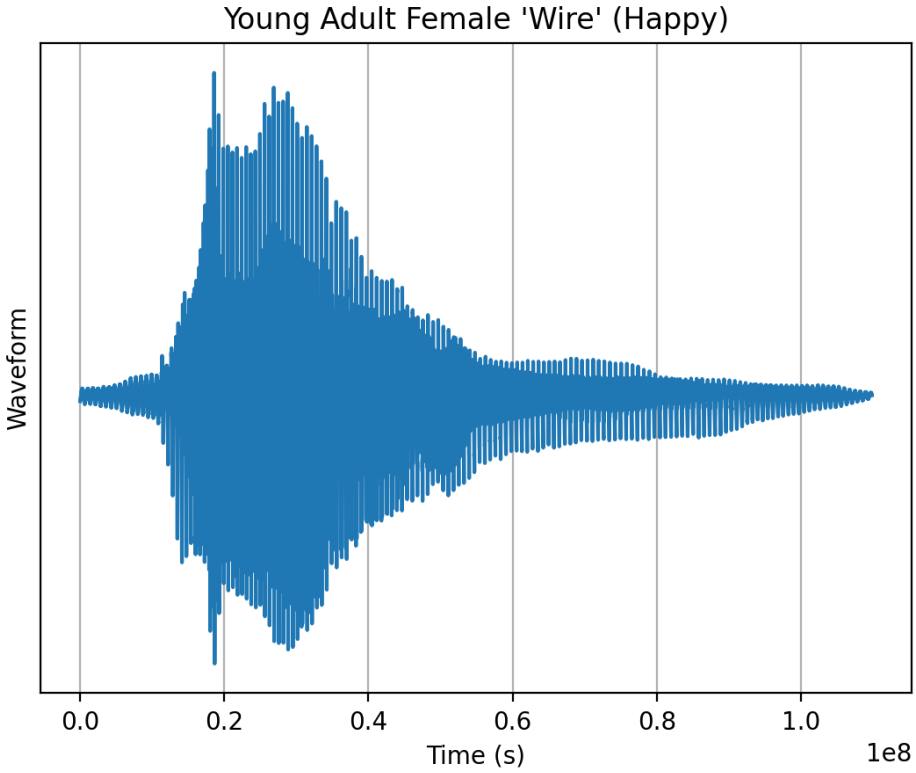
2.1 Dataset Origins

In addition to the results published by the University of Toronto (Dupuis & Pichora-Fuller 2010), the full research dataset was made available. The Toronto Emotional Speech Set (TESS) contains nearly 3000 audio recordings created for the purposes of the study. The audio files were recorded by two distinct voice actors (aged 26 and 64 years) recording a list of 200 one-syllable words embedded in the sentence "Say the word _____." in each of the following emotional tones: anger, disgust, fear, happiness, pleasant surprise, sadness, and neutral (lack of any emotional tone). The actors gave their consent for their voices to be recorded and released to the public for scientific research, and they received compensation for their contribution.

The limited nature of the dataset causes our analysis to not focus on building a generalized model that can detect emotion on any given voice; rather, we hope to provide that given the samples taken from the two voice actors, we may be able to produce a personalized model to detect emotional by changes in individual timbre, proving that the methods we will employ may be generalizable given data on a sufficient number of individuals.

2.2 Data Cleaning

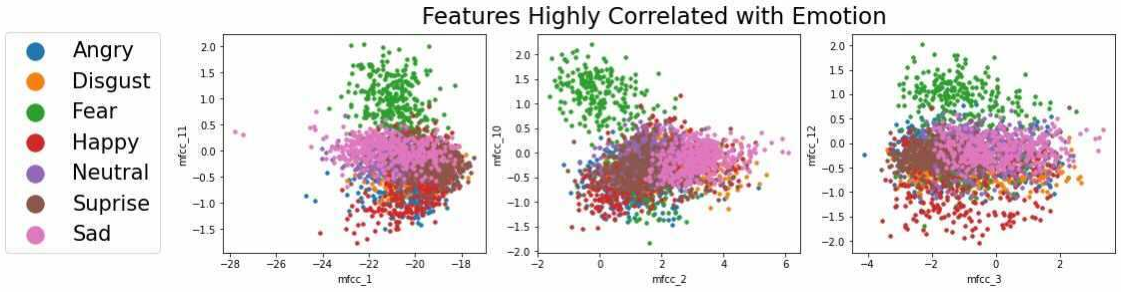
From the beginning of our project, we knew it would be less impressive if we included the identical first three words "Say the word" in each of our data samples. So, we went through each sample and clipped the first 75% of the duration of each audio sample. Since all 4 words in the sentence are mono-syllabic, this approach worked well. We spot-checked 20 random samples from each actor and each emotion, which amounts to 10% of our dataset, to make sure that we were not capturing any of the three previous words in each sentence. The audio clipping was successful.



