SpeakFeel: Emotion Recognition In Speech

Milestone Three: Model Building and Evaluation

Bach Le, Kien Tran, Sauryanshu Khanal and Sike Ogieva.

Project Overview

The objective of this project is to build a neural network to classify human speech data by emotion. We use the audio speech-only portion of the Ryerson Audio-Visual Database of Emotional Speech and Song [3] to train and evaluate our model.

Earlier, we looked at literature to find and consider and evaluate earlier work and results on this topic. We also collected this data, analyzed it for quality, and augmented the voice files by adding noise to, stretching, shifting and changing the pitch of the audio file. This time, we were extra careful to augment only the training data.

In this milestone, we do a final review of related literature, and leverage the details to build several models, train them on our augmented data, adjust their hyperparameters and evaluate them.

Reviewing Related Works

First, we analyze the techniques that were used in other related works and select for patterns which would benefit our model.

Model 1**:** This work introduces a deep learning architecture based on Long Short-Term Memory (LSTM) networks. LSTM units are designed to maintain information in a 'memory-like' component over long sequences. We believe that this will be crucial for our project since speech data that involves the emotional context or cues might span several time steps. The LSTM can retain this information and use it to make more accurate predictions about the emotional state expressed in the speech.

- During training, the LSTM uses Backpropagation Through Time, which allows it to learn weights based on the error contributions of each timestep in the input sequence. This is vital for adjusting the model based on the temporal progression of speech patterns.
- The LSTM structure with gating mechanisms (input, forget, and output gates) helps in moderating the flow of gradients during backpropagation. This control helps prevent the gradients from vanishing (becoming too small to make any significant change) or exploding (becoming too large, leading to unstable training dynamics).
- With 1024 units, the LSTM layer in our model can generate a rich and nuanced set of features representing different dynamics and variations in speech. This is particularly important for emotion recognition. It also indicates a high capacity to learn from complex data structure.

Model 2: In reviewing the existing literature, particularly the approach detailed in Model 2 for Speech Emotion Recognition (SER) using a convolutional neural network (CNN), several key insights can be drawn that may influence the development and optimization of our model.

- This model employs multiple Conv1D layers with an increasing number of filters. This is a strategic approach as it allows the model to extract and learn from more complex patterns progressively. To further optimize these layers in our own project, we could experiment with different kernel sizes and strides to better capture the nuances in the feature maps produced at each layer

Drawing on these works, we can further experiment with integrating LSTM with CNN layers.

SpeakFeel: Emotion Recognition In Speech

Milestone Three: Model Building and Evaluation

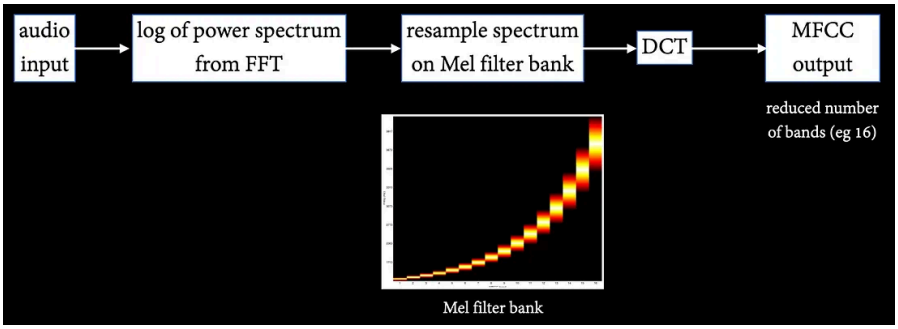
Bach Le, Kien Tran, Sauryanshu Khanal and Sike Ogieva.

Feature Extraction

We start by converting our audio files into numerical forms, which we can feed into a neural network or build a kNN model around. We have two major options of:

1. Mel Frequency Cepstral Coefficients (MFCCs)

These are the classic numerical features for sound modeling and having been designed to mimic human hearing, they are most common features used in voice-recognition systems. The MFCCs of an audio file are a small set of features (usually about 10 - 20) which describe the overall shape of the spectral envelope [1]. In order to be thorough, we extract 40 MFCCs per audio file for our project.



2. Chroma Short Term Fourier Transform

Unlike the all-encompassing MFCCs, chroma features are designed to more closely represent pitch, specifically. They are more common in music detection, rather than human voice detection. However, we have still tested models built on this feature, and achieved similar results to MFCC.

Other features we found in literature include: Zero Cross Rate Energy, Entropy of Energy , Spectral Centroid , Spectral Spread, Spectral Entropy, Spectral Flux, Spectral Roll off, Chroma Vector and. Chroma Deviation [2]. Since we already achieved a high accuracy with the most popular voice model features – MFCCs and Chroma Vectors– the marginal benefits of trying other features are low.

Data Processing Considerations

Apart from what features to extract, we had two other questions concerning data processing. Since our audio files were of slightly differing lengths (but all around three seconds), we wanted to know whether there would be some benefit in padding the data with silence to have them be all the same size before we extracted the MFCCs. We found that this was not only unnecessary but to be avoided, as models trained on the padded data performed horribly.

