HIDDEN-ARTICULATOR MARKOV MODELS: PERFORMANCE IMPROVEMENTS AND ROBUSTNESS TO NOISE

Matt Richardson, Jeff Bilmes and Chris Diorio

University of Washington
{mattr@cs, bilmes@ee, diorio@cs}.washington.edu

ABSTRACT

A Hidden-Articulator Markov Model (HAMM) is a Hidden Markov Model (HMM) in which each state represents an articulatory configuration. Articulatory knowledge, known to be useful for speech recognition [4], is represented by specifying a mapping of phonemes to articulatory configurations; vocal tract dynamics are represented via transitions between articulatory configurations.

In previous work [13], we extended the articulatory-feature model introduced by Erler [7] by using diphone units and a new technique for model initialization. By comparing it with a purely random model, we showed that the HAMM can take advantage of articulatory knowledge.

In this paper, we extend that work in three ways. First, we decrease the number of parameters, making it comparable in size to standard HMMs. Second, we evaluate our model in noisy contexts, verifying that articulatory knowledge can provide benefits in adverse acoustic conditions. Third, we use a corpus of side-by-side speech and articulator trajectories to show that the HAMM can reasonably predict the movement of the articulators.

1. INTRODUCTION

Hidden Markov Models are a popular method for speech recognition. Commonly, a left-to-right topology is used, where each phoneme is represented by a sequence of states, typically three [16]. In this topology, each state simply represents a portion of a phoneme. This acoustic-based model for speech recognition does not directly incorporate any knowledge of the source that produced the speech.

In contrast, we know that speech is formed by the glottal excitement of a human vocal tract consisting of articulators which shape and modify the sound in complex ways. Since this system is limited by physical constraints, it could allow us to construct a more realistic model of speech to improve speech recognition. Such a model could have many advantages such as being better able to predict co-articulation effects, since they are due to physical limitations and energy-saving shortcuts in articulator movement [9]. Furthermore, by modeling articulators, we can allow asynchrony between their movements, which may more accurately model the production of speech [4]. Finally, because articulatory configurations are shared across multiple phonetic conditions, such a model may need less training data than a model without such information.

There has been much interest in incorporating articulatory knowledge into speech recognition. Gupta and Schroeter [8] discuss the analysis-by-synthesis approach, which attempts to estimate the parameters of the Coker [3] model, which is based on articulatory features. The analysis-by-synthesis work is often targeted toward speech compression, where the quality of the synthesis is more important than the accuracy of the estimated parameters. The inverse mapping problem, the mapping of acoustic features to articulatory configurations, is discussed in [1]. In [10] Kirchhoff demonstrates how to use artificial neural networks to estimate articulatory features from acoustic features. The HAMM can also be cast as a factorial HMM [14] which has been attempted for speech recognition [11] without the use of articulatory knowledge. We chose to implement the HAMM using a standard HMM with a constrained state space equal to the Cartesian product of the components, as this allows us to use standard HMM algorithms for training and testing.

The rest of the paper is as follows: Section 2 presents the model and describes how it is initialized and trained. Section 3 presents experimental results.

2. THE MODEL

A Hidden-Articulator Markov Model is based on the human articulatory system. Suppose we have N articulators in our model, and each articulator, *a*, can be in one of M_a positions. An *articulatory configuration* is an N-element vector $C=\{c_1,c_2,\ldots,c_N\}$, where c_a is an integer $0\le c_a<M_a$. A HAMM is an HMM in which each hidden state represents a particular articulatory configuration. The details of the model can be found in our previous work [13].

2.1 Articulatory Space

A word is a sequence of articulator targets. In mapping words to articulator configurations, we make the simplifying assumption that words can be modeled as a sequence of phonemes, each of which is mapped to a sequence of one or more articulatory configurations.

Using Edwards [6], we devised an articulatory feature space using eight features (see Table 1). Each phoneme's articulatory characteristics were manually examined to determine the best mapping into our feature space. The phoneme may be mapped into one, or a sequence of articulatory configurations. We developed static constraints to limit the possible articulatory configurations. Some of these constraints rule out physical impossibilities, while others disallow states that are physically possible but unlikely in American English. The static constraints reduced the number of HAMM states from 25,600 to 6,676. Finally, we developed dynamic constraints which impose continuity and maximum velocity restrictions on the articulators. Details on the exact phoneme mapping, static constraints, and dynamic constraints can be found in our previous work [13].

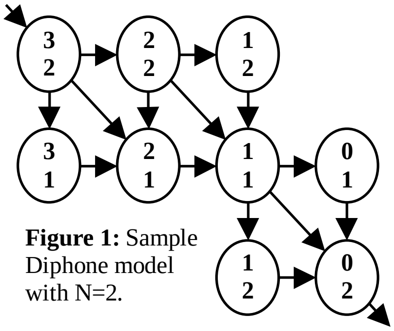

Figure 1: Sample Diphone model with N=2.

We constructed models for each diphone appearing in the training, development, and test set. To construct a diphone, we list the sequence of articulatory targets from the first target of the first phoneme to the last target of the second phoneme. The states in between are filled in and allowable transitions are added. Figure 1 shows a sample diphone from phoneme { [3 2] → [1 1] } to { [0 2] }. Notice how the HAMM allows asynchrony in the articulator movements, whereby one articulator may move with or without other articulators moving. We believe this de-synchronization allows the HAMM to more accurately model speech production. In addition, many different diphones may contain the same intermediate articulatory state, leading to a large amount of state sharing between diphones.

2.2 Training

The HAMM is trained using the Baum-Welch algorithm. An HMM is constructed for each diphone; word models are created by concatenating diphone models. This allows the training to learn transition probabilities on a diphone level. We used segmental k-means clustering to produce the initial parameter settings for each articulatory state corresponding to a phoneme. Initial parameters for the interim states were initialized by interpolation. Details can be found in our previous work [13].

3. EXPERIMENTS AND RESULTS

Speech recognition results were obtained using PHONEBOOK, a large-vocabulary, phonetically-rich, isolated-word, telephone-speech database [12]. All data is represented using 12 MFCCs plus c_0 and deltas resulting in a 26 element feature vector sampled every 10ms. In the HAMM, each state uses a mixture of two diagonal covariance Gaussians. Additionally, we define a model, *4state*, which is a standard left to right, diagonal Gaussian HMM with 4 states per phoneme and with 24 mixtures per state.

The training and test sets are as defined in [5]. Test words do not occur in the training vocabulary, so test word models are constructed using diphone models learned during training. Training was considered complete when the log-likelihood difference between successive iterations fell below 0.2%.

3.1 Reducing the number of Parameters

One disadvantage of the HAMM is its large state space and therefore number of parameters. We thus removed states during training that had low state occupation probabilities. During each training iteration, a state *i* was removed from a diphone if:

$$\gamma_i < \sum_{j=1}^{N} \frac{\gamma_j}{N\tau} \text{ , where } \quad \gamma_i = \sum_t \gamma_i(t) \qquad \gamma_i(t) = p(Q_t = i \mid X)$$

Where *N* is the number of diphone states in the diphone, *i* represents a state, Q_t is the hidden state random variable, and X is the entire observation set. τ is what we call the *state vanishing ratio (SVR)*. When a state is removed, any transitions to it are proportionately re-directed to all of possible direct successors.

Feature	Abbr.	M	Low →	High	Formula
Jaw Separation	Jaw	4	closed	open	UI_Y – LI_Y
Lip Separation	Lip	4	closed	open	UL_Y – LL_Y
Lip Rounding	Rnd	4	round	wide	none
Tongue Body	BF	5	back	fwd.	-TB_X – BN_X
Tongue Body	LH	4	low	high	TB_Y – BN_X
Tongue Tip	Tip	5	low	high	TT_Y – BN_Y
Velic Aperture	Vel	2	closed	open	-V_Y – BN_Y
Voicing	Voic	2	off	on	laryn. c_0 energy

Table 1: Articulatory dimensions. M denotes the number of quantization levels. Formulas are given for translating from recorded MOCHA [15] data to our articulatory space (see Section 3.5). All values except laryngograph energy come from the EMA data.

Models were trained initially using a large SVR, τ=10^{20}. After training converged, the SVR was decreased and models were re-trained until convergence. As a final step, states were removed if they existed only in untrained diphones. We did not implement mapping from untrained diphones to trained diphones. Instead, we depend on the shared nature of the model to predict untrained diphones (see section 3.4).

Figure 2 shows the effect of various SVRs on the number of model parameters, as well as on the word error rates (WER). As expected, when SVR decreases so do the number of parameters, but unexpectedly we also found a WER improvement. After determining the ideal SVR on the development set (τ =10^5) , we tested the pruned model on the test set. As Table 2 shows, the pruned model has 51% fewer parameters, but shows a 16-24% relative WER reduction. Later experiments use this reduced model.

In previous work [13], we verified that the HAMM uses the articulatory knowledge to its advantage by showing it outperforms a similarly constructed model containing no articulatory knowledge. To construct such a model, we used a random mapping of phonemes to articulatory features. That is, for all phonemes, for all a, c_a is set to a random value between 0 and M_a. The same static and dynamic constraints are still applied. Here, we re-verify these findings after both the HAMM and the random models have been pruned using the SVR technique. The results are summarized in Table 2. Each of the models (one HAMM, five random) was pruned with a SVR of 10^2, 10^3, 10^5, 10^{10}, and 10^{20}. The SVR which achieved the lowest WER on the 75 and 150 word development sets was then used for the test set. The HAMM significantly out-performs the random models (p<0.01). The HAMM also has significantly fewer parameters than the random models (p<0.01).

3.2 Model Combination

The HAMM performs worse than the *4state* model. We hypothesize that since it is based on articulatory knowledge, the HAMM makes different errors, and thus a combination of the two will have superior performance.

There are a variety of model combination techniques. One simple way is by a weighted sum of their log-likelihoods. The weight given to each model represents a prior confidence in its

Model	**75**	**150**	**300**	**600**	**params**
unpruned HAMM	3.23%	4.67%	6.69%	9.03%	520k
pruned HAMM	2.46%	3.77%	5.47%	7.56%	255k
pruned random models	3.18% ± 0.08%	4.48% ± 0.11%	6.53% ± 0.15%	8.83% ± 0.17%	388k ± 27k
4state	1.45%	2.79%	4.04%	5.76%	203k
pruned HAMM + 4state	0.99%	1.80%	2.79%	4.17%	458k

Table 2: WER Results on the test set for various lexicon sizes. Random model results are given as mean ± standard error (over 5 models). The pruned HAMM does better in both WER and number of parameters than before pruning, as well as in comparison with random models. The last entry is the combined model, which out-performs all other models tested.

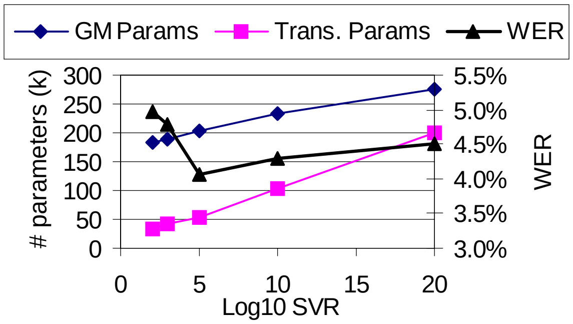

Figure 2: Effect of varying SVR. Shown are the number of Gaussian mixture (GM) parameters, transition parameters, and word error rate on the development set.

accuracy. If the model errors are independent, this will result in a higher accuracy [2]. We gave the HAMM model a weight of 1, and found the optimal *4state* model weight (searching in increments of 0.5) based on the development set to be 2.5. On the test set, the combined model achieves a 28-35% WER improvement over the *4state* model alone (see Table 2). This demonstrates that a HAMM can give practical gains when used in combination with a standard model.

3.3 Noise

A potential advantage of articulatory based HMMs is robustness to noise. Table 3 compares the performance of the models in a 15dB SNR additive noise environment. Interestingly, the HAMM and the *4state* model achieve comparable WER in this case. We believe the articulatory knowledge assists the HAMM by being more attuned to the speech-like information contained in the signals. Again, we combined the two models, using a weight of 1 for both (the optimum on the development set), and obtained a 23-26% relative WER improvement over the 4state model alone.

3.4 Diphone Models

Because the HAMM is diphone-based and the *4state* model is monophone-based, our experiments may exhibit a bias against the *4state* model. To ensure that our experiments were fair, we built diphone *4state* models, called *4state-d1*, and *4state-d2* with 1 and 2 diagonal Gaussian components per state, respectively. We also constructed a new reduced test set which is the full test set minus any words which contain at least one diphone that appeared in the training set less than 10 times. On average, the reduced test set is 12% smaller than the full test set, both in utterances and lexicon size. The reduced set is necessary for testing the *4state-d* models. By comparing the results between *4state* on the full and reduced test sets, we find that the reduced test set is simpler. We have verified that the words which were removed were no greater than average in causing errors, and thus the error reduction in the reduced test set is due to the reduction in lexicon size.

Note that the relative WER increase in going from the reduced to the full test set is lower for the HAMM than it is for the *4state* monophone model, which implies the HAMM does not have a disproportionately larger number of errors in the words containing untrained diphones. This suggests that the HAMM does a reasonable job at predicting unseen diphones. Also note that the performance of the 4state-d models is similar to the 4state model. This suggests that we have not been unfair in our comparison of the HAMM to the 4state model, even though the 4state model is only a monophone model while the HAMM is a diphone model.

3.5 Real Articulatory Data

A Viterbi path using our HAMM is an estimation of articulatory feature values throughout an utterance. To show that our model reasonably predicts articulator movements, we compare the Viterbi path with recordings of articulator motion. The articulator data comes from the MOCHA [15] database, which contains both speech and the measured time-aligned articulator trajectories. Data for two speakers, a female (fswe0) and a male (msak0), is currently available. The formulas for converting from the MOCHA data to our articulator feature space are given in Table 1. Note that in the MOCHA database, positive x-direction is toward the back of the vocal tract, away from the teeth, and positive y-direction is up, toward the roof of the mouth.

Table 5 shows the correlation coefficients between our articulatory feature predictions with that of the measured MOCHA features. All values greater than 0.01 are statistically significant ($p<0.01$). As can be seen, the diagonal entries tend to have the highest correlation. Table 5 also presents the correlation of the measured MOCHA features with themselves. This table demonstrates which correlations between features are expected, due to the physical behavior of the articulators. For instance, the strong negative correlation between the estimated jaw opening parameter with the measured lowness of the tongue is normal, as it also occurs within the measured data. The estimated and measured feature correlations generally agree.

There are a multitude of reasons why these correlations are not higher. First, the MOCHA data is recorded at 16kHz but PHONEBOOK is telephone-quality. Second, our model was trained using isolated word speech but MOCHA is continuous speech. Third, our quantization of articulatory features as represented in the hidden state space is not necessarily linear. Also, MOCHA is British and PHONEBOOK is American English. Nevertheless, the correlations indicate that the HAMM is indeed representing ar-

Model	**75**	**150**	**300**	**600**
HAMM	15.40%	20.63%	26.16%	32.43%
4state	14.65%	20.70%	26.76%	33.68%
combined	10.91%	15.60%	20.61%	25.86%

Table 3: WER results on the test set in the presence of 15db SNR additive noise for various lexicon sizes.

ticulatory information, and that the Baum-Welch algorithm has not re-assigned the state meanings during training.

4. DISCUSSION

We plan to extend this work by adding more articulatory knowledge, with rules for phoneme modification that arise as a result of physical limitations and shortcuts in speech production, as was done in [7] (for example, vowel nasalization). Such rules may help speech recognition systems in the presence of strong coarticulation, such as in conversational speech.

We would also like to use the MOCHA database in the training process. We believe it could help perform model initialization, determine better articulatory feature mappings, and more realistic constraints on articulator dynamics.

We have presented results demonstrating the practical usefulness of a HAMM. We accomplished a reduction in model size by 51%, while achieving a reduction in WER of 16-24%. By combining with a standard HMM model, we accomplish a 28-35% WER reduction relative to the HMM model alone, resulting in the lowest WER for PHONEBOOK that we are aware of. In the presence of noise, we improved on recognition over a standard HMM by 23-26%.

ACKNOWLEDGEMENTS

This work was supported by a Packard Foundation Fellowship.

		Measured Feature							**Measured Feature**						
		Jaw	Lip	BF	LH	Tip	Vel	Vce	Jaw	Lip	BF	LH	Tip	Vel	Vce
Estimated Feature	Jaw	.36	.21	-.22	-.29	-.31	.18	.20	.21	.15	-.14	-.18	-.21	.03	.15
	Lip	.14	.36	-.12	-.08	-.06	-.06	-.03	.07	.27	-.08	-.07	-.01	-.11	-.08
	BF	-.17	.15	.22	-.02	.23	-.10	-.12	-.22	.03	.03	.04	.28	.08	-.13
	LH	-.44	-.07	.14	.36	.43	-.19	-.22	-.32	-.01	.05	.23	.31	-.02	-.14
	Tip	-.18	-.11	-.06	.11	.36	.03	-.04	-.06	-.02	.02	.02	.20	.11	.04
	Vel	-.08	-.12	.09	.08	.08	.29	.22	.01	-.06	.10	.06	.02	.23	.28
	Vce	.21	.09	-.09	.00	-.16	.16	.61	.23	.14	-.05	-.08	-.13	.16	.60

		Measured Feature							**Measured Feature**						
		Jaw	Lip	BF	LH	Tip	Vel	Vce	Jaw	Lip	BF	LH	Tip	Vel	Vce
Measured Feature	Jaw	1.0	.40	-.23	-.31	-.62	.24	.35	1.0	.50	.08	-.40	-.65	.01	.33
	Lip	.40	1.0	.09	.08	-.17	.06	.19	.50	1.0	.12	.02	-.18	-.05	.25
	BF	-.23	.09	1.0	.13	.01	-.23	-.14	.08	.12	1.0	.08	-.10	.07	-.08
	LH	-.31	.08	.13	1.0	.45	-.19	.00	-.40	.02	.08	1.0	.55	-.12	-.09
	Tip	-.62	-.17	.01	.45	1.0	-.13	-.27	-.65	-.18	-.10	.55	1.0	.06	-.19
	Vel	.24	.06	-.23	-.19	-.13	1.0	.23	.01	-.05	.07	-.12	.06	1.0	.16
	Vce	.35	.19	-.14	.00	-.27	.23	1.0	.33	.25	-.08	-.09	-.19	.16	1.0

Table 5: Correlations of estimated vs. measured articulator positions of female (upper-left) and male (upper-right) data. Correlations of measured articulator positions vs. themselves in female (lower-left) and male (lower-right) data. Measurements are from MOCHA, estimates are from the pruned HAMM Viterbi path.

Model	**test set**	**75**	**150**	**300**	**600**	**param**
4state	full	1.45%	2.79%	4.04%	5.76%	203k
4state	reduced	1.08%	2.18%	3.31%	5.08%	203k
4state-d1	reduced	1.39%	2.29%	3.48%	4.79%	217k
4state-d2	reduced	1.13%	1.91%	2.86%	4.10%	425k
HAMM	full	2.46%	3.77%	5.47%	7.56%	255k
HAMM	reduced	2.08%	3.25%	4.92%	7.02%	255k

Table 4: Comparison of diphone and non-diphone systems on full and reduced test sets. The reduced test set contains no words with untrained diphones.

REFERENCES

[1] G. Bailly, et al. (1992). "Inversion and Speech Recognition," Signal Processing VI: Proceedings of EUSIPCO-92, vol.1 pp.159-164

[2] C. Bishop (1995), *Neural Networks for Pattern Recognition.* (Oxford University Press)

[3] C. Coker (1976). "A model of articulatory dynamics and control," Proc. IEEE 64, 452-60.

[4] L. Deng and D. Sun (1994). "Phonetic Classification and Recognition Using HMM Representation of Overlapping Articulatory Features for all classes of English sounds," ICASSP, 1994, pp.45-8

[5] S. Dupont, et al. (1997). "Hybrid HMM/ANN systems for training independent tasks: Experiments on PHONEBOOK and related improvements," ICASSP, 1997, pp.1767-70

[6] H.T. Edwards (1997), *Applied Phonetics: The Sounds of American English* second edition. (Singular, San Diego)

[7] K. Erler and G.H. Freeman (1996). "An HMM-based speech recognizer using overlapping articulatory features," J. Acoust. Soc. Am. 100, pp.2500-13

[8] S. Gupta and J. Schroeter (1993). "Pitch-synchronous frame-by-frame and segment-based articulatory analysis by synthesis," J. Acoust. Soc. Am. 94:5, Nov. 1993, pp.2517-30

[9] W.J. Hardcastle and N. Hewlett (eds.) (1999) *Coarticulation. Theory, Data and Techniques*. (Cambridge)

[10] K. Kirchhoff (1998). "Combining articulatory and acoustic information for speech recognition in noisy and reverberant environments", Proceedings of ICSLP, 1998, pp.891-4

[11] B. Logan and P. Moreno (1998). "Factorial HMMs for acoustic modeling," ICASSP, 1998 , pp.813-6

[12] J. Pitrelli, et al. (1995). "PHONEBOOK: A phonetically-rich isolated-word telephone speech database" ICASSP, 1995 pp.101-4

[13] M. Richardson, J. Bilmes, C. Diorio (2000). "Hidden-Articulator Markov Models for Speech Recognition" ASR2000.

[14] L. Saul and M. Jordan (1999). "Mixed Memory Markov models: decomposing complex stochastic processes as mixtures of simpler ones," Machine-Learning 37:1, Oct 1999, pp.75-87

[15] Wrench (2000). "A Multichannel/Multispeaker Articulatory Database for Continuous Speech Recognition Research," Workshop on Phonetics and Phonology in ASR, Saarbruecken, Germany, 2000, to appear.

[16] S. Young. "A Review of Large-vocabulary Continuous-speech Recognition," IEEE Signal Processing Magazine, 13:5, Sept. 1996, pp.45-57