Weak representational bias and the discovery of linguistic categories from speech waveforms

Ying Lin
Department of Linguistics
University of Arizona
yinglin@u.arizona.edu

September 21, 2006

1 Introduction

1.1 Strong and weak representational bias

How does a child start to learn the sound patterns of her native language? There are two diametrically opposed approaches to this problem. According to one approach, the learner always looks for a symbolic representation in input, and such representation is based on a set of universal phonetic features (Chomsky and Halle, 1968). Since phonetic segments – or more intuitively “speech sounds” – are treated as bundles of these features, the universal feature-based representation also implicitly assumes a temporally discrete, segmental input to the learner. In this paper, this perspective will be referred to as strong representational bias. In contrast, the connectionist approach (Elman and Zipser, 1989) claims that the basis of phonological acquisition is the ability to extract statistical information from speech signals. At the level of categories, networks have been trained to simulate the perceptual change experienced by infants after exposure to input distributions (Guenther and Gjaja, 1996). Some believe that phonetic segments are an “emergent” property (Plaut and Kello, 1999) in the learning system, and that universal features offer little insight regarding how phonological acquisition can proceed from acoustic signals. Hence, connectionism illustrates a class of approaches that consider no representation bias as necessary for the learner to accomplish the task.

This paper takes an intermediate standpoint between the two views, which we have named weak representational bias. Intuitively, the current approach treats phonological acquisition as a problem of “breaking the speech code”\(^1\), where the input is raw speech signals, and the “code” is made of abstract representations similar to those of adult language. The emphasis on the “code”, or a highly constrained hypothesis space, is shared by the weak and the strong

\(^1\)This analogy has also been used by other researchers, e.g. recently in (Kuhl, 2004).
representational bias. However, rather than handing the code to the learner, we also stress that the learner needs to make use of statistical information in the input to select from a set of candidates. Ideally, such information need to be extracted directly from the speech signal, instead of from the symbolic representations imposed on the signal. In this respect, the current work shares with connectionism the emphasis on statistical learning. Moreover, we would like the learner to be unsupervised with respect to the abstract representation itself, since under normal learning conditions, such information is unavailable to infants. Currently, neither the strong bias nor the no bias approach has addressed this concern.

How could an unsupervised learner use statistics to discover the discrete structures within continuous signals? Imagine we have exposed a learner to signals such as the one shown in Figure 1, and the learner’s task includes forming phonological categories from these signals, as well as inferring the underlying sequences of phonological categories. Without assuming that the learner comes with a set of phonological units and a method for linking these to their acoustic manifestation, a possible strategy is to hypothesize tentative boundaries between different units, and to group similar sections of the spectrogram as one category. For example, in Figure 1, the category “1” stands for a group of sounds that start with silence, then abruptly changes into a wide-band signal that characterizes noise; while sounds in category “2” share an energy profile with peaks appearing in similar frequency bands. Although the learner may not be informed that category “1” and “2” are officially named “stops” and “vowels” by phoneticians, the lack of such knowledge is not crucial for associating a symbolic representation with each signal.

![Figure 1: Spectrogram of a speech signal for the word “kitty”](image)

Since in some aspects, the problem of phonological acquisition as solved by infants is analogous to “decoding” spectrograms without prior knowledge, the above thought experiment is instrumental for showing how unsupervised learning from signals is possible: because the speech signal changes continuously, some (perhaps rudimentary) knowledge of how the “alphabet” of the system is encoded as acoustic signals is necessary for the learner to map the signal to a sequence of units. On the other hand, different parts of the signal also need to be linked to the underlying “code”, so that the learner would know how each unit in the alphabet manifests in the signal. Although this description of the problem sounds circular, it actually suggests that learning could proceed using an
iterative strategy: first, the learner forms preliminary categories ("alphabet") by clustering different portions of the signal according to its acoustic properties, and encodes these properties in statistical models. Subsequently, improved models of phonological categories help to map the signal to a sequence of units ("code"). The learner thus goes back and forth until settling on a set of acoustic models for categories as well as abstract representations for words.

Figure 2: Flow chart of iterative learning

Figure 2 illustrates three kinds of computations in the iterative learning process. *Segmentation* finds the best way of dividing the speech signals into a sequence of units. The search space consists of all possible sequences formed using the alphabet. Two sources of information are used in the search: one is the acoustic signal itself, the other is the learner’s knowledge of the phonology, encoded in models of phonological categories and of phonotactics. In *clustering*, the learner groups the speech sounds into discrete categories, and uses acoustic models to encode these sounds according to the grouping. The third process takes a set of abstract representations as input, and produces the phonotactic model that summarizes the sequential patterns in the language. Different from the training of connectionist networks, these computations are based on the probability calculus, and each step leads to a better fit of the model to the distribution observed in the data. Starting from the acoustic segmentation of the speech waveform, the iterative learning algorithm eventually converges to a set of abstract representations and acoustic models. In a different domain, similar strategies have also been adopted to the data-driven design of speech units in speech processing systems (Bacchiani, 1999).

1.2 Relationship with previous work

The most important distinction between the strong and weak representational bias is whether the input speech signal is treated as specified in terms of segments and features as in the adult language. Instead of having the input fully specified, weak representation bias can be expressed as the following more general assumptions about input signals:

- The basic units of speech signals are discrete phonological categories.
• Each spoken utterance is combinatorial in terms of a sequence of phonological categories.

Compared to the strong representation bias, the weak bias seems to be more compatible with empirical research. First of all, the strong bias perspective does not account for the way in which infants' knowledge of sound patterns is influenced by experience. Previous research has shown that although infants are born as universal perceivers, they gradually become sensitive to the inventory and sequential patterns of sounds in their language (Werker and Tees, 1984; Jusczyk et al., 1993). To a large extent, such information must be encoded in the input speech signal directed to the infants, yet the strong bias approach makes little use of them. Second, contrary to an early account that attributes the learning of distinctive features to the recognition of minimal pairs (Jakobson, 1941), it has been noted that early words do not include sufficient minimal pairs (Charles-Luce and Luce, 1990; Maye and Gerken, 2000). Moreover, the view that early lexical representations are not segmental also has a long history in child language research (Jusczyk, 1997). In the literature, it is common to find descriptions of children’s lexical representations such as “holistic”, “phonological idioms”, “frozen forms”, “speech formulas” and “unanalyzed wholes”.

How can we connect this “holistic” view of early phonological representation with another popular view that sees adults’ and older children’s lexical representations as based on segments and features (Gerken, Murphy, and Aslin, 1995)? Weak representation bias connects these two seemingly contradictory views of early word representation by highlighting the role of statistical learning. With proper constraints on the hypothesis space, the symbolic representation is a result of learning from acoustic speech data. The direct empirical support for this view comes from two pioneering studies that examine the possibility of learning phonemic categories without access to minimal pairs (Maye and Gerken, 2000; Maye, Werker, and Gerken, 2002). In the experiment, subjects were told that they would learn about sounds in a new language, and were presented with a set of stimuli synthesized from the continuum between /d/ and /(s)t/. The distribution in the stimuli used in the two groups are shown in Figure 3. As their results show, the mono-modal and bi-modal distributions have different effects on phonetic discrimination, thus suggesting that distribution learning alone can lead to the discovery of linguistic categories.

\footnote{For the sources of these terms, see (Leopold, 1947; Ingram, 1979; Peters, 1983; Walley, 1993).}
Based on these findings, we formalize the constraint that the sound inventory of a language consists of discrete categories in a *mixture model*, a kind of probabilistic model based on the explicit assumption that multiple disjoint sources contribute to the variability in the data. Applying probability calculus to a discrete structure, mixture models are a natural candidate for modeling linguistic categories with gradient memberships. Such representational bias is the key to demonstrating the feasibility of learning language-like phonological categories in an unsupervised manner.

Although Maye et al’s study focuses on the problem of learning phonological units from word-initial positions, we are also interested in extending the general idea of phonological acquisition as a type of statistical learning to the more realistic scenario of learning from holistic utterances. In contrast to a popular connectionist view that segmental structure is an emergent property of the network (Elman and Zipser, 1989), we take the opposite position that the combinatorial structure bias must be inherent to the learning model. In the current implementation of the model, such bias is formalized as a Markov chain structure over the sequence of segments, with transition probabilities corresponding to phonotactic patterns. The unsupervised learning of this Markov chain interacts with the phonological categories in an iterative way, as illustrated in Figure 2: the mixture model and the Markov chain-based phonotactic model are used to perform segmentation of holistic words into a sequence of categories, while the result of segmentation is used to update both the mixture and the phonotactic model. Bootstrapped from an acoustic segmentation procedure, the iterative learning algorithm eventually converges to a set of phonological categories as well as symbolic representations of words. As confirmed in our experiment using acoustic speech data, such a weakly constrained model is capable of learning from holistic words rough phonological representations that can be compared to adult language.
The organization of this paper follows from our characterization of the weak representational bias in the learner. First we will introduce the mixture model as a probabilistic method of modeling discrete categories, then we will deal with the problem of learning sequential structure from continuous word signals. Our learning algorithm will be tested in the subsequent experiment, followed by a discussion.

