A Hidden Markov Model Approach to Distinguishing Between Non-Prototypical Displays of Boredom and Interest

by

Justine Heritage

Submitted in partial fulfillment of Graduation Requirements
for the Computer Science Major
Dickinson College, 2013 – 2014

Professor John MacCormick, Supervisor

May 13, 2014

Abstract

A Hidden Markov Model Approach to Distinguishing Between Non-Prototypical Displays of Boredom and Interest

by
Justine Heritage

Affective computing is an emerging field of study in human-computer and human-robot interaction. As artificial intelligence continues to advance, there is a perceived need for developing methods to detect, recognize, process, and replicate human emotion in machines. Current emotion recognition and labeling mechanisms have achieved accurate performance on detecting "prototypical" human emotions such as happiness, sadness, and anger (Mower et al., 2009, p. 2). Furthermore, existing approaches generally utilize data containing acted emotional expressions (Zeng, 2008, p. 41). However computer recognition of naturalistic, nuanced emotion is still relatively undeveloped. This paper presents a Hidden Markov Model (HMM) approach to distinguishing between naturally occurring instances of two non-prototypical emotions: boredom and interest. We examine the vocal activation levels (VAD) and fundamental frequencies (F_0) of recorded speech using 3- and 5-state models, each under two different discretization methods. While the 5-state model using the first binning technique for F_0 performed best, accurately classifying 64.8% of the testing data, the average recognition rate for all models was 54.4%.

Chapter 1

Introduction

Affective computing is a quickly growing area of research in human-computer interaction. As computers become more central in daily human-human interaction, crafting systems that have “human-centered designs instead of computer-centered designs” (Zeng, 2009, p. 39) will allow computers to assume a more useful role in the lives of their users. For example, an application of this research is in the realm of educational technology. Teachers and students are increasingly utilizing online instruction and tutoring tools instead of the traditional face-to-face setup of a classroom environment. While these online tools are convenient and provide many of the same utilities found in a classroom, they are missing one crucial feature: the capacity to instantly adjust a lesson based on active student feedback. In a traditional classroom, a teacher has the ability to alter his or her presentation of material if students are disengaged or unchallenged by the lecture. Were computers able to recognize this emotional feedback, online tutoring systems could adjust to match the students’ needs and increase the effectiveness of the lesson.

This paper explores the question of how computers can learn to distinguish between two emotional states: boredom and interest. In a classroom setting such as the one described above, these emotions have very similar visual and audio expressions. The response to these emotions, however, should obviously be very different. Early attempts to solve this problem focused on identifying a small subset of basic emotional states (i.e. happiness, anger, sadness, disgust, fear and surprise) acted in a controlled laboratory setting. More recently, research has progressed to identifying more complex emotions that are non-performed (Zeng, 2009, p. 40-41). To fully understand the state of current research, we will begin by discussing several current issues in affective computing: psychological foundations, data considerations, and classification techniques. I will next outline the framework for my own research, present my findings, and discuss possibilities for the future of this research and other research in the field.

Chapter 2

Background

2.1. Psychological Foundations

While affective computing is a fairly recent area of interest in computer science, human psychologists have long been studying emotional expression. Psychology has formed the basis for much of the research done in the field so far, and thus is an important starting point for discussion. According to the American Psychological Association, emotion is "[a] complex pattern of changes, including physiological arousal, feelings, cognitive processes, and behavioral reactions, made in response to a situation perceived to be personally significant" (APA, 2013). Psychologists tend to agree that emotions are "coherent [clusters] of components" that can assume a finite number of emotional states, all with "distinct facial expressions" and different "underlying dimensions of vocal expression" (Silvia, 2006, p. 14, 16). It is this distinctness that allows humans to recognize emotions in others. Under these assumptions, it is feasible that computers could learn to detect and respond to human emotions with accuracy equal to or exceeding human recognition rates.

