Parametric Trajectory Mixtures for LVCSR

Man-hung Siu, Rukmini Iyer, Herbert Gish and Carl Quillen

BBN Systems and Technologies
msiu@bbn.com

ABSTRACT

Parametric trajectory models explicitly represent the temporal evolution of the speech features as a Gaussian process with time-varying parameters. HMMs are a special case of such models, one in which the trajectory constraints in the speech segment are ignored by the assumption of conditional independence across frames within the segment. In this paper, we investigate in detail some extensions to our trajectory modeling approach aimed at improving LVCSR performance: (i) improved modeling of mixtures of trajectories via better initialization, (ii) modeling of context dependence, and (iii) improved segment boundaries by means of search. We will present results in terms of both phone classification and recognition accuracy on the Switchboard corpus.

1. Introduction

One limitation of the hidden Markov models, which are the most widely used models to represent speech, is the modeling assumption that features are conditionally independent given the state sequence. In our previous paper [1], we proposed the parametric trajectory model which exploits the time dependence of speech frames by representing the speech features of a speech segment as Gaussian mixtures with time-varying parameters. We have shown the effectiveness of this model in a vowel classification task on the Timit database. The same approach can also be applied for large vocabulary continuous speech recognition (LVCSR).

In this paper, we investigate in detail three extensions of the mixture trajectory model approach aimed at improving LVCSR performance: (i) improved modeling of mixtures of trajectories, (ii) modeling of context-dependent phones, and (iii) improved segment boundaries via a dynamic programming search strategy within the n-best rescoring framework [2].

2. Polynomial Trajectory Model

Given a speech segment with a duration of N frames, where each frame is represented by a D dimensional feature vector, the segment can be expressed in matrix notation as:

$$\mathbf{C} = \begin{bmatrix} c_{1,1} & \cdots & c_{1,D} \\ c_{2,1} & \cdots & c_{2,D} \\ \vdots & & \vdots \\ c_{N,1} & \cdots & c_{N,D} \end{bmatrix} = \begin{bmatrix} \underline{\mathbf{C}}_1 & \cdots & \underline{\mathbf{C}}_D \end{bmatrix} \tag{1}$$

and modeled, as:

$$\mathbf{C} = \mathbf{ZB} + \mathbf{E} \tag{2}$$

where $\mathbf{Z}$ is a $N \times R$ design matrix that specifies the type of model to use, $\mathbf{B}$ is a $R \times D$ trajectory parameter matrix, and $\mathbf{E}$ is a residual error matrix. R is the number of parameters in the trajectory model: $R = 1$ for constant, $R = 2$ for linear, and $R = 3$ for quadratic trajectories.

Given the segment model in Equation 2, the next step is to solve for the model parameters, which we can do on each phone separately. Assuming that the errors are independent and identically distributed (normal with covariance Σ), the Maximum Likelihood (ML) estimate of the trajectory parameter matrix, $\hat{\mathbf{B}}_k$, is given by the linear least squares estimate:

$$\hat{\mathbf{B}}_k = \left[\mathbf{Z}_k'\mathbf{Z}_k\right]^{-1}\mathbf{Z}_k'\mathbf{C}_k, \tag{3}$$

for a segment k with data matrix $\mathbf{C}_k$, and design matrix $\mathbf{Z}_k$.

With $\hat{\mathbf{B}}_k$ estimated, the residual error covariance matrix for the segment, $\hat{\Sigma}_k$, is given by:

$$\hat{\Sigma}_k = \frac{\hat{\mathbf{E}}_k'\hat{\mathbf{E}}_k}{N_k} = \frac{\left(\mathbf{C}_k - \mathbf{Z}_k\hat{\mathbf{B}}_k\right)'\left(\mathbf{C}_k - \mathbf{Z}_k\hat{\mathbf{B}}_k\right)}{N_k}, \tag{4}$$

where N_k is the number of frames in segment k.

