Audio Feature Classification in Voice Recordings

Calli Fuller, Ethan Bokelberg, Garrett Faucher, Finn Dority

Opportunities with Audio Data

- Lots of text to speech AI, but communication has nuances beyond just the words being said
- Emotion is an important identifier to what people are saying, and gender is a useful classifier for identifying speakers, and can make identifying emotions easier as well
- Identifying emotion of a statement can help deaf people or others relying on closed captions better understand the context and meaning of it

Data Set

We used the Ryerson Audio-Visual Database of Emotional Speech and Song as our dataset for the project. This dataset is widely used in machine learning projects working with voice data, and is well-labeled, which makes getting it into a form we can feed into an algorithm straightforward

The labelling is in the WAV filenames, with features such as emotion of the recording and gender of the actor included, making it easy to assign the target classification values for each clip.

Data Set Examples

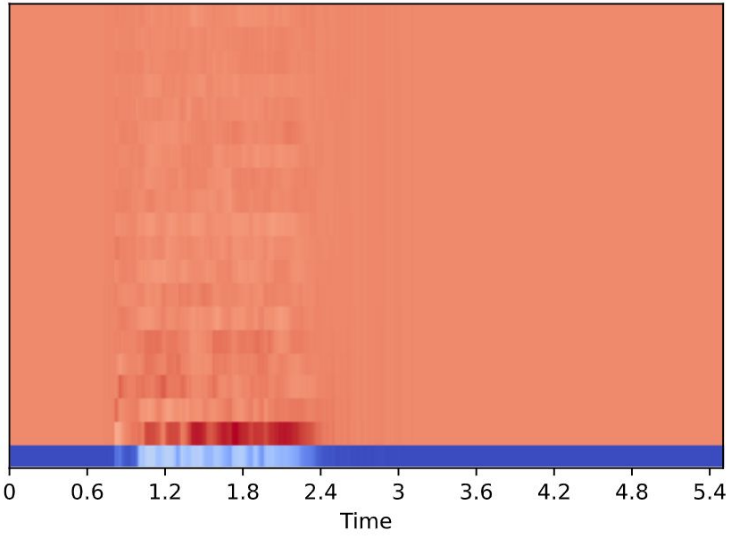
Male: Dogs at the Door

- Neutral 🔊
- Happy 🔊
- Disgust 🔊

Female: Kids Talking By The Door

- Neutral 🔊
- Happy 🔊
- Disgust 🔊

MFCCs



MFCCs, or Mel-Frequency Cepstral Coefficients, are a commonly used representation of sound, which was why we chose to use them as the form in which our data was fed into the algorithms.

- Librosa Python library
- MFCCs recommended by other papers
- MFCC data needed to be uniform in size
- Padding added to end of audio clips

Machine Learning Algorithms

We decided that with a larger group, we could cover more ground in terms of exploring which algorithms provide the best results for classification, so we had each member work with a different algorithm and compared results to see which was worth pursuing further in terms of optimizing performance.

This turned out to be the Neural Networks, which was not particularly surprising, as we knew from the published papers that we read that Neural Networks are a popular choice when working with audio data.

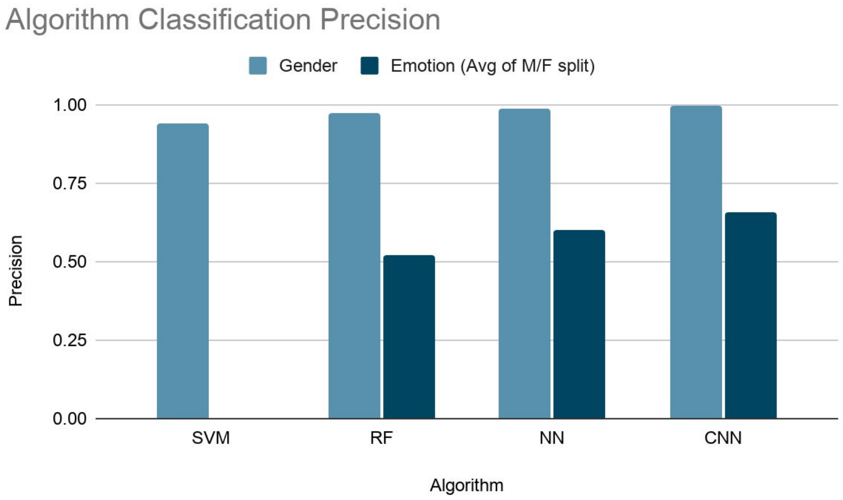
- Support Vector Machines
- Random Forest Classifier
- Neural Network
- Convolutional Neural Network

Algorithm Comparison

This graph shows a good comparison of the performance we found for our algorithms, as you can see, the precision for gender classification is fairly high for all of them, but the Neural Network and Convolutional Neural Network were able to get almost 100% precision.