This paper takes particular interest in the emotional states of boredom and interest. Extensive research on the distinct physical cues for these emotions has been completed in psychology research. For example, facial expressions that indicate interest include "slightly lowered or raised eyebrows, raised lower eyelids, and parted lips and a dropped jaw" (Silvia, 2006, p. 17). Additionally, an interested subject might show decreased levels of blinking, gaze diversion, and head movement. While these visual cues are important for emotion, auditory cues are also important, and are the focus of this project. Since human "[r]ecognition of emotions from vocal features tends to be high" (Silvia, 2006, p. 19), it is important to also consider the auditory features that indicate interest and boredom in computational approaches to emotion detection. Psychologists characterize “bored speech as ‘generally slow and monotonous.’ […] The base level of pitch declines, speech rate declines, the voice’s pitch becomes less variable, and shows a smaller range of frequency. Interested speech, in contrast, shows a quicker rate of speech and a higher range of frequency” (Silvia, 2006, p.19). Thus, taking measurements of the pitch and energy of audio data will prove useful in attempting to distinguish between boredom and interest from audio recordings. While there are solid psychological foundations for labeling emotional data, researchers are currently struggling with several prominent issues concerning data collection and analysis.

2.2. Data Considerations

Gathering and processing data has proven to be a major challenge for affective computing research. Collecting a sufficient amount of real human data to train and test machine learning algorithms or other tools can be “time consuming, error prone, and expensive” (Zeng, 2009, p. 43). Considering the progression of research in the field so far, even when enough data can be collected it is difficult to induce the desired emotions in human subjects in a way that is useful for analysis. Additionally, recent research has brought to light the need for multi-modal data when classifying emotions.

The majority of existing methods for discriminating between different emotions use acted emotional data. For the purposes of current research, this data is inadequate for producing meaningful results. Typically, acted data is obtained by asking subject to mimic a particular emotion using the same facial expression and tone of voice as they would in a real-world display of that emotion. Emotion detection on acted data is generally quite good, but this is more a reflection of the exaggeration and control exhibited in acted emotions than the quality of the classification systems. Acted data “differs in visual appearance, audio profile, and timing from spontaneously occurring behavior” and “fail[s] to generalize to the subtlety and complexity” of naturalistic emotional expression (Zeng, 2009, p. 43). Additionally, acted data is typically obtained in a laboratory setting with constant lighting and body position. Spontaneous emotional data, on the other hand, is more prone to visual and auditory noise, making it difficult to process with the same precision as acted data. Researchers from the Massachusetts Institute of Technology conducted an experiment that attempted to distinguish between smiles that were born out of frustration and delight. They used five different classification techniques on two sets of audiovisual data. The first set contained 45 acted expressions and the second contained 72 spontaneous expressions. Despite the larger sample size in the spontaneous corpus, the average accuracy of the classifiers on the spontaneous data was only 41.67% compared to 82.34% for acted data (Hoque, McDuff, and Picard, 2012, p. 6). A related issue is person-dependent results. Since the number of subjects that can be included in a data set is fairly limited because of time and cost constraints, many systems for emotion detection have some dependence on a particular subject or set of subjects and do not translate to individuals that were not included in the original study (Zeng, 2009, p. 42). Additional research is needed to improve the accuracy of classifiers on spontaneous data in a non-person-dependent context for emotion detection systems to be integrated into real-world applications.

The second major concern in collecting data revolves around modality. Much of the early research in affective computing and emotion detection considered only unimodal data, evaluating either the facial expression of the subjects or their vocal utterances. More recently, researchers have been shifting their attention to audiovisual data and attempting to determine the temporal relationships between visual and auditory emotional cues. Multi-modal approaches to emotion classification are still topics of research themselves. For example, Busso et al. from the University of Southern California attempted a multi-modal analysis of emotion recognition using both facial expressions and speech. Their system fused the modalities at the decision-level, meaning they modeled the input data for each medium independently and the combined the results to make a final decision about the emotional label for that data. Their experiments showed a marked improvement in recognition over examining video and speech completely independently (Busso et al., 2004, p. 205). However other researchers assert that "the assumption of conditional independence between audio and visual streams in decision-level fusion is incorrect and results in the loss of information of mutual correlation between the two modalities" (Zeng, 2009, p.50). To account for the temporal relationship between audio and visual emotional data, researchers such as Sebe et al. have implemented full fusion systems using Bayesian networks and Hidden Markov Models (2006, p. 1138). While there is neither a definitive psychological or computational confirmation of the benefits of analyzing multimodal data over unimodal data, most emerging research focuses efforts on the former.