The likelihood of an observed segment k, $L(\hat{\mathbf{B}}_k, \hat{\Sigma}_k|\mathbf{B}_m, \Sigma_m)$ with estimated trajectory mean $\hat{B}_k$ and covariance $\hat{\Sigma}_k$ given the model mean B_m and model covariance Σ_m can be expressed as:

$$\begin{aligned} L(\hat{\mathbf{B}}_k, \hat{\Sigma}_k|\mathbf{B}_m, \Sigma_m) \;=\; \mathrm{l}(k|m) \;=& \\ (2\pi)^{-\frac{DN_k}{2}}\,|\Sigma_m|^{-\frac{N_k}{2}} \cdot \exp\left(-\frac{N_k}{2}\mathrm{tr}\left[\Sigma_m^{-1}\hat{\Sigma}_k\right]\right) \cdot& \\ \exp\left(-\frac{1}{2}\mathrm{tr}\left[\mathbf{Z}_k(\hat{\mathbf{B}}_k - \mathbf{B}_m)\Sigma_m^{-1}(\hat{\mathbf{B}}_k - \mathbf{B}_m)'\mathbf{Z}_k'\right]\right).& \end{aligned} \tag{5}$$

The above formulation can be extended to estimate parameters of an M-component mixture model. These trajectory parameters include the means and covariance of the mixture components $\hat{B}_m$, $\hat{\Sigma}_m$ and the mixture weights denoted by $p(m)$ for $1 \leq m \leq M$. The likelihood of a segment k, $1 \leq k \leq K$, in a given set of K segments, can be expressed as:

$$L(\hat{\mathbf{B}}_k, \hat{\Sigma}_k) = l(k) = \sum_m^M p(m)L(\hat{\mathbf{B}}_k, \hat{\Sigma}_k|\mathbf{B}_m, \Sigma_m).$$

The ML solution can be found using the Expectation-Maximization algorithm (EM) resulting in the following reestimating equations.

1. Prior probability for model m:
$$p^i(m) = \frac{1}{K}\sum_{k=1}^{K} p^i(m|k) \tag{6}$$
2. Trajectory parameter for model m:
$$\mathbf{B}_m^i = \left[\sum_{k=1}^{K} p^i(m|k)\mathbf{Z}_k'\mathbf{Z}_k\right]^{-1} \left[\sum_{k=1}^{K} p^i(m|k)\mathbf{Z}_k'\mathbf{Z}_k\hat{\mathbf{B}}_k\right] \tag{7}$$
3. Covariance matrix for model m:
$$\Sigma_m^i = \frac{\sum_{k=1}^{K} p^i(m|k)\left(\mathbf{C}_k - \mathbf{Z}_k\mathbf{B}_m^i\right)'\left(\mathbf{C}_k - \mathbf{Z}_k\mathbf{B}_m^i\right)}{\sum_{k=1}^{K} p^i(m|k)N_k} \tag{8}$$
4. Likelihood $l^i(k|m)$ using Equation 5
5. Posterior probability of the model given segment.
$$p^{i+1}(m|k) = \frac{\mathrm{l}^i(k|m)p^i(m)}{\sum_{j=1}^{M}\mathrm{l}^i(k|j)p^i(j)} \tag{9}$$

3. Improvements for LVCSR

We investigated three different areas to improve the trajectory models for LVCSR. First, we compared three different clustering methods which can be used to initialize the mixture models. Second, we extended the model to represent context-dependent phones. Third, we developed a search algorithm to improve phonetic segment boundaries in both training and recognition.

3.1. Initializing the Mixtures

Good initial models are essential to the estimation of mixtures. In our previous paper [3], we described the agglomerative clustering approach where segments are clustered based on pair-wise likelihood ratio distances between segments. Computing the pair-wise distances is expensive ($O(N^2)$ where N is the number of segments for a phone). This limits its extension to large amount of training data. It is also difficult to prune the dendrogram to form clusters, for which heuristic thresholds are required.

In this paper, we introduce two approaches that are variations of the k-means algorithm and compare them with the agglomerative approach. The k-means algorithm can be described by its three major steps: 1) initial clustering, 2) estimation of centroids, and 3) partitioning the data given the centroids. In the first approach, k-means-I, the data is modeled by parametric trajectories. In the second approach, k-means-II, the data is modeled by non-parametric trajectories.

