LARGE VOCABULARY MANDARIN SPEECH RECOGNITION WITH DIFFERENT APPROACHES IN MODELING TONES

Eric Chang, Jianlai Zhou, Shuo Di, Chao Huang, Kai-Fu Lee

Microsoft Research China

5F. Beijing Sigma Center, No. 49. Zhichun Road, Haidian District, Beijing 100080, P.R.C.

{echang, jlzhou, i-chaoh}@microsoft.com

ABSTRACT

Large vocabulary continuous Mandarin speech recognition has been an important problem for speech recognition researchers for several reasons [1], [3]. First of all, it is a tonal language that requires special treatment for the modeling of tones. There are five tones in Mandarin which are necessary to disambiguate between confusable words. Secondly, the difficulty of entering Chinese by keyboard presents a great opportunity for speech recognition to improve computer usability. Previous approaches to modeling tones have included using a separate tone classifier [1] and incorporating pitch directly into the feature vector [3]. In this paper, we describe a large vocabulary Mandarin speech recognition system based on Microsoft's Whisper system. Several alternatives in modeling tones and their error rates on continuous speech are compared.

The experimental result shows a character error rate of 7.32% on a test set of 50 speakers and 1000 sentences when no special tone processing is performed in the acoustic model. When the final syllable model set is expanded to include tones, the error rate drops to 6.43% (error rate reduction of 12.2%). When pitch information and the larger final syllable set are used in combination, the error rate is 6.03% (cumulative error rate reduction of 17.6%). This result suggests that other sources of information such as energy and duration can also contribute toward disambiguating between different tones.

1. INTRODUCTION

The Microsoft Whisper speech recognition system [4] is a flexible senone-based recognizer that has previously been converted to recognize Japanese [5]. We have extended the system to model the different tones in Mandarin. The system uses context dependent semi-syllabic units for modeling Mandarin syllables. A total of 6000 senones with 8 Gaussians per senone are used in the acoustic model, with the assignment of senones to semi-syllabic units determined through decision tree based clustering.

Three variations in modeling tones are studied. In the first case, no specific tone modeling in the acoustic model is performed. Instead, a powerful language model is used as the sole method for disambiguating between tonally confusable words. In the second case, the final syllable model set is expanded to model the 5 tones separately. However, the feature vector of the system is not modified to take pitch into account. Lastly, a fast pitch extractor that runs in real time was developed. The pitch track obtained with the pitch extractor is smoothed and added to the feature vector along with its delta and double delta components. In Section 2, the acoustic phone sets that were used in this study are described. In Section 3, we present the two pass pitch extraction algorithm and its evaluation. In Section 4, we describe the corpora and the system used in the experiments and the experimental results. Finally, we conclude in Section 5.

2. ACOUSTIC UNIT SELECTION

There have been many different acoustic representations for Mandarin in recent years. For example, there have been syllable based approach, syllable initial/final approach, and preme/toneme approach [3]. In this study, we selected the syllable initial/final approach and then expanded only the syllable final set according to tones. Table 1 lists the syllable initial and final units that we used in this work. The acoustic unit set was constructed in consultation to previous phonological studies of Mandarin [2]. For syllables with no consonants, such as *a*, *e*, *er*, and *o*, we use a psudo-initial syllable so that the representations are (ga a), (ge e), (ger er), and (go o) respectively. Syllables *chi*, *ri*, *shi*, and *zhi* are represented as (chi ib), (r ib), (sh ib) and (zh ib) respectively. To distinguish the different tongue palate locations during production, syllables *ci*, *si*, and *zi* are represented as (c if), (s if), and (z if) respectively. In addition, we include a silence phone and a garbage phone to model the background. So we have a total of 187 phone models for the large phone set experiments. For the small phone set, we have a total of 66 phone models.