2.3. Classification Techniques

Researchers have applied many machine learning and modeling techniques to the problem of computational emotion detection. In their study attempting to distinguish between frustrated and delighted smiles, researchers utilized five different classifiers including Support Vector Machine (SVM), Hidden Markov Model (HMM), and Hidden-state Conditional Random Field (HCRF) approaches. This particular study "automated the process of extracting temporal facial features." The time-automation resulted in the SVM “[outperforming] HMM and HCRF” for their dataset (Hoque, McDuff, and Picard, 2012, p. 5). However, “[i]n naturalistic behavior, the affective states of a person change at a rate much slower than the typical rate at which video is recorded […]. Hence, there is a high probability that consecutive recorded instants of expression represent the same affective content” (Meng and Berthouze, 2011, p. 378). In other words, while automating the time aspect may produce good results on acted data (as was the case in the frustrated vs. delighted smiles study), in spontaneous emotional expressions it becomes more necessary to consider temporality. Recent studies in affective computing have shown that HMMs are valuable tools for solving many problems in affective computing. The subsequent sections provide an overview of how HMMs work in general as well as their specific applications in emotion detection.

2.4. General Overview of Hidden Markov Models[1]

Consider the following scenario: you are locked in a windowless office building for an indefinite period of time, but you want to know whether it is hot or cold outside. As you cannot directly observe the weather, you must rely on other observations to determine the outside temperature. If you notice that the co-worker who sits next to you is wearing a heavy jacket when she comes in to work, you might assume that it is cold out. Similarly, if your co-worker comes in without a jacket, you may decide that the weather is warm. You may also be able to assume that if your observations from the previous day indicated that it was hot outside, there is a certain likelihood that the current day is either hot or cold. Over time, you could make adjustments to your assumptions about the outside weather to the point where you could fairly accurately predict the temperature based on your co-worker’s attire without ever leaving your building.

[1] The details in this section (except the sample, which is self-formulated) were taken from Stamp (2012) and Ramage (2007).

Problems such as these can be solved using Hidden Markov Models (HMMs). HMMs are statistical models that, given some set of partial observations over a time period, can determine the likelihood of being in a particular state at a particular time. HMMs are based on the idea that the probability of being in a given state at time t depends only on the state at time $t-1$ (i.e. a first-order Markov process). Therefore, if we have some set of initial probabilities for each possible state of our data, as well as the probabilities of observing a particular feature given these states, we can train a model to make inferences about state from new observations.

In a HMM, the initial state probabilities are represented by a N-vector π where N is the number of distinct states in the sequence. For our example above, we could have:

$$\pi = \begin{bmatrix} 0.6 & 0.4 \end{bmatrix}$$

This vector indicates that there is a 60% chance that we will start our observations on a hot day, and a 40% chance we will start them on a cold day.

Another important element of a HMM is the transition probability matrix A, where A is a $N \times N$ matrix and N is the number of distinct states in the model. Let us again consider the weather example. The following matrix is a transitional matrix for hot days (H) and cold days (C):

$$A = \begin{array}{c} H \\ C \end{array} \begin{bmatrix} 0.7 & 0.3 \\ 0.4 & 0.6 \end{bmatrix}$$

Here, the entry $A_{0,0} = 0.7$ represents that the probability that if the state at time t is hot, there is a 70% chance that the state at time $t+1$ will be hot as well. Similarly, the entry $A_{1,0} = 0.4$ indicates that if yesterday was cold, there is a 40% chance that today will be hot.

The second element of a HMM is the observation matrix B, an $N \times M$ matrix where N is the number of distinct states and M is the number of different observations that can be made about that state. Continuing our example, let B_{ij} represent the probability of your co-worker wearing no jacket (NJ), a light jacket (LJ), or a heavy jacket (HJ) given the outside temperature:

$$B = \begin{array}{c} \\ H \\ C \end{array} \begin{array}{c} \begin{array}{ccc} NJ & LJ & HJ \end{array} \\ \begin{bmatrix} 0.7 & 0.2 & 0.1 \\ 0.1 & 0.4 & 0.5 \end{bmatrix} \end{array}$$

Each entry in matrix B represents the probability of observing some value in a particular state. In this example $B_{0,0} = 0.7$ indicates that the probability of observing no jacket on a hot day is 70%, while $B_{1,1} = 0.4$ means there is a 40% chance that it is cold outside if your co-worker is wearing a light jacket.

From our example, suppose we observed our co-worker's attire for four days and recorded the following observations $O = (1, 0, 2, 1)$ where $NJ = 0$, $LJ = 1$, and $HJ = 2$. Given our HMM $\lambda = (A, B, \pi)$, we can adjust the values of each matrix using the forward-backward algorithm (alpha/beta-pass), briefly outlined in Algorithm 2.1.