3.2. K-means-I

The segments are modeled using parametric trajectory models which is consistent with what is used in the latter EM training. In fact, this k-means clustering approach is very similar to EM. To avoid under-trained Gaussians, clusters containing less than a minimum number of segments (10) are merged to the closest cluster, while the cluster with the largest variance is split into two clusters.

Initialization The partitioning of the data is initialized based on the segments' duration. This is motivated by the fact that segments with different durations may indicate different speaking rates or pronunciations and their trajectories can be quite different.

Estimating a centroid The centroids are estimated using Equations 6-8.

Re-partitioning of data Re-partitioning of the data is done based on the posterior probability as defined in Equation 9. Instead of computing the complete likelihood, only the trajectory fit is considered in the likelihood computations. Specifically, Equation 5 is modified as

$$\hat{L}(\hat{\mathbf{B}}_k, \hat{\Sigma}_k|\mathbf{B}_m, \Sigma_m) = \exp\left(-\frac{1}{2}\mathrm{tr}\left[\mathbf{Z}_k(\hat{\mathbf{B}}_k - \mathbf{B}_m)\Sigma_m^{-1}(\hat{\mathbf{B}}_k - \mathbf{B}_m)'\mathbf{Z}_k'\right]\right). \tag{10}$$

Equation 9 is modified to make a hard decision.

$$p^{i+1}(m|k) = \begin{cases} 1 & \text{if } l^i(k|m)p^i(m) \geq l^i(k|\hat{m})p^i(\hat{m}), \quad \forall_{\hat{m}\neq m} \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

3.3. K-means-II

Instead of using parametric trajectories, this approach uses non-parametric trajectories formed by linear interpolating the segments to the same length. Instead of using likelihood, a weighted Euclidean distance is used.

Initialization Multiple random initial centroids are selected and are normalized to a fixed length M by means of linear interpolation. M is selected as a fixed percentile of the segment durations. Denote $\mathcal{L}(x_i|M)$ as the interpolated version of the segment x_i of length n_i to a new length M. The k-th frame of the interpolated segment $\mathcal{L}(x_i|M)$ is expressed as,

$$\mathcal{L}(x_i|M)[k] = (1-\alpha) \times x_i[\hat{k}] + \alpha \times x_i[\hat{k}+1],$$

where
$$\hat{k} = \left\lfloor \frac{k-1}{M-1} \times (n_i - 1) + 1 \right\rfloor,$$
$$\alpha = \frac{k-1}{M-1} \times (n_i - 1) + 1 - \hat{k},$$

and $\lfloor x \rfloor$ denotes the integral value of x.

Estimating centroids A new centroid c_j is computed as the arithmetic mean of interpolated segments,

$$c_j = \frac{\sum_{x_i \in c_j} \mathcal{L}(x_i|M)}{\sum_{x_i \in c_j} 1},$$

and the duration of the centroid, d_j is the average duration of all the segments.

Re-partitioning of data Segments are assigned to the nearest centroid measured by a distance metric $d(x_i, c_j)$ that takes into account the the Euclidean distance and the duration difference between the interpolated segment and the centroid. Denoting the segment and centroid durations as n_i and d_j, this distance is defined as,

$$d(x_i, c_j) = \frac{\|\mathcal{L}(x_i|M) - c_j\|_2}{M} + \frac{\|\mathcal{L}(c_j|n_i) - x_i\|_2}{n_i} + w\|n_i - d_j\|_2,$$

where w is a pre-defined weight which is selected empirically and is the same for all phones.

3.4. Context Dependent Modeling

Context-dependent models, such as triphone models, have long been shown to improve speech recognition performance in HMM based recognizer. An important issue with context-dependent models is the sharing of parameters among different models to avoid insufficient training. Our approach to parameter sharing in triphone modeling is to allow all triphones with the same middle phone to share the same set of Gaussian mixture components but with different mixture weights. This method of parameters sharing is also used in the Byblos recognizer [4], and is called phonetically tied mixture model (PTM). The context dependent model training involves the following steps:

1. Estimate the context-independent models and use them as the 0-th iteration context dependent model. That is, $p_t^0(m) = p_(m)$, $\hat{B}_m^0 = \hat{B}_m$ and $\hat{\Sigma}_m^0 = \hat{\Sigma}_m$, where $1 \leq m \leq M$ and t refers to the triphone. Since all triphone shares the same B and Σ, they are not index by the triphone t.
2. At each iteration, re-compute the model parameters using Equation 9- 8 with $p(m)$ replaced by $p_t^i(m)$ and the posterior probability $p^i_(m|k)$ by $p_t^i(m)$.

