Hidden Markov Model for Emotion Detection in Speech

Cyprien de Lichy, Pratyush Havelia, Raunaq Rewari

***Abstract*—This paper seeks to classify speech inputs into emotion labels. Emotions are key to effective communication and their accurate detection in speech gives way to numerous applications that improve human computer interaction, leading to advancements in various other fields. In order to achieve this, a Gaussian Hidden Markov Model (HMM) is used, to take into account the temporal nature of the speech signals. A Gradient boosting machine using decision trees approach has also been tried and has shown to have achieved a high accuracy of 84.4%. The "Emotional Prosody Speech database", which consists of "acted" speech samples was used for training.**

I. Introduction

EMOTIONS are one of the most essential components of communication. They affect education, decision-making, personal relationships and human perception in general. With the growing dependency on technology and recent innovations in artificial intelligence, it has become imperative that human-computer interaction should become very efficient and accurate. Since emotions are a vital part of effective communication, there is currently a lot of dedicated research in improving emotion detection from human actions. Additionally, as speech is the most widely used method of communication, our team has decided to dedicate our efforts to working on an efficacious learning algorithm that would contribute towards improving human-computer interaction. [1]

Besides human-computer interaction, we feel some of the areas impacted most by this research include emotion detection support for selectively gifted individuals, emotion control training for professionals, public speaking training routines, decryption of vocal messages and cross-lingual communication.

The objective of the paper is to classify an audio sample into one of the 6 emotions used during training- Anger, Boredom, Disgust, Happy, Neutral and Sadness. First, a Gradient Boosting Machine (GBM) with decision trees using global features extracted from the utterances was trained to get some first insight into which features are important and get a first measure of accuracy. Features related to pitch, energy, MFCCs (Mel Frequency Cepstral Coefficients) for the entire duration of the audio sample were used as inputs to train the GBM. In contrast, the HMMs require low-level features that can be extracted from a short time period within the utterance (a frame). One HMM was trained for each of the six emotion categories, with the inputs to each HMM being pitch, energy features extracted for each frame in the audio sample. This ensures that our model takes into account the temporal variation in an audio sample.

II. Related Work

There has been significant research towards detecting emotions in speech recently. Researchers have experimented with various algorithms like Support Vector Machines [1], artificial neural networks, linear discriminant analysis, k-nearest neighbors [2] and hidden markov models [3], [4], [5].

Based on the consistently high accuracies achieved by past researchers using HMM's: 80% with 6 emotion tags [3], 86% with 7 emotion tags [4] and 78% with 6 emotion tags [5], we decided to carry out our research using HMM's with 6 emotion tags: **Anger, Boredom, Disgust, Happy, Neutral** and **Sadness**.

III. Data Set and Features

The first and arguably the most important part of the project was to get a suitable dataset. We decided to use "acted" speech instead of using "natural" speech as it is less noisy. We decided to use the "Emotional Prosody Speech and Transcripts" database [8], courtesy of the Linguistics Department at Stanford, because of good articulation of words and proof of previous success for other authors using this dataset.

An important aspect of tackling any machine learning problem is to pre-process the data to get a clean dataset, devoid of observable errors. The dataset contained a significant number of corrupt sound files which we had to remove as they lead to invalid features. The dataset also contained multiple entries where the labels did not make sense, like "mamamama", "package" etc. After pruning the data set, we were left with 4676 audio samples to further divide into training and testing sets. We used an 80-20 split to divide the data into training and test set. This is where we feel our approach of setting up a pipeline (using IPython Notebook) in place helped us continuously experiment.

The initial set of emotions contained 14 emotions (and the neutral state). Because this is a high number of classes, we decided to focus only on a few categories of emotion. We needed a way to naturally club some categories together and so decided to run a k-means on the data by hiding the labels to see if there is a natural aggregation of some emotion categories. After playing with the parameters of the k-means algorithm, using different subsets of features for training, we realized that no pattern is clearly seen in the data. For example, one of the 6 clusters that we obtained had a very high concentration of both "dominant" labelled and "passive" labelled training examples. It does not make sense for two opposite emotion categories to be in the same cluster. We eventually decided to combine categories based on our understanding of which emotions can

be put together. As described later in the paper, this leads to a low classification error on our test set.