Algorithm 2.1: The forward-backward algorithm[2]

1. α-pass: the probability of the observation sequence up to time t. Let $\alpha_{0,i}$ be the probability of seeing the value of O_0 given the initial state distribution π_i and observation matrix B.

```
$for\ t \leftarrow 0, 1, \dots, T-1$ {
    $for\ i \leftarrow 0, 1, \dots, N-1$ {
        $for\ j \leftarrow 0, 1, \dots, N-1$ {
            $\alpha_{t,i} = \alpha_{t,i} + \alpha_{t-1,i} A_{j,i}$
        }
        $\alpha_{t,i} = \alpha_{t,i} B_{i,O_t}$
```

[2] Algorithm adapted from Stamp (2012)

```
        }
    }
```

2. β-pass: the probability of observing all future states after the state at time t. Let $\beta_{t-1,i} = 1$.

```
    $for\ t \leftarrow 0, 1, \ldots, T-2 \{$
        $for\ i \leftarrow 0, 1, \ldots, N-1 \{$
            $for\ j \leftarrow 0, 1, \ldots, N-1 \{$
                $\beta_{t,i} = \beta_{t,i} + (A_{i,j} B_{j,\mathcal{O}_{t+1}} \beta_{t+1,j})$
            }
        }
    }
```

After the forward-backward algorithm completes, the probabilities in each matrix are adjusted using a learning algorithm. The learning algorithm can vary, but it essentially recomputes the probabilities for each entry in A and B in the following manner:

- A: the ratio of the expected number of transitions between $state_i$ and $state_j$ to the expected number of transitions between $state_i$ and any other state
- B: the ratio of the expected number of times the model is in $state_i$ with some $observation_k$ to the total number of times the model is expected to be in $state_i$

The final steps of the model involve comparing the negative log of the probability of seeing the observations given the current model to that of the previous iteration of the algorithm (with log $(P(\mathcal{O}|\lambda)) = -\infty$ for the initial iteration). If the new log probability is greater than the old log probability and the algorithm has not exceeded its maximum number of iterations, we recurse through the algorithm starting with the α-pass to further refine the probabilities in A and B. Otherwise, the model is returned with its final probability distributions for the given observation sequence. For the above example, running the model for five iterations results in the following matrices:

$$\pi = [0.8943 \quad 0.1057]$$

$$A = \begin{bmatrix} 0.5813 & 0.4187 \\ 0.5031 & 0.4969 \end{bmatrix}$$

$$B = \begin{bmatrix} 0.4090 & 0.4764 & 0.1146 \\ 0.2068 & 0.0942 & 0.6990 \end{bmatrix}$$

Using our given observation sequence $\mathcal{O} = (1, 0, 2, 1)$, the adjusted model tells us that we have a roughly 89% chance of beginning our observations on a hot day, slightly larger chances of having two consecutive hot or cold days than of transitioning from a hot or cold day to the other, and that it is 69% likely to be a cold day if we observe a heavy jacket.

2.5. Application of HMMs to Emotion Detection

Hidden Markov Models are ideal for making inferences about emotional states because emotions change over time and are in part determined by making observations about particular features such as a person's vocal signals and facial expressions. For the purposes of this project, the possible states a subject could be in at a given time are interested and bored ($N = 2$). The next step in utilizing a HMM is determining what observable features to use in the model. Due to time constraints, this project focuses only on audio features.

As HMMs require discrete observations, any audio features must be binned into categories. For example, a useful feature in characterizing audio data is fundamental frequency. As mentioned in Section 2.1, bored speech might be characterized by declining frequency over the course of the audio. To consider frequency, we first have to segment the audio data into discrete time segments. Each segment could be characterized of having low, medium, or high average pitch. These pitch assignments would form our observation sequence that we would input to the model along with the matrices A, B, and π. Since there is no way to determine the initial probabilities of these matrices, it is necessary to randomize the value of each entry in the three matrices as follows:

$\pi_i \approx 1/N$

$A_{i,j} \approx 1/N$

$B_{j,k} \approx 1/M$

It is important to note that none of the entries should be set to the exact value of the given ratio because "exactly uniform values will result in a local maximum from which the model cannot climb" (Stamp, 2012, p. 9). Given a good learning algorithm with randomized input data, the model should be able to adjust the values of the matrices to output the maximum likelihood probabilities of the state sequence extracted from the audio data.