Syllable Initial	b, c, ch, d, f, g, ga, ge, ger, go, h, j, k, l, m, n, p, q, r, s, sh, t, w, x, y, z, zh
Syllable Final	a, ai, an, ang, ao, e, ei, en, eng, er, i, ia, ib, ian, iang, iao, ie, if, in, ing, iong, iu, o, ong, ou, u, ua, uai, uan, uang, ui, un, uo, v, van, ve, vn
Syllable Final with Tone	a(1-5), ai(1-4), an(1-4), ang(1-5), ao(1-4), e(1-5), ei(1-4), en(1-5), eng(1-4), er(2-4), i(1-5), ia(1-4), ib(1-4), ian(1-5), iang(1-4), iao(1-4), ie(1-4), if(1-4), in(1-4), ing(1-4), iong(1-3), iu(1-5), o(1-5), ong(1-4), ou(1-5), u(1-5), ua(1-4), uai(1-4), uan(1-4), uang(1-4), ui(1-4), un(1-4), uo(1-5), v(1-4), van(1-4), ve(1-4), vn(1-4)

Table 1: Syllable initial and final units used for experiments with tone and without tone. (Numbers following syllable final units indicate the range of tones represented; the number 5 represents the neutral tone.)

3. PITCH EXTRACTION

3.1 Constraints in Practical System

Although there are many pitch extraction algorithms, previous work [6] which compares the performance of the different algorithms shows that no one is absolutely better than the others. On the other hand, for practical use of dictating text, real-time display of recognized text is desirable. With real-time speech recognition systems, the front-end module generates acoustic feature vectors including Mel-scale cepstrum coefficients and pitch as speech waveform is entered into the system. Therefore, there is no look-ahead buffer of data that can be used to improve on pitch extraction accuracy. In addition to working in real time, the pitch extraction algorithm must be computationally efficient due to the limited amount of resources that can be devoted to front-end computation.

3.2 Algorithm Description

Our target is to design a fast and robust pitch tracker. For a complete pitch tracker, often there are three major components: 1) A preprocessor, which removes some background noise and unreasonable frequency components in the frequency domain, 2) A F0 candidates estimator, which seeks the candidates of the true period, and 3) A post-processor, the best candidate is selected and the F0 is refined in this stage.

In the above three components, the F0 candidate estimator is the most time consuming, because variant forms of correlation are calculated in this stage. To speed up the traditional F0 candidate estimator, a two-pass procedure is employed. The idea is to use the fastest algorithm for finding N possible F0 candidates in the first pass, and then apply more powerful algorithm to re-score these N candidates in the second pass. Usually, N is much smaller than the whole estimating range of possible F0 values. As a result, computation is reduced dramatically with limited accuracy loss.

In our implementation, the DC bias is estimated and subtracted from each speech frame in pre-processing. For the F0 candidate estimator, we select the average magnitude difference function (AMDF) [6] as the estimator in the first pass, and normalized cross correlation function (NCCF) [7] in the second pass. Because AMDF consists of the subtraction as following, it is faster than other algorithms.

$$D_{i,k} = \sum_{j=m}^{m+n-1} \left| s_j - s_{j+k} \right|, k = 0, 1, \text{L}, K-1 \qquad (3.1)$$

Where, s_j and s_{j+k} are jth and (j+k)th sample in the speech waveform, $D_{i,k}$ represents the similarity of ith speech frame and its adjacent neighbor with interval of k samples.

The normalized cross correlation function can be expressed as:

$$\phi_{i,k} = \frac{\sum_{j=m}^{m+n-1} s_j s_{j+k}}{\sqrt{e_m e_{m+k}}} \quad k = 0, 1, \text{L}, K-1; \quad i = 0, 1, \text{L}, M-1 \qquad (3.2)$$

Where

$$e_m = \sum_{l=m}^{m+n-1} s_l^2 \qquad (3.3)$$

Because the value of NCCF is independent of the amplitude of adjacent speech frames, the NCCF overcomes the shortcomings of the other F0 candidate estimators described in [6], [7], but computation is increased.

During post-processing, dynamic programming is applied to select the best F0 and voicing state candidates at each frame based on a combination of local and transition costs.

