Isolated English Letter Speech Recognition

Konstantinos G. Derpanis

April 5, 2004

1 Introduction

This report contains a summary of several strategies used in building an isolated-word speech recognition system. The acoustics of each word is modeled as a Gaussian mixture, continuous density hidden Markov model (CDHMM). The speech database used, called *ISOLET* [1], consists of 7800 spoken English letters, two productions of each letter by 150 speakers. The database is organized into five equal subsets, (*ISOLET-1*, *ISOLET-2*, *ISOLET-3*, *ISOLET-4* and *ISOLET-5*). For the purposes of training only the first production of each letter from *ISOLET 1-4* was used and for evaluating the strategies all data in *ISOLET-5* was used. To instantiate the strategies the HTK HMM toolkit [3] was used.

This report is organized as follows: Section 2 describes the grammar used in all experiments, Section 3 considers modeling whole word HMMs (i.e., one HMM per word), Section 4 considers modeling the language as the concatenation of sub-word models, where the basic units are the individual sounds of the language called phones and Section 5 provides a discussion of the results of each of the experiments.

2 Grammar

The goal of the system was to build an isolated English letter speech recognition system. The grammar used for all experiments consisted of 26 paths between start and end nodes, where each path contained a letter from the English alphabet see (Fig. 1). In addition, for each of the experiments (except for the experiment detailed in section 3.1) a silence state was added to the beginning and the end of each path. This was done to explicitly model the silence portions present in the *ISOLET* speech signals (see [1] for details).

3 Word-based modeling

In the following subsections a summary of two approaches for word-based HMM modeling will be given. The first approach considered in subsection 3.1 considers the entire speech signal as the pronunciation of the letter (i.e., silence included). The second approach detailed in subsection 3.2 considers each speech signal as consisting of a letter preceded and proceeded by an approximately equal silence interval (roughly 80 milliseconds [1]).

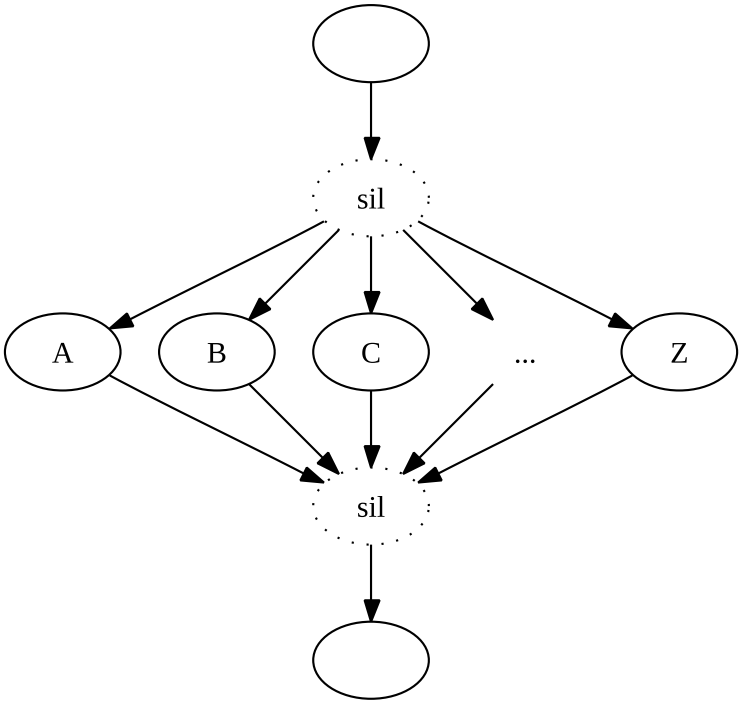

Figure 1: Grammar. Depicted is the general grammar used in all experiments. Note that the experiment detailed in subsection 3.1 omits the usage of sil (i.e., silence node).

3.1 Non-silence model

The first experiment consisted of modeling each of the speech signals (i.e., letter pronunciations) as an HMM. The grammar for this experiment is depicted in Fig. 1, where the sil nodes are omitted. By not explicitly modeling the silence transitions results in considering the silence portions as part the pronunciation of the speech signals. One would expect that more states would be required to successfully model each letter compared to the situation where the silence was modeled separately as done in subsection 3.2; the experimental results affirm this hypothesis, see Table 2. The rest of this subsection summarizes the training, evaluation and the formatting of the output results for this experiment; see Fig. 2 for a summary of the HTK processing stages.

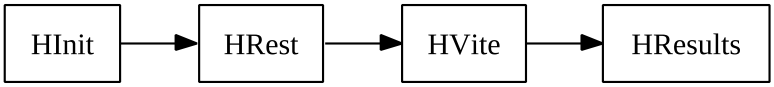

Figure 2: Depicted is a summary of the HTK processing stages for isolated word training (i.e., *HInit* and *HRest*), recognizer evaluation (i.e., *HVite*) and formatted result output (i.e., HResults).

To initialize the parameters of the HMM models *HInit* was first used. The basic principle behind *HInit* considers the HMM as a generator of speech observations. If the states that generated each of the observations in the training data were known, then the means and variances of the observation distributions for each state could be estimated. The realization of this principle (i.e., *HInit*) is done by first uniformly segmenting the training data and associating each successive segment with each successive state. Next, Viterbi segmentation is used to find the most probable sequence of states for all the data. Following the segmentation, the HMM parameters are re-estimated. The Viterbi segmentation and parameter re-estimation steps are iterated until the parameters have converged (for more details see section 8.2 in [3]). For the problem at hand, each HMM model was initialized by considering only its corresponding training data (e.g., using only training data for speech signal A for HMM model A).

To complete the estimation of the HMM models, the tool *HRest* was used. *HRest* takes the initialized models of *HInit* as input and outputs the final estimates of each model. The principles and operations behind the two tools are very similar in that they try to find an assignment of the data to the HMM states in an iterative fashion. The main difference is that through the use of Viterbi training *HInit* makes a hard decision in the state assignment of each observation, whereas *HRest* uses Baum-Welch to make a soft decision for the assignment (for the mathematical details see [3] section 8.2 and [2]). As with *HInit*, the estimation of each of models are done using its corresponding data.

Following the estimation step is the recognition step. For the recognition step the *HVite* utility was used. *HVite* takes as input a recognition network and a set of transcribed testing data[1] and outputs classifications for the respective data. The recognition network consists of a word-level network (see Fig. 1), a dictionary, and a set of trained HMMs. To arrive at the

[1] For this project the transcriptions were taken directly from the filenames of the data.

classification *HVite* uses the Viterbi algorithm to find the path (i.e., letter in the current context) in the recognition network with the largest probability.

The final step consisted of formatting the output of *HVite* using the *HResults* tool. For the purposes of this project the evaluation of each system's performance was based on the global percentage correct (% Correct), defined as,

$$\% \text{ Correct} = \frac{H}{N} \times 100 \tag{1}$$

where H is the number of correctly classified test cases and N is the total number of test cases. In addition, *HResults* has the ability to output a confusion matrix[2]. This feature proved helpful in isolating problematic cases (see section 4 for more details).

3.2 Silence model

In the previous subsection the silence present in the speech signal was implicitly modeled by the HMM representing each letter. The next experiment consisted of extending the previous strategy by adding silence nodes before and after the pronunciation of each letter (see Fig. 1). The silence node was treated as a separate word in the dictionary and thus modeled by an HMM. A consequence of the extension is that the training phase introduced in subsection 3.1 was replaced by embedded training. This subsection summarizes the training step. For a pictorial summary of the all the steps for this experiment see Fig. 3.

To initialize the parameters *HCompV* was first used. The basic strategy implemented by *HCompV* is to make all models equal initially and move straight to embedded training. This is accomplished by equating the local mean and variance parameters of the Gaussians of each state to the global mean and variance. Unlike *HInit* this approach does not require labeled training data.

To complete the estimation of the HMM models, the *HERest* tool was used. In short, *HERest* performs a single iteration of Baum-Welch re-estimation of the whole set of HMM models simultaneously. For each piece of training

[2] A confusion matrix is a matrix containing information about the actual and predicted classes. Each cell in the matrix represents the number of elements classified as class m when the actual class was n. The diagonal of the matrix represents the correctly classified elements.

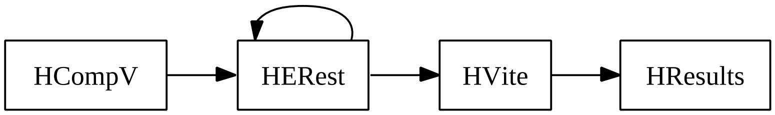

Figure 3: Depicted is a summary of the HTK processing stages for embedded training (i.e., *HCompV* and iterated *HERest*), recognizer evaluation (i.e., *HVite*) and formatted result output (i.e., *HResults*).

data, the corresponding phone models are concatenated and the forward-backward algorithm is used to collect state occupancy statistics, mean and variance statistics. When all the training data has been processed, the statistics are collected and the model parameters are re-estimated. This process was iterated five times (as suggested in class) to avoid over fitting to the training data.

4 Phoneme-based modeling

In this section a brief summary of two experiments using phoneme-based word models will be given. The motivation for the phonemic representations was the observation that a subset of the letters that shared an elemental sound were being confused (i.e., prominent when viewing results using a confusion matrix), for example the letters B and V. With the phonemic models the hope was that the elemental sounds (i.e., phones) would be better modeled due to the data sharing in the embedded training phase and thus improve overall classification.

The steps used for the phonemic-modeling approach are almost exactly as those detailed in subsection 3.2 and summarized in Fig. 3. The exception is that the dictionary of the language contained the phonemic transcriptions of each of the letters as given in Table 1. The transcriptions for all letters were arrived at by using the British English BEEP pronouncing dictionary[3]; with the exception of the letter Z since ISOLET follows the American pronunciation. The results of this experiment (see discussion in section 5) looked promising except for the letters A and E. These letters only contain one phoneme and were being confused in cases where their respective phoneme was a constituent of another word. The confusion may be a result of impre-

[3] Available by anonymous ftp from: svr-ftp.eng.cam.ac.uk/pub/comp.speech/dictionaries/beep.tar.gz.

Letter	Phones	Letter	Phones
A	ey	O	ow
B	b iy	P	p iy
C	s iy	Q	k y uw
D	d iy	R	aa r
E	iy	S	eh s
F	eh f	T	t iy
G	jh iy	U	y uw
H	ey ch	V	v iy
I	ay	W	d ah b l y uw
J	jh ey	X	eh k s
K	k ey	Y	w ay
L	eh l	Z	z iy
M	eh m	SILENT	sil
N	eh n		

Table 1: Listed are the phonemic transcriptions for each of the letters of the English alphabet (plus a transcription for silence) used for the phoneme models.

cise segmentation of the data and/or co-articulation effects. To address this issue the second phoneme experiment replaced the phone transcription of the letters A and E with distinct phones. The idea was that only the training data for the letters A and E would contribute to the training of these models and thus would be modeled better.

5 Discussion

In this section a short summary and discussion of the experimental evaluation is given.

Performance was measured by using the percent correct measure given in Eq. (1). To find the best percent correct score various combinations of the number of HMM states, the number of mixtures and the degrees of freedom of the covariance matrix (i.e., diagonal vs. full) were investigated. For the word based model, the number of states $\in [1, \ldots, 10]$ and the number of mixtures $\in [1, \ldots, 10]$. For the phone models, the number of states was fixed to 3 (as suggested in class and in [3]) and the number of mixtures $\in [1, \ldots, 10]$. All experiments used the first production of each letter from *ISOLET 1-4* for training and all the data in *ISOLET-5* for testing.

The top result for each of the strategies is summarized in Table 2. As can be seen the best result was 98.44% obtained by using the *Phone Model 2*, where the covariance matrix was full. Interestingly, in [1] the authors report that the best performance they achieve using the same subset of data for training and testing as presented here was 95%.

Further improvements may be had by using the gender information provided in the *ISOLET* data set to improve trained models. This would be accomplished by building separate models for male and female speakers. Additionally, leveraging the georgraphical information of the speakers may improve results.

References

[1] R. Cole, Y. Muthusamy, and M. Fanty. The ISOLET spoken letter database. Technical report, Dept. Comp. Sci., Oregon Graduate Institute, Nov. 1994.

[2] L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. In *Proceedings of the IEEE*, volume 77, pages 257–286, Feb 1989.

[3] S. Young, D. Kershaw, J. Odell, D. Ollason, V. Valtchev, and P. Woodland. *The HTK Book*. Microsoft Corp., 3.1 edition, July 2000.

Word Model I						*Word Model II*					
Diagonal Covariance			Full Covariance			Diagonal Covariance			Full Covariance		
% Correct	Number of States	Number of Mixtures	% Correct	Number of States	Number of Mixtures	Correct	Number of States	Number of Mixtures	Correct	Number of States	Number of Mixtures
94.42	8	3	95.64	6	2	96.05	7	1	97.84	4	1
						96.05	7	2	97.84	4	2
						96.05	7	3	97.84	4	3
						96.05	7	4	97.84	4	4
						96.05	7	5	97.84	4	5
						96.05	7	6	97.84	4	6
						96.05	7	7	97.84	4	7
						96.05	7	8	97.84	4	8
						96.05	7	9	97.84	4	9
						96.05	7	10	97.84	4	10

Phone Model I						*Phone Model II*					
Diagonal Covariance			Full Covariance			Diagonal Covariance			Full Covariance		
% Correct	Number of States	Number of Mixtures	% Correct	Number of States	Number of Mixtures	% Correct	Number of States	Number of Mixtures	% Correct	Number of States	Number of Mixtures
94.91	3	1	97.78	3	1	95.64	3	1	98.44	3	1
94.91	3	2	97.78	3	2	95.64	3	2	98.44	3	2
94.91	3	3	97.78	3	3	95.64	3	3	98.44	3	3
94.91	3	4	97.78	3	4	95.64	3	4	98.44	3	4
94.91	3	5	97.78	3	5	95.64	3	5	98.44	3	5
94.91	3	6	97.78	3	6	95.64	3	6	98.44	3	6
94.91	3	7	97.78	3	7	95.64	3	7	98.44	3	7
94.91	3	8	97.78	3	8	95.64	3	8	98.44	3	8
94.91	3	9	97.78	3	9	95.64	3	9	98.44	3	9
94.91	3	10	97.78	3	10	95.64	3	10	98.44	3	10

Table 2: Depicted is a summary of the results (i.e., largest % Correct) with the accompanying parameters (i.e., number of states, number of mixtures and covariance type) for the four experiments conducted. Here *Word Model I* refers to the experiment summarized in subsection 3.1, *Word Model II* refers to subsection 3.2, *Phone Model I* refers to the experiment using the phone transcription given in Table 1 and outlined in section 4 and *Phone Model II* refers to the extension of *Phone Model I* outlined in section 4.