Figure 2.5 shows the flow of data through our emotion detection system (adapted from Liu and Wang (2011)). From the figure, we can see that HMMs follow a typical machine-learning pattern. After obtaining data and extracting the necessary features, a training set is subdivided into features and inputted to the model so it can learn (in this case, by adjusting the probability matrices). After training the models, we can input sample data directly into the decision-making portion of the system and return a label for the emotion.

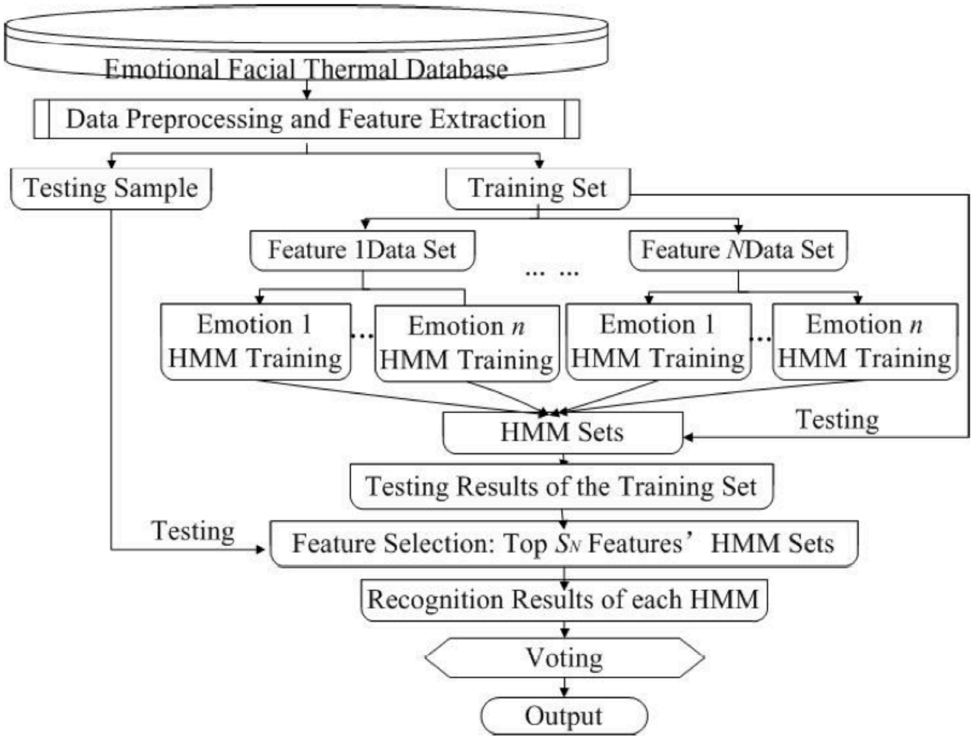

Figure 2.1: The framework for our emotion detection system adapted from Liu and Wang (2011)

Chapter 3

Methods

3.1. Data Collection, Feature Extraction, and Discretization

As mentioned in Section 2.2, gathering appropriate data for affective computing research is challenging. The Carnegie Mellon Kids Corpus provides a large database of audio recordings of children ages 6-9 reading short sentences about topics they might normally learn in school. While the data from this set is unimodal, it does include the audience that applications of this research would target. A total of 87 recordings were compiled from the database. Most of the data (50 recordings) were used to train the HMM while the rest was used for testing. Each data point was manually labeled as either bored or interested to validate the results of the model. The Voicebox Toolkit for MATLAB was used to extract two salient audio features: vocal activation levels (VAD) and fundamental frequency (F_0).

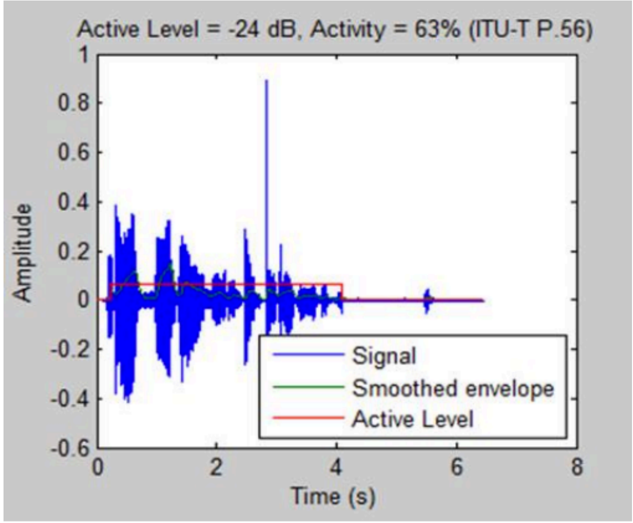

Figure 3.1: The output of `activlev()` plotted over the speech signal. The red line represents the activation level output.