2 Discreteness bias and mixture models

2.1 Finite mixture models

An important step in the iterative learning scheme in Figure 2 is about learning categories: if the learner knows how the signal is divided into speech sounds, is it possible to form phonological categories from instances of speech sounds? As noted above, there is some evidence from Maye and colleagues’ work that shows this feat is possible. In the modeling literature, the problem of identifying discrete categories from unlabeled data has a long history and is usually known as clustering. A popular method for clustering categorical data is by using finite mixture models. The basic philosophy of mixture modeling can be illustrated with the example in Figure 1: the bi-modal distribution is assumed to be formed by “mixing” data from two categories. The mixing is done as follows: first, a coin is flipped to pick a category, then an example is drawn from the distribution specified by that category. The probability of the coin flip is referred to as the prior probability, and it determines the proportion of each category in the data. This proportion, as well as the parameters of the distribution associated with each category, determine the distribution characterized by the mixture model in the following manner: each category contributes to the probability of an example, and the probability that a mixture model assigns to a stimulus is calculated by summing up the individual contribution of each category, weighted by the prior probability. The way in which the model generates a distribution over the space of possible stimuli for the mixture of two categories is illustrated in Figure 4.

![Figure 4: A mixture model for two categories](image-url)
Clearly, Figure 4 rests on an implicit assumption that the learner knows how many categories appear in the data. Although it is still unclear how human learners (such as those in Maye et al’s experiments) solve this problem, our current strategy is to treat how to decide the number of categories (also referred to as the “model selection” problem) as a separate problem and discuss it in Section 4.1. Even with such a simplification, the finite mixture model still captures a crucial aspect of the learning problem: the outcome of the coin flip is not available for the learner, and the unsupervised learning task implies that the categories and the memberships of each example must be inferred at the same time.

The mixture model also allows us to answer questions about generalization, using the language of probability. Suppose that for the example considered above, we are given the two models $M_1$ and $M_2$ that account for the data, as well as the prior weights attributed to each model. Then for a new stimulus, its category membership can be expressed with a pair of numbers between 0 and 1 that add up to one. For example, while the membership of a stimulus can only be either $(0, 1)$ or $(1, 0)$ under a categorical scheme, mixture models allow for category memberships such as $(0.5, 0.5)$ (halfway between $M_1$ and $M_2$) or $(0.3, 0.7)$ (a slight preference towards $M_2$). Those numbers in the membership coding are calculated in a way that bears similarity to the calculation of posterior probability, which takes into account both the sizes of the categories and the probability each category model assigns to the stimulus (see equation (2) in Appendix A for more details). In situations where more than two categories are needed to account for the data, the membership can be extended to a vector of any finite length\(^3\). Hence the mixture-model based categorization also provides a flexible, formal characterization of the ambiguity in categorization problems that involve multiple categories.

### 2.2 Learning parameters of the mixture model

Since a mixture model with a fixed number of parameters corresponds to a probability distribution, learning the model from a distribution implies selecting parameter values based on the data available to the learner. Although there are many ways of fitting the parameters values to the data, a popular choice in the modeling literature states that the resulting model should make the probability (or “likelihood”) of the data as large as possible, if the model explains the distribution represented by the data\(^4\). It is worth noting that learning parameters, in the sense used here, is separate from representational bias. While learning means choosing a particular candidate among a class of models, representational bias constrains the hypothesis space (the possible distributions) in which learning takes place. If the representational bias is inadequate, then the learner may generalize poorly no matter which strategy is used to select a hypothesis, because the true model does not live in the hypothesis space defined by the representational bias.

---

\(^3\)For 3 categories, the membership can take a variety of values, e.g. $(0, 1, 0)$, or $(0.3, 0.2, 0.5)$.

\(^4\)In statistics, this approach is also called “maximum likelihood”.
In computation, the learning of mixture models uses an iterative method called Expectation-Maximization (EM) algorithm (Dempster, Laird, and Rubin, 1977), a technique widely used in probabilistic modeling. The key steps in the EM algorithm have intuitive interpretations in light of the connection between unsupervised and supervised learning:

**E-step (categorization):** for each stimulus, compute its membership with respect to each category from the mixture components and weights. Because the membership takes a gradient form and can be seen as fractions of the stimulus, this step can be seen as assigning “partial” labels to data, or a way of making a “soft” categorization.

**M-step (model updating):** following the maximum likelihood principle, use the partially labeled stimuli to update models of each category as if the learning is supervised. In doing so, the contribution of each stimulus towards the model is weighted by the fraction suggested by the label. For example, a stimulus labeled (0, 1) has no contribution to $M_1$, yet another one labeled (0.5, 0.5) contributes equally to both models. In the meantime, the weights (or the relative size of the category) are also recalculated by congregating the fractions. As the result of M-step, new parameter values for the categories as well as weights will be carried over to the next iteration of the E-step.

The E-step and M-step address two important problems of probabilistic models, namely: inference – assigning tentative labels to the data, and learning – updating parameters of the model. By alternating between inference and learning, the algorithm is able to make gradual improvements, until a local maximum is reached. Detailed discussion of this algorithm can be found in many standard texts on statistical learning\(^5\), and a version of this algorithm is used for the current model.

### 2.3 The mixture components for modeling speech segments

Although the mixture model as a broad concept can be applied to many domains in which the data naturally consist of discrete categories, components for the mixture model must be chosen according to the domain of application. With inappropriate bias, the probabilistic model will poorly characterize the distribution of stimuli. Since the stimuli in the current study consist of acoustic speech signals, a minimal requirement for the mixture component is being able to capture certain characteristics of speech. For this purpose, a number of tools and modeling strategies from automatic speech recognition are integrated in the mixture model. Three such strategies are reflected in the model:  

\(^5\)For example, (Duda, Hart, and Stork, 2000) or (Hastie, Tibshirani, and Friedman, 2002).
Short-time acoustic modeling: although certain subsets of speech sounds can be characterized along one or two phonetic dimensions, acoustic modeling needs to provide a parametrization of a broad range of sounds, considering the fact the input to phonological acquisition is rarely as selective as in the controlled studies. Since little is known about infants’ perceptual representation of speech sounds, acoustic modeling of the current work uses cepstral parameters – a type of smoothed spectral energy profile that is standard in speech processing. In particular, a mel-filter bank analysis is applied prior to the cepstrum calculation to reflect the non-linear frequency resolution of the human ear, and the cepstral parameters are weighted with a window to de-emphasize the contribution of voice quality, such as pitch and other speaker characteristics. The result of acoustic modeling is 13 cepstral coefficients for each short-time analysis window, with the first coefficient characterizing the signal energy.

Dynamic acoustic modeling: the short-time analysis does not capture the spectral change between successive analysis windows. To compromise this weakness, it is also a standard practice to augment the cepstral parameters obtained above with local linear regression coefficients by comparing adjacent analysis windows (Furui, 1986). The result of such analysis is adding another 13 dimensions to the static cepstral vector, resulting in 26 dimensions for each frame of short time-analyzed speech.

Hidden Markov models: a speech sound is generally not stationary, but contains multiple points of acoustic change and has different durations in different contexts. Moreover, a speech sound typically does not have a fixed length, therefore making it difficult to map speech sounds into a space with a fixed dimension. An important tool for modeling this type of data, used extensively in speech recognition systems, is the hidden Markov model (HMM). HMM is essentially a Markov model equipped with extra machinery to handle variability. The data is again assumed to be generated in two steps: first, a state sequence is generated from the underlying/“hidden” Markov chain by following transitions of the chain; then the observed data sequence is generated from the output distribution of each state. In the speech applications, it is common to specify a Gaussian mixture output distribution for each state of the HMM. The present work uses a constrained architecture of HMM as shown in Figure 5, and all output distributions are set to a 2-Gaussian mixture.

Notice that the one-dimensional space for the stops in Figure 3 is rather abstract. The ambient space of the stimuli used in Maye et al.’s experiments has a much higher dimension. Generally speaking, it is an open problem whether low-dimensional phonetic representations can be found for all relevant speech sounds.

Also called “acceleration” or “delta” features.