The use of context dependent model also raises the question about back-off in the case of un-observed triphones. In our current implementation, un-observed triphones back off left and right context models and if neither have been observed, we back off to the context-independent models.

3.5. Better Segment Boundaries

To compute the likelihood using the parametric trajectory model, segment boundaries are needed. Because the trajectory representation models all the frames in a segment jointly, error in segment boundaries affect the fit for the whole segment. Ideally, one should search for the best segmentation among all possibilities but that is computationally expensive. Instead, we devise a simple search algorithm that allows the segmentation to vary up to d frames from the nomial starting boundaries such as the Viterbi alignment generated by the HMM recognizer. This search algorithm can also be used iteratively to improve the the training segmentation and the n-best segmentation.

For a segment x_k of phone t_k, denote $x_k(a, b)$ as the variation of k that begins a frames earlier and ends b frames later where $-d \leq a, b \leq d$. Thus, $x_k = x_k(0, 0)$. The search process involves two steps: 1) computes the likelihoods of $p(x_k(a, b)|t_k \;\; \forall_{k, -d \leq a, b \leq d}$, and 2) uses dynamic programming to search for the best segmentation. In step (1), we potentially need to evaluate $(2d+1)^2$ likelihoods for each segment. In reality, the number of unique segments is much fewer because we can remove ill-formed segments, such as segments with less than 3 frames in a quadratic model. Furthermore, in n-best rescoring, a segment variation in one n-best hypothesis frequently overlaps with another variation in another n-best hypothesis and hence, caching computed segment scores significantly speeds up computation. In step (2), we uses dynamic programming search to find the best path among all the valid paths. This is achieved by imposing constraints in the transitions between segments that restricting that all frames are accounted for and no frame is represented for two different segments.

Model configuration	phone class. acc.
non-mixture gender indpt.	35.1%
non-mixture gender dept.	38.9%
32 mixture, gender dept. agglo.	53.1%
32 mixture, gender dept. k-means-I	51.0%
32 mixture, gender dept. k-means-II	53.8%
256 mixture, gender dept. k-means-II	56.0%
32 mixture, context dept k-means-II	54.1%

Table 1: Results of phone classification

4. Experiments

We trained our models using 20 hours of speech from the Switchboard Corpus. Our test consists of 7 switchboard conversations that contain 35 minutes of speech. Phonetic segmentation for training and test data are generated by the Byblos [4] recognizer. We use the 14 cepstral coefficients, the normalized energy, and their first and second order differences. In all experiments reported here, quadratic trajectories and diagonal covariances are used.

4.1. Phone Classification

We performed a series of phone classification experiments that differentiates between 47 phones (no silence or nonspeech). Phone classification is a good way of measuring model improvement independent of the segmentation issues. As shown in Table 1, we begin with a primitive gender-independent model without mixture. By using gender dependent models, we can improve the phone classification by 3%. The use of mixture models with a maximum of 32 components gave a significant gain, improving the phone classification accuracy by more than 10%. We compared the three clustering methods and found them quite comparable. The k-means- II method has a slight performance advantage and is more computationally efficient as compare to the dendrogram approach.

We improved the model further by increasing the number of mixture components from 32 to 256 as well as moving to context dependent models. Adding more mixture components helps but the gain is not as dramatic as going from non-mixture to 32 mixtures. The context dependent models differed from the context independent models in having context dependent mixture weights in training and test. In phone classification, since the context information is not known in test, context-independent weights are used. However, because context information was used during EM training of the Gaussian components, the resulting model is slightly better and improved phone classification slightly.