Vocal activation level refers to the amount of time a subject spent actively speaking over the duration of the recording. This data was obtained using the `activlev()` function, which returns an *n*-vector of zeros and ones where *n* is the length of the recording in terms of the sampling frequency. A value of one in the vector indicates that the subject was actively speaking at that time, and a zero means the subject was not speaking. Figure 3.1 shows a typical output vector from running `activlev()` on one of the recordings plotted over time. For our purposes, we interpret high VAD levels as a sign of interest and lower VAD levels as a sign of boredom.

After the VAD was extracted from each recording, it was sent to a Java program `ActivLevClassifier.java` for discretization. The first step was to divide the VAD vector into 100-element segments to allow the activation measurements to be meaningful in terms of real time. The average value of each of these segments was taken and placed into a new array to be assigned to different observations. Table 3.1 shows the different approaches to discretizing the VAD data. We ran experiments on 3-observation (Low, Medium, and High activation levels) and 5-observation systems (Very Low, Low, Medium, High, and Very High activation levels), each with two different methods of binning (Bin1 and Bin2). The VAD data followed a roughly uniform distribution, so the bins were selected over uniform intervals. The classifications of each element were stored in an array for input into the HMM.

VAD	**Bin1**	**Bin2**
3-observation	Low ($avg_{VAD} \leq 0.90$) Medium ($0.90 < avg_{VAD} \leq 0.95$) High ($0.95 \leq avg_{VAD}$)	Low ($avg_{VAD} \leq 0.95$) Medium ($0.95 < avg_{VAD} \leq 0.97$) High ($0.97 \leq avg_{VAD}$)

5-observation	Very Low ($avg_{VAD} \leq 0.90$) Low ($0.90 < avg_{VAD} \leq 0.93$) Medium ($0.93 < avg_{VAD} \leq 0.96$) High ($0.96 < avg_{VAD} \leq 0.99$) Very High ($0.99 < avg_{VAD}$)	Very Low ($avg_{VAD} \leq 0.96$) Low ($0.96 < avg_{VAD} \leq 0.97$) Medium ($0.97 < avg_{VAD} \leq 0.98$) High ($0.98 < avg_{VAD} \leq 0.99$) Very High ($0.99 < avg_{VAD}$)

Table 3.1: Four methods by which VAD data was classified for input into the HMM

Fundamental frequency refers to the lowest frequency of a wave and has been shown in past research to be a good indicator of emotional state. Since we want to examine the magnitude of the fundamental frequencies, it was necessary to convert the data from being a function of time to a function of frequency. This was accomplished using the Fast Fourier Transform function `fft()` (typical output shown in Figure 3.2). For this feature, the division of data into 100-element segments happened before moving to classification so that we could calculate the fundamental frequency at each time step rather than over the course of the whole recording.

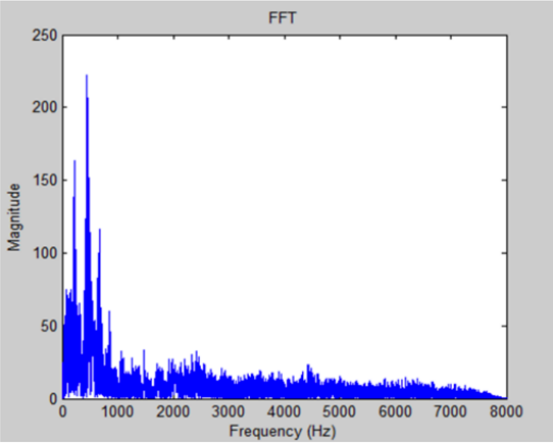

Figure 3.2: The output of `fft()` when run on a typical recording.

