

# HMM-GMM Acoustic Models for Speech Recognition

Term project for EECS 281A

Arlo Faria

SID: 14944274

arlo@cs.berkeley.edu

## Abstract

This project explores statistical models and methods employed for automatic speech recognition. A graphical model describes the speech process with phonetic-acoustic HMM-GMM units; in particular we present several ways to model continuous acoustic observations with Gaussian approximations. We develop a simplified experiment using low-dimensional features, which provide convenient conceptualization and analysis. In addition to discussion of state-of-the-art implementations, this work investigates two performance measures based on alignment accuracy and entropy.

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Experimental Methodology</b>	<b>2</b>
2.1	The TIMIT Corpus . . . . .	2
2.2	Evaluation metrics . . . . .	2
2.2.1	Average entropy . . . . .	2
2.2.2	Alignment accuracy . . . . .	3
<b>3</b>	<b>Acoustic processing</b>	<b>4</b>
3.1	Formant features . . . . .	4
3.2	Standard speech feature extraction . . . . .	5
<b>4</b>	<b>Modeling observations</b>	<b>5</b>
4.1	Multivariate Gaussian models . . . . .	6
4.1.1	Maximum likelihood parameter estimation . . . . .	6
4.1.2	Assumption of diagonal covariance . . . . .	7
4.1.3	Mixture models . . . . .	7
4.2	Experiment . . . . .	8
4.2.1	Analysis . . . . .	10
4.2.2	Comparison to existing results . . . . .	11
<b>5</b>	<b>Temporal modeling</b>	<b>13</b>
<b>6</b>	<b>Conclusion</b>	<b>13</b>

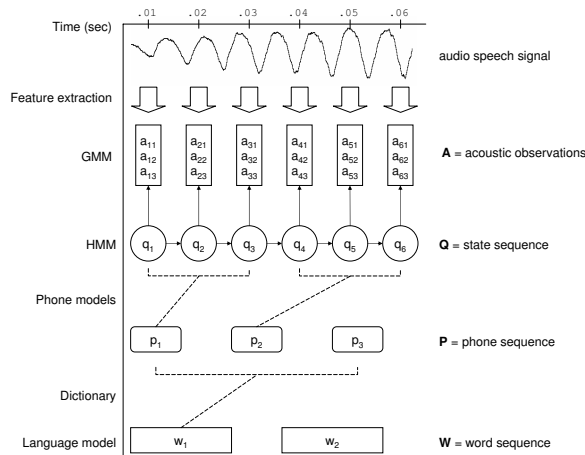


Figure 1: Speech modeling in a typical automatic speech recognition system.

## 1 Introduction

The speech recognition problem has a well-known noisy-channel formulation in which a speaker’s intended words  $\mathbf{W}$  are encoded into an acoustic speech signal  $\mathbf{A}$ . The task of the recognizer is to model this process and retrieve the hypothesis word string  $\hat{\mathbf{W}}$  that is most probable given the acoustic evidence:

$$\hat{\mathbf{W}} = \arg \max_{\mathbf{W}} P(\mathbf{W}|\mathbf{A}) \quad (1)$$

We can frame this problem differently by applying Bayes’ formula:

$$\hat{\mathbf{W}} = \arg \max_{\mathbf{W}} \frac{P(\mathbf{A}|\mathbf{W})P(\mathbf{W})}{P(\mathbf{A})} \quad (2)$$

When the observation  $\mathbf{A}$  is fixed, we seek the maximum *a posteriori* estimate:

$$\hat{\mathbf{W}} = \arg \max_{\mathbf{W}} P(\mathbf{A}|\mathbf{W})P(\mathbf{W}) \quad (3)$$

The likelihood  $P(\mathbf{A}|\mathbf{W})$  can be computed with an acoustic model, while the prior  $P(\mathbf{W})$  is given by a language model. We will investigate the acoustic models that underly many speech recognition architectures: Hidden Markov Models (HMM) with outputs parameterized by Gaussian Mixture Models (GMM). The language model, used to describe word sequences, can take various forms: for example, a task grammar could be prescribed for dialog systems, while stochastic n-gram models would be more suitable for large-vocabulary transcription. Although they are significant components of speech recognition systems, this paper will not provide a treatment of language modeling techniques and search strategies for word decoding.

The acoustic model relates words to their realization as speech. Typically, a pronunciation dictionary is used to map a sequence of words  $\mathbf{W}$  to a sequence of phonemes  $\mathbf{P}$ . These correspond to elementary phone units which define a state transition table for the hidden state sequence  $\mathbf{Q}$ . Each state generates a frame of the observation sequence  $\mathbf{A}$ , whose features correspond to acoustic properties of the speech signal. This hierarchical structure is depicted in Figure 1.

We present experiments to compare a variety of acoustic models; the task and evaluation criteria are given in Section 2. Section 3 describes front-end processing for deriving speech features. Ways to model acoustic observations are discussed in Section 4, and temporal models follow in Section 5.

We will develop an example using a simple set of features and models, discussing results of experiments and relating them to more substantial systems. Due to the difficulty of implementation and time constraints, we apologize that some results are not available at present. (Currently, the HMM model is problematic...)

## 2 Experimental Methodology

The experiments in this paper are unorthodox because we do not build a full-scale system to evaluate word-level recognition performance. Rather, we exploit a specially-labeled corpus to build a variety of small models, and present an alternative to the standard evaluation based on word error rate.

### 2.1 The TIMIT Corpus

The data used in these experiments are drawn from the TIMIT corpus of read speech [1]. We selected 1750 unique sentences<sup>1</sup> from among 440 speakers in the training set and 35 speakers in the test set. The corpus contains relatively good quality audio recordings, sampled at 16 kHz in quiet studio conditions.

Although pronunciation dictionaries can provide a distribution over multiple pronunciations for a given word, the TIMIT dictionary is a one-to-one mapping of words to phonemic pronunciations<sup>2</sup>. That is,  $P(\mathbf{P}|\mathbf{W}) = 1$  for some unique phone sequence. This deterministic pronunciation model enables us to effectively narrow the scope of our acoustic models to the process relating  $\mathbf{P}$  and  $\mathbf{A}$ . Note that even under this constraint, word decoding is still difficult because the inverse mapping  $\mathbf{P} \rightarrow \mathbf{W}$  is not one-to-one.