2.2.1 Figure 1: Waveform of one audio sample from the TESS dataset of the young adult female saying "wire" with "happy" emotion.

2.3 Feature Extraction

To retrieve from the audio files and key descriptive information that is less prohibitive than the entire file we make use of the Open-Source Python Library for Audio Signal Analysis, PyAudioAnalysis (Giannakopoulos 2015), to extract key variables from each signal. Collected data includes metrics such as average frequencies per time-step, spectral values, Mel Frequency Cepstral Coefficients, different normalized energies and the degree to which each of the 12 western musical frequencies are present. We decided to simply the model and to define the timestep as the length of the audio file. This allowed us to analyze the entire file as a whole as well as reduce our data from a time series matrix for each file down to a vector of values. Our code was specified to extract the 33 unique features that were not dependent on having multiple steps to measure. Further analysis could be pursued to lower the timestep to further segment the signal to extract more of the above-mentioned key metrics for each signal segmentation, but after preliminary testing, it seemed that one segmentation for the two second audio file was sufficient to have interesting results and impressive accuracy scores. The package PyAudioAnalysis left no missing data to manage. Our initial visualizations of these features indicated that we would need to augment the dataset, since none of the features correlated very strongly with emotion. Our scatter plots showed that the emotions almost entirely overlapped eachother, except for disgust.