Although the output distribution is also a mixture, it only models one time slice of speech, and is at a different level than the mixture model of hidden Markov models.
To sum up, components in our finite mixture model are hidden Markov models with a left-to-right structure including within-state transitions. The output probability distributions are assumed to be mixture distributions of 26 dimensions, which includes both the static and dynamic cepstral features. With regard to learning, an EM algorithm is used to learn the parameters of such a mixture model, in a way that is very similar to the description in 2.2. For the equations used in the algorithm, the reader is referred to Appendix D for more details.

2.4 Example: 2-way phonetic class distinction discovered by mixture models

To illustrate the effect of using a mixture model to separate sound categories with different spectral characteristics, we present the result of a clustering experiment in Figure 6. The data is taken from the TIMIT database (Garofolo, 1988), a phonetically transcribed corpus that has been used in much speech research. It consists of read utterances recorded from 630 speakers from 8 major regions of American English, of which the data from the New England dialect region will be used. Because the goal of TIMIT was to train phone-based systems, the reading materials were chosen to be phonetically-balanced, and all data were manually transcribed by expert phoneticians. For this experiment, about 7000 phonetic segments from 22 speakers are used as input to a mixture of two hidden Markov models. The segments are selected from the waveforms according to the expert transcriptions, but no label is used. After 20 iterations of unsupervised learning, each segment is associated with a posterior probability vector, as discussed in Section 2.1, and we identify the segments with the category corresponding to the largest posterior probability.

For each phonetic label in Figure 6, the position of the vertical bar indicates the percentages of the acoustic segments that were assigned to the left and right cluster. The segments are labeled with a phonetic alphabet that is similar to
IPA (see Table 7 in Appendix F for more details). The naming of the clusters can be arbitrary since they are discovered by the learner. But as a mnemonic, the clusters are named as “approximant” and “non-approximant”, based on a subjective interpretation of the members. For example, the bars corresponding to the voiced interdental fricative “dh” represent the result that 95% of acoustic segments labeled “dh” were assigned to cluster “non-approximant”. Finally, some consonants and all the vowels are consolidated into one row for better display.

![Figure 6: The first partition: [approximant]](image)

It can thus be seen that the distributions of the segments fit our expectations of contextual phonetic variation: while the majority of sounds fall into one of the two classes near-categorically, the ones that are well-known to exhibit significant variation have a somewhat intermediate status. An example of such a sound is the labio-dental fricative [v], which is often produced as an approximant in an inter-vocalic environment. Distributions such as the one shown in Figure 6 can be interpreted in terms of a non-categorical version of phonological features (Ladefoged, 2001).

9 Notice another option for naming this feature is to call it [sonorant] and the two classes “obstruent” and “sonorant”. However since nasals are generally considered sonorants, we will try to avoid the confusion by using approximant instead.
3 Implementing the combinatorial bias in the sequence model

3.1 Building sequential structure into words

A finite mixture of hidden Markov models is capable of clustering pre-analyzed instances of speech sounds into acoustically distinct categories. As demonstrated in 2.4, meaningful linguistic categories can be discovered via unsupervised learning. Here we would like to take the work one step further, and apply the mixture model directly to unanalyzed signals corresponding to individual words. However, a completely unconstrained model is unlikely to achieve this task, since it is unclear how mixture models, which are assumed to generate “a bag of sounds”, can possibly produce signals corresponding to words that consist of continuous streams of sounds.

The current proposal for bridging this gap is constraining the word representation to be a sequence of elementary categories. Arguably, the sequential structure is a universal property of the sound structure of any spoken utterance. Although a combinatorial bias has been imposed on possible word representations\(^{10}\), such an approach is much less constrained than the strong representational bias for two reasons: first, as mentioned in Section 2, the learner does not have much prior knowledge about the categories that form the word representations other than the discreteness bias; Second, the learner also does not know what specific categories are needed to represent each word. In fact, the only constraint imposed on the sequential structure is the largest possible number of units per word\(^{11}\), and categories and word representations must be learned at the same time. Therefore, the problem of “breaking the speech code” is made far less trivial than with the strong bias approach.

Our commitment to the combinatorial bias naturally relates to another interesting aspect of phonological acquisition – the phonotactics of a language, i.e. the kind of knowledge that makes an English speaker aware that *blick* is a better word than *bnick*. Although a linguistically motivated approach to phonotactics can be considerably more complex, phonotactics is modeled as transition probabilities between categories\(^{12}\). The advantage of this simplified phonotactic model is mainly the simplicity in computation, since the maximum likelihood estimate of the transition probabilities simply follow from the bigram counts of the sequences.

To connect the sound categories with the sequential structure, we augment the mixture of hidden Markov models with two more sets of parameters that represent phonotactics and segmentation, respectively. The purpose of these parameters is to relate acoustic signal of words to the mixture model, which provides an account of the units: once the learner knows how to break words

---

\(^{10}\) Thus, representations such as connection weights in a network are eliminated.

\(^{11}\) The main motivation for imposing this constraint is purely computational, since extremely short segments lead to slow convergence and undesirable local maxima.

\(^{12}\) Note that the transition probabilities between categories are distinct from the transition probabilities within hidden Markov models, which belong to parameters of the units.
into sequences of units, new values will be assigned to the segmentation parameter, and the probability of words can be calculated from its constituents and the phonotactic probability. Details of this model are presented in Appendix B. An important simplifying assumption underlying the derivation of this model is that once their respective segmentations are known, the word tokens are independent of one another, regardless of whether they are the instances of the same lexical item. Therefore we can imagine the current approach places phonological acquisition at a pre-lexical stage: units and phonotactics are learned only from acoustic signals, without any concept of lexicon.

3.2 Learning in the sequential model

The same maximum likelihood framework as in 2.2 is used to learn the model parameters from a given set of holistic words. Just like the EM algorithm, the learning algorithm is also iterative in nature, motivated by the inter-dependence of the three variables – category, phonotactics and segmentation:

1. Knowledge of phonotactics and phonetic categories affects segmentation. In particular, segmentation is determined jointly from the acoustic score calculated from the unit models and the phonotactic score.

2. Phonotactics needs to be re-estimated from segmentation. When segmentation of each word is known, the resulting unit sequences should be used to derive a better estimate of phonotactics.

3. Once a word has been segmented, then a set of sound instances become available for updating the category models. Thus, the learning problem is reduced to the previous scenario of “bag of sounds”.

The inter-dependence of these variables leads to an iterative learning algorithm, which also consists of three steps, each updating a separate parameter:

1. Find the optimal segmentation of each word, using the category models and phonotactics;

2. Update the phonotactic model with the transition probabilities calculated from the new sequences;

3. Update the category models using the segments resulting from the new segmentation.

Each step increases the likelihood by updating one set of parameters, and iterating all the steps also brings the likelihood to a local maximum. Details of the algorithm are presented in Appendix C, and a simple proof of convergence can be found in (Lin, 2005). Although in the infant literature, no clear conclusion has been reached regarding the time order of the development of categories, phonotactics and segmentation, a computational model such as ours would have to improve its hypothesis in an iterative manner, precisely because of the inter-dependence as described above.
3.3 Bootstrapping with initial acoustic segmentation

In general, iterative learning methods such as the one used in Section 3.2 require a reasonable initial estimate of the parameters, since the starting points partly determine the local maximum to which the algorithm will eventually converge. In our case, a starting point means either a set of initial values for parameters in the mixture model, or an explicit segmentation of all the word signals. Since holistic words are the only available data in the beginning, an initial estimate of the category models is obtained by learning from a set of initial segments. These segments are results of a procedure called acoustic segmentation – a segmentation of the waveform without using any category models.

The goal of acoustic segmentation can be seen as finding acoustic discontinuities, or landmarks\(^{13}\) in the speech signal. Following earlier work in speech recognition (Svendson and Soong, 1987; Bacchiani, 1999), acoustic segmentation is posed as the problem of dividing waveforms into a fixed number of relatively stationary regions. In computation, the solution of this problem can be obtained through dynamic programming, together with an appropriate measure of acoustic change. The requirement that the number of segments needs to be fixed is motivated by computational reasons, since the total cost found by dynamic programming decreases with a higher number of segments. Figure 7 shows the acoustic segmentation of the word “yumyum” (including the silence before and after the word), with the total number of segments set to 4 different values. Details of this algorithm are given in Appendix E.

