A New Dynamic HMM Model for Speech Recognition

Feili Chen (1) and Eric Chang (2)

(1). chenfeil@public1.sta.net.cn
(2). echang@microsoft.com

(1). Shanghai JiaoTong University and Bell-lab Joint Lab
Department of Electronic Engineering, SJTU
(2). Microsoft Research China
5F, 49 Zhichun Road, Haidian District, Beijing, China

Abstract

In this paper, we describe a new method to perform speech recognition based on Dynamic HMM architecture. Pitch values are treated as hidden layer and used to modify the parameters of observation probability functions. The results show that the new model achieves approximately 10 percent relative error reductions both in base-syllable recognition task and tonal syllable recognition task. The new method can be used compatibly with conventional HMM based EM training algorithm and Viterbi decoding algorithm.

1. Introduction

Speech can be described as sequences of discrete states. Within a fixed state, speech is treated as stationary stochastic signals, which is easier to calculate the observation probability. This assumption ignores the dynamics and time sequential specifics of speech, which cause the ability of modeling distorted.

Some new methods are introduced to compensate it: Trend-hmm model with discriminative training is used in [1][2] follow the hypothesis that the speech change according with time; HDM (Hidden Dynamic models) is used in [3] to describe the dynamics of the speech refer to articulation features, and etc.

These models cannot avoid the fact that when consider the dynamics, the model need to trace back several time intervals to describe the dynamic sequence. It breaks the basic presumption of Viterbi decoder. Stack decoding or rescoring are used instead, which need much higher computation load.

We introduce here a new method to model the dynamic property of the speech to improve speech recognition performance. Pitch is used as a hidden and dynamic state space description to indicate the continuous variation of the feature space. In our new model, stationary stats have been replaced by dynamic states, with the pitch contours being added as the parameter of within state dynamics. The Gaussian mixtures are now modifiable and various in the same state. The relationship between Gaussian mixtures and pitch values are trained under EM steps. The decoder algorithm is same as conventional Viterbi decoding algorithm.

2. Pseudo-pitch contour

Pitch, as the Fundamental frequency (F0), is an important feature of speech. It contains various kind of information, such as emotion, prosody, speaking rate, grammar and even the position of the sentence. In Chinese speech, pitch acts as another important role: It decides the tone of Chinese syllable.

Pitch is still seldom used in speech recognition. This is mostly due to (1) the difficult of fundamental frequency extraction: it is often weak or missing and frequently confused at voiced/unvoiced speech, half or double errors exist everywhere. (2) The discontinuity of the feature space. Fundamental frequency does not exist in unvoiced speech and it is unstable in the boundary between voiced speech and unvoiced speech [4].

We have developed new methods to fix the errors of the raw-extracted pitch contour, and converted it into continuous, smooth and stable feature stream, which can be used to describe the dynamic property of speech.

Figure 1. The pitch contour (a. Wave; b. raw pitch; c. Supervised extension; d. Unsupervised extension)

The raw pitch contour is first checked under a forward-back algorithm to delete the half-double errors based on the continuity assumption (within the segment of voiced speech).

In unvoiced speech segments, pseudo-pitch values are used to fill in the position where fundamental frequency is unavailable—we suppose that it still keeps the continuous tendency of speech even when the pitch cannot be extracted.

The ‘fill’ algorithm is different under 2 different conditions:

1. The segmentation of speech is available. Thus we clearly know the beginning and the end of a syllable. An extended function has been generated. Figure 1.c shows the result of extension.

2. The segmentation is unavailable. We use a cubic curve to fill the empty part of pitch (Figure 1.d). The value and the 1st-order derivative of point A, B are used to estimate the parameter of the cubic curve.
3. Architecture of dynamic HMM model

The dynamic HMM model is described in Figure 2. Pitch is an input value with every frame of observation feature. Each frame of observation feature belongs to a dynamic state. The dynamic state is decided by the stationary state (mostly a state of a phone), and the input value (pitch).

![Figure 2. The architecture of Dynamic HMM](image)

In a segment of speech, the observation may belong to the same stationary state, but due to the different pitch values, they belong to different dynamic states.

Except for the existence of dynamic state and the input value, the whole architecture is nearly the same as the conventional HMM model. Our purpose is the same: to find out the best stationary state to the given observation sequences.

3.1. Mathematics description of the model

Replace the stationary states with dynamic states, the likelihood function of given observation sequence and state sequence should be:

\[ L(X \mid D) = \prod_{n=1}^{N} p(x_n \mid d_n) a_{s_n,s_{n+1}} \]

(1)

Here \( X \) is the observation sequence \( \{x_1, x_2, \ldots, x_N\} \), and \( D \) is the dynamic state sequence \( \{d_1, d_2, \ldots, d_N\} \). \( a_{s_n,s_{n+1}} \) is the transition probability from \( s_n \) to \( s_{n+1} \).

Assuming that the transition probability matrix between states has no relationship with input value, we can focus on the description of observation probability function:

Given the observation probability under a fixed dynamic state:

\[ p(x \mid d) = p(x \mid s, A) \]

(2)

Here \( X \) is the observation feature of a frame, \( A \) is the input value (can be regarded as part of observation feature), \( d \) is the dynamic state and \( s \) is the stationary state.

When using Gaussian distribution as the observation probability, the relationship between stationary states and dynamic states becomes to the relation between their parameters. For dynamic states,

\[ p(x \mid d) = \frac{(2\pi)^{-\frac{1}{2}} |\Sigma|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(x - \mu)\Sigma^{-1}(x - \mu)\right)}{\exp\left(-\frac{1}{2}(x - \mu_d)\Sigma^{-1}(x - \mu_d)\right)} \]

(3)

Here \( x \) is the observation; \( \mu_d \) and \( \Sigma \) are the mean and covariance matrix of the dynamic state. For the dynamic state belongs to the same stationary state \( s \), we can write the mean of the dynamic state as a function of input value \( A \). Thus we have:

\[ \mu_d = F(A) = F_0 + A^\ast \mu_a + O(A^2) \]

(4)

Assuming that the high order effects are small, we can approximate the relationship within the neighborhood as a linear function, \( (\mu_d \text{ is used instead } F_0) \)

\[ \mu_d = F_0 + A^\ast \mu_a = \mu_a + A \mu_a \]

(5)

Combining of (3) and (5), and using const \( C \) instead the first item, we can update the observation probability function as:

\[ p(x_i \mid d) = C \exp\left(-\frac{1}{2}(x_i - \mu_a - A \mu_a)\Sigma^{-1}(x_i - \mu_a - A \mu_a)\right) \]

(6)

3.2. The extended EM algorithm

The model can be trained in EM steps. Here we have the auxiliary function defined as \( Q(\Phi, \Phi) \), \( \Phi \) is the original model, and \( \Phi \) is the new model. Given the observation set as \( X = \{X_1, X_2, \ldots, X_P\} \).

The auxiliary function can be written as:

\[ Q(\Phi, \Phi) = \sum_{X, S} p(X, S \mid \Phi) \log p(X, S \mid \Phi) \]

(7)

\( S^T \) is the set of all available state sequences. The models can be updated as finding the optimized parameters of \( \Phi \) to maximize \( Q(\Phi, \Phi) \). An EM algorithm is used here:

1. E-step: Computation of auxiliary function \( Q(\Phi, \Phi) \) based on \( \Phi \).
2. M-step: Set the derivation function of (7) to zero, we have the equation:

\[ \frac{\sum_{n} \xi_n(j)(x_n - \bar{\mu}_j)A_n}{\sum_{n} \xi_n(j)A_n} \]

(8-a)

\[ \bar{\mu}_j = \frac{\sum_{n} \xi_n(j)(x_n - \bar{\mu}_j)}{\sum_{n} \xi_n(j)} \]

(8-b)

\( \bar{\mu}_j \) and \( \bar{\mu}_j \) are the updated model parameters of state \( j \).

x_i and \( A_i \) are the observation feature and input value of frame \( n \). Then we have the update functions as:

\[ \Delta \mu_{ij} = \bar{\mu}_{ij} - \mu_{ij} = \frac{\alpha - \beta k_i}{1 - k_i \ast k_2} \]

(9-a)

\[ \Delta A_{ij} = \bar{A}_{ij} - A_{ij} = \frac{\beta - \alpha k_2}{1 - k_i \ast k_2} \]

(9-b)

\[ \mu_{ij} = \frac{\sum_{i} \xi_i(j)(x_i - \mu_{ij} - A_i\mu_{ij})}{\sum_{i} \xi_i(j)} \]

(9-c)
\begin{equation}
\beta = \frac{\sum_{t} \xi_{j}(t)(x_{t} - \mu_{j} - A_{i}\mu_{a_{j}})A_{i}}{\sum_{t} \xi_{j}(t)A_{i}^{2}} \tag{9-d}
\end{equation}

\begin{equation}
k_{i} = \frac{\sum_{t} \xi_{j}(t)A_{i}}{\sum_{t} \xi_{j}(t)}, \quad k_{j} = \frac{\sum_{t} \xi_{j}(t)A_{i}}{\sum_{t} \xi_{j}(t)}, \quad r = 1 - k_{i} \cdot k_{j} \tag{9-e}
\end{equation}