Usually, a lattice structure is organized which consists of N voiced candidates with the pitch value calculated by the estimator described above and an unvoiced assumption in each frame. For each speech frame, the local cost is the NCCF value or score for every candidate assumed to be a voiced segment, and the average NCCF score for the unvoiced. The transition cost takes into account many factors, such as ratio of energy, ratio of zero crossing rate, Itakura distance, and difference of F0 between two adjacent speech frames.

3.3 Evaluation Results

In the ideal case, a physical device measurement such as larynograph should be used to evaluate the performance of the pitch tracker. However, a large database of speech from many speakers and the corresponding larynograph recording are not available to us. Our solution is to select the commonly used pitch tracker made by Entropic to generate the reference pitch track. We use 70,000 sentences enunciated by 250 male speakers and 100 female speakers as our testing data. The comparison result of pitch trackers between Entropic and MSRCN is listed in Table 2. The result shows that the two pass pitch tracker that we developed is approximately 20 times faster than the Entropic pitch tracker with limited accuracy loss. Also, while there are some absolute differences between the pitch tracks extracted by the Entropic system and our system, the pitch contour is more important for tone recognition. In later experiments that incorporate pitch into the feature vector, there was no error increase when the pitch track from our two pass system was used instead of the pitch tracks extracted by the Entropic pitch tracker.

	Entropic	MSRCN
Accuracy	100%	94%
Speed	1.15	0.057

Table 2: The speed value in the table is the ratio of time spent by each pitch tracker divided by the speech duration.

4. EXPERIMENTS

The basis of our work is a state of the art speech recognition system, Whisper, which we have enhanced by adding specific features that are beneficial for recognizing tonal languages such as Mandarin.

4.1 System Description

Our contributions in developing the Mandarin recognition system include refining acoustic models and Chinese language models. In this section, we will characterize our progress by percent error rate reduction.

4.1.1 Feature representations

At present, MFCC based feature is the most popular feature used in speech recognition systems. For Mandarin, as we discussed above, pitch and its dynamics should be provided in the feature vector to model the tones.

In our system, a feature vector with 36 dimensions is used. The 36 dimensions consist of:

- Energy based feature (E, ΔE, ΔΔE)
- MFCC based feature (12MFCC, 12ΔMFCC, 6ΔΔMFCC)
- Pitch based feature (F0, ΔF0, ΔΔF0)

In our early experiments, we found that when the extracted pitch track is directly added to the feature vector, no accuracy improvements were found. A smoothing process is necessary to make pitch information useful in continuous speech recognition. There are several smoothing methods, but due to the real time feedback constraint for better user interface, some specific compromises are made. For example, we should deal with every frame of speech without looking ahead. For voiced speech segment, the smoothed value is:

$$P_t^{'} = \log 10(p_t) + x \tag{4.1}$$

For the unvoiced,

$$P_t^{'} = P_{t-1}^{'} + \lambda(P_{aver}^{'} - P_{t-1}^{'}) + x \tag{4.2}$$

Where, p_t represents the real pitch value in time t, and $P_t^{'}$ the smoothed pitch value. $P_{aver}^{'}$ is a running average calculated from previous history or training data, λ is a constant determined through experiments. X is a small random value that can prevent the variance of the Gaussians models from being zero.

4.1.2 Detailed acoustic modeling by parameter sharing

Decision tree has been successfully used for improved sharing of HMM parameters in many speech recognition systems. In decision tree based clustering, a binary tree is built for each state of every phone. Each tree has a yes or no phonetic question at each node. In our system, a set of questions is prepared based on Chinese phonetics [2]. There are 187 questions that summarize the enunciation property of Mandarin. Clustering at a state level provides the freedom to use a larger number of states for each tri-phone model. In our system, we use 6000 senones with 8 Gaussians in each senone.

4.1.3 Language modeling

A stochastic grammar such as bigram or trigram provides an *a priori* estimate of the probability of each word in context to its preceding words. We used a trigram language model during the decoding process.