![Figure 7: Acoustic segmentation of the word “yumyum”. The total number of segments is set to 3,5,6,8](image)

In the experiment reported below, the total number of segments used in the initial acoustic segmentation is set to be equal to the largest number of segments (10) used in the later stages of iterative learning. After acoustic segmentation

\(^{13}\)The original definition of landmarks (Stevens, 1998) refers to locations of change in feature values. But here we are using landmarks to loosely refer to possible segmental boundaries in the signal.
has been performed on each word, the set of acoustic segments is used to train the category models in an unsupervised manner, as described in Section 2.2 on the learning of mixture models. Since acoustic segmentation is merely a way of bootstrapping the category models, we do not make any claims about whether there is a parallel in the speech development of infants. Instead, we remain open to the idea that other mechanisms can provide learners with the knowledge of elementary sound units.

4 Experiment: directly learning segmental representation from wholistic words

The first experiment reported here is based on the same set of stimuli from the TIMIT database as used in the clustering experiment in 2.4. However, an important difference lies in the use of segment boundaries that are included with the original spoken corpus. Unlike the experiment in Section 2.4, the segment boundaries provided in TIMIT are completely ignored in the learning stage of this experiment, and are only used in later comparisons with the results found by the learner. Consequently, the learning algorithm only has 2068 word-level acoustic stimuli\(^{14}\) as input, and proceeds as follows:

1. The acoustic segmentation algorithm in 3.3 is applied on the holistic words to obtain an initial set of segments for each word;
2. Using the initial segments as input, the learning algorithm for discrete mixture models produces an initial set of category models;
3. The sequential learning algorithm in 3.2 realizes the updating of unit models, segmentation and phonotactics.

The last two steps are iterated until the convergence criterion is met, as illustrated in the flow chart of Figure 2.

4.1 Coarse-to-fine search for multiple levels of categories

An important detail in the second step is how to find a desired number of categories, from only the unlabeled initial segments. As an illustration of a simple strategy, one could set these models to random initial values, with the hope that many different random starting points could help avoid very poor local maxima. However, this strategy fails to utilize the fact that speech sounds in human language often form natural classes — phonetic categories with an inherent hierarchical organization. For example, obstruents and sonorants are two basic natural classes. Within obstruents, one can further divide them into continuants (such as fricatives) and non-continuants (such as stops).

\(^{14}\)A few function words are excluded because they were identified with very short duration (< 10ms) in TIMIT transcription.
Another problem we have left un-addressed in 2.2 is that the size of the mixture must be given and fixed throughout the learning. This was necessary in the first step of model development because we were dealing with the problem of searching within a parameter space of fixed dimension. However, it is reasonable to expect the learner to expand the parameter space to consider other hypotheses that include larger numbers of categories, since human languages often utilize sound systems that are rich in the categories that naturally form a hierarchical structure.

To address these problems, we extend the clustering method introduced in 2.2 by incorporating a coarse-to-fine search heuristic that intends to discover new fine-grained categories within coarse-grained ones. In each step, a binary grouping is first carried out within one of the existing categories (initially, 1 category), followed by a round of re-learning, in which membership of each instance is adjusted according to the updated model. After a given class is partitioned into two sub-classes, members of the old class are subsumed under the new classes, thereby increasing the number of clusters by 1. Although the old class is not explicitly retained in the new mixture\(^{15}\), the hierarchical structure is preserved through the order in which the new classes are added to the mixture: the initial values of the new classes are assigned based on parameters of the old class. Therefore classes resulting from the same parent are generally more similar than the non-siblings in the parameter space. As an example, Figure 8 shows the category hierarchy resulting from step 2 of the current experiment:

![Figure 8: Coarse-to-fine category hierarchy](image)

As seen from the figure, the categories are indexed using the path from the root category to the leaf. For example, category 11 and 12 are obtained by

\(^{15}\)Note, however, that it is in principle possible to keep all the parent categories as well as the new ones, though this may result in complications regarding membership of samples.
splitting the stimuli that belong to category 1 (and category 1 is discarded afterwards). After 5 steps of binary splitting of categories, the final model consists of 6 categories, namely those occupying the leaves in Figure 8. The partition of speech sound in terms of these categories resemble the broad phonetic classes based on manners of articulation. Later, Section 4.5 provides a quantitative assessment of such a correspondence.

It should be noted that other strategies also exist for integrating the coarse-to-fine search heuristics with the iterative learning paradigm. For example, instead of finding all six categories in Figure 8 in one step, one may add one category at a time, after each stage of iterative learning has reached convergence. Although such a strategy is more computationally intensive, in practice, no significant difference was found compared to the approach adopted here.

4.2 Convergence

Since each iteration of the learning algorithm increases the likelihood of the data by updating the model parameters, convergence occurs until no significant gain in likelihood is achieved by further iterations. For the current experiment, it takes about 10 iterations for convergence to occur. Because the iterative algorithm is only guaranteed to find a local maximum, multiple runs of the algorithm were conducted after acoustic segmentation is carried out on the word signals\textsuperscript{16}. The likelihood in each iteration, together with the total number of segments, is shown in Figure 9 for one of these runs:

\textsuperscript{16}However, we note that these runs do not return significantly different results.
4.3 Examples of linguistic representations learned by the model

Among the many ways in which our learner can be evaluated, one option is to directly examine the linguistic representation of holistic words returned by the algorithm. After the learning algorithm has converged, we can examine the parameter values in the model by examining the analyses of the word signal implied by these values. In particular, the values of the segmentation parameters are most informative since they allow us to directly inspect the representations of acoustic words. A few examples of such representations are shown in Figure 10. The top panels show the spectrograms of the example words, overlaid with the representations learned by the model. The labels on each segment are the final categories shown in Figure 8. For the purpose of comparison, the bottom panels indicate the TIMIT transcriptions for each word, produced by expert transcribers.
Figure 10: The learned and TIMIT segmental representations for the words “speech” and “features”.
4.4 Evaluation of segmental boundaries

The examples in Figure 10 suggest that although each cluster corresponds to a number of different sounds according to the TIMIT transcription, the time resolution of the clusters generally coincide with the segments identified by expert transcribers. To give an estimate of how well the broad class boundaries are aligned with the TIMIT transcription, precision and recall are calculated based on the best segmentation of each word. In calculating the alignment, an error tolerance of 3 frames\(^\text{17}\) is allowed between the broad class boundaries and the expert-produced transcription. Two sets of data are used in the assessment: training data are the same ones used for learning, while test data are taken from a new set of words that the learner has not seen in the learning stage. The test data are recorded from new speakers from the same dialect area who are not included in the learning data, and also contain a certain number of new words not seen in the learning data. By separating the training and test sets, we are hoping to see whether the knowledge acquired by the learner can be generalized to new words spoken by new speakers. The result, shown in Table 1, suggests that the learned word representations consist of approximately segment-sized units.

<table>
<thead>
<tr>
<th></th>
<th>Precision</th>
<th>Recall</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training</td>
<td>75.6%</td>
<td>81.0%</td>
</tr>
<tr>
<td>Test</td>
<td>78.0%</td>
<td>77.1%</td>
</tr>
</tbody>
</table>

Table 1: Precision and recall on the segment boundary detection task

4.5 Evaluation of the broad classes

Another observation based on Figure 10 is that the clusters correspond to a grouping of the speech sounds based on their spectral similarity, sometimes referred to as broad classes in phonetics and phonology. For example, the cluster labeled 111 roughly correspond to the broad classes of fricative sounds, while the cluster 112 correspond to the plosive sounds. Based on subjective interpretation of the broad classes, a correspondence between these classes and segment labels used in TIMIT is set up manually, as shown in Table 2. IPA equivalents of these segments can be found in Appendix F. It should be noted that such correspondences are made without regard to the specific phonetic contexts where these segments occur, a factor that has major influence on the acoustic properties of speech sounds.

\(^{17}\)which corresponds to about 10 ms since a frame is taken every 3 ms from the speech signal.
Table 2: Benchmark mapping from the broad classes to the TIMIT segments

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>111</td>
<td>fricative (f v th dh s sh z zh hh)</td>
</tr>
<tr>
<td>112</td>
<td>plosive (p t k b d g jh ch q epi)</td>
</tr>
<tr>
<td>12</td>
<td>nasal (em en nx eng ng m n)</td>
</tr>
<tr>
<td>211</td>
<td>central vowels (ae ay ux uh ah ax)</td>
</tr>
<tr>
<td>212</td>
<td>back sonorant (aa aw ao el r l w ow oy er axr)</td>
</tr>
<tr>
<td>22</td>
<td>front high sonorant (iy ih eh ey y ix)</td>
</tr>
</tbody>
</table>

Such correspondences are taken as the basis for constructing a benchmark for evaluating the broad classes. Since the broad classes and the TIMIT segments do not always agree in number, the broad class evaluation is done in a manner similar to continuous speech recognition: for each word, the learned broad class sequence is aligned with the benchmark sequence using a dynamic programming procedure, with the standard weights assigned to deletion, insertion and substitution\(^ {18}\). After the sequences are aligned, the result on the training and test set can be also summarized with the following statistics, a metric that is also borrowed from continuous speech recognition.

\[
\text{correct\%} = \frac{\text{Corr} - \text{Sub} - \text{Del}}{\text{Total}} \\
\text{accurate\%} = \frac{\text{Corr} - \text{Sub} - \text{Del} - \text{Ins}}{\text{Total}}
\]

Here Corr stands for the number of correctly aligned labels, Sub stands for the aligned labels that differ, and Del and Ins stand for the number of deleted and inserted labels after alignment is performed, respectively. These summary statistics are reported in Table 3. Since the alignment of learned and reference label sequences is sensitive to weights used in the dynamic programming procedure, and the benchmark is chosen without properly considering the effect of phonetic contexts, these numbers should be taken as a qualitative, rather than quantitative assessment of the model.