$r$ is an important factor in function (9). It is in the range of 0.1 and decides the relationship between $\mu_{a_{j}}$ and $\mu_{j}$. We can set it to zero if the pitch value is constant, which means there is no dynamic property of the model (We can set $\mu_{a_{j}}$ to zero to constrain the state to stationary), the update function will be:

\begin{equation}
\Delta \mu_{a_{j}} = \mu_{a_{j}} - \mu_{j} = \sum_{n} \xi_{n}(j)(x_{n} - \mu_{j}) \tag{10}
\end{equation}

It is same as the conventional update functions. In this occasion, the model will reduce to conventional HMM model. A threshold of $r$ has been set. If $r$ is beyond the threshold, the dynamic states will be used. Otherwise we still use stationary states instead. The threshold usually is set to 0.1. In our experiments, with threshold as 0.1, less than 10% of the states are stationary.

3.3. Decoding algorithm

The Viterbi decoding is the same as conventional algorithm except for the computation of observation probability. The difficult is how to calculate the pseudo-pitch contour. There are different algorithms according to different conditions:

1. Suppose the speech has been segmented. This is the best condition. The pitch value can be extracted, checked and extended using the method in section 1.1.

2. Only the pitch value is available, pitch value is extracted, checked and a cubic curve is used to fill the empty part of pitch values to calculate the probability.

3. Pitch value is unavailable when decoding. We can use the estimation of pitch value instead.

As the observation feature and the probability function has been given, we can find the probability function of pitch value as:

\begin{equation}
p(A) = p(x, A \mid S)/ p(x \mid S) \tag{11}
\end{equation}

Here $p(x \mid S) = \int p(x, A \mid S) dA$

We can find that the probability distribution of pitch is a Gaussian too, with mean and variance as:

\begin{equation}
\bar{A} = \frac{\sum_{i} (x_{i} - \mu_{i})\mu_{a_{i}}\sigma_{i}^{2}}{\sum_{i} \sigma_{i}^{2} \mu_{a_{i}}^{2}}, \quad \sigma_{A}^{2} = \sum_{i} \sigma_{i}^{2} \mu_{a_{i}}^{2} \tag{12}
\end{equation}

The subscript $i$ means the dimension of the feature. We use $\bar{A}$ as the estimation of pitch contour to calculate the observation probability of given features.

4. Experiment and results

4.1. Corpus

The experiments we have done are based on two corpora. One is a speaker dependent corpus, and the other is speaker-independent. The speaker-dependent corpus contains 12016 utterances for training, and 250 utterances as testing. This corpus is manually segmented. Detail labeling has been done to the corpus, with every pronunciation checked.

The speaker-independent corpus contains 250 male speakers with approximately 200 utterances from each speaker. The test set is selected from Chinese 863 test corpus, with 30 speakers and 10 utterances per speaker.

4.2. Baseline model

The speech we use is sampled at 16 KHz. Each frame is 25 ms in length, with 10 ms shift. The feature is 39-order vector: the 12 cepstral coefficients, cepstral energy and their first and second order differences [6]. We use Chinese Mandarin tonal syllable as baseline unit. The phone set contains 27 initial and 157 tonal-finals [7].

The whole dictionary contains 1667 tonal syllables. The HMM models are context dependent triphone models. Every tri-phone is three-state, left to right and ergodic. We use Gaussian mixture with 8 diagonal Gaussian as observation probability distribution function and use syllable-loop as the word-net of decoding.

4.3. Results for speaker-dependent model

Table 1 is the result of speaker-dependent model. In the case of speaker-dependent model, we have the segmentation information both for training and test corpus. The result shows, when we compute the pitch contour according to segment information (Suppose it is the best condition), the new model gain the best result (as the case of supervised training and testing, Test1).

The new model achieves around 10% error reduction, both for tonal-syllable and base-syllable.

Table 1 is the result of speaker-dependent model. In the case of speaker-dependent model, we have the segmentation information both for training and test corpus.

<table>
<thead>
<tr>
<th></th>
<th>WER-tonal (%)</th>
<th>Error reduction (tonal) (%)</th>
<th>WER-toneless (%)</th>
<th>Error reduction (toneless) (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Base-line</td>
<td>23.50</td>
<td>6.37</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Test1</td>
<td>20.59</td>
<td>12.3%</td>
<td>5.81</td>
<td>8.9%</td>
</tr>
<tr>
<td>Test2</td>
<td>21.49</td>
<td>8.44%</td>
<td>6.08</td>
<td>4.55%</td>
</tr>
<tr>
<td>Test3</td>
<td>21.06</td>
<td>10.38%</td>
<td>5.64</td>
<td>11.46%</td>
</tr>
<tr>
<td>Test4</td>
<td>21.04</td>
<td>10.39%</td>
<td>5.26</td>
<td>17.4%</td>
</tr>
</tbody>
</table>

Table 1 is the result of dynamic model (SD)(Test1: training and testing with segment information; Test2: training and testing without segment information (using cubic curve) Test3: training without segment information and testing using estimated pitch; Test4: training with segment information and testing using estimated pitch)
conclusion that the performance of the new model depends on the accuracy of the pitch contour.