The SVM classifying on only gender was from our initial pre-proposal results, and as you can see the other algorithms improved on it.

It also shows the clear improvement in performance on emotion classification in the CNN compared to the other algorithms.



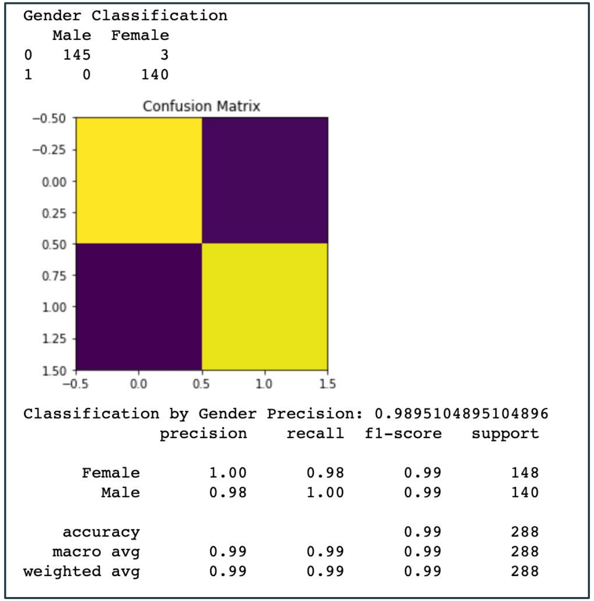
Non-Neural Network Results

As previously mentioned, our initial results were using an SVM to classify on only gender, which gave us a precision of approximately 93%. This was improved upon in the Random Forest, and that itself was further improved on by the Neural Network and Convolutional Neural Network.

There was a noticeable difference in results between gender classification and emotion classification present from the Random Forest, which gave a precision of 97% for gender classification and 47% for emotion classification. Like in the neural networks, separating the data by gender prior to classifying by emotion showed an improvement in performance, with precision for male samples increasing to 50% and increasing to 54% in female samples for an average total precision of 52%. However, this was still noticeably worse performance than the neural networks, so we elected to focus on optimizing those for further improving results.

NN Results

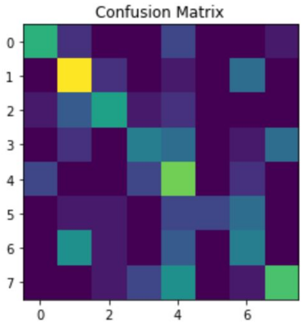
Speaker Gender Classification



```
Gender Classification
   Male  Female
0   145       3
1     0     140
```

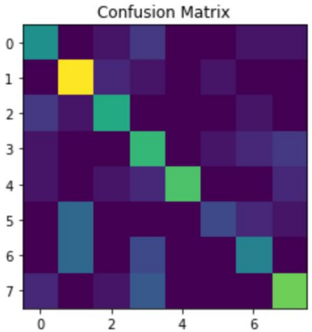
```
Classification by Gender Precision: 0.9895104895104896
              precision    recall  f1-score   support

      Female       1.00      0.98      0.99       148
        Male       0.98      1.00      0.99       140

    accuracy                           0.99       288
   macro avg       0.99      0.99      0.99       288
weighted avg       0.99      0.99      0.99       288
```

Male Emotion Classification:



Precision = 54%

Female Emotion Classification:

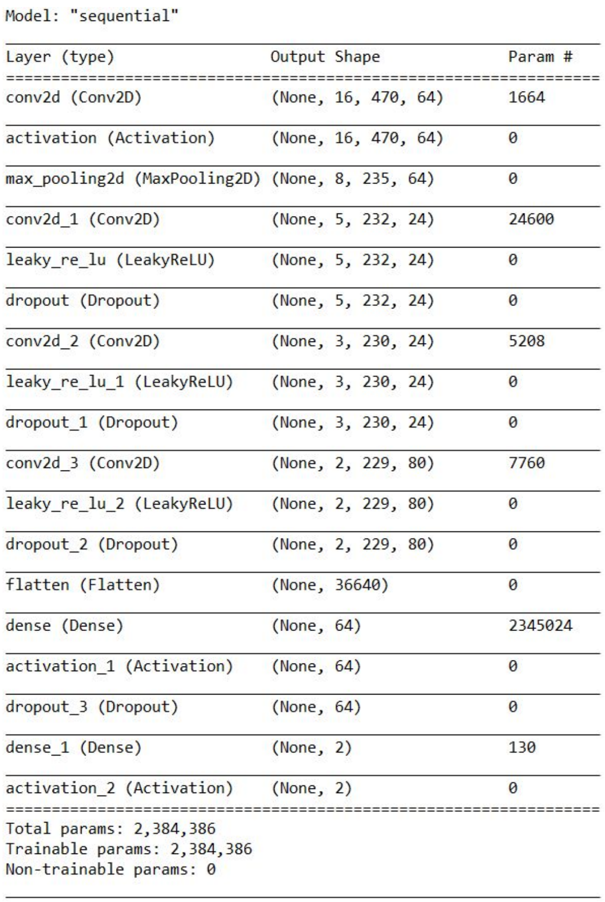


Precision = 62%

0 = Neutral	4 = Angry
1 = Calm	5 = Fearful
2 = Happy	6 = Disgust
3 = Sad	7 = Surprised

CNN Architecture

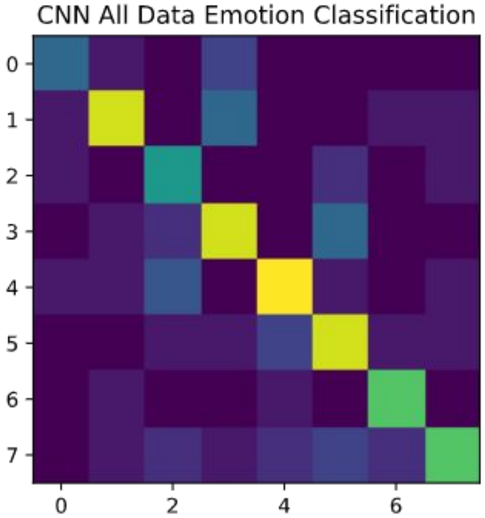
- Four convolutional layers
  - # of filters ranged from 24 to 80
  - Leaky ReLU
  - Dropout ranging from 0.1 to 0.5
  - One max pooling after first layer
- Layers flattened and two dense layers after
- Softmax output layer
  - 8 units, one for each emotion
- Used categorical cross entropy and Adam optimizer
- Hyperparameter tuning automated via KerasTuner
  - GPU access for TensorFlow
  - Quick iteration
  - Made for an odd but accurate architecture



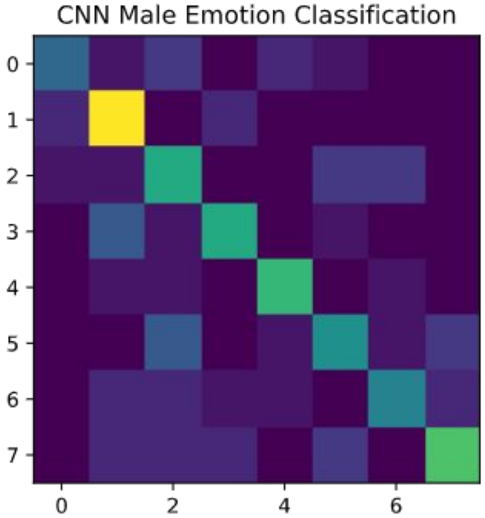
```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d (Conv2D)              (None, 16, 470, 64)       1664
_________________________________________________________________
activation (Activation)      (None, 16, 470, 64)       0
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 8, 235, 64)        0
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 5, 232, 24)        24600
_________________________________________________________________
leaky_re_lu (LeakyReLU)      (None, 5, 232, 24)        0
_________________________________________________________________
dropout (Dropout)            (None, 5, 232, 24)        0
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 3, 230, 24)        5208
_________________________________________________________________
leaky_re_lu_1 (LeakyReLU)    (None, 3, 230, 24)        0
_________________________________________________________________
dropout_1 (Dropout)          (None, 3, 230, 24)        0
_________________________________________________________________
conv2d_3 (Conv2D)            (None, 2, 229, 80)        7760
_________________________________________________________________
leaky_re_lu_2 (LeakyReLU)    (None, 2, 229, 80)        0
_________________________________________________________________
dropout_2 (Dropout)          (None, 2, 229, 80)        0
_________________________________________________________________
flatten (Flatten)            (None, 36640)             0
_________________________________________________________________
dense (Dense)                (None, 64)                2345024
_________________________________________________________________
activation_1 (Activation)    (None, 64)                0
_________________________________________________________________
dropout_3 (Dropout)          (None, 64)                0
_________________________________________________________________
dense_1 (Dense)              (None, 2)                 130
_________________________________________________________________
activation_2 (Activation)    (None, 2)                 0
=================================================================
Total params: 2,384,386
Trainable params: 2,384,386
Non-trainable params: 0
_________________________________________________________________
```