Once we had the data set cleaned and divided into training and testing sets, we used **PRAAT**, a software package for phonetics analysis in speech, to extract out an initial set of features using the scripting language within the application. The features included:

- Fraction voiced frames
- Mean and standard deviation of pitch
- Minimum and maximum pitch
- Mean abs slope of pitch
- Time of minimum and maximum pitch
- Mean and standard deviation of intensity
- Minimum and maximum intensity
- Time of minimum and maximum
- Mean and standard deviation of first 3 formants
- Mean and standard deviation of first 13 MFCC's

It intuitively makes a lot of sense to have different kinds of features related to pitch and intensity of the audio sample. The features related to these quantities would make more sense on carefully studying III

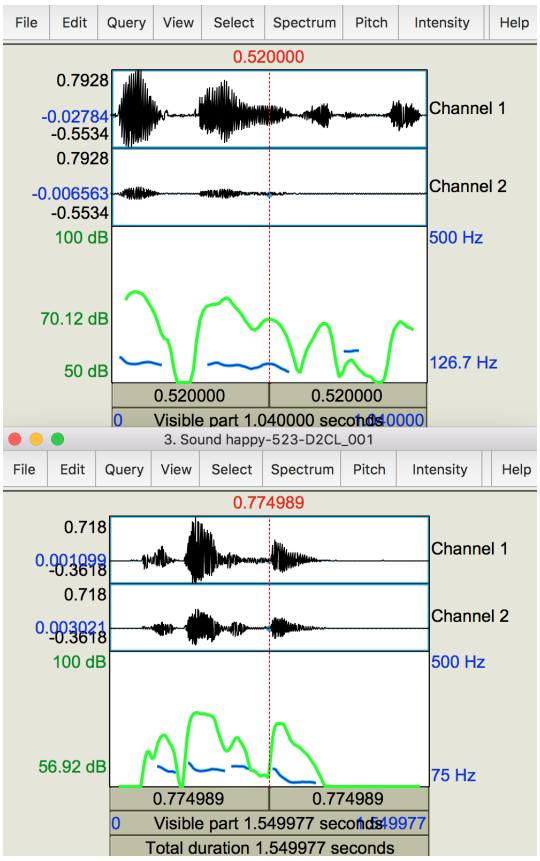


Fig. 1. Figure showing differences in values for pitch, intensity for Sadness (top) and Happiness (bottom)

Formants are resonances of vocal tract and estimation of their location and frequencies seemed important to us and has been seen to be used over the years. Another set of features that we were interested in were the MFCCs, which together make up an MFC. Mel Frequency Cepstrum (MFC) is a representation of linear cosine transform of a short-term log power spectrum of speech signal on a non-linear Mel scale of frequency. A lot of focus was given on Pitch and Intensity due to the importance given to these features by past research [6]. By doing our own feature analysis, we found that mean pitch, minimum intensity and the mean of MFCC 2 (Mel Frequency Cepstral Coefficients) are clear indicators of emotion tags on speech samples. We plotted graphs for the mean (represented by scatter dots) and standard deviation (represented by SD bars) for each emotion and analyzed the trends to see how each feature changes with different emotions. We noticed that the curves for mean pitch, minimum intensity and MFCC 2 follow a set trend. Therefore, we deduced that these features are important for accurate emotion detection.