The segmented F_0 data was then inputted to a Java class, `FFTClassifier.java`. This classifier works similarly to the VAD classifier, with the addition of data about the subject's sex into the classification process. Since the average fundamental frequencies for male and female children differ, in order to accurately bin the F_0 measurements we had to consider sex before the classification process. The average fundamental frequency for male children age 6-9 is around 254 Hz, while the average for female children in the same age range is around 266 Hz (Nicollas et al, 2008, p. 672). Table 3.2 presents the 3- and 5-observation binning methods (Bin1 and Bin2) used for discretizing F_0 data. The first number in the bin definitions is for male subjects and the second is for females. Selecting the bins for F_0 was slightly more challenging as fundamental frequency data does not follow a normal distribution, but a positive skew. Again, the final classifications of the data into these discrete observations was stored in an array for input into the HMM.

F_0	Bin1	Bin2
3-observation	Low ($F_0 \leq 252/255$) Medium ($252/255 < F_0 \leq 260/283$) High ($260/283 \leq F_0$)	Low ($F_0 \leq 252/255$) Medium ($252/255 < F_0 \leq 260/283$) High ($260/283 \leq F_0$)
5-observation	Very Low ($F_0 \leq 233/246$) Low ($233/246 < F_0 \leq 245/252$) Medium ($245/252 < F_0 \leq 257/257$) High ($257/257 < F_0 \leq 269/263$) Very High ($269/263 < F_0$)	Very Low ($F_0 \leq 233/246$) Low ($233/246 < F_0 \leq 245/252$) Medium ($245/252 < F_0 \leq 257/257$) High ($257/257 < F_0 \leq 269/263$) Very High ($269/263 < F_0$)

3.2. Implementation of the Hidden Markov Model

The HMM outlined in Sections 2.4, 2.5 was completed using an expectation-maximization algorithm for learning and model re-estimation. The algorithms were implemented in Java and tested on sample data from Stamp (2012). After obtaining the arrays of observations, we began training using 50 of the audio recordings, half of which were manually labeled as bored and half of which were labeled as interested. Each feature had its own HMM.

The VAD model, HMM_{VAD}, was given as input the observation arrays from the `ActivlevClassifier`, $\pi = [0.49 \quad 0.51]$ (indicating there is a roughly 50% chance that the subject will start in either state), $A = \begin{bmatrix} 0.60 & 0.40 \\ 0.40 & 0.60 \end{bmatrix}$ (indicating that a subject is slightly more likely to remain in the same emotional state at time *t* as they were at time *t* – *1*, and the observation matrix B. For the 3-observation experiments (Low, Medium, High), we initialize B as $B = \begin{bmatrix} 0.20 & 0.30 & 0.50 \\ 0.50 & 0.30 & 0.20 \end{bmatrix}$. For the 5-observation experiments (Very Low, Low, Medium, High, Very High), we initialize B as $B = \begin{bmatrix} 0.09 & 0.11 & 0.20 & 0.29 & 0.31 \\ 0.31 & 0.29 & 0.20 & 0.11 & 0.09 \end{bmatrix}$. Both of these matrices indicate that there is a higher chance the subject is interested if we observe higher activation levels, and a higher chance that the subject is bored if we observe lower activation levels. Note that based on the psychological foundations of this project, making such assumptions about the initial probabilities is valid. The model for F_0, HMM_{F0}, was also given the observation array from `FTTClassifier` as input, as well as the same π, A, and B (since observing high frequency also indicates the subject is more likely interested).

Each HMM ran for all 50 training inputs, updating the observation and transition matrices. We then moved to the testing phase, which examines the observations extracted from the remaining 37 recordings. Due to time constraints, a pre-implemented Viterbi algorithm was adapted for use in this model and given the observation sequence and adjusted π, A, and B as input[3]. The Viterbi algorithm is often used for decision-making on the output

[3] Algorithm obtained from the website of Dr. Paul Fodor, Stony Brook University.

Hidden Markov Models as it finds the most likely path of hidden states given an observation sequence. The final state output in the sequence – either bored or interested - for each recording was considered to be the emotional label.

Chapter 4

Results and Discussion

4.1. Results

Table 4.1 summarizes the results of running the training and testing on the 3- and 5-observation methods with both Bin1 and Bin2. Note that a label is considered accurate if it matched the manual label given to the recording at the start of the experiment. The average percentage of accurate recognitions when considering activation levels was 50%, while the average accuracy when considering F_0 was slightly higher at 58.8%. The best performing approach was the 5-observation Bin1 method for F_0, which accurately labeled 64.8% (24 of 37) of the recordings. For VAD, the 3-observation approach yielded a 54.1% average accuracy rate and the 5-observation approach resulted in only a 46.0% accuracy rate. For F_0, the average accuracy under 3-observations was 58.15 and 59.5% for 5-observations.