This corpus was also chosen because it is accompanied by manually aligned phonemic transcriptions. For each utterance in the corpus, the reference word string  $\bar{\mathbf{W}}$  was mapped to the reference phonemic transcription  $\bar{\mathbf{P}}$ . A trained linguist then aligned this transcription to corresponding times in the speech signal. Note that this labeling procedure is crucially different from asking a linguist to create a phonetic transcription of the speech signal, unconstrained by a fixed phonemic transcription<sup>3</sup>. We denote the reference alignment as  $(\bar{p}_1, \dots, \bar{p}_t, \dots, \bar{p}_N)$  where each frame  $t$  of the speech signal has  $\bar{p}_t$  as its phone label.

### 2.2 Evaluation metrics

Research in speech recognition is predominantly guided by a single measure of performance: word error rate, the minimal edit distance between hypothesized and reference word strings, in terms of weighted deletions, insertions, and substitutions. While the measure is a clear indicator of overall system performance, it is in some ways problematic. Experimentation is expensive because full recognition is a complicated procedure, and it can be difficult to interpret the effects caused by modifying components that are not directly connected to the ultimate output of the system. This has led to an incremental development strategy that has been criticized [2], and the proposal of alternative tasks [3].

We hereby introduce two new measures, hopefully meaningful indicators of the quality of an acoustic modeling approach. It would be desirable to analyze how well an acoustic model matches an acoustic observation sequence  $\mathbf{A}$  (of length  $N$ ) to its given phonemic transcription  $\mathbf{P}$  (of length less than  $N$ ).

#### 2.2.1 Average entropy

We might try to utilize the duration-averaged log likelihood of an acoustic observation sequence:

$$\text{ALL} = \frac{1}{N} \log P(\mathbf{A}|\mathbf{P}) = \frac{1}{N} \log \sum_{\mathbf{Q}} P(\mathbf{A}|\mathbf{Q})P(\mathbf{Q}|\mathbf{P}) \quad (4)$$

Although we will see that these quantities are computable, they are not good for comparing models with different types of features because the magnitude of the acoustic likelihood depends very much on the size of the feature space. This measure might still be useful when the features are fixed, though.

---

<sup>1</sup>To make the corpus more “natural”, we remove repeated sentences. Our data include all phonetically-diverse sentences (SI), and one selection from each of the phonetically-compact sentences (SX). The dialect sentences (SA) were omitted.

<sup>2</sup>Actually, some words have multiple pronunciations that depend on semantics (eg. past and present tenses of *read*). In these cases, the correct disambiguation was manually provided.

<sup>3</sup>There is a subtle distinction between *phonetic* vs. *phonemic*: a *phone* can be any specified speech sound, whereas *phonemes* refer to the distinguishable speech units of a language. We apologize if this is confusing, pedantic, or worse – misused.

One might also attempt to examine the posterior probability of a phone sequence:

$$P(\mathbf{P}|\mathbf{A}) = \frac{P(\mathbf{A}|\mathbf{P})P(\mathbf{P})}{P(\mathbf{A})} = \frac{P(\mathbf{A}|\mathbf{P}) \sum_{\mathbf{W}} P(\mathbf{P}|\mathbf{W})}{\sum_{\mathbf{W}} P(\mathbf{A}|\mathbf{W})} \quad (5)$$

Unfortunately, the terms for unconditional probabilities involve (infinite) summation over all possible word sequences. Unless phone posteriors are computed directly, as with recurrent neural networks [4], these kinds of measures are at best approximated [5].

An alternative we propose is based on the notion of entropy, with the intuition that highly discriminant distributions are more representative of the data. Clearly, this is an unjustified assumption if a low-entropy distribution could be entirely unrelated to the data. Yet, this intuition is supported by experiments that demonstrate a correlation between entropy and performance in terms of word error rate [6].

As discussed above, a distribution over the sequence  $\mathbf{P}$  is difficult to compute. However, we will see that the HMM framework provides efficient inference algorithms for  $P(q_t|\mathbf{P}, \mathbf{A})$ , the local posterior probability of a particular state given the phone and observation sequences. If each state belongs to exactly one phone model, we can easily derive the related phone posterior at frame  $t$ :

$$P(p_t|\mathbf{P}, \mathbf{A}) = \sum_{q_t \in p_t} P(q_t|\mathbf{P}, \mathbf{A}) \quad (6)$$

We define the *per-frame phone entropy*:

$$H_t = - \sum_{p_t} P(p_t|\mathbf{P}, \mathbf{A}) \log_2 P(p_t|\mathbf{P}, \mathbf{A}) \quad (7)$$

Let the measure ENT be the average of this entropy:

$$\text{ENT} = \frac{1}{N} \sum_{t=1}^N H_t \quad (8)$$

This represents how many bits of phonemic information are carried by each frame of  $\mathbf{A}$ . Using a reference alignment, we let  $\text{ENT}_V$  and  $\text{ENT}_{NV}$  be average over the vowel and non-vowel segments.

### 2.2.2 Alignment accuracy

Given an observation  $\mathbf{A}$  that is fixed to  $\mathbf{P}$ , the most likely state sequence  $\hat{\mathbf{Q}} = (\hat{q}_1, \dots, \hat{q}_t, \dots, \hat{q}_N)$ :

$$\hat{\mathbf{Q}} = \arg \max_{\mathbf{Q}} P(\mathbf{Q}|\mathbf{P}, \mathbf{A}) \quad (9)$$

Because each state corresponds to precisely one phone and generates an acoustic observation at each time step, the state sequence specifies an alignment  $(\hat{p}_1, \dots, \hat{p}_t, \dots, \hat{p}_N)$ , where state  $\hat{q}_t$  belongs to the phone model  $\hat{p}_t$ . This joint maximization does not necessarily imply that the posterior probability of  $\hat{p}_t$  is maximal at every frame. If we used such a locally maximized alignment, where each  $\hat{p}_t = \arg \max_{p_t} P(p_t|\mathbf{P}, \mathbf{A})$ , then the alignment sequence might not be compatible with the given transcription  $\mathbf{P}$ .