In most conditions, fundamental frequency is difficult to extract in real time. We have to predict the F0 instead. Thus the pitch contour is estimated using equation (12).

The experiment achieves nearly the same result as the supervised model. The tonal syllable recognition is a little worse in this case. But the base-syllable recognition task achieves better results. This may due to the inaccurate computation of the estimated pitch contour. When using estimated pitch value, the precision of pitch extraction and extension does not affect the performance of recognition. The training procedure becomes the most important factor.

For comparison, the same experiment has been done on the model trained supervised. The result of tonal syllable is nearly the same as the result from unsupervised training and using estimated pitch in testing. But the result for base-syllable recognition achieves the best result, about 17% of error reduction, even better than the result of supervised training and testing.

4.4. Results for speaker-independent model

Since we have not the segment information for the speaker-independent corpus. The experiment of SI model only has the result of un-supervised training and testing.

> Table 2. Result of Dynamic model (SI) (Test1: Training and testing without segment information; Test2: Training without segment information and testing using estimated pitch; Test3: Add pitch as one-dimension feature.)

<table>
<thead>
<tr>
<th></th>
<th>WER-tonal (%)</th>
<th>Error reduction (tonal)</th>
<th>WER-toneless (%)</th>
<th>Error reduction (toneless)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Base-line</td>
<td>44.22</td>
<td>23.36</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Test1</td>
<td>38.14</td>
<td>13.75%</td>
<td>20.44</td>
<td>12.5%</td>
</tr>
<tr>
<td>Test2</td>
<td>39.78</td>
<td>10.04%</td>
<td>20.47</td>
<td>12.37%</td>
</tr>
<tr>
<td>Test3</td>
<td>34.61</td>
<td>21.7%</td>
<td>22.51</td>
<td>3.64%</td>
</tr>
</tbody>
</table>

The experiment of training and testing without segment information gains nearly the same improvement as SD model. It achieves about 10% error reduction both in tonal syllable and base-syllable (Test 1).

We use estimated pitch value instead of the extracted pitch value. The model is trained without segment information (Test 2). We find that the tonal-syllable error rate increases a little. But the base-syllable is almost the same.

Another experiment (Test3) is using the pitch (including the derivate and accelerator) as extra dimensions of feature vectors, training and decoding using common HMM models. The corpus, both training set and test set, and the phone-set is same as baseline model. The result of Test3 achieves better result on tonal syllable recognition, but less improvement on base-syllable recognition. This result is reasonable, since the tone of Chinese syllable mainly depend on pitch contour.

4.5. Results under noisy condition

We have the SD model tested under the noise condition. White noise has been added to the test wave files.

The SNR is 20db. The model we use is still the model used for SD model. The result is shown in Table 3.

> Table 3. Result under Noise Condition (Test1: Test with segment information; Test2: Test with estimated pitch value)

<table>
<thead>
<tr>
<th></th>
<th>WER-tonal (%)</th>
<th>Error reduction (tonal)</th>
<th>WER-toneless (%)</th>
<th>Error reduction (toneless)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Base-line</td>
<td>40.14</td>
<td>23.46</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Test1</td>
<td>38.51</td>
<td>4.06%</td>
<td>22.92</td>
<td>2.30%</td>
</tr>
<tr>
<td>Test2</td>
<td>38.40</td>
<td>4.33%</td>
<td>21.64</td>
<td>7.76%</td>
</tr>
</tbody>
</table>

The result shows that the new model has a little improvement under noise condition. And the model using estimated value achieves nearly 8% of error reduction for base-syllable recognition task.

5. Discussions and conclusion

The new model has been shown to improve recognition accuracy. Both in SD and SI model, for both tonal and base-syllable it can achieve 10% of error reduction.

Further analysis found that the performance of the new model depends on the computation of estimated pitch contours. When the segment information is available, the new model achieves even better result. The model can also be used when pitch extraction is impossible when performing speech recognition in real time. An estimated pitch contour can be used.

We also find that when using estimated pitch value, the model can improve the accuracy of base-syllable. It is even better than performing recognition with extracted pitch contours. In the case of absence of pitch value, we can regard it as hidden layer to describe the dynamics of speech.

Further experiment should be done on speaker-adaptation. And this model also can be extended to other prosodic features such as formant.

6. Reference