The second question was how much we should augment our dataset which originally contained 1_440 files. At first, we increased that to 3_600 files and achieved accuracies around 40% on an LSTM network. But generating up to 7_200 files in total gave us much better accuracies of around 60%. Finally, we

settled on 12_000 files, which also provided an increment in accuracy, but this improvement was not as significant as the 3_600 to 7_200 jump.

Model Design

We built several models to determine how several architectures performed on the problem of speech emotion recognition. Each of them was trained on our carefully augmented dataset, and evaluated with accuracy, precision, recall, F1 scores, as well as classification reports and confusion matrices. Regularization was achieved by early stopping, drop-out layers, batch normalization and the monitoring and adjustment of the learning rate.

K Nearest Neighbours

We started by building a baseline kNN model to evaluate our neural networks against.
Cross-validation was performed on an array of possible k values using ten splits and the F1 score as a metric, to find that k = 1 makes for the best number of neighbors, and that F1 steeply drops off as k is increased.
Then we fit a kNN model on our training data. It gave us an accuracy of 55%, which is considerably better than we would expect for say, a random classifier (that is ⅛ or 12.5%).

Multilayer Perceptron

Our first neural network was a fully-connected, feed-forward one. MLPs are good at capturing interactions between features at a global level. They are also versatile and adaptable, and we thought their ability to solve our problem was worth exploring.
Our model had five hidden layers and 131_688 trainable parameters, and achieved an accuracy of 56%.

Convolutional Neural Network

Given their success in capturing spatial relationships in data and our reference to earlier work, we thought a CNN might be well-suited to capturing the temporal relationships in our MFCCs. Our CNN had seven hidden layers and 110_216 trainable parameters, as we wanted to exploit the capability of CNNs to improve with depth. It achieved an accuracy of 45%.

Long Short Term Memory Neural Network

Given the conventional nature of LSTMs in solving sequence prediction problems, like text, speech and video classifications, and the promising results we observed in related works, we expected a marked improvement in performance. Our pure LSTM model had a single hidden layer of 1024 nodes and achieved an accuracy of 54%.

Combined (CNN + LSTM) Neural Network

We hoped that leveraging the strengths of both the LSTM and CNN architectures in a single model would prove valuable in capturing both the textural and temporal nuances of speech. This model had three Conv1D layers and one LSTM layer and it achieved an accuracy of 64%.

SpeakFeel: Emotion Recognition In Speech

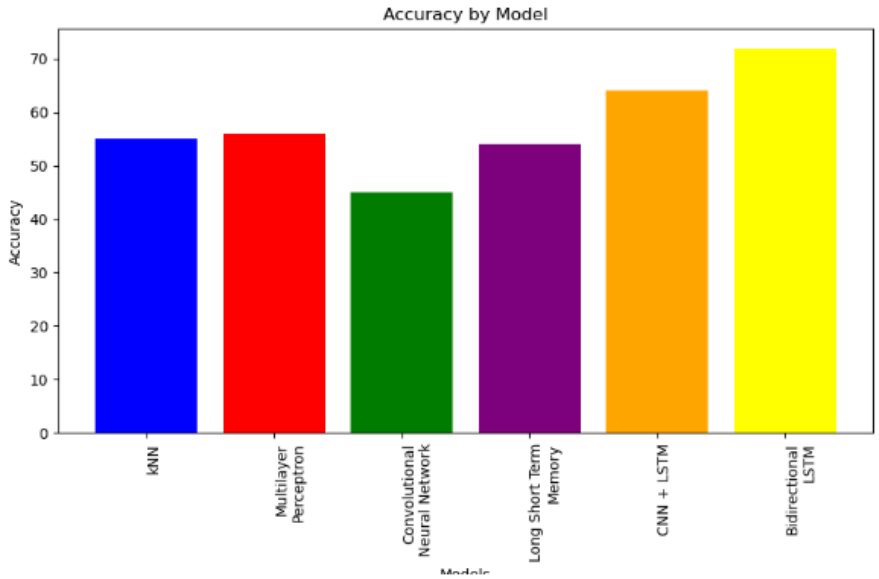
Milestone Three: Model Building and Evaluation

Bach Le, Kien Tran, Sauryanshu Khanal and Sike Ogieva.

Bidirectional LSTM Neural Network

Bidirectional LSTMs are an extension of traditional LSTMs where the input data is processed in both forward and backward directions (two LSTMs), which means that the network has information from past (backward) and future (forward) states simultaneously.

We infer that this additional context led to the best performance we saw, in this milestone– an accuracy of 72%.



Citations

1. Intuitive understanding of MFCCs. The mel frequency cepstral coefficients… | by Emmanuel Deruty | Medium
2. Audio signal feature extraction and clustering | by Pipe Runner | Project Heuristics | Medium
3. RAVDESS Emotional Speech Dataset | Kaggle
4. Ravdess + Crema + Tess + Savee Speech Emotion Datasets on Kaggle
5. SER Notebook - AbdelRahman on Kaggle
6. SER Notebook - Aditya on Kaggle
7. Our Project On Github