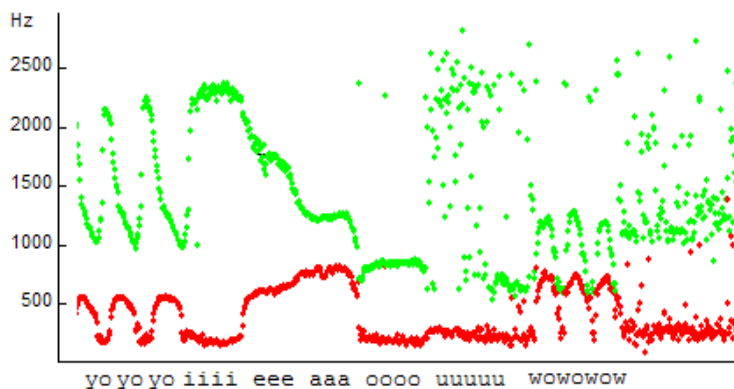


Figure 2: First and second formant estimates (F2 is always higher, by definition) produced with the Snack Sound Toolkit [8]. Note that F2 of the the vowel *u* is very low, and the tracker has difficulty locating it. Note also that the semivowels *y* and *w* are part of diphthongs characterized by dynamic formants.

Given a reference and hypothesized alignment, the accuracy measurement is straightforward:

$$\text{ACC} = \frac{1}{N} \sum_{t=1}^N \delta(\hat{p}_t = \bar{p}_t) \quad (10)$$

Similarly,  $\text{ACC}_V$  and  $\text{ACC}_{NV}$  are alignment accuracies for subsets of vowels<sup>4</sup> and non-vowels, respectively.

The alignment metric is useful for supervised tasks, where we aim to reproduce human performance. However, it is possible that a hypothesized alignment will differ from the reference in a meaningful way. Especially in unsupervised learning schemes, perhaps the model units will not reflect the salient phonemic distinctions we intend – yet, these data-driven models could be a more apt characterization of speech.

### 3 Acoustic processing

We wish to determine whether a low-dimensional, linguistically-motivated set of speech features can be applied to the speech recognition task. Such a feature set is desired because it allows one to intuitively understand the modeling process, and because their dimensionality reduces computational complexity.

#### 3.1 Formant features

Without explication of articulatory and acoustic phonetics [7], we briefly note that human speech production is approximated by a source-filter model: vibration of the vocal cords provides a periodic source signal, while the configuration of the vocal tract simulates an open-tube filter. In particular, some vowel sounds can be principally characterized by two of the vocal tract’s resonant frequencies, which phoneticians call the first formant (F1) and second formant (F2).

While there is a subjective element involved in locating formants, there are automated tools which can reasonably estimate formant frequencies from an audio signal. Figure 2 demonstrates the formant estimation tool that was used to derive the features used in this paper’s experiments: a 2-dimensional feature containing estimates of the center frequencies of the first and second formant.

<sup>4</sup>We consider recall for steady-state non-schwa vowels: *iy ih eh ae aa ao uw*

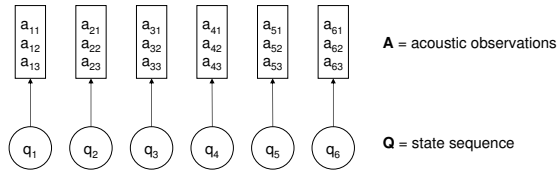


Figure 3: Acoustic model for Section 4.

### 3.2 Standard speech feature extraction

The signal processing front-end of an automatic speech recognition system converts a one-dimensional audio amplitude signal into a more manageable feature representation. Samples generally belong to a multi-dimensional continuous feature space; time is discretized, with standard frames corresponding to 25ms windows stepped at 10ms intervals. There are a few notable exceptions such as vector-quantized feature spaces [9], and long-term temporal features [10].

Common acoustic features are based on spectral properties of the speech signal. Discuss MFCC and PLP features. use features based on mel-frequency cepstral coefficients [11]. A typical MFCC base feature comprises 13 dimensions: 12 cepstral coefficients and energy.

Because speech is a dynamic process, a basic feature vector is often appended with sets of features corresponding to its first and second order derivatives. These  $\Delta$  and  $\Delta\Delta$  features multiply the dimensionality of the acoustic feature space: many features have 26 or 39 dimensions.

In addition to the acoustic models, the signal-processing front-end can provide some amount of robustness to noise in the input signal. For example, it may be possible to filter out reverberant noise caused by the room acoustics. Also, because convolution of a noise signal in the time domain is additive in the frequency domain, the cepstral mean can be averaged over an utterance or speaker, then subtracted to adjust the feature vectors to each have mean zero.

The spectral characteristics of speech produced by an open-tube model depends on two parameters: velocity of sound and the length of the vocal tract. Because vocal tract length varies, especially between male and female populations, many speech recognition systems employ vocal tract length normalization to re-scale the frequency axis when calculating features. Estimation of an appropriate scaling factor is typically done by choosing from a discrete set of values so as to maximize the likelihood of acoustic observations.

## 4 Modeling observations

Let us begin to describe an acoustic model by considering  $P(a|q)$ , the *emission* probability that a given state generates an acoustic observation. Specifically, we will provide a parameterization of this continuous distribution by approximating with Gaussian models. Although we treat these numbers as probability masses, in fact they are actually densities; this is acknowledged by the field as a “convenient fiction” [12].