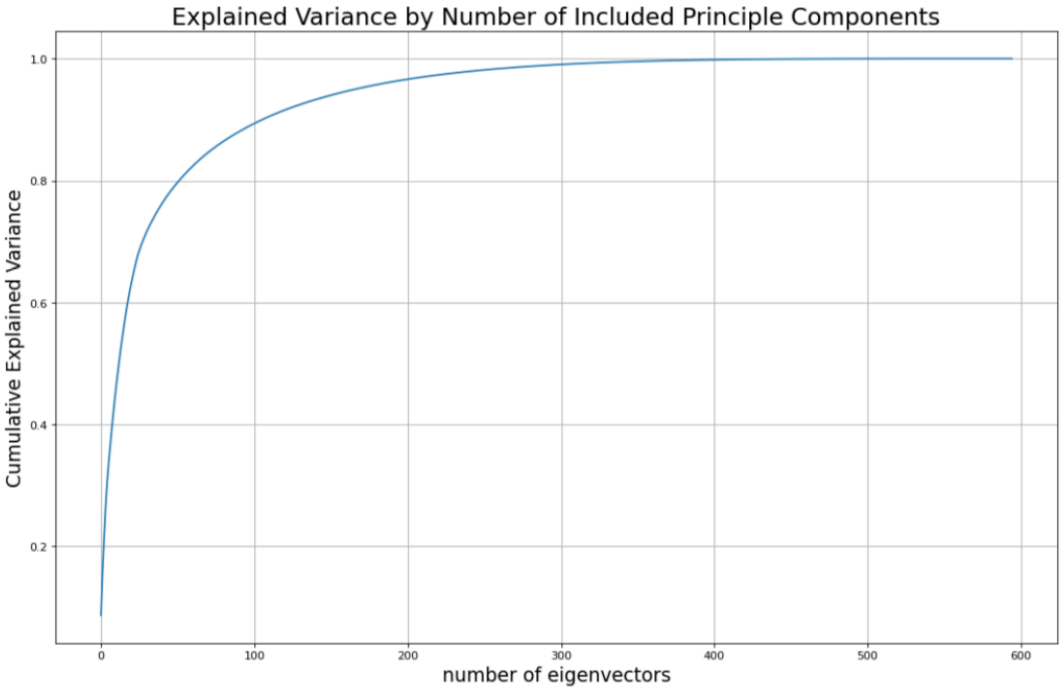


2.3.1 Figure 2: Scatterplots of Features Highly Correlated with Emotion

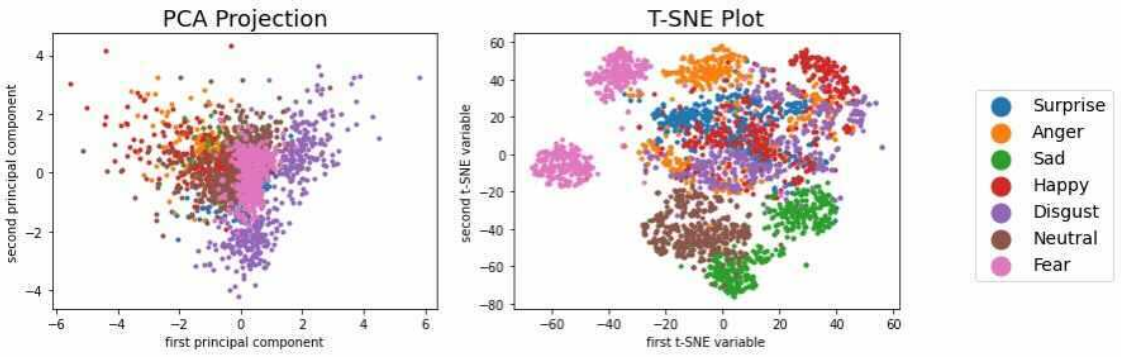
3 Methods and Models

3.1 Preprocessing

To begin, we analyzed the 33 pyAudioAnalysis features of our dataset, and determined that most of the features had no correlation to emotion, which was expected. After running a few preliminary models with the 33 features from pyAudioAnalysis, we determined that we needed to include more terms to allow for quasi-nonlinear correlations between the features. We augmented our dataset by included interaction terms (multiplying each combination of 2 terms together) and squared terms, which expanded the dataset to nearly 600 features. This allowed us to capture some of the non-linear relationships that were occurring in our data. This quickly aided our random forest algorithms to converge to a consistent accuracy without hours of training by making more data available at the growth of each new tree. These interactions, while not extremely useful with the choice of a Random Forest Classifier, aided both in clustering algorithms implemented later and in building a better principal components analysis (PCA) reconstruction. The PCA decomposition of our data will be used to remove 2% of the explained variance, ostensibly removing noise from our data to increase further the out-of-sample performance. Before the PCA decomposition, however, feature scaling was included to standardize the mean and variance of each feature to 0 and 1, respectively. As shown in Figure 1, much of the explained variance, once standardized, is given by the first dozen principal components, and the majority of the principal components are not required to reach 98% of the explained variance, showing evidence of much noise in the dataset.



3.1.1 Figure 3: Percentage Variance Explained by Principal Components



3.1.2 Figure 4: Principal Component Analysis and T-SNE Data Visualizations

3.2 Model Choice and Implementation

To answer each of our research questions, we decided to create 4 different models to measure the potential for different collections of data and desired outcomes.

The first model is labeled with and predicts on emotion alone, regardless of actor identity. This aligns with our primary research question. The second model is labeled by emotion and actor identity, and predicts only on emotion. We realized that providing labels for actor identity is a beneficial middle step in the process of predicting emotion. The third model is trained on one actor's voice and predicts emotion on the other actor's voice. This model shows that one voice does not generalize well to an unknown voice, and that a large, diverse training set will be needed to create a generalizable algorithm. The fourth model is labeled for and predicts the actor's identity, with both

actors and all emotions in the training and test sets. This model could act as an intermediate step between the audio and the emotion prediction, to determine which samples to train the emotion prediction model on based on the test set, since the emotion prediction model (Model 1) works best when the voices it is tested on voices similar to the voices it was trained on.

Model 1:

```
Not labeled by actor
Trained on 50% of the data
Target Classification: Emotion
```

Model 2:

```
Labeled by actor
Trained on 50% of the data
Target Classification: Emotion
```

Model 3:

```
Trained on all of one actor's samples (50% of the data)
Target Classification: Other actor's emotion
```

Model 4:

```
Trained on 50% of the data
Target Classification: Speaker
```

When we first approached the problem of creating a model to analyze emotions, we were unsure of which model would classify the emotions the best. Accordingly, we started with simple, linear models, and then moved to more advanced models.