<table>
<thead>
<tr>
<th></th>
<th>Correct%</th>
<th>Accurate%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training</td>
<td>68.3%</td>
<td>49.3%</td>
</tr>
<tr>
<td>Test</td>
<td>62.6%</td>
<td>42.1%</td>
</tr>
</tbody>
</table>

Table 3: Summary statistics of broad class evaluation

There is a noticeable difference between the correct and accuracy scores in Table 3. Although a detailed analysis is difficult, one major factor is the fact that in the learned representation, the closure of the stop and the release are often merged as one unit, as can be seen from the examples in Figure 10. There are other ways of creating benchmarks for evaluating the learned representations, yet they are not explored here.

\(^ {18}\)The set of weights are the same as the ones used in the standard NIST scoring package: 3, 3, and 4 for insertion, deletion and substitution, respectively.
4.6 Comparison with a supervised learner

Due to the lack of a previous study with similar goals, we also compared our result with a supervised learner, which uses not only the same model but also labeled data. This involves not only using the segment boundary information in TIMIT, but explicitly training each of the 6 HMMs with the data labeled according to Table 2. The supervised learner is clearly artificial with respect to the goal of modeling language acquisition, but this type of comparison is useful for us to assess the best possible performance that can be achieved with the model and the contrived evaluation metric used in the previous section.

The supervised learner is trained in the following manner: for each category, an HMM is directly trained\(^\text{19}\) with the TIMIT segments explicitly labeled according to Table 2, and the phonotactic model is learned directly from the TIMIT transcriptions. In the testing stage, a word signal is presented to the learner for analysis, and the same benchmark as in 4.5 are used to evaluate the representation that the learner imposes on the word signal. The results are shown in Table 4:

<table>
<thead>
<tr>
<th></th>
<th>Correct%</th>
<th>Accurate%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Train</td>
<td>75.6%</td>
<td>61.7%</td>
</tr>
<tr>
<td>Test</td>
<td>73.0%</td>
<td>60.8%</td>
</tr>
</tbody>
</table>

Table 4: Result of the supervised broad class identification task

Training the model with completely labeled data does return better results, especially in terms of the accuracy measure. Such difference can be mostly ascribed to the insertion errors caused by the closure and release of the stop segments, where the supervised learner does much better than the unsupervised one. It is also worth reflecting on the lack of superior performance in the supervised task, which may indicate a fundamental limitation of the context-free broad classes. We posit that the lack of clear criteria for defining these knowledge-based classes accounts for a large percentage of the error reported for both tasks.

4.7 Phonotactic parameters learned from the data

Finally, values of phonotactics parameters, defined as the transition probabilities between the six broad classes, are displayed in Table 5. Each number represents the transition probability from the category occupying the row to the category occupying the column. It is difficult to directly assess the phonotactics learned by the model, since the learning data does not contain a sufficiently large number of English words, and unlike in previous work, transition probabilities are calculated with respect to broad phonetic classes rather than individual phonemes. However, to illustrate how this model can be related to the empirical work on the acquisition of phonotactics, we calculated the phonotactic scores of two sets

\(^{19}\)This training uses the standard EM algorithm, as described in (Rabiner, 1989).
of monosyllabic non-words used by Jusczyk and colleagues (Jusczyk, Luce, and Charles-Luce, 1994), based on the results in Table 5. The calculation was done as follows: first, the monosyllabic words used in Experiment 1 of Jusczyk et al’s study were converted to broad class-based representations, using the correspondence presented in Table 2; second, based on the respective broad class-based representations, the phonotactic score of each word was calculated by multiplying the bigram probabilities listed in Table 5. For example, for the word [dIz], its phonotactic score was calculated as $0.136 \times 0.099 = 0.0135$. The result of this calculation is shown in Table 6. An one-tailed t-test shows that the difference between the high probability items and the low probability items is significant ($p < 0.001$).

<table>
<thead>
<tr>
<th>Fricative</th>
<th>Stop</th>
<th>Nasal</th>
<th>Central</th>
<th>Back</th>
<th>Front</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fricative</td>
<td>0.023</td>
<td>0.103</td>
<td>0.042</td>
<td>0.112</td>
<td>0.116</td>
</tr>
<tr>
<td>Stop</td>
<td>0.215</td>
<td>0.056</td>
<td>0.095</td>
<td>0.091</td>
<td>0.141</td>
</tr>
<tr>
<td>Nasal</td>
<td>0.074</td>
<td>0.149</td>
<td>0.023</td>
<td>0.070</td>
<td>0.093</td>
</tr>
<tr>
<td>Central</td>
<td>0.128</td>
<td>0.228</td>
<td>0.206</td>
<td>0.058</td>
<td>0.132</td>
</tr>
<tr>
<td>Back</td>
<td>0.042</td>
<td>0.131</td>
<td>0.116</td>
<td>0.264</td>
<td>0.095</td>
</tr>
<tr>
<td>Front</td>
<td>0.099</td>
<td>0.147</td>
<td>0.232</td>
<td>0.157</td>
<td>0.071</td>
</tr>
</tbody>
</table>

Table 5: Transition probabilities between the 6 broad classes learned from data

<table>
<thead>
<tr>
<th>Phonotactic score</th>
<th>High Prob Group</th>
<th>Low Prob Group</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.025 (0.015)</td>
<td>0.010 (0.005)</td>
</tr>
</tbody>
</table>

Table 6: Phonotactic scores of the monosyllabic items in experiment 1 of Jusczyk et al’s study

Clearly, we cannot make any claims about whether this brings support to our proposed learning mechanism, since there are many other competing theories and ways of assigning phonotactic scores (Bailey and Hahn, 2001) that may be consistent with the results in (Jusczyk, Luce, and Charles-Luce, 1994). However, these results show that although broad class-based representation has not been explored in the infant literature, explanations based on such representations cannot be simply excluded from the discussion.

5 Discussion

“Initial bias vs. statistics” is often thought of as a dichotomy in the discussion of language acquisition. As many have become aware, such a dichotomy prevents us from gaining further insights in the subject for a number of reasons. First, both sides fail to address the important question of how abstract linguistic representations are learned from empirical data. Instead, it was dismissed as a non-question by both sides of discussion: either the learning problem is
regarded as trivial, or the abstract representation is seen as artificial. Second, the dichotomy presumes a statistical learner can be free of initial bias, which is a false assumption itself. Results in statistical learning theory show that no machine can learn an arbitrary set of functions (Vapnik and Chernovenkis, 1971; Blumer et al., 1989). As an instance of non-parametric models, neural nets have their own bias and one can prove when it reaches its limits (Baum and Haussler, 1989). A crucial problem with low-bias models is how to achieve good generalization with a relatively small amount of learning data\(^{20}\). Considering that sensory stimuli (such as speech signals) often live in high dimensions, networks with a special architecture are often necessary (Waibel et al., 1989). It is yet an open question whether there is any connection between the bias in such learners and the kind of representation adopted in linguistic analysis.

Rather than dichotomizing initial bias and statistics, the current study demonstrates how a statistical learner, equipped with appropriate bias, can bridge the gap between statistics and abstract representation. We can summarize the underlying mechanism of such a discovery as two kinds of processes. One is the grouping of similar sounds into classes, the other is the counting of these sound classes from a signal varying with time. Since a fundamental characteristic of speech signals is the lack of cues that separate different speech sounds, grouping and counting must be done together, with the help of processing that unveils the underlying structure of the signals as inferred from the current state of knowledge. The probabilistic model used in the present study can be seen as a marriage of symbolic and probabilistic components, and has been used extensively in modern pattern recognition. Their interaction is best elucidated by Grenander’s mathematical theory of patterns (Grenander, 1996): the algebraic component of the model specifies the regularity – in our case, combinatorial structure of words – while the probabilistic component specifies the inherent variability in the acoustic signal. A probabilistic model with appropriate representational bias, we argue, is the key to the reconciliation of the “holistic” and symbolic views of children’s phonological representations. As shown in our experiment, by relaxing the strong representational bias, the weakly constrained model is capable of learning segmental-like representations of phonological words. Although it is almost intuitive that a learner with discreteness and combinatorial biases can perform such as task, it takes a mathematical model and simulation to show that this strategy is viable, based on the kind of input similar to the ones directed to infants. It remains an open question as to whether such bias is present in non-human learners that are capable of some type of statistical learning.