For this section, we relax some conditions so that our acoustic models isolate the emission probabilities. Let the set of phones  $\mathcal{P}$  have a bijective correspondence to the set of states  $\mathcal{Q}$ . Let the state transition probabilities equal the *a priori* probabilities, and let the state sequence be independent of a given transcription:

$$\begin{aligned} P(q_t|q_{t-1}) &= P(q_t) \\ P(\mathbf{Q}|\mathbf{P}) &= P(\mathbf{Q}) \end{aligned} \tag{11}$$

In effect, we have removed temporal dependencies from the model, as in Figure 3. For such a model, the state posterior at frame  $t$  can be localized:

$$P(q_t|\mathbf{P}, \mathbf{A}) = P(q_t|a_t) = \frac{P(a_t|q_t)P(q_t)}{P(a_t)} = \frac{P(a_t|q_t)P(q_t)}{\sum_{q \in \mathcal{Q}} P(a_t|q)P(q)} \tag{12}$$

Our assumed independence of states also allows the most likely state sequence to be found by locally maximizing the state posterior at each frame:

$$\begin{aligned}\hat{\mathbf{Q}} &= \arg \max_{\mathbf{Q}} P(\mathbf{Q}|\mathbf{P}, \mathbf{A}) = \prod_{t=1}^N P(q_t|\mathbf{P}, \mathbf{A}) \\ &= (\hat{q}_1, \dots, \hat{q}_t, \dots, \hat{q}_N), \text{ where } \hat{q}_t = \arg \max_{q_t} P(q_t|\mathbf{P}, \mathbf{A})\end{aligned}\tag{13}$$

However, to compute the acoustic likelihood:

$$P(\mathbf{A}|\mathbf{P}) = \sum_{\mathbf{Q}} P(\mathbf{A}|\mathbf{Q})P(\mathbf{Q}|\mathbf{P}) = \sum_{\mathbf{Q}} \prod_{t=1}^N P(a_t|q_t)P(q_t)\tag{14}$$

This is an intractable computation because the summation over possible state sequences is  $O(|\mathcal{Q}|^N)$ .

## 4.1 Multivariate Gaussian models

Suppose our feature representation is an  $n$ -dimensional vector of continuous values,  $a_t = [a_{t1}, \dots, a_{tn}]^T$ . A suitable approximation to  $P(a|q)$  is the multivariate Gaussian parameterized by a  $n \times 1$  mean vector  $\mu_q$  and a symmetric  $n \times n$  covariance matrix  $\Sigma_q$ :

$$P(a|q) \sim \mathcal{N}(a; \mu_q, \Sigma_q) = \frac{1}{(2\pi)^{n/2} |\Sigma_q|^{1/2}} \exp \left\{ -\frac{1}{2} (a - \mu_q)^T \Sigma_q^{-1} (a - \mu_q) \right\}\tag{15}$$

### 4.1.1 Maximum likelihood parameter estimation

This moment parameterization of Gaussian models can be trained by maximum likelihood estimation if we employ a supervised procedure. Because there is one state per phone, the reference alignment  $(\bar{p}_1, \dots, \bar{p}_t, \dots, \bar{p}_N)$  deterministically maps to a labeled state sequence. We let the data set  $\mathcal{D}_q$  be the subset of acoustic training samples labeled for state  $q$ . Each element of this set is independently and identically distributed, so the data log likelihood is formed as:

$$l(\mu_q, \Sigma_q | \mathcal{D}_q) = -\frac{|\mathcal{D}_q|}{2} \log |\Sigma_q| - \frac{1}{2} \sum_{a \in \mathcal{D}_q} (a - \mu_q)^T \Sigma_q^{-1} (a - \mu_q)\tag{16}$$

The log likelihood function is concave, so to estimate  $\hat{\mu}_q$  we take the derivative with respect to  $\mu_q$  and set it to zero. This is simply the sample mean:

$$\hat{\mu}_q = \frac{1}{|\mathcal{D}_q|} \sum_{a \in \mathcal{D}_q} a\tag{17}$$

Using this result, we can estimate  $\hat{\Sigma}_q$  similarly:

$$\hat{\Sigma}_q = \frac{1}{|\mathcal{D}_q|} \sum_{a \in \mathcal{D}_q} (a - \hat{\mu}_q)(a - \hat{\mu}_q)^T\tag{18}$$

The state prior probabilities are empirical frequencies:

$$\hat{P}(q) = \frac{|\mathcal{D}_q|}{\sum_{q' \in \mathcal{Q}} |\mathcal{D}_{q'}|}\tag{19}$$

For some experiments, however, we will set these priors to the uniform distribution  $P(q) = |\mathcal{Q}|^{-1}$ , so as to investigate the influence of the acoustic likelihood alone.

### 4.1.2 Assumption of diagonal covariance

Especially as the dimensionality of the feature space increases, the multivariate Gaussian model can become difficult to compute, particularly the operations involving the  $n \times n$  covariance matrix. For this reason, most speech recognition systems specify that  $\Sigma_q$  be a diagonal matrix, assuming that feature components are all mutually independent. This simplification lets the diagonal entries correspond to the sample variances  $\sigma_{qi}$  of each component  $a_i$ , which we can now store in a  $n \times 1$  vector:

$$\hat{\Sigma}_q = [\sigma_{q1}, \dots, \sigma_{qi}, \dots, \sigma_{qn}] \text{ where } \sigma_{qi} = \frac{1}{|\mathcal{D}_q|} \sum_{a \in \mathcal{D}_q} (a_i - \hat{\mu}_{qi})^2 \quad (20)$$

The likelihood  $P(a|q)$  is then a product of univariate Gaussians:

$$P(a|q) \sim \prod_{i=1}^n \mathcal{N}(a_i; \mu_{qi}, \sigma_{qi}) \quad (21)$$

One reason why MFCC features are used in speech recognition is because this diagonal covariance assumption is justified – the components of cepstral features are significantly de-correlated. For our formant features, such an assumption is not necessarily warranted. There is a degree of correlation between the first and second formant, due to the fact that both resonant frequencies depend on the vocal tract length.

### 4.1.3 Mixture models

We have made a strong assumption in modeling output distributions with Gaussian models. Although in many cases the Gaussian is an appropriate approximation, it would be better to model arbitrary distributions over the feature space. To this end, Gaussian mixture models are used to represent multimodal distributions; given enough mixtures, a GMM should be able to closely approximate any distribution.