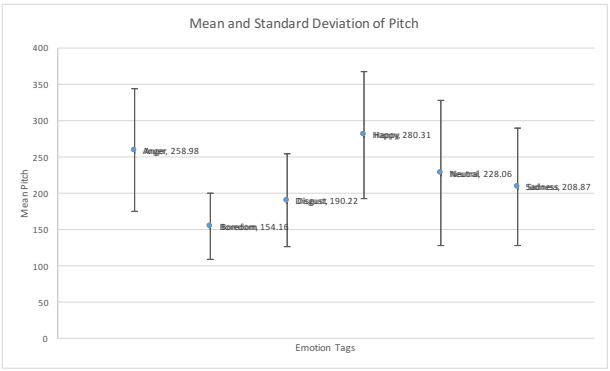


Fig. 2. Mean Pitch trends with the 6 emotion tags

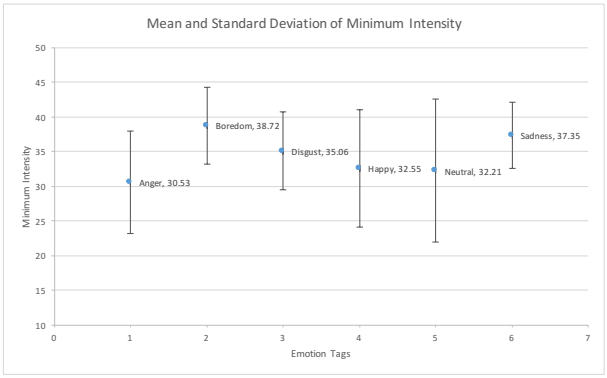


Fig. 3. Minimum Intensity trends with the 6 emotion tags

A PRAAT script was written and executed to extract out the above mentioned featured from the audio samples in the dataset. This feature set was treated to a little more cleaning which mostly included deletion of training samples where the features were not properly retrieved.

An important point to note is that the above features are **global features**, global statistics of the audio sample. These features would work well for certain learning algorithms (like Gradient Boosting Machines), but are not appropriate for using in Hidden Markov Models, as HMM's require low-level features that capture short-time behavior(see the section on Hidden Markov Models). These features are called **temporal**

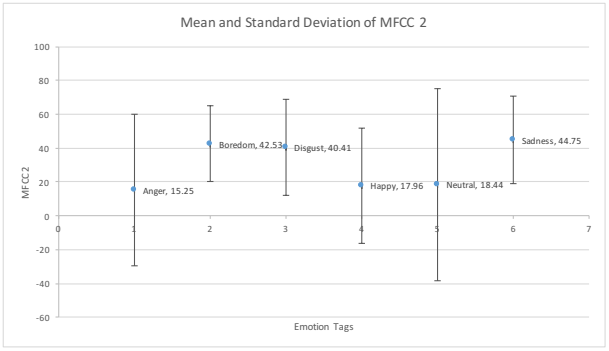


Fig. 4. MFCC 2 trends with the 6 emotion tags

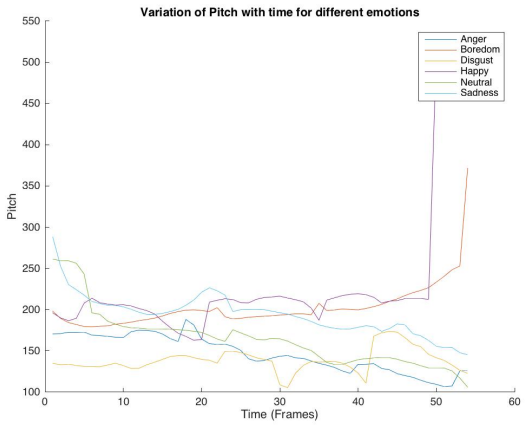


Fig. 5. Variation of Pitch with time for different emotions

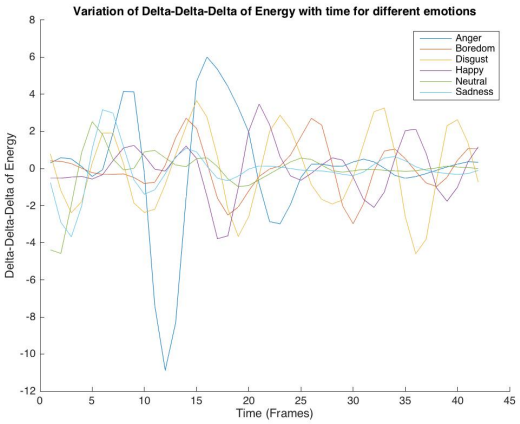


Fig. 6. Variation of Delta-Delta-Delta of Energy with time for different emotions

features. We extracted out the following temporal features using a PRAAT script:

- Pitch
- Delta of Pitch
- Delta-Delta of Pitch
- Energy
- Delta of Energy
- Delta-Delta of Energy
- Delta-Delta-Delta of Energy

The 'delta' features are approximate time derivatives (also known as differential coefficients) of the corresponding base features. For example, the delta of Pitch is the **differential coefficient** of pitch, the delta-delta of pitch is the **acceleration coefficient** of pitch, and so on.