Before closing, we would like to discuss a few issues regarding how our model can be related to phonological development. First, by focusing on the listener, our study has ignored the role of speech production. This simplification is justified by the fact that perception precedes production in phonological development. It is true that the current model does not have the ability to distinguish

\(^{20}\)This problem has also been related to the bias-variance dilemma in statistics (Geman, Bienenstock, and Doursat, 1991).
certain phonetic classes by place of articulation (e.g. the difference between [t] and [k]), a basic kind of distinction that is believed to be acquired early in acquisition. However, it is worth noting that the approach taken in this thesis can also be applied to articulatory data and be further used to explore the relation between perception and production. Second, we are also aware of the controversy regarding the basic units of speech, and arguments in favor of units other than segments, such as syllable-like units. Regarding this issue, our results show the possibility of reaching the segment level without learning syllables as an intermediate stage. Each holistic word is presented as a waveform, and segment-sized broad classes are hypothesized as the only level of structure that generates the waveform. These broad classes are presumed to be relatively homogeneous in time. Therefore, it is somewhat expected that syllables do not emerge as the results of learning since they are temporally complex, and the simple 3-state hidden Markov model used in our study does not have enough capacity to model these complex units. However, once proper models that better characterize the temporal dynamics within syllables become available, it will also be possible to learn syllable-sized units within the same framework. The third problem – model selection – arises when the number of units need to be decided. In the coarse-to-fine search heuristic used in 4.1, no further division is performed beyond the six categories, because it does give rise to new categories that can be interpreted as phonetically-motivated natural classes. Since there is no limit on how far the division can proceed (in the extreme case, each stimulus would have its own category), several options can be considered, including using model complexity to balance the goodness of fit, Bayesian averaging of different hypotheses, or incorporating some top-down influence such as the lexicon (Lin, 2005). Since there are an enormous number of ways in which categories can arise from the data, it remains to be seen what additional strategies human learners employ in learning of linguistic categories.

We would also like to clarify that since our model is based on fairly simple assumptions (e.g. Markov chains augmented with noisy observations), these models are intended to be a first approximation of the underlying mechanism of phonological acquisition. For example, complex segments, such as stops, are characterized by perceptual cues within the segment as well as ones without the segment\[21\]. By strictly segmenting the acoustic signal as a linear string, our model fails to utilize the shared information encoded within the transition between adjacent segments. Moreover, in order for the model to discover more fine-grained phonological categories, contextual variation due to co-articulation also needs to be treated by the probabilistic model. Even at the level of broad phonetic classes, the boundaries between different categories can be significantly blurred by co-articulation in connected speech\[22\]. None of these problems can be satisfactorily treated by a model that is based on a “beads-on-string” view of speech. Considering these limitations, there are two directions in which the

\[21\] For stops, these cues can include the silence or voicing before the release, the burst, and the formant trajectory at the onset of the vowel.

\[22\] In the experiments reported above, this is further confounded by the lack of an evaluation metric for the categories that emerge from clustering.
current work may be extended. If it is indeed possible for a statistical learner to discover the adult-like linguistic representations from the input signal, then we must further constrain the model so that it more accurately characterizes the possible space of input speech signals. On the other hand, considering the potential differences between infants and adults with regard to speech perception, it is worth exploring whether the current model can be used to formulate hypotheses related to abstract representations used in infant speech perception.

A Formulation of the finite mixture model

The degree to which the model (denoted by $\theta$ hereafter) predicts a given set of data (denoted by $y$) is characterized by a probability function, called the likelihood function. Given a finite mixture model, the likelihood $p(y|\theta)$ can be written in the following form:

$$p(y|\theta) = \lambda_1(\theta)f_1(y|\theta) + \lambda_2(\theta)f_2(y|\theta) + \cdots + \lambda_M(\theta)f_M(y|\theta)$$  (1)

The symbol $\theta$ denotes the collection of unknown parameters in the model, which includes all the categories. The number of mixture components is given by $M$, a fixed integer. The distribution functions $f_i(y|\theta), i = 1, \cdots, M$ each characterize a separate category, or a component of the mixture. They are usually chosen from the same family of distributions. The probability that a sound is drawn from component $m$ is $\lambda_m(\theta)$, which is also part of the model parameter, and is subject to the constraint $\sum_m \lambda_m(\theta) = 1$. Alternatively, one may view $\lambda_m$ as a prior probability, which determines the proportion of data that is generated by the $m$-th component. Seeing $\lambda_m$ as prior probability reflects the assumption that a larger category is more likely to account for the data than a smaller one, given the same likelihood.

An intuitive way of interpreting the probabilistic function (1) is thinking of the mixture model as a device for generating data: first, one of the $M$ categories is picked by throwing a $M$-faced dice with probability $(\lambda_1, \cdots, \lambda_M)$; then a sound is generated from one of the components (which themselves are generative devices) $f_1, \cdots, f_M$ according to the category membership of the sample. This process is then repeated independently until a specified sample size is obtained, thereby producing a “bag of sounds”. Since we have started from a known parameter $\theta$ and obtained a random sample $y$, this particular way of modeling a distribution is often called a “generative” approach in statistical modeling literature. As can be seen from the above example, the specific choice of generative model constrains the range of distribution that can be accounted for by the model.

Once the probability model in (1) is fully specified, a gradient categorization is realized through the following quantity:

$$w^i_y = \frac{\lambda_i(\theta)f_i(y|\theta)}{\sum_{j=1}^M \lambda_j(\theta)f_j(y|\theta)} \propto \lambda_i(\theta)f_i(y|\theta), \quad i = 1, \cdots, M$$  (2)
Here the weight $w_i^y$ is a direct combination of the prior probability $\lambda_i(\theta)$ and the likelihood $f_i(y|\theta)$. Due to the similarity of (2) and the Bayes formula, it is informative to think of $w_i^y$ as a posterior probability\textsuperscript{23}, i.e. the chance that a stimulus $y$ comes from each of the model in the mixture. We can represent the contribution of each component towards the stimulus $y$ as a vector $(w_1^y, \ldots, w_M^y)$, and $\sum_i w_i^y = 1$. This vector representation of categorization reflects the fact that a stimulus can be ambiguous between different categories. When one of $w_i^y$, $i = 1, \ldots, M$ is equal to 1 and the rest go to zero, then this gradient representation is equivalent to a categorical one.

\section*{B Derivation of the sequential model}

In this section, we extend the mixture of hidden Markov models in the previous section with two more sets of parameters, and consider a new probabilistic model with the following objective function:

$$p(W|U, P, S)$$  \hfill (3)

Here $W = \{w_1, w_2, \ldots, w_n\}$ stands for the set of all unanalyzed word tokens, obtained in the form of sequences of acoustic parameters; $U$ stands for parameters of category models; $P$ stands for parameters of phonotactics – a transition matrix between categories; and finally $S = \{s_1, \ldots, s_n\}$ stands for segmentation for each word being considered. As a variable, segmentation not only specifies the boundary of each segment, but also the unit label for each segment in the given word. Given the basic setup of the model, the next step is to apply the independence assumption, and express (3) as follows:

$$p(W|U, P, S) = p(w_1|U, P, s_1)p(w_2|U, P, s_2) \cdots p(w_n|U, P, s_n)$$

$$\log p(W|U, P, S) = \sum_k \log p(w_k|U, P, s_k)$$  \hfill (4)

The conditional independence assumption states that the word tokens $w_1, \ldots, w_n$ are assumed to be independent of one another, given their respective segmentation. The basis of this conditional independence assumption is our focus on the pre-lexical stage of phonological acquisition, since no concept of lexicon is invoked by the model.