For language model training, we used a large text corpus that contains 1.6 billion characters. The content of the corpus comes from many different domains including newspaper articles, novels, web texts, and technical documents. There are 52,000 words in our vocabulary, and the size of the language model is approximately 124 MB. The detailed process of building the language model is described in [8].

4.2 Experimental Results

Several experiments have been done to demonstrate the improvements step by step. In particular, we will show the result of using a small phone set, a large phone set and a large phone set with pitch in the feature vector.

4.2.1 Data

Having a lot of data is essential for establishing a modern state of the art speech recognition system. We collected a speech database including 500 speakers (half male, half female), with 200 sentences per speaker. The scripts read by the speakers were carefully selected to ensure broad triphone coverage. The data were recorded at a 16k sampling rate and 16 bits per sample. These data are training data for all of our experiments. All the speakers were recruited in the Beijing area.

We also collected a testing database of 1000 sentences from 50 speakers, 25 males and 25 females, with 20 sentences per speaker. The average perplexity of the sentences is less than 200 based on our language model. For convenience, we will call the male test set as m-msr, and the female test set as f-msr.

4.2.2 Experiments

In order to observe how well the tone is modeled, we constructed a baseline system with a small phone set that contains 66 phoneme like units. In small phone set, only syllable initials and non-tone-specific syllable finals of Mandarin syllables are used and no tone information is represented.

To study whether tones can be distinguished without incorporating pitch into the feature vector, we used a large phone set as described in Section 2, but keeping the feature vector the same as the baseline system. Lastly, we added pitch based feature in the feature vector by the method discussed above.

The error rate of each test set is shown in Table 3. The experimental results show a character error rate of 7.32% on average when no special tone processing is performed in the acoustic model. When the final syllable model set is expanded to include tones, the error rate drops to 6.43% (error rate reduction of 12.2%). When pitch information and the larger final syllable set are used in combination, the error rate is 6.03% (cumulative error rate reduction of 17.6%). More than 17% error rate reduction on average is achieved by introducing pitch based feature and the large phone set. The error rate reductions are consistent across different gender. The improved accuracy with the larger tone-dependent syllable final set even without the inclusion of pitch information matches the result previously presented in [9]. This shows that spectral information present in the MFCC feature vector also contains information for discriminating between different tones.

	Female	Male	Average
Small Phone Set	6.35	8.28	7.32
Large Phone Set without Pitch	5.64	7.21	6.43
Large Phone Set with Pitch	5.35	6.71	6.03

Table 3: Error rate on each test set.

Another series of experiments showed us that the amount of improvement is different for each pitch based feature such as pitch, delta pitch and double delta pitch. We used the male large phone set model with pitch and left only one of the three pitch-based features used at a time. Then we repeated the decoding experiments using the same male test set described above. The error rate of each configuration is shown in Figure 1. Comparing the results of using each pitch-based feature separately with the original result using all three pitch based features, it is clear that the delta pitch parameter is the most important factor in improving accuracy.

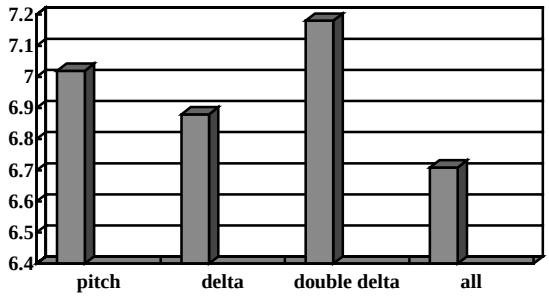

Figure 1: The error rate for each feature configuration on the male test set.

5. CONCLUSION

Three variations in modeling tones are studied. In the first case, no specific tone modeling in the acoustic model is performed. Instead, a powerful language model is used as the sole method for disambiguating between tonally confusable words. In the second case, the final syllable model set is expanded to model the 5 tones separately. However, the feature vector of the system is not modified to take pitch into account. Lastly, a fast pitch extractor that runs in real time was developed . The pitch track obtained with the pitch extractor is smoothed and added to the feature vