UCLA Working Papers in Phonetics, No. 103, 64-70

Learning phonetic features from waveforms

Ying Lin
(yinglin@ucla.edu)

Abstract

Unsupervised learning of broad phonetic classes by infants was simulated using a statistical mixture model. With the phonetic labels removed, hand-transcribed segments from the TIMIT database were used in model-based clustering to obtain data-driven classes. Simple Hidden Markov Models were chosen to be the components of the mixture, with Mel-Cepstral coefficients as the front-end. The sound classes were found by iteratively partitioning the clusters. The results of running this algorithm on the TIMIT segments suggest that the partitions may be interpreted as gradient acoustic features, and that to some degree, the resulting clusters correspond to knowledge-based phonetic classes. Thus, the clusters may reflect the preliminary phonological categories formed during language learning in early childhood.

1 Introduction

An important change that occurs during early phonological development is that an infant changes from a "universal" perceiver to a language-specific one[7]. It is widely believed that one of the underlying mechanisms is the ability to learn sound prototypes from distributions of sounds[2]. Although unsupervised learning of sound prototypes has been simulated using connectionist models with artificial data, no model is yet available that takes real speech signals as input. The current study is a first step in building a computational model for the human-like learning of sub-lexical units from acoustic signals, using tools from Automatic Speech Recognition and statistical learning. Assuming for this first step that a phone-level segmentation is given, we study the technical problem of using a statistical mixture model to cluster a set of unlabelled acoustic segments. To handle the challenge that acoustic segments are non-stationary and have variable durations, simple Hidden Markov Models (HMM) were chosen to be the components of the mixture, using Mel-cepstral coefficients as the parameterized representation of speech.

2 Method

2.1 Mixture model

The basic intuition behind mixture models is that the observed data may be generated by different sources, each captured by a separate component model. The way to decide which component best accounts for the data is by comparing the data's likelihood given each component. Under the mixture model assumption, the likelihood has the general form:

$$p(d|M) = \sum_{i=1,\cdots,N} p(m_i) \cdot p(d|m_i) \tag{1}$$

where each m_i is often a parametric model that serves as a mixture component of M, and $p(m_i)$ is the prior probability over all components $\{m_1, \cdots, m_N\}$ of M. The intuitive interpretation of $p(m_i)$ is the relative size of the subset of data that is attributed to m_i, and $p(d|m_i)$ corresponds to likelihood of the data given m_i. Another important notion is the posterior probability of each mixture component, defined as:

$$p(m_i|d) = \frac{p(m_i)p(d|m_i)}{\sum_{i=1,\cdots,N} p(m_i) \cdot p(d|m_i)} \tag{2}$$

Intuitively speaking, the posterior probability represents to what extent the data d is explained by component m_i. When this value approaches 1, it means that almost certainly, m_i is responsible for d.

2.2 Learning algorithm

The learning problem of the mixture model is addressed by the well-known Expectation-Maximization algorithm [9]. The EM algorithm iterates over the following two steps until the data likelihood stops increasing:

1. E-step: given current estimates of $\{p(m_i)\}$ and m_i, compute the posterior probabilities using (2);
2. M-step: weight each datum with the posterior $p(m_i|d)$, and update m_i and $p(m_i)$ with the discounted data.

One perspective on this algorithm is provided by some proposals on exemplar-based category learning [4]. E-step can be viewed as determining the membership of a new exemplar with regards to each class using the pre-stored exemplars, while M-step can be viewed as shifting the centers of the exemplar clouds by updating the contribution of each exemplar. The main difference between our model and exemplar-based models is in essence similar to the one between template-based and statistical speech recognition: rather than storing all exemplars and using some template-matching techniques to determine similarity, we assume exemplars are generated by a mixture of models and use likelihood to measure similarity.

2.3 Mixture of HMMs

In principle, any probabilistic model that can be used to approximate (2) for time series data can serve as a component of the mixture. Therefore, the choice of models was not limited to HMM. We chose HMM because it is relatively easy to implement, not because we considered it the best model for acoustic segments.

The main challenge in clustering speech segments was that segments may have different lengths and are not stationary. Rather than mapping all segments to a fixed dimension [5], we used a mixture of HMMs to model the whole segments. The use of HMMs in clustering speech was considered in [3], but the mixture of HMMs was first applied to the clustering of motion data [1].

The algorithm for training a mixture of HMMs involves some minor modifications to the regular Baum-Welsh algorithm [6]. Assuming the output probability of each state is computed from a Gaussian mixture, the E-step includes the following formulae (the use of symbols also follows [6]):

$$p(O^{(s)}|\lambda_m) = \sum_i \alpha_t(i)\beta_t(i) \tag{3}$$

$$\xi_t^{(s,m)}(i,j) = \frac{\alpha_t(i) \cdot a_{ij} b_j(o_{t+1}^{(s)}) \cdot \beta_{t+1}(i)}{p(O^{(s)}|\lambda_m)} \tag{4}$$

$$\gamma_t^{(s,m)}(i) = \frac{\alpha_t(i)\beta_t(i)}{p(O^{(s)}|\lambda_m)} \tag{5}$$

$$\gamma_t^{(s,m)}(i,k) = \gamma_t^{(s,m)}(i) \cdot \frac{c_{i,k}^{(m)} N(o_t^{(s)}, \mu_{i,k}^{(m)}, \Sigma_{i,k}^{(m)})}{\sum_j c_{i,j}^{(m)} N(o_t^{(s)}, \mu_{i,j}^{(m)}, \Sigma_{i,j}^{(m)})} \tag{6}$$

$$p(\lambda_m|O^{(s)}) = \frac{p(\lambda_m)p(O^{(s)}|\lambda_m)}{\sum_j p(\lambda_j)p(O^{(s)}|\lambda_j)} \tag{7}$$

$\alpha_t(i), \beta_t(i)$ are the regular forward and backward probabilities computed from model parameters. a_{ij} are transition probabilities. $b_j(o_t^{(s)})$ are output probabilities. $N(o_t^{(s)}, \mu_{i,k}, \Sigma_{i,k})$ are the Gaussian components in the output probabilities. In (4)(5)(6), the extra subscripts m and s indicate that there is a separate counter for each pair of HMM and observation sequence. (7) calculates the posterior probability.

As mentioned in 2.2, the M-step uses the posterior probability to weigh each sufficient statistics counter in (4)(5)(6), and updates the parameters of a given model using the weighted sum of all counters associated with this model. The formulae include:

$$a_{ij}^{(m)} = \frac{\sum_s p(\lambda_m|O^{(s)}) \sum_t \xi_t^{(s,m)}(i,j)}{\sum_s p(\lambda_m|O^{(s)}) \sum_t \gamma_t^{(s,m)}(i)} \tag{8}$$

$$\mu_{i,k}^{(m)} = \frac{\sum_s p(\lambda_m|O^{(s)}) \sum_t \gamma_t^{(s,m)}(i,k) o_t^{(s)}}{\sum_s p(\lambda_m|O^{(s)})} \tag{9}$$

$$\Sigma_{i,k}^{(m)} = \frac{\sum_s p(\lambda_m|O^{(s)}) \sum_t \gamma_t^{(s,m)}(i,k)(o_t^{(s)} - \mu_i)(o_t^{(s)} - \mu_i)^T}{\sum_s p(\lambda_m|O^{(s)})} \tag{10}$$

$$c_{i,k}^{(m)} = \frac{\sum_s p(\lambda_m|O^{(s)}) \sum_t \gamma_t^{(s,m)}(i,k)}{\sum_j \sum_s p(\lambda_m|O^{(s)}) \sum_t \gamma_t^{(s,m)}(i,j)} \tag{11}$$

$$p(\lambda_m) = \frac{\sum_s p(\lambda_m|O^{(s)})}{\sum_s \sum_j p(\lambda_j|O^{(s)})} \tag{12}$$

(8)(9)(10)(11) updates the corresponding parameters (transition probabilities $a_{ij}^{(m)}$ between states, the means $\mu_{i,k}^{(m)}$, covariances $\Sigma_{i,k}^{(m)}$ and weights of the Gaussian mixture $c_{i,k}^{(m)}$) for each HMM component in the mixture, and (12) updates the prior probability over the mixture components.

Note that running the algorithm for the first time requires an initial estimate for the HMM parameters and for the prior probability. The K-Means algorithm based on the Itakura-Saito distortion [6] was used for such purpose. Using this method, every acoustic segment was mapped to the LPC vector of its centroid spectrum and the initial clustering was done on all the LPC vectors.

K-means:

	1	2	3
ɪ n	757	335	355
ɹ i	243	421	303
ɑ ɹ	86	140	497

HMM mixture:

	1	2	3
ɪ n	1437	5	5
ɹ i	27	900	40
ɑ ɹ	13	41	669

	1	2	3
water	2	58	276
she	230	170	8
ask	137	179	11

	1	2	3
water	0	3	333
she	405	3	0
ask	11	316	0

Table 1: *Comparisons of clustering methods*

Table 1 shows two comparisons of the clustering methods, using 3 diphones and 3 words respectively. Columns $1, 2, 3$ represent the cluster indices. We can see that when the units contain significant dynamics, the HMM mixture achieves a much better separation of different units than the K-Means algorithm.

2.4 Iterative refinement of the mixture model

Due to the complex form of the likelihood function, finding the global maximum in the likelihood space can be very difficult. The heuristic that we used to approximate the global maximum is to start with a small number of clusters, and then split them successively to obtain the desired number of clusters. The criterion for choosing which cluster to split is again based on likelihood.

The intuition of Algorithm 1 is that new categories first emerge from the largest or the most heterogeneous subset of data. Thus it may be viewed as a strategy for inductively learning the sound categories from unlabelled data.

Our clustering experiment was conducted on the manually transcribed TIMIT database. The training of the HMM mixture was implemented by modifying the HTK source code, and the successive splitting

Algorithm 1 Successive cluster splitting

1: Train a mixture of k HMM's
2: **repeat**
3: **for** each cluster C_i **do**
4: Split C_i into n clusters and obtain a new mixture model, record the gain in likelihood
5: **end for**
6: Choose the split that maximally increases the likelihood
7: Retrain the new mixture model on all data
8: **until** stopping condition is satisfied

algorithm was implemented in Matlab. All HMMs are 3-state, left-to-right, with a 2-Gaussian mixture modelling the output distribution of each state. Mel-cepstral coefficients (13) [8] together with the delta features (13) [6] were used as the parameterized representation of speech signals. This representation allowed us to focus on the spectral envelope instead of the speaker information. With the phonetic labels removed, 7166 acoustic segments from 22 speakers in TIMIT were clustered. Starting with 2 clusters, 5 partitions were found. Each partition replaced the old cluster with 2 new clusters, thereby resulting in a total of 6 clusters. The distribution of phonetic labels over the clusters was calculated after each partition and retraining.

3 Results

Figures 1 – 5 illustrate how the phonetic segments are divided into two new clusters at each partitioning step. The phonetic labels use symbols from the TIMIT phonetic alphabet. For each phonetic label, the position of the vertical bar indicates the percentages of the acoustic segments that were assigned to the left and right cluster. For example in Figure 1, the bars corresponding to the voiced interdental fricative "dh" represent the result that 95% of acoustic segments labelled "dh" were assigned to cluster 1 ("obstruent") and 5% were assigned to cluster 2 ("sonorant"). The clusters were named using prefix coding. For example, a parent cluster named 12 was split into daughter clusters 121 and 122. To save space, each figure displays the subset of labels with more than half of the segments falling in the parent cluster. For example, labels included in Figure 3 (cluster 21 and 22) were those that have been mostly assigned to cluster 2 ("sonorants") in Figure 1.

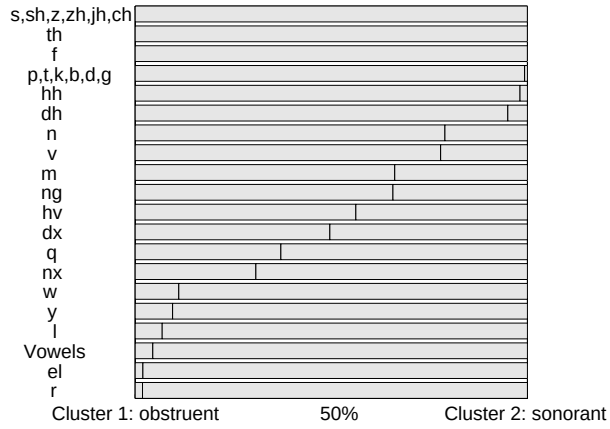

Figure 1. *The first partition*[1]*: [sonorant]*

[1]Some phonetic labels are consolidated for better display.

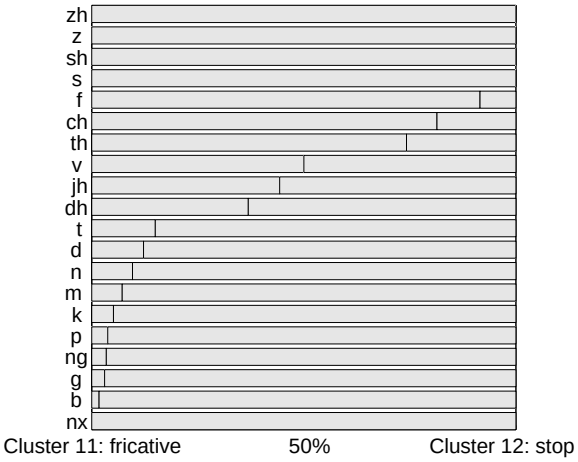

Figure 2. *The second partition of obstruents: [fricative]*

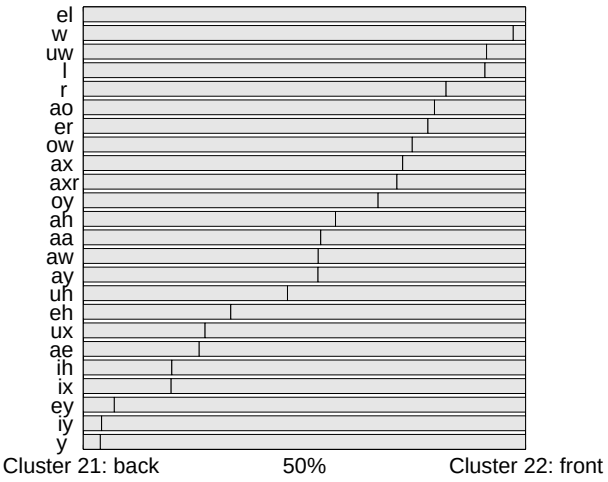

Figure 3. *The third partition of sonorants: [back]*

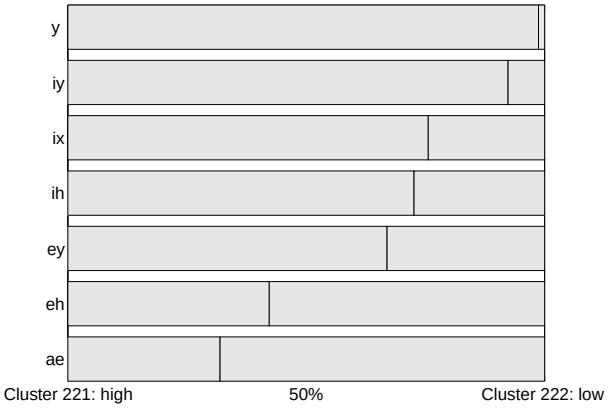

Figure 4. *The fourth partition of front sonorants: [high]*

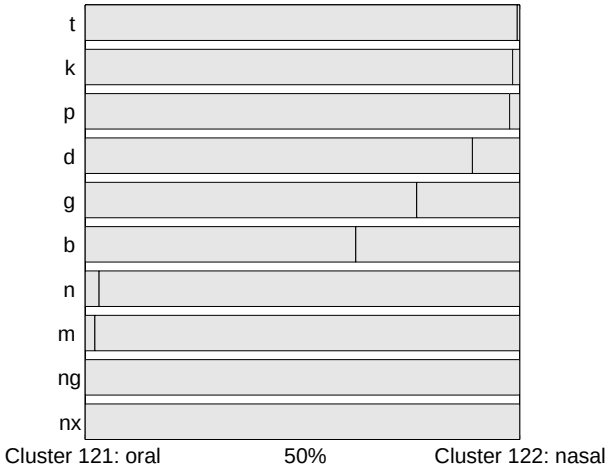

Figure 5. *The fifth partition of stops: [nasal]*

The division of phonetic segments at each split suggests that the splits may be interpreted as gradient, distinctive acoustic features that distinguish two classes of sounds by the general shapes of their spectral envelopes. For convenience, these features were named using linguistic terms. The percentages may depend on the distribution of sounds in the training data set, but they reflect some general patterns of contextual variation in phonetic segments. Take the voiced labiodental fricative [v] as an example. The fact that in continuous speech, [v] is often produced as an approximant without significant frication noise is reflected by the ambiguous status of [v] in Figure 1 and Figure 2. Another example is the distribution of [w],[ɹ],[l] and [l̩]. They all fall into the category of sonorants that have a low F2, which may coincide with a primitive phonetic category in early child language.

To further investigate the nature of these classes, an evaluation was also conducted by creating 6 reference labels for the 6 broad phonetic classes obtained above: *fricative/affricate, plosive, nasal, back sonorant, high front sonorant* and *central sonorant*. These reference labels were completely based on linguistic knowledge. The percentage of the data-driven labels that match the knowledge-based labels was calculated. Moreover, a test set was constructed from 7 speakers from the same TIMIT dialect area. The results are reported in Table 2. Considering that the mixture model was learned in a completely unsupervised manner, its performance on the phone classification task was, as expected, reasonable. The similarity between the training and test set suggests that our results reflect general patterns rather than those specific to the training set.