First, we tried a one-vs-rest logistic (logit) classifier with the 33 original pyAudioAnalysis terms. This model correctly classified 77% of the data samples, which is less than what we wanted to acheive. Second, we tried a gradient-boosted tree model and an XGBoost model, which both performed very well with about 84% accuracy. Interestingly, using a Random Forest Classifier (RFC), we acheived slightly higher accuracy rates, and so we decided to conduct our further research using RFCs. We then decided to analyze our RFC model by looking at the individual error rates for each of the canonical emotions, which gives us further insight into our SER problem, which we discuss in the next section.

Preliminary Cross-Validated Models

	Model 1	Model 2	Model 3a	Model 3b	Model 4
Logistic ElasticNet (One v. Rest)	0.8378	0.859	0.439	0.389	0.94
Gradient Boosted Tree	0.8378	0.84	0.34	0.318	0.922
XGBoost	0.8357	0.844	0.491	0.49	0.93
Random Forest	0.8407	0.851	0.35	0.363	0.97
			Trained on Person 1	Trained on Person 2	

3.2.1 Figure 5: Preliminary Results using only 33 original pyAudioAnalysis features

3.3 Hyperparameter Tuning

Due to the exploratory nature of the study, the hyperparameter tuning for each model was modest. A grid search showed that the best number of trees was between 200 and 500 and the best method of determining how many of the n features to include in each tree $\sqrt{n}$. In further model tuning, leaf size and experimenting with entropy loss could result in a better model instead of not restricting leaf size or defaulting to gini loss, and would be advantageous to build a more performative model: however, more than a perfect model, we are much more interested in how performance differs between machine learning algorithms when data such as personal fixed effects (individual labeling of data by person) are and aren't available.

4 Results

4.1 Model Performance

After training each specified model, certain things became apparent. Firstly, when moving from Model 1 to Model 2, indicating the actor's identity seemed to aid prediction only slightly. The reason for this becomes apparent when comparing the first two model specifications to the two versions of Model 3, trained in turn on person 1 before testing on person 2, then person 2 to train for predicting for person 1. Both versions of the third model perform approximately as poorly chance in at least one emotion with an overall success rate only double that of chance. It appears that the metrics used to gauge the voice's tone and emotions are expressed by one person so very different from those of the other person, resulting in abysmal prediction out of a single person sample. The voices are so distinct that Model 4 was able to correctly predict the samples' owner slightly over 97% of the time. We predict that a larger dataset with more diverse voices would only exacerbate the issues of training on voices that are dissimilar to the voices the algorithm is testing.

Random Forest Classifier with Interaction Terms

	Model 1	Model 2	Model 3a	Model 3b
Overall Accuracy Score	0.8407	0.85	0.3507	0.3628
Happy	0.66	0.645	0.02	0.025
Surprise	0.773	0.792	0.565	0.585
Anger	0.777	0.756	0.17	0.065
Sad	0.963	0.963	0.115	0.56
Disgust	0.851	0.894	0.585	0.575
Neutral	0.894	0.935	0.85	0.58
Fear	0.979	0.979	0.15	0.15
			Trained on Person 1	Trained on Person 2

	Model 4
	0.97
Person 1	0.9682
Person 2	0.9718

4.1.1 Figure 6: RFC Classification broken down by emotion for each model

Most satisfactorily, the accuracies of Models 1 and 2 do show that it is very possible to train a model to understand various emotions. The plurality of the errors in both models was the model's confusion between "Happy" and "Pleasant Surprised," which is very understandable as the writers are not quite sure of the difference themselves, and once the two emotions are combined, the accuracy jumps to above 90%.

Additionally, we see that six of the seven canonical emotions saw higher classification rates when

the actor identity was included in the training data. This suggests that fine-tuning or specific sampling of the training data to match the target voice could be beneficial in future research. We also see that the classification rates plummeted accross all seven emotions when the model was trained entirely on one dataset and tested on the other. This is indicative of over-fitting to the voices in our dataset, which was expected. We cannot expect these models to perform well having only been trained on voices dissimilar to the target subject's voice.