The calculation [7] of delta- coefficients is given by the following equation:

$$d_t = \frac{\sum_{n=1}^{N} n(c_{t+n} - c_{t-n})}{2\sum_{n=1}^{N} n^2} \tag{1}$$

With temporal features, we encountered several frames in the audio samples where the speaker was not speaking, or **unvoiced frames**. These frames resulted in undefined values for pitch when extracting them out from PRAAT. To tackle this issue, we simply removed the unvoiced frames from our analysis and made a naive assumption that the voiced frames would be sufficient enough to derive a learning pattern. As a manual exercise to verify our assumption, we plotted these features with time (or frames) for each emotion.

Figures 4 and 5 show the graphs of Pitch and Delta-Delta-Delta of Energy with time for the six emotion tags for randomly picked audio samples from each emotion. To be more robust, and because of the high sensitivity to noise of the delta calculations, a low-pass filter is used on the signal first. There are slight variations in trends which are compelling enough to try a sophisticated learning algorithm like HMM's and potentially get meaningful results.

IV. Methods

We use HMMs to classify an audio sample into one of the six categories because our problem can be thought of as a sequence of observations over time where an input audio file is divided into time-frames. A hidden markov model assumes that the observation at any time, t is generated using some process, whose true state is hidden from us. It also assumes the Markov property that the hidden state (z), at any time t, is only dependent on the hidden state at the time $t-1$, where t in our case is a particular frame in the audio sample and also that the hidden states are discrete valued. Each hidden state can transition in the next frame to either itself or to another hidden state i.e. $p(z_n|z_n-1)$. The matrix that contains this probability of transition from one state to the next is called the transition matrix. Another component of HMMs is the emission matrix which gives information about the conditional distributions of the observed variables (x) from a specific state i.e $p(x_n|z_n)$. The joint distribution is given as follows:

$$p(x_1, x_2, ...x_N, z_1, z_2, ...z_N) = p(z_1)[\prod_{n=2}^{N} p(z_n|z_n-1)] \prod_{n=1}^{N} p(x_n|z_n) \tag{2}$$

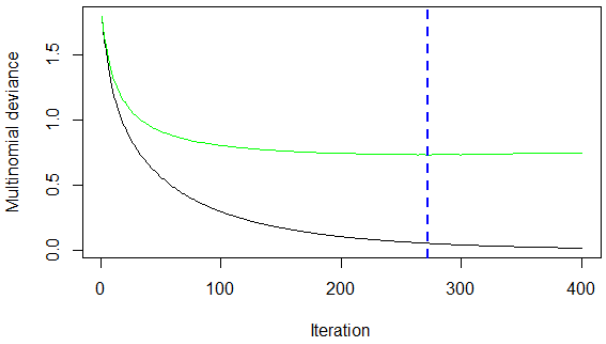
We used a continuous density and, more particularly, we approximated the emission probability by a mixture of Gaussians which is commonly done in speech recognition. Like

mentioned previously, we train one HMM for each emotion, first all the models are initialized using the global mean and variances on the training set for each feature (flat start), then the Baum-Welch Algorithm [10] is used to re-estimate the parameters ofthe model: the transition probabilities, and the parameters of the Gaussian mixtures (the weight of each mixture, the mean vector and the covariance matrix) for each state. At the time of prediction, we output the emotion corresponding to the maximum likelihood model for an audio sample from the test set, using the Viterbi Algorithm [9].

V. Results

A. Gradient Boosting Machine

For our first model based on the global features of the utterances, we used Gradient boosting with decision trees as it is an efficient algorithm that allows us to gain some insights about the importance of each feature via relative influence calculations. We trained the model using 400 trees with an interaction depth parameter of 10, then we selected the minimum number of trees at which no further improvement is observed using 5-fold cross-validation. The number of trees chosen is 272, see the following plot where the green curve represents the CV error and the black curve represents the training error:



The results obtained are pretty good, we get an overall accuracy of 83.44% (which is good for a 6 classes classification problem) on the test set and we get the following confusion matrix:

TABLE I
Confusion Matrix

Prediction / Truth	ang	bor	dis	hap	neu	sad
anger	38	0	3	5	2	2
boredom	0	17	1	0	0	0
disgust	2	3	65	5	1	3
happy	3	0	2	42	0	3
neutral	3	9	11	6	293	17
sadness	2	6	5	5	3	95

The confusion matrix analysis gives us helpful insights into the performance of GBM. The highest F1 score of **0.918** was achieved by the emotion **Neutral** and the lowest F1 score of **0.642** was for emotion **Boredom**. This suggests that the algorithm performed well overall with a high average F1 score. This also tells us that the difference in F1 scores for different emotions might be because of uneven data distribution and/or rough feature selection. For instance the **Neutral** emotion had the largest training samples and therefore the algorithm had more learning bandwidth for Neutral emotion prediction. We intend to work on these factors next.

TABLE II
Confusion Matrix Analysis

Emotion/Metric	Recall	Precision	F1 Score
Anger	0.79	0.76	0.78
Boredom	0.49	0.94	0.64
Disgust	0.75	0.82	0.78
Happy	0.67	0.84	0.74
Neutral	0.98	0.86	0.92
Sadness	0.79	0.82	0.81

GBM also allows us to gain some insight about the importance of the features for the emotion recognition task since the relative influence of each feature can be computed and used as a criterion for feature selection. The top features are (by order of importance):

- Min intensity
- Mean 2nd MFCC
- Mean 0th MFCC
- Standard deviation 9th MFCC
- Standard deviation 0th MFCC
- Mean Pitch
- Standard deviation Pitch

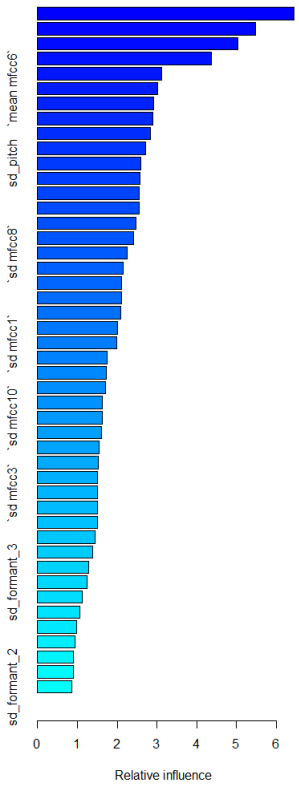
The plot below shows the relative influence of each feature. These results show that the Pitch, Intensity and spectral features (especially the MFCCs) are good predictors of the emotional state while some features like the formants and the fraction of voiced frames are not. This preliminary study will allow us to focus on certain sets of features when we will be working on HMMs, spectral, pitch and intensity features can be computed for each frame in the audio file and thus can be easily cast in a HMM framework.

B. HMM

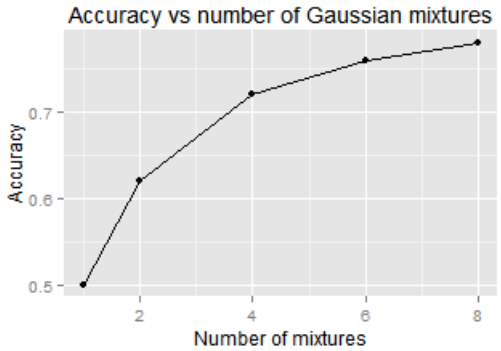
After training the GBM we observed that the MFCCs are one of the most iportant features, however as they tend to represent the phonetic content of the utterance (which is independent of the emotional state) it does seem enough to consider the MFCCs, and, as expected when we trained our first HMMs on the data using exclusively MFCCs and their tiem derivatives (delta and delta-delta) we couldn't get more than 55% accuracy). Thus we decided to take into account other features which seem relevant to us. Besides using the Pitch and energy related features, we also used the following:

- Linear predictive coefficients
- Jitter and shimmer (which carry information about voice quality and harmonicity)
- Zero-crossing-rate

Then we used a left-right HMM structure for each HMM, allowing jumps of at most 2 states. For the number of mixtures used, we started from only 1 mixture per state and then



we increased the number of mixtures by splitting and re-estimating the models. We got to 8 mixtures as we couldn't see a real improvement on our validation set after this, see below for a plot showing the improvement of the model with the number of mixtures used:



We used 64 states to capture the temporal complexity of the speech. We finally obtain an overall accuracy of 78%, see below the following confusion matrix:

VI. Conclusions and Future Work

In this project we could really get some understanding about how emotions are transmitted in speech, and we were able to classify them with good accuracy (about 80% which is pretty high considering that it is a 6 classes classification problem), which compares with results from the literature. This was a pretty ambitious project, as none of us had dealt with speech before, we had to use matlab to compute some of the derived features (like the deltas), we also had to deeply understand how HMMs work, how to train them, how to optimize the models to get a good accuracy. We also had to figure out how to use the Hidden Markov Model to emotion recognition as this is a very specific problem.

TABLE III
Confusion matrix

Prediction / Truth	ang	bor	dis	hap	neu	sad
anger	65.9 %	0 %	5.6 %	20.4 %	2 %	1.3 %
boredom	0 %	66.8 %	1 %	0 %	0 %	0 %
disgust	12.2 %	4.2 %	65 %	8.2 %	3 %	3.9 %
happy	4.9 %	0 %	9.3 %	55.1 %	1 %	1.3 %
neutral	14.6 %	16.7 %	7.4 %	8.2 %	87.8 %	6.7 %
sadness	2.4 %	12.3 %	13 %	8.2 %	6.7 %	86.8 %

References

[1] T. Pfister. Emotion Detection from Speech. *Gonville and Caius College*. Computer Science Tripos 2010

[2] D. Ververidis and C. Koutropoulos. Emotional speech recognition: Resources, features and methods. *Speech Communication 48*. 2006.

[3] A. Nogueiras, A. Moreno, A. Bonafonte and J,B. Marino. Speech Emotion Recognition using Hidden Markov Models. *Eurospeech 2001 - Scandinavia*

[4] B. Schuller, G. Rigoll, and M. Lang. Hidden Markov Model-Based Speech Emotion Recognition. *2003 IEEE International Conference on Acoustics, Speech and Signal Processing*, April 6-10, 2003, Hong Kong.

[5] T. L. Nwe, S.W. Foo and L. Silva. Speech Emotion Recognition using Hidden Markov Models *Speech Communication*, November 2003.

[6] R. Cowie, E. Douglas-Cowie, N. Tsapatsoulis, G. Votsis, S. Kollias, W. Fellenz and J.G. Taylor. Emotion Recognition in Human-Computer Interaction. *IEEE Signal Processing Magazine*. 2001. *IEEE*. 2000.

[7] Mel Frequency Cepstral Coefficient (MFCC) Tutorial. Practical Cryptography. *http://practicalcryptography.com/miscellaneous/machine-learning/guide-mel-frequency-cepstral-coefficients-mfccs/*.

[8] Liberman, Mark, et al. Emotional Prosody Speech and Transcripts LDC2002S28. DVD. Philadelphia: Linguistic Data Consortium, 2002.

[9] Hidden Markov Models and the Viterbi algorithm *http://www.cs.cornell.edu/ ginsparg/physics/INFO295/vit.pdf*

[10] Derivation of Baum-Welch Algorithm for Hidden Markov Models *http://www.cs.berkeley.edu/ stephentu/writeups/hmm-baum-welch-derivation.pdf*