Data set	Speakers	Phones	Percentage
Train	22	7166	69.17
Test	7	2084	67.61

Table 2: *Percentage of phones that match the knowledge-based reference labels*

4 Discussion and future work

The current study demonstrates the possibility of using statistical ASR tools for the purpose of modelling acquisition of phonetic categories. This work will be extended in two directions. First, instead of using manually-transcribed segments, we would like to segment the word signals and learn sound categories at the same time. Second, we would also like to replace phone-level optimization with lexicon-level optimization, and highlight the connection between lexical growth and sub-lexical units.

5 Acknowledgements

The author would like to thank Pat Keating, Abeer Alwan and Yingnian Wu for their comments.

References

[1] J. Alon et al. "Discovering clusters in motion time-series data," in *Proc. CVPRC*, 2003.

[2] J. Maye, J. F. Werker, and L. Gerken, "Infant sensitivity to distributional information can affect phonetic discrimination," *Cognition*, vol. 3, no. 82, pp. B101–B111, 2002.

[3] B. Raj, R. Singh, and R. Stern, "Automatic generation of subword units for speech recognition systems," *IEEE Trans. Speech and Audio Proc.*, vol. 10, no. 2, pp. 89–99, Feb 2002.

[4] K. Johnson, "Speech perception without speaker normalization: An exemplar model," in *Talker Variability in Speech Processing*, K. Johnson and J. W. Mullennix, ed. 1997

[5] F. Korkmazskiy, B. H. Juang, and F. Soong, "Generalized mixture of HMM's for continuous speech recognition," in *Proc. ICASSP97*, pp. 1443–1446, 1997

[6] L. Rabiner and B. H. Juang, *Fundamentals of Speech Recognition.* Prentice Hall, 1993

[7] J. F. Werker and R. C. Tees, "Cross-language speech perception: Evidence for perceptual reorganization during the first year of life," *Infant Behavior and Development*, no. 7, pp. 49–63, 1984.

[8] S. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentence," *IEEE Trans. ASSP*, vol. 28, pp. 357–366, 1980.

[9] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum liklihood from incomplete data via the EM algorithm," *J. Royal Stat. Soc.*, vol. B, no. 39, pp. 1–38, 1977.