Experiments	traj.	traj + HMM
baseline (HMM)	–	42.4%
random	46.0%	–
256 mix. context dept. model	44.5%	42.4%
+ re-segment tst only	44.9%	42.3%
re-segment trn only	44.5%	42.1%
re-segment trn & test	44.8%	42.2%

Table 2: Results of recognition

4.2. Recognition

Recognition is done in the n-best rescoring framework. The parametric trajectory models were used to re-order the n-best generated by the Byblos system. This n-best rescoring paradigm [2] can be considered a very limited search around the neighborhood of the HMM's one best answer. It searches for the best linear combination of a number of other scores. In our case, these scores includes the HMM acoustic likelihood, language model likelihood, number of words, phones and silences and the trajectory model score. For each test sentence, 1000 n-best hypothesis are generated by the Byblos system using the phonetically tied mixture (PTM) model. This HMM model is quite similar to the context dependent trajectory model in that both 1) share approximately the same number of diagonal covariance mixture components (256), 2) use the same parameter sharing mechanism (PTM), 3) train with the same data, and 4) use gender-dependent models. The models are initialized by k-means-II method.

In Table 2, we tabulate the results of rescoring n-best with and without the HMM acoustic likelihood. We also rescored using a random model where we replace the trajectory score with a randomly generated score. The effect of searching better phonetic boundaries by allowing the boundaries to vary 1 frame from the HMM alignment (denoted re-segment in Table 2) is inconclusive. While it seems to help the performance when combined with the HMM scores, it degrades the rescoring performance without HMM scores. Re-segmenting the phone boundaries in training helps the rescoring with HMM further, resulting in an 0.3% absolute improvement over using the HMM alone.

4.3. Discussion

The results in Table 1 shows that we can progressively improve the model by being more specific and adding more parameters. Comparing the three clustering methods, the agglomerative based approach is quite comparable with the k-means-II method but its computational requirement is much bigger. In fact, we have to subsample some of the phones that have more than 5000 training segments. The k-means-I method is not performing as well probably because the duration based initialization is too restrictive. We plan to continue to pursue this approach in future by using either random initialization or partition the duration based on counts. The effect of using the context-dependent model is small because context information is not used in phone classification.

The performance of rescoring n-best using the random score is not terrible because the n-best is a in effect a small search space around the HMM 1-best answer. Furthermore, the optimization process uses additional scores such as language model scores that can provide a powerful indication for the good answers among the restricted set of n-best. Comparing the n-best rescoring performance of using the trajectory model to the random score, the trajectory models are about 2.0% better. This shows that the trajectory model is providing useful knowledge. Comparing the trajectory model with the HMM, the trajectory model is performing about 2% worse by itself.

There are several issues that we plan to pursue. The first issue is further improvements in the context dependent model. We notice that the the contribution of the mixture weights in trajectory model is much smaller than in HMM making the context dependent model quite weak. In HMM, the mixture weight is applied for each frame in the sentence while in trajectory model, it is applied once per phone. So, we may need to re-scale the contributions of the mixture weights to make it more effective. The second issue is the use of duration information. From our experience, use of duration information can improve phone classification significantly. In our current experiments, we do not use it to avoid rescoring n-best with too many new parameters. The third is further understanding of the effect of segmentation. Because the trajectory models represent the whole phone, the goodness of the segment boundary is important. It is not clear whether the initial segmentations are problematic or that variation by a couple of frames is insufficient.

In summary, we presented different ways to improve the parametric trajectory models for LVCSR in this paper. In particular, we showed that the use of mixture model, context dependent model and better initialization techniques can improve the trajectory models as measured on phone classification accuracy. We also presented a efficient algorithm for improving segment boundaries in both training and n-best rescoring. By combining trajectory model with an HMM model in the n-best rescoring paradigm, we can improve recognition by 0.3%.

5. Reference

1. H. Gish and K. Ng, "A segmental speech model with applications to word spotting", In *Proc. ICASSP, 1993*,pp. 447-450.
2. M. Ostendorf, A. Kannan, O. Kimball, et al., "Integration of diverse recognition methodologies through reevaluation of n-best sentence hypotheses", In *Proc. DARPA Speech and Natural Language Workshop*, pages 83–87, 1991.
3. H. Gish and K. Ng, "parametric trajectory models for speech recognition", In Proc. *ICSLP 1996*, pp. 466-469.
4. L. Nguyen *et al.*, "The 1994 BBN Byblos speech recognition system," In *Proc. SLS Technology Workshop*, pp. 77–81, 1994.