VAD	**Accuracy (%)**
3-state, Bin1	51.4
3-state, Bin2	56.7
5-state, Bin1	40.5
5-state, Bin2	51.4

F_0	**Accuracy (%)**
3-state, Bin1	56.8
3-state, Bin2	59.5
5-state, Bin1	64.8
5-state, Bin2	54.1

Table 4.1: The labeling accuracy rates on the testing data.

4.2. Discussion and Future Work

While some of the results presented in the previous section are promising, no method performed significantly better or worse than a 50% accuracy rating (which would be analogous to assigning each recording an emotional label at random). However we can obtain some valuable information from these results. Overall, F_0 produced a larger number of accurate labels, which is unsurprising given the results of past research. On average, the vocal activation level only correctly labeled half of the recordings, showing that it is not particularly useful to consider VAD independently of other audio features.

It is suspected that because vocal activation is essentially a binary measurement, introducing more observation levels did not lead to an improvement in accurate recognition rates. For F_0, an essentially continuous measurement, the number of accurate labels increased slightly when more observations levels were included because it provided the model with more information. After completing the experiments, it became clear that without a more in-depth statistical analysis, the use of different binning thresholds would not be particularly helpful for the purpose of comparison across different features and different numbers of observations. Because the amount of data available for this project was so small, it is difficult to take advantage of the statistical distributions of vocal activation and fundamental frequency that would allow for meaningful comparisons of binning methods. Were more data used, a more robust analysis could be done to determine why some binning thresholds were more useful than others.

Were expansions to be made on this project in the future, more focus should be placed on fusing the information about activation levels and frequency so decisions can be made about a subject's emotional state considering both features together rather than independently. Additionally, a more robust feature set would be helpful in increasing the amount of information known about a recording. Another major block in this project was the availability of data. The data from the CMU Kids Corpus were of young children reading, so some of the information about the emotional state of the subject could have been misleading as children do not read as fluidly as adult subjects. Ideally, it would be beneficial to collect a large amount of data specifically for this project, but for the scope of this project the CMU database was sufficient. Finally, one of the biggest improvements that could be made to this project would be to implement a more sophisticated learning algorithm in the HMM. The expectation-maximization algorithm is the naïve approach, so it is possible that better results could be obtained under a different training method.

References

American Psychological Associatoin (2013). Glossary of Psychological Terms. Retrieved Nov 11, 2013. http://www.apa.org/research/action/glossary.aspx#e.

Busso, C. et al. (2004). Analysis of Emotion Recognition Using Facial Expressions, Speech and Multimodal Information, *6th ACM International Conference on Multimodal Interfaces*, 205-211.

Fodor, Paul D. (2007). "Viterbi.java." Stony Brook University. Retrieved April 5, 2014. http://www.cs.stonybrook.edu/~pfodor/old_page/viterbi/Viterbi.java.

Hoque M., McDuff, D., and Picard, R. (2012). Exploring Temporal Patterns in Classifying Frustrated and Delighted Smiles. *IEEE Transactions on Affective Computing*, 3(3), 1-13.

Meng, H. and Bianchi-Berthouze, N. (2011). Naturalistic Affective Expression Classification by a Multi-stage Approach Based on Hidden Markov Models, *Affective Computing and Intelligent Interaction*, 6975, 378-387.

Mower, E. et al. (2009). Interpreting Ambiguous Emotional Expressions. 3rd International Conference on Affective Computing and Intelligent Interaction and Workshops. Amsterdam, NE: IEEE.

Ramage, D. (2007). *Hidden Markov Models Fundamentals* [PDF document]. Retrieved from http://cs229.stanford.edu/section/cs229-hmm.pdf.

Sebe, N., Cohen, I., Grevers, T., and Huang, T. (2006). Emotion Recognition Based on Joint Visual and Audio Cues. *Pattern Recognition*, 1, 1136-139.

Silvia, P. (2006). *Exploring the Psychology of Interest*. New York, NY: Oxford University Press.

Stamp, M. (2012). *A Revealing Introduction to Hidden Markov Models* [PDF document]. Retrieved from http://www.cs.sjsu.edu/~stamp/RUA/HMM.pdf.

Zeng, Z. (2009). A Survey of Affect Recognition Methods: Audio, Visual, and Spontaneous Expressions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(1), 39-58.