CNN Results

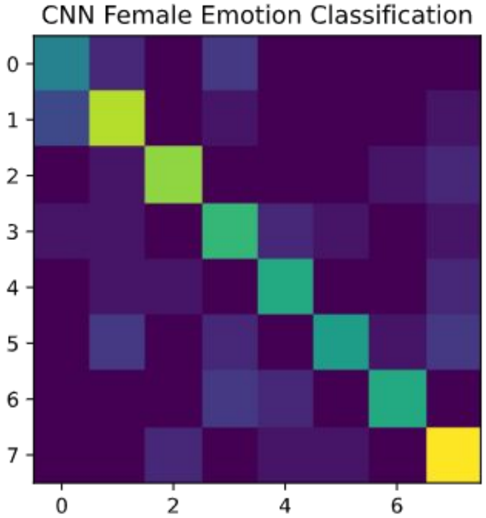
0 = Neutral
1 = Calm
2 = Happy
3 = Sad
4 = Angry
5 = Fearful
6 = Disgust
7 = Surprised


CNN All Data Emotion Classification

Precision = 63.8%


CNN Male Emotion Classification

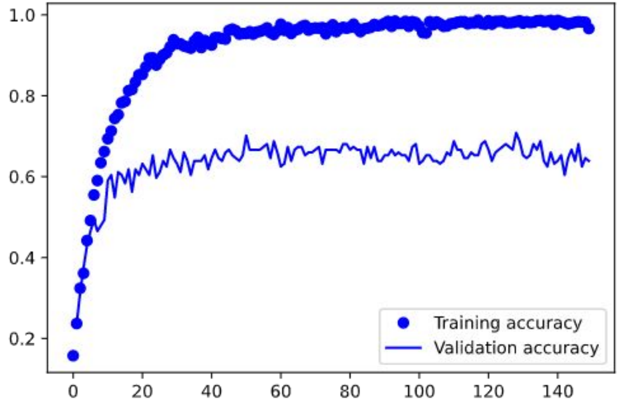
Precision = 61.1%

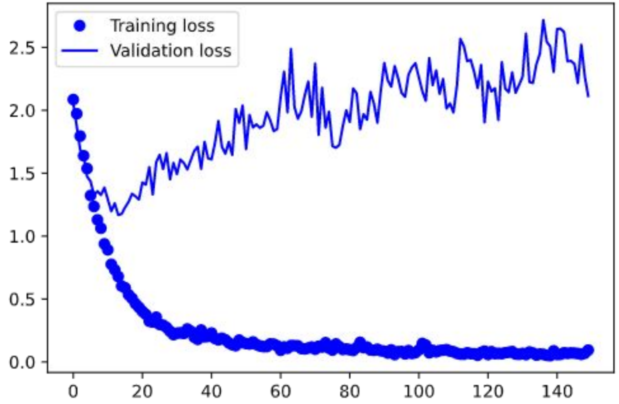

CNN Female Emotion Classification

Precision = 70.1%

CNN Issues

- Small dataset made avoiding overfitting difficult
- Dropout didn’t help
- Test train split didn’t help
- Early stopping was helpful sometimes
  - Inconsistent results
  - Model certainly overtrained


All Data Emotion Classification, Training and Validation Accuracy


All Data Emotion Classification, All Data Training and Validation Loss

Algorithm Evaluation vs Other Work

In terms of evaluating our results, our recorded 71% accuracy on emotion classification is ever-so-slightly better than some other attempts to do this same task - the highest accuracy we've found in the wild was 70.2%, done by the SRM Institute of Science and Technology, which was done with a Multi-Layer Perceptron.

As such, we can say with fair confidence that our algorithm, although not perfect, is more than competent for the software and computing capabilities that we have available today.

For comparing via gender classification, we found a similar project that reached 99% accuracy through an ensemble learning technique. For reference, our neural network also reached a 98.95% accuracy on this task. It's hard to get much better than this, and we're very proud of our success there as well.

Algorithm Evaluation vs Humans

In terms of algorithm evaluation versus humans, nobody's managed to match humans on emotion recognition, but we're definitely getting there.

According to a study published in June this year by Adi Lausen and Kurt Hammerschmidt, humans have an approximate 80% accuracy on this exact topic - participants were given **7** emotions to choose from to decide what a sample expressed, as compared to our 8, so this study very closely matches what our algorithm was given to choose between.

What's worth mentioning is that although humans are bad at detecting surprise, our algorithm is bad at detecting anger and good at detecting surprise.

Questions?