The most straightforward approach to optimizing the function (4) is performing maximization in each coordinate, thereby giving rise to the following three steps:

$$U \leftarrow \arg \max_U \{\log p(W|U, P, S)\}$$  \hfill (5)

$$P \leftarrow \arg \max_P \{\log p(W|U, P, S)\}$$  \hfill (6)

$$S \leftarrow \arg \max_S \{\log p(W|U, P, S)\}$$  \hfill (7)

\textsuperscript{23}A better working definition of $w_i^y$ is the “expected” or average fraction of the stimulus $y$ as contributed by the $i$-th model. This is an important part in the discussion of the EM algorithm.
Since each step increases the likelihood, it is clear that iterating these three steps will bring the likelihood to a local maximum.

**Solution of (5):** here we need another independence assumption: given a segmentation of a word and models of categories, the probability of the word is a product of the component sounds, multiplied with the transition probability of the phonotactic Markov chain. In terms of log likelihood, this means each term in (4) can be written as:

$$
\log p(w_k|U, P, s_k) = \sum_j \log p(\omega_j^{(k)}|U, P, s_k) + \sum_j \log a_{q_j^{(k)}, q_{j+1}^{(k)}}
$$

(8)

$$\omega_j^{(k)}, j = 1, \ldots, n_k$$ represent the sequence of sounds specified by the segmentation $$s_k$$; $$q_j^{(k)} \rightarrow q_{j+1}^{(k)}$$ represents a transition within word $$w_k$$, from the state occupied by the j-th sound to the (j+1)-th sound in the phonotactic model; $$a_{q,q'}$$ represents the transition probability from state $$q$$ to $$q'$$ in the phonotactic model $$P$$. When $$P$$ is fixed, and $$s_k$$ is also fixed for each $$w_k$$, (5) is equivalent to just maximizing:

$$
\sum_k \sum_j \log p(\omega_j^{(k)}|U, P, S_k) = \sum_k \sum_j \log p(\omega_j^{(k)}|U)
$$

(9)

The equality holds given the assumption that the sounds are individually generated from category models. In order to maximize (9), we simply have to pool all the segments, treat them as conditionally independent given the model parameters, and fit the category model $$U$$ to the set of segments. Appendix D describes the necessary equations for implementing this step.

**Solution of (6):** recall that $$P = \{a_{q,q'} : q, q' \in Q_p\}$$, collecting the second term (since the first term is independent of $$P$$) in (8):

$$
\arg \max_{P} \{P(W|U, P, S)\} = \arg \max_{P} \sum_k \sum_j \log a_{q_j^{(k)}, q_{j+1}^{(k)}}
$$

(10)

(10) shows that the solution to (6) is obtained by counting and normalizing the observed transitions of the Markov chain.

**Solution of (7):** Because of the independence assumption described in (4), the solution to (7) can be obtained by individually solving for $$s_k$$ from:

$$
\arg \max_{s_k} \{p(w_k|U, P, s_k)\}
$$

(11)

i.e. the desired $$s_k$$ is the maximum likelihood segmentation of each word $$w_k$$. Here $$U$$ and $$P$$ are analogous to the “acoustic model” and “language model” in speech recognition systems, and the well-known Viterbi algorithm can be used to decode the word signal into a sequence of categories by combining the acoustic score and the phonotactic score.
C General algorithm

The discussion presented in Section 3.2 can be stated formally as the following algorithm, which is used in the training of the model:

**Algorithm 1** Training

**Require:** a set of acoustic words $W = \{w_1, \cdots, w_N\}$; total number of categories $M > 0$; $\epsilon > 0$

1: for each $k = 1, \cdots, N$ do
2: $(s_k, \{\omega_j^{(k)}\}) \leftarrow \text{acoustic\_segment}(w_k)$
3: end for
4: $U^{(0)} \leftarrow \text{clustering\_EM}(\{\omega_j^{(k)}\})$
5: initialize $P^{(0)}$: $a_{q,q'} = \frac{1}{M}, \forall q, q' \in Q$
6: $t \leftarrow 0$
7: repeat
8: for each $k = 1, \cdots, N$ do
9: $(s_k, \{\omega_j^{(k)}\}, \{q_j^{(k)}\}) \leftarrow \text{viterbi}(w_k, U^{(t)}, P^{(t)})$
10: end for
11: $P^{(t)} \leftarrow \text{ML\_estimate}(\{q_j^{(k)}\})$
12: $U^{(t+1)} \leftarrow \text{clustering\_EM}(\{\omega_j^{(k)}\})$
13: $t \leftarrow t + 1$
14: until $\log p(W|U^{(t+1)}, P^{(t+1)}, S^{(t+1)}) - \log p(W|U^{(t)}, P^{(t)}, S^{(t)}) < \epsilon$

The algorithm iterates the coordinate-wise ascent procedure on each of the parameter coordinates: $U$, $P$, and $S$, until the likelihood gain is less than the threshold $\epsilon$. The subroutine clustering\_EM is also iterative, and it implements the EM algorithm for training the mixture of HMMs, described in Appendix D.

D The EM learning algorithm for a mixture of HMMs

The notation used here follows $O^{(s)} = o_1^{(s)}, o_2^{(s)}, \cdots, o_T^{(s)}$, where $s$ is the sample index, and $T$ is the the length or the number of frames of the speech segment. The mixture model is parameterized by the following set of variable tuples:

$\{(\lambda_m, (a_{ij})^{(m)}), (\mu_{i,k})^{(m)}, (\Sigma_{i,k})^{(m)}, (c_{i,k})^{(m)}): m = 1, \cdots, M\}$, $A_{mn}$.

**E-step:**

$$
\xi_t^{(s,m)}(i,j) = \frac{\alpha_t^{(s,m)}(i) \cdot a_{ij}^{(m)}(i) \cdot b_j^{(m)}(o_t^{(s)}) \cdot \beta_t^{(s,m)}(i)}{p(O^{(s)}|\theta_m)} \quad (12)
$$

$$
\gamma_t^{(s,m)}(i) = \frac{\alpha_t^{(s,m)}(i) \beta_t^{(s,m)}(i)}{p(O^{(s)}|\theta_m)} \quad (13)
$$
\begin{align}
\gamma_t^{(s,m)}(i,k) &= \gamma_t^{(s,m)}(i) \cdot \frac{c_t^{(m)} N(o_t^{(s)}, \mu_{i,k}, \Sigma_{i,k})}{\sum_j c_t^{(m)} N(o_t^{(s)}, \mu_{i,j}, \Sigma_{i,j})} \\
\omega_s^m &= E(O^{(s)}, \theta) \gamma_t^{(s,m)} = \frac{p(\theta_m) p(O^{(s)}|\theta_m)}{\sum_j p(\theta_j) p(O^{(s)}|\theta_j)}
\end{align}

Some explanations of the notations: $\gamma_i^{(s,m)}$ is the likelihood of sequence $O^{(s)}$ given mixture component $\theta_m$. $a_t^{(m)}$ are transition probabilities associated with model parameter $\theta_m$. $b_{ij}(\cdot)$ are output probabilities associated with model parameter $\theta_m$. $N(o_t^{(s)}, \mu_{i,k}, \Sigma_{i,k})$ are the likelihood functions of the normal components in the output mixture distribution $b_{ij}(\cdot)$.

In (12)(13)(14), the extra subscripts $m$ and $s$ indicate that there is a separate counter for each pair of HMM and observation sequence. (12) calculates the average number of transitions from state $i$ to state $j$ in model $\theta_m$ when sequence $O^{(s)}$ is observed. (13) calculates the “soft” segmentation of frame $t$ to state $i$ of model $\theta_m$ when sequence $O^{(s)}$ is observed. Finally, there are two steps that compute the “fractions”: (14) calculates the fractions of individual frames in $O^{(s)}$ as assigned to each component in the normal mixture in state $i$ of model $\theta_m$; (15) calculates the fractions of sequences as assigned to each component in the HMM mixture. Notice that we have two different mixing mechanisms – one on the frame level, the other on the sequence level. It is the flexibility of the mixture model that allows them to be combined in one model.

**M-step:** Due to the normality assumption, the solution to the M-step can be expressed as a series of weighted-sums.

\begin{align}
a_{ij}^{(m)} &= \frac{\sum_s w_s^m \sum_t \xi_t^{(s,m)}(i,j)}{\sum_s w_s^m \sum_t \gamma_t^{(s,m)}(i)} \\
\mu_{i,k}^{(m)} &= \frac{\sum_s w_s^m \sum_t \gamma_t^{(s,m)}(i,k) o_t^{(s)}}{\sum_s w_s^m \sum_t \gamma_t^{(s,m)}(i,k)} \\
\Sigma_{i,k}^{(m)} &= \frac{\sum_s w_s^m \sum_t \gamma_t^{(s,m)}(i,k) (o_t^{(s)} - \mu_s)(o_t^{(s)} - \mu_s)^T}{\sum_s w_s^m \sum_t \gamma_t^{(s,m)}(i,k)} \\
c_{ij}^{(m)} &= \frac{\sum_s w_s^m \sum_t \gamma_t^{(s,m)}(i,j)}{\sum_j \sum_s w_s^m \sum_t \gamma_t^{(s,m)}(i,j)} \\
\lambda_m(\theta) &= \frac{\sum_s w_s^m}{\sum_s \sum_j w_s^j}
\end{align}

---

24The reader is referred to standard tutorials, e.g. (Rabiner, 1989), for more details about the definitions of forward and backward probabilities.
The M-step further illustrates the idea outlined in 2.2: the “fractions” of individual frames are used to weight the sufficient statistics in (17) and (18); in addition, the “fractions” of sequences are used to weight the sufficient statistics in (12)(13)(14). The parameters of a given model are then updated using the weighted sum of all counters associated with this model. Specifically, (16) updates the transition probabilities \( a_{ij}^{(m)} \) between states; (17) updates the means \( \mu_{i,k}^{(m)} \) in the normal mixtures; (18) updates the covariance matrices \( \Sigma_{i,k}^{(m)} \) in the normal mixtures; and (19) updates the mixing priors of the normal mixture \( c_{i,k}^{(m)} \). Finally, on the level of the HMM mixture, (20) updates the mixing priors of the mixture components.

E Acoustic segmentation and the dynamic programming solution

When the number of segments \( M \) is fixed, acoustic segmentation can be formalized as another optimization problem:

Given \( 1, \cdots, N \) frames of speech and \( M > 1 \), find \( s_0 < s_1 < \cdots < s_M \) subject to constraints \( s_0 = 1, s_M = N \), such that

\[
\sum_{i=0}^{M-1} d(X_{s_i,s_{i+1}})
\]

is minimized, where \( d(X_{s,t}) \) is a function that measures the cost of the segment \( X_{s,t} = (x_s, x_{s+1}, \cdots, x_t) \).

\( M_1 \) – the number of segments in the word – serves as a free parameter. When the value of \( M_1 \) is fixed, a solution to this problem can be obtained by applying a dynamic programming strategy: let \( D(m, n) \) be the minimal cost for \( n \) frames to be divided into \( m \) segments, then:

\[
D(1, n) = d(X_{1,n}), n = 1, \cdots, N \\
D(m, n) = \min_{m < t \leq n} (D(m-1, t) + d(X_{t,n})), m > 1, n = 1, \cdots, N
\]  

(21)

The dynamic programming formulation has been used in previous work (Svendson and Soong, 1987; Bacchiani, 1999).

The segment boundaries are found by backtracking from the last boundary that minimizes \( D(M, N) \). In Svendson and Soong (1987), \( d \) is chosen to be the Itakura-Saito distortion (Itakura, 1975), which has connections with an all-pole model of speech production. However, since it is expensive to compute the Itakura-Saito distortion, we followed a simpler approach and let \( d \) be the following function (Bacchiani, 1999):

\[
d(X_{s,t}) = \max_{\mu} (\log(p(x_s, \cdots, x_t) | \mu, \Sigma_0)), \text{ where } \Sigma_0 = \text{cov}(X_{1,N});
\]

(22)

where \( \Sigma_0 \) is the diagonal covariance matrix calculated using the whole sequence of data \( X_{1,N} \). \((\mu, \Sigma_0)\) are the parameters of each Gaussian distribution
over the MFCC features, where \( \mu \) is unknown.\[ \max_{\mu} \log(p(x_s, \cdots, x_t | \mu, \Sigma_0)) \]
can be calculated using \( \mu = \hat{\mu} \), the maximum likelihood estimate of \( \mu \) from the sequence \( X_{s:t} \). In practice this produces similar results to the metric using the Itakura-Saito distortion.

F Correspondence between the TIMIT alphabet and IPA

The transcriptions of the TIMIT database are based on the ARPABET, a phonetic alphabet proposed for speech recognition purposes. Symbols of the alphabet roughly correspond to phonemes of English, although some notable allophones are also included. The correspondence of ARPABET and the International Phonetic Alphabet is shown in the table below. Six symbols: \( \text{bc1}, \text{dc1}, \text{gc1}, \text{pc1}, \text{tc1}, \text{kc1} \) represent the closure part of a stop or affricate and therefore do not have any IPA equivalents.
<table>
<thead>
<tr>
<th>TIMIT</th>
<th>IPA</th>
<th>Example</th>
<th>TIMIT</th>
<th>IPA</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>b</td>
<td>b</td>
<td>bee</td>
<td>hv</td>
<td>h</td>
<td>ahead</td>
</tr>
<tr>
<td>d</td>
<td>d</td>
<td>day</td>
<td>l</td>
<td>l</td>
<td>lay</td>
</tr>
<tr>
<td>g</td>
<td>g</td>
<td>gay</td>
<td>r</td>
<td>r</td>
<td>ray</td>
</tr>
<tr>
<td>p</td>
<td>p</td>
<td>pea</td>
<td>w</td>
<td>w</td>
<td>way</td>
</tr>
<tr>
<td>t</td>
<td>t</td>
<td>tea</td>
<td>y</td>
<td>j</td>
<td>yacht</td>
</tr>
<tr>
<td>k</td>
<td>k</td>
<td>key</td>
<td>el</td>
<td>l</td>
<td>bottle</td>
</tr>
<tr>
<td>dx</td>
<td>r</td>
<td>muddy</td>
<td>iy</td>
<td>i</td>
<td>beet</td>
</tr>
<tr>
<td>q</td>
<td>?</td>
<td>bat</td>
<td>ih</td>
<td>i</td>
<td>bit</td>
</tr>
<tr>
<td>jh</td>
<td>dʒ</td>
<td>joke</td>
<td>ch</td>
<td>ɛ</td>
<td>bet</td>
</tr>
<tr>
<td>ch</td>
<td>tj</td>
<td>choke</td>
<td>ey</td>
<td>ɛ</td>
<td>bait</td>
</tr>
<tr>
<td>s</td>
<td>s</td>
<td>sea</td>
<td>ae</td>
<td>æ</td>
<td>bat</td>
</tr>
<tr>
<td>sh</td>
<td>ʃ</td>
<td>she</td>
<td>aa</td>
<td>ɑ</td>
<td>bott</td>
</tr>
<tr>
<td>z</td>
<td>z</td>
<td>zone</td>
<td>aw</td>
<td>ɔo</td>
<td>bout</td>
</tr>
<tr>
<td>zh</td>
<td>ʒ</td>
<td>azure</td>
<td>ay</td>
<td>ɑ</td>
<td>bite</td>
</tr>
<tr>
<td>f</td>
<td>f</td>
<td>fin</td>
<td>ah</td>
<td>ʌ</td>
<td>but</td>
</tr>
<tr>
<td>th</td>
<td>θ</td>
<td>thin</td>
<td>ao</td>
<td>ə</td>
<td>bought</td>
</tr>
<tr>
<td>v</td>
<td>v</td>
<td>van</td>
<td>oy</td>
<td>ɔr</td>
<td>boy</td>
</tr>
<tr>
<td>dh</td>
<td>ð</td>
<td>then</td>
<td>ow</td>
<td>ɔ</td>
<td>boat</td>
</tr>
<tr>
<td>m</td>
<td>m</td>
<td>mom</td>
<td>uh</td>
<td>ʌ</td>
<td>book</td>
</tr>
<tr>
<td>n</td>
<td>n</td>
<td>noon</td>
<td>uw</td>
<td>u</td>
<td>boot</td>
</tr>
<tr>
<td>ng</td>
<td>ŋ</td>
<td>sing</td>
<td>ux</td>
<td>ʊ</td>
<td>toot</td>
</tr>
<tr>
<td>em</td>
<td>m</td>
<td>bottom</td>
<td>er</td>
<td>ɛ</td>
<td>bird</td>
</tr>
<tr>
<td>en</td>
<td>ə</td>
<td>button</td>
<td>ax</td>
<td>ɔ</td>
<td>about</td>
</tr>
<tr>
<td>eng</td>
<td>ŋ</td>
<td>washington</td>
<td>ix</td>
<td>ɪ</td>
<td>debit</td>
</tr>
<tr>
<td>nx</td>
<td>r</td>
<td>winner</td>
<td>axr</td>
<td>r</td>
<td>butter</td>
</tr>
<tr>
<td>hh</td>
<td>h</td>
<td>hay</td>
<td>ax-h</td>
<td>ə</td>
<td>suspect</td>
</tr>
</tbody>
</table>

Table 7: TIMIT alphabet-IPA correspondence table

References


33