An output probability approximated by a GMM with  $M$  mixture components can be specified as:

$$p(a|q) = \sum_{i=1}^M \pi_i \mathcal{N}(a; \mu_i, \Sigma_i) \quad (22)$$

where  $\pi$  is a multinomial distribution over the mixture components, such that the mixture weights sum to one:  $\sum_{i=1}^M \pi_i = 1$ . Note that mixture models require more parameters: for each state  $q$ , we must store  $M$  mixture weights, mean vectors, and covariance matrices. An  $n$ -dimensional GMM with full covariance matrix and  $M$  mixtures is characterized by  $M(n + n^2 + 1)$  parameters.

The parameters of the GMM can be learned with the Expectation-Maximization algorithm. Starting from an initial guess for the parameters, we iteratively calculate posterior probabilities of mixture components and use these to update parameters. Let  $\{\pi_i^{(t)}, \mu_i^{(t)}, \Sigma_i^{(t)}\}$  be the parameter values for mixture  $i$  at the current iteration  $t$ . For the E-step, let  $\tau_i^{(t)}(a)$  be the posterior probability that sample  $a$  was generated by the  $i$ -th mixture component:

$$\tau_i^{(t)}(a) = \frac{\pi_i^{(t)} \mathcal{N}(a; \mu_i^{(t)}, \Sigma_i^{(t)})}{\sum_{j=1}^M \pi_j^{(t)} \mathcal{N}(a; \mu_j^{(t)}, \Sigma_j^{(t)})} \quad (23)$$

Summing over the training data  $\mathcal{D}$ , we get the expected counts  $c_i^{(t)}$  of mixture  $i$ :

$$c_i^{(t)} = \sum_{a \in \mathcal{D}} \tau_i^{(t)}(a) \quad (24)$$

In the M-step, mixture weights are updated with the relative frequency of expected counts:

$$\pi_i^{(t+1)} = \frac{c_i^{(t)}}{\sum_{j=1}^M c_j^{(t)}} = \frac{c_i^{(t)}}{|\mathcal{D}|} \quad (25)$$

Gaussian parameters are updated with weighted analogues of the assignments from Eqs. (17) and (18):

$$\mu_i^{(t+1)} = \frac{1}{c_i^{(t)}} \sum_{a \in \mathcal{D}} \tau_i^{(t)}(a) \cdot a \quad (26)$$

$$\Sigma_i^{(t+1)} = \frac{1}{c_i^{(t)}} \sum_{a \in \mathcal{D}} \tau_i^{(t)}(a) \cdot (a - \hat{\mu}_i^{(t+1)})(a - \hat{\mu}_i^{(t+1)})^T \quad (27)$$

These iterative updates of the EM algorithm perform coordinate ascent on the log likelihood of the data.

A very important practical consideration is the initialization of the mixtures. We could begin with mixtures placed randomly or uniformly over the feature space; or we might estimate initial means with a K-means algorithm. This is sometimes done for large mixture models with  $M = 256$ , for example. For acoustic models with fewer mixtures, new mixtures can be created with a splitting routine:

1. Begin with a single-mixture Gaussian.
2. For each mixture  $i$ :
  3. Create two new mixtures  $i'$ ,  $i''$ :
    - Divide the mixture weights:
 
$$\pi_{i'} \doteq \pi_{i''} \doteq \pi_i / 2$$
    - Separate the means: (Let  $k = 0.2$ , for example.)
 
$$\mu_{i'} \doteq \mu_i + k\sigma_i \quad \mu_{i''} \doteq \mu_i - k\sigma_i$$
    - Copy the covariance matrix (or variance vector):
 
$$\Sigma_{i'} \doteq \Sigma_{i''} \doteq \Sigma_i$$
  4. Re-train the GMM parameters using the EM algorithm.
5. Repeat from Step 2 until:
  - Desired number of mixtures is reached,
  - Or EM log-likelihood doesn't significantly improve..

Given a GMM with a large number of mixtures, we might want to reduce the size of the parameterization by removing mixtures with very low weight. We could also merge similar mixtures. If  $\mu_i \approx \mu_j$ , we can remove one component and reset the other's mixture weight to be  $\pi_i + \pi_j$ .

## 4.2 Experiment

As a demonstration of the techniques described in this section, we implemented and tested several variations of the multivariate Gaussian model. These experiments also serve to evaluate the performance of two-dimensional formant features, and to assess the utility of our alignment and entropy measures.

A single-mixture Gaussian model was built where parameters were trained with the closed-form maximum likelihood estimates. A 2-mixture GMM was then created by splitting the single-mixture Gaussian and parameters were retrained with the EM algorithm. This model was then further split to form a 4-mixture GMM, and re-trained once more. Alignment accuracy and average per-frame phone entropy are given in Tables 1 and 2, where the prior probability of phones is either fixed to the uniform distribution or empirically estimated. Results are grouped into vowel/non-vowel classes and individually for the ten best-scoring phones. For these subgroups the alignment accuracy is recall, the number of correct frames divided by the number of reference frames. The entropy of a subgroup is averaged over the frames as labeled in the reference alignment.

Figure 4 shows the placement of single-mixture Gaussians in the F1-F2 formant feature space, for models trained with full and diagonal covariance matrices. Seven vowels (iy ih eh ae aa ao uw), three voiceless fricatives (sh f s), and a silence phone<sup>5</sup> (sil) are displayed. Symbols are placed at the model means, overlaid with 10% error ellipses – roughly speaking,  $P(a \text{ inside ellipse } | q) = 0.10 \approx P(|a - \mu_q| < 0.25\sigma_q)$ .

<sup>5</sup>The corpus technically denotes this as **h#**, the utterance boundary – which is mostly silence. There is also a short pause phone (**pau**) but its occurrence is infrequent in read speech.

Phone	Accuracy	Entropy	Phone	Accuracy	Entropy
All	0.108	5.31	All	0.168	4.67
Vowels	0.215	5.41	Vowels	0.317	4.84
Non-vowels	0.086	5.29	Non-vowels	0.136	4.63
y	0.701	5.23	iy	0.779	4.71
aa	0.608	5.13	sil	0.641	4.69
f	0.397	5.00	aa	0.620	4.67
s	0.385	4.90	s	0.581	3.94
iy	0.335	5.38	eh	0.393	4.96
ey	0.322	5.56	ao	<b>0.131</b>	4.53
ae	0.221	5.55	ae	0.113	4.99
ao	<b>0.197</b>	4.94	l	0.107	4.72
aw	0.183	5.28	ey	0.075	4.96
eh	0.143	5.53	r	0.056	4.89

Uniform prior

Empirical prior

Table 1: Performance of the single-mixture multivariate Gaussian model. The prior probability of phones is uniform on the left, and estimated from data on the right.

Phone	Accuracy	Entropy	Phone	Accuracy	Entropy
All	0.114	4.86	All	0.159	4.24
Vowels	0.262	5.08	Vowels	0.322	4.51
Non-vowels	0.083	4.81	Non-vowels	0.125	4.19
aa	0.503	4.53	iy	0.663	4.51
ux	0.409	5.26	aa	0.609	4.12
y	0.388	5.08	s	0.587	4.15
ao	<b>0.373</b>	4.46	eh	0.301	4.66
s	0.341	5.13	sil	0.288	3.15
iy	0.315	5.21	l	0.255	4.49
f	0.299	5.15	pcl	0.249	3.49
ae	0.265	5.16	ae	0.243	4.62
pcl	0.249	3.98	ao	<b>0.235</b>	4.04
ey	0.199	5.36	dcl	0.223	4.39

Uniform prior

Empirical prior

Table 2: Performance of the four-mixture multivariate GMM. The prior probability of phones is uniform on the left, and estimated from data on the right.

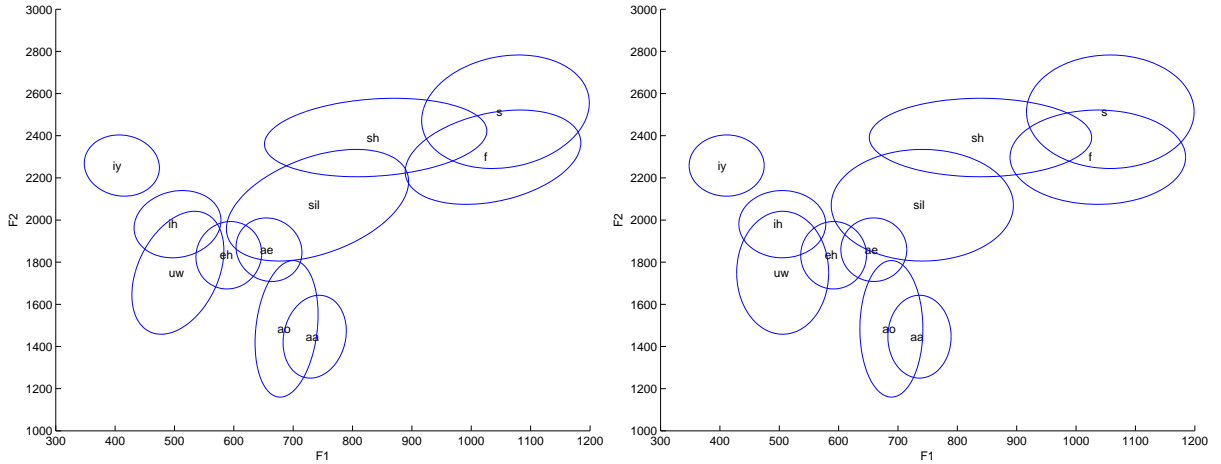


Figure 4: Distributions in the formant feature space, with full (left) and diagonal (right) covariance matrices.

#### 4.2.1 Analysis

First, there are several unsurprising things to note about the results. As expected, formant features are generally better for recognizing vowel sounds than most consonants. The use of an empirical prior distribution over phones is clearly superior to a uniform prior. Also, to some extent the multiple-mixture Gaussian models perform better than the single-mixture models.

The assumption of diagonal covariance is a simplification that trades off precision of modeling versus computational efficiency. Because these experiments used features and models with relatively minimal dimensions, the diagonal covariance assumption provided no significant computational advantage; moreover, the effect on performance was also negligible. Nonetheless, visual inspection of Figure 4 confirms the hypothesis that formant features might have a slightly positive correlation.

Examining the placement of phones on the F1-F2 space, it is reassuring to see that the models for vowels are consistent with the charts given in phonetics textbooks [7], and classic studies on the relation of vowels to formants [13]. It also makes sense that the model for the silence phone has high variance and is in the center of the formant space; this is because the formant tracker output is essentially random.

But in any experiment, the most meaningful results are the unexpected ones:

- One would not expect formant features to characterize voiceless fricatives because these sounds have no formants, per se – yet these phones are remarkably well-recognized. Fricatives, such as the alveolar /s/ and the labio-dental /f/ are essentially high-frequency noise caused by air being forced through a narrow constriction (i.e. almost whistling). Most of this is spread over the 3-4 kHz range – much higher than the range of estimated F1 and F2; for the lower frequencies, the spectrum is identical to silence<sup>6</sup>. However, the formant estimates for these sounds is higher than for silence (Figure 4). After investigation of the formant-tracking software, the reason was discovered: the algorithm actually tracks the first four formants, and imposes dependence among them. Thus, the high-frequency fricatives causes the tracker to find high F3 and F4, which “pulls up” the lower two formants.
- The mean F2 for the back vowels, such as /u/ and /ao/, are too high. Their large variance in that dimension also suggests a high degree of uncertainty. Again, this is due to the formant tracker: when F2 is low, and especially when it is close to F1, the formant tracker incorrectly tracks F2 with what should be F3. This problem, however, is largely alleviated by turning to mixture models.

Consider the histogram of F2 values for the vowel /ao/ (pronounced as in “all”), which are given in Figure 5. The data are at least bimodal, where the two main modes correspond to the true F2 and

<sup>6</sup>This is why fricatives are difficult to recognize in low-bandwidth telephone speech.

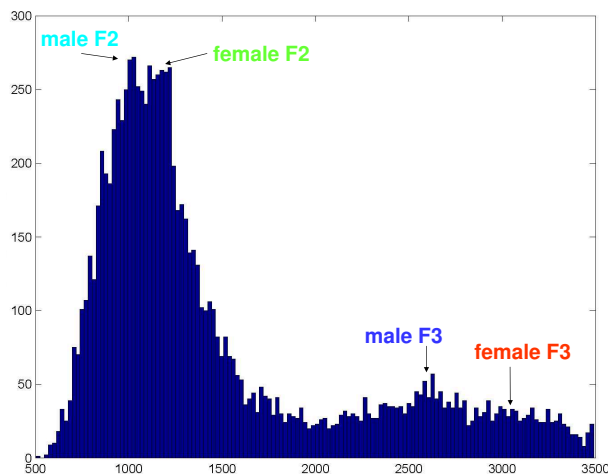


Figure 5: Histogram of F2 values for the vowel /*ao*/.

the erroneously tracked F3. As well, each mode appears to comprise two close but distinct modes; these are likely the male and female populations exhibiting different spectral characteristics due to physiology. Since resonant frequencies are inversely related to vocal tract length, the higher of the pair of modes can be interpreted as the female population<sup>7</sup>.

We can see that this multimodality is well modeled by a mixture model, as illustrated in Figure 6. Starting from a poorly fitting single Gaussian, the mixture-splitting procedure locates the two main modes after a few iterations of EM. Splitting again, the new mixtures refine the model to capture the gender-dependent characteristics. This splitting procedure, in which new mixtures are spawned nearby previous mixtures, seems ideally suited to these situations with recursively modal topology. Note also that after splitting mixtures, the EM algorithm increases the data log-likelihood to levels that might have been unattainable with fewer mixtures. The end result, as evidenced in Tables 1 and 2, is that the model for /*ao*/ becomes much better.

- Alignment accuracy and entropy are not exactly correlated. It was anticipated that vowels, having smaller variance and higher accuracy, would exhibit low entropy over their frames because their models would be correctly discriminant; consonants would display higher entropy because their distributions would be less peaked. The opposite was true in many cases, because there were many vowels and other phones competing in a small, crowded region of the formant space. By contrast, the entropy during other sounds, such as silence and fricatives, were located in relatively empty regions of the formant space where other models attributed low probability. Thus, despite the fact that their distributions were not as peaked, for these phones the entropy was often lower and contributed to better accuracy.

#### 4.2.2 Comparison to existing results

As previously stated, the measures used in the work are not standard. They were chosen to allow evaluation of model components without having to scale up to a fully-integrated speech recognizer.

In these experiments the measure of alignment accuracy is based entirely on posterior probabilities that are temporally independent. Since there is actually no relation to the phone sequence, it might be more apt to call it the phone recognition accuracy<sup>8</sup>. In the author’s opinion, this is an unfortunate choice of terminology: it implies that there is some phonetic ground truth in the reference. This cannot be the case

<sup>7</sup>The TIMIT data is approximately 70% male, a ratio which is more evident in the two F3 modes.

<sup>8</sup>In fact, this is the *de facto* standard when working with TIMIT data.

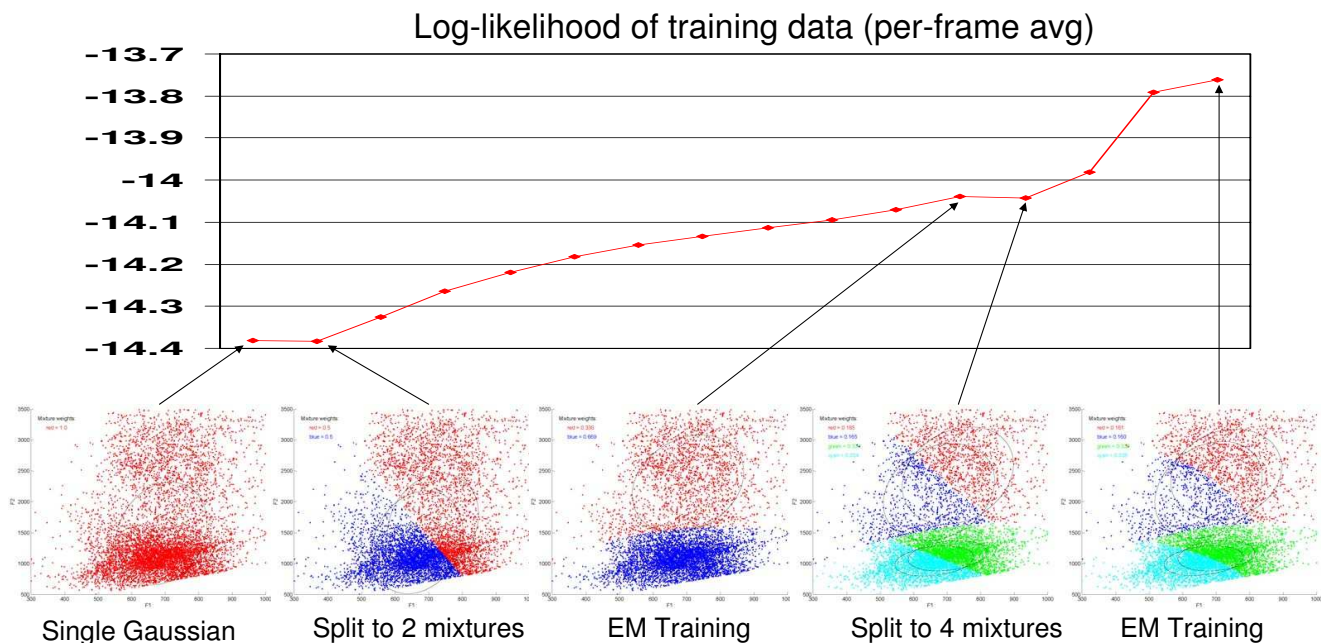


Figure 6: Mixture-splitting and EM training for the /ao/ model.

if the manual labeling process was constrained by a phonemic transcription of the utterance. The human annotator was not recognizing speech, so much as aligning to a given set of symbols. For precisely this reason, though, the labels provide an ideal benchmark against which to compare a Viterbi forced alignment. That is, the alignment accuracy measure was really posited with the intent of evaluating inference in the HMM model of the next section.

To place these results in the context of other work, phone recognition of the TIMIT corpus was of interest in some of the early HMM phone modeling experiments [14], but today is principally used in developing connectionist phone classifiers. These phone-labeled data provide targets for error back-propagation training of neural networks, and so the phone recognition task is natural when a forward pass through the network outputs posterior distributions of phones. Following the convention established in the first experiments, the 61 phonemes of the TIMIT transcriptions are conflated to a set of 39 phones; we did not perform such a reduction in these experiments, though in retrospect it would have been more informative to do so.

For an HMM with vector-quantized observations [14], phone recognition accuracy is reported around 65% for context-independent phone models. With context-dependency, which captures the allophonic nature of the phonemic alignments, recognition improves to over 70%, slightly better than the performance of expert human spectrogram readers. Neural networks can achieve up to 75% [15], and very recent work in conditional random fields has yielded TIMIT phone accuracies close to 80% [16]. Considering that all these approaches make use of temporal dependencies (neural networks use several frames of acoustic context), the performance our naive Gaussian classifier is not entirely discouraging and indicates that integrating the HMM-GMM system will likely yield significant improvement.

## 5 Temporal modeling

[Note:] I regret that I had some difficulties implementing the HMM portion of the HMM-GMM system, but I leave this unfinished section here to show what the next step will be, and to preserve the intended structure of this project. The next step, of course, is to define a topology for the HMM phone units. These can then be concatenated in the order given by a phone transcription, so as to define the HMM state transition matrix. The alignment accuracy metric requires an algorithm to find the most likely state sequence. The entropy metric requires an algorithm for computing posterior state probabilities from incomplete evidence.

Fortunately, there is a set of recursive algorithms used for efficient HMM inference and parameter estimation. Each of these is simply a specific instance of a generalized technique learned in CS281A:

- The forward and forward-backward algorithms, which can compute overall likelihoods and posterior probabilities from HMMs, are instances of SUM-PRODUCT.
- The Viterbi algorithm, which finds the most probable path through the HMM, is MAX-PRODUCT.
- Baum-Welch re-estimation of parameters is the EM algorithm: in the E-Step, the posterior probabilities of states are computed by the forward-backward algorithm.

Alternatively, an HMM can be easily converted to a form compatible with the junction tree framework.

## 6 Conclusion

This project was an enriching opportunity to explore the speech recognition problem formulated in terms of graphical models. In particular, a mixture of multivariate Gaussians proved effective in modeling acoustic observations in a continuous feature space. Through a demonstration using very low-dimensional speech features, it was possible to analyze the acoustic model in a manner consistent with linguistic knowledge, as well as from a more quantitative perspective.

From the results of experiments, it seems possible that formant features could be used for some limited variants of automatic speech recognition. The acoustic match may be good enough for a small-vocabulary task, such as number recognition. Formant features could also be employed for didactic purposes, illustrating acoustic modeling concepts with an intuitive example, as in this work.

In the near future, I look forward to completing the full implementation of an HMM-GMM acoustic model. This will give a better determination of whether formant features are in fact useful, and if the proposed performance measures are meaningful.

## References

- [1] L. F. Lamel, R.H. Kassel, and S. Seneff. Speech database development: Design and analysis of the acoustic-phonetic corpus. In *Proc. DARPA Speech Recognition Workshop*, pages 100–109, 1986.
- [2] H. Boullard, H. Hermansky, and N. Morgan. Towards increasing speech recognition error rates. *Speech Communication*, 18(3):205–231, 1996.
- [3] B. Chen, Ö Cetin, G. Doddington, N. Morgan, M. Ostendorf, T. Shinozaki, and Q. Zhu. A CTS task for meaningful fast-turnaround experiments. In *Proc. RT-04 Workshop*, 2004.
- [4] A. Robinson, M. Hochberg, and S. Renals. IPA improved modeling with recurrent neural networks. In *Proc. ICASSP*, 1994.
- [5] T.J. Hazen and I. Bazzi. A comparison and combination of methods for OOV word detection and word confidence scoring. In *Proc. ICASSP*, 2001.

- [6] J. Barker, G. Williams, and S. Renals. Acoustic confidence measures for segmenting broadcast news. In *Proc. ICSLP*, 1998.
- [7] P. Ladefoged. *A Course in Phonetics*. Harcourt, Orlando, 2001.
- [8] The snack sound toolkit. [www.speech.kth.se/snack/](http://www.speech.kth.se/snack/).
- [9] L.R. Bahl, P.V. deSouza, P.S. Gopalakrishnan, and M.A. Picheny. Context-dependent vector quantization for continuous speech recognition. In *Proc. ICASSP*, 1993.
- [10] B. Chen, Q. Zhu, and N. Morgan. Learning long-term temporal features in LVCSR using neural networks. In *Proc. ICSLP*, pages 612–615, 2004.
- [11] B. Gold and N. Morgan. *Speech and Audio Signal Processing*. John Wiley and Sons, New York, 2000.
- [12] S. Young et al. *The HTK Book*. Cambridge University Press, Cambridge, 2001.
- [13] Gordon Peterson and Harold Barney. Control methods in a study of the vowels. *Journal of the Acoustical Society of America*, 24(2):175–184, March 1952.
- [14] K. Lee and H. Hon. Speaker-independent phone recognition using Hidden Markov Models. *IEEE Trans. on Acoustics, Speech, and Signal Processing*, 37(11):1641–1648, 1989.
- [15] N. Strom. A tonotopic artificial neural network architecture for phoneme probability estimation. In *Proc. of the IEEE Workshop on Speech Recognition and Understanding*, pages 156–163, 1997.
- [16] A. Gunawardana, M. Mahajan, A. Acero, and J.C. Platt. Hidden conditional random fields for phone classification. In *Proc. Interspeech*, pages 1117–1120, 2005.