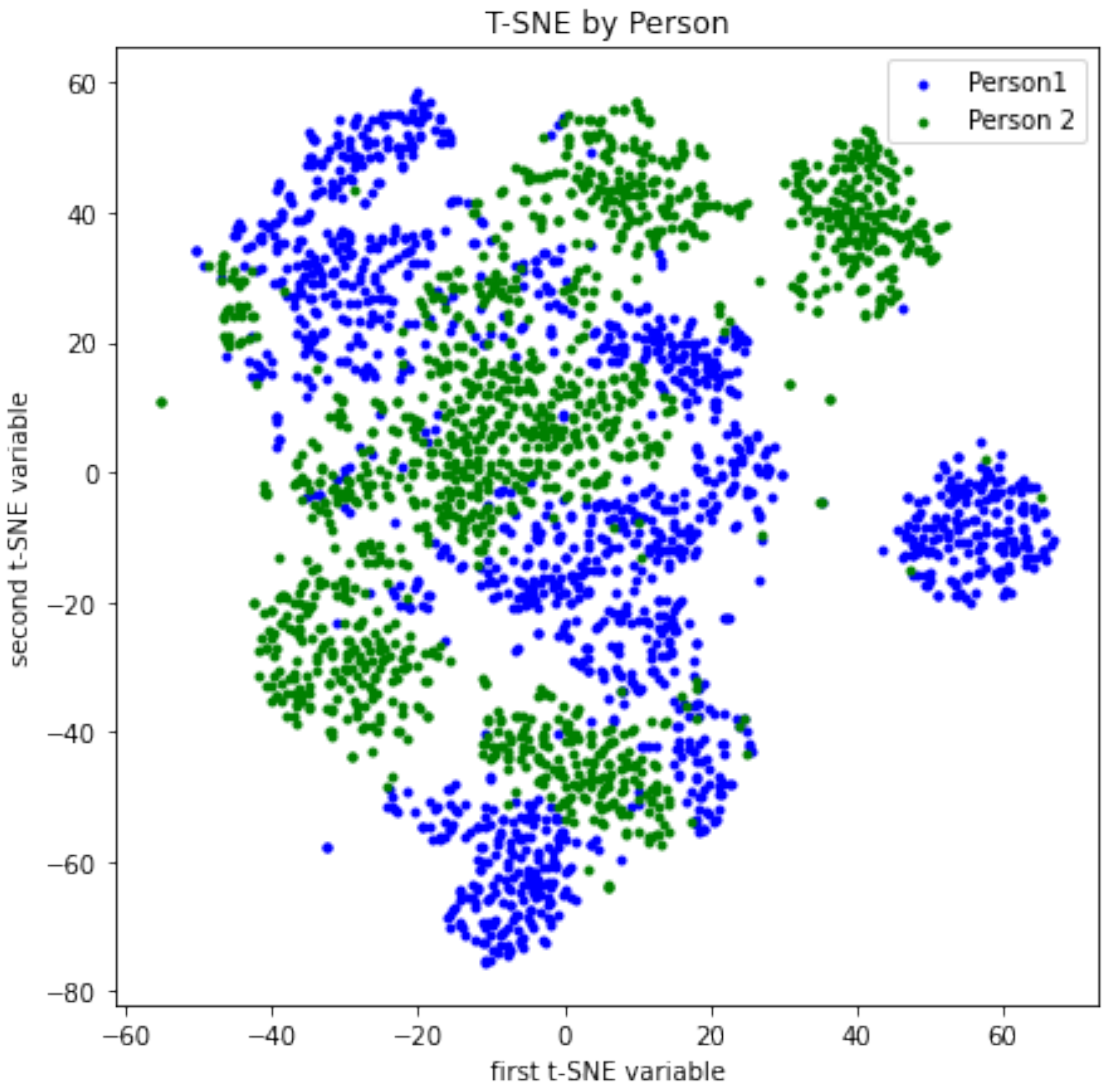
While in low dimensions the data is quite noisy and the Random Forest approach works by moving along key features and usefully segmenting the data into purer groups. While impossible to visualize the 200+ featurespace, two dimensional transformations (PCA and t-Distributed Stochastic Neighbor Embedding) are shown in Figure 4 above, indicating that there are in fact somewhat separable clusters in the featurespace that reveal decent identification strategies.

5 Conclusion

5.1 Implications

Using clean, well-labeled training data from our test subjects, our algorithms were able to predict on the seven canonical emotions remarkably well, proving our hypothesis true. There is a lot more to unpack here besides the 85% accuracy achieved by the random forest algorithm.

First, certain aspects of our dataset made the classification process more simple than it would be in real life. There is an obvious benefit of training and testing on just one or two peoples' voices. Although we made sure that none of the test data was included in the training set, models 1, 2, and 4 were trained and tested on the same two peoples' voices. This is what caused the ~14 visible clusters in our T-SNE plot: both people had their own cluster for each of the 7 emotions. This is even more apparent if we color our T-SNE plot by person, as we can see in figure. However, this limitation in our dataset is likely representative of how this algorithm would be implemented in a product. Just as Siri learns a specific person's voice by having them say "Hey Siri" a few times during setup, our theoretical medical assistant would be provided with a few pieces of person-specific training data to use to fine-tune whatever pretrained algorithm it had been trained on.



5.1.1 Figure 7: T-SNE plot colored by person

On the other hand, some aspects of our data were more limiting than they would be in real life: our theoretical medical assistant would likely have access to more descriptive audio files. Since our dataset is composed of random single-syllable words, our algorithm cannot learn features pertaining to the prosody (rhythm) or tempo (speed) of a person's speech, which change depending on the emotion the speaker is feeling. Real-world usage of such an algorithm would include entire sentences and phrases, providing more rich features than can be extracted from single-syllable words. Additionally, it is worth mentioning that the majority of errors made by the algorithm would not be considered grave ones in most situations. As mentioned in section IV.A, the emotions "Happy" and "Pleasant Surprise" are quite similar sounding, and also have similar meanings. The most critical emotions to classify correctly for a medical assistant would likely be "Anger," "Sadness," and "Fear," all of which could be warning signs of distress. The combined accuracy of these 3 emotions for model 2 is 89.93%. Although we didn't study this scientifically, we were ourselves less than perfect at guessing between "Happy" and "Pleasant Surprise" when listening to a single sample of our dataset. Regardless, we accomplished our goal for a 'proof of concept' of this algorithm.

We believe that given time, and funding, we could build a dataset and general model capable of classifying the emotions of a general user who has provided a few audio samples to fine tune the algorithm.

5.2 Ethics

While our intentions for this project and the hypothetical development of an emotionally intelligent medical virtual assistant are pure, this algorithm has potential to be misused and the data that it would use must be safely protected. While the data we used is present in the public domain for use, the data that an assistant would collect would be personally identifying and therefore sensitive. To safeguard against hacking, our assistant would be equipped with a low-power machine learning processor and on-board memory so that the computation using the collected audio data can happen locally, on the user's device. This way, the device can come pre-loaded with a pre-trained base algorithm that can be fine-tuned by the device as it receives more and more data from the user. We also believe that keeping the algorithm proprietary would protect against mal-use of the algorithm. We can see how an emotion classifier could be used as a loss function on a neural net that could be optimized for producing negative emotions within people. While this is hypothetical, it is terrifying to think that someone could use an emotion classifier to induce personal trauma. For that reason, we would not release the code or parameters used by the model.

6 Bibliography

Dupuis, K., & Pichora-Fuller, M. K. (2010). Use of affective prosody by young and older adults. Psychology and Aging, 25(1), 16–29. https://doi.org/10.1037/a0018777

Giannakopoulos T (2015) pyAudioAnalysis: An Open-Source Python Library for Audio Signal Analysis. PLoS ONE 10(12): e0144610. https://doi.org/10.1371/journal.pone.0144610

Mustaqeem, & Kwon, S. (2019). A CNN-Assisted Enhanced Audio Signal Processing for Speech Emotion Recognition. Sensors (Basel, Switzerland), 20(1), 183. https://doi.org/10.3390/s20010183

Pichora-Fuller, M. Kathleen; Dupuis, Kate, 2020, "Toronto emotional speech set (TESS)", Scholars Portal Dataverse, V1, https://doi.org/10.5683/SP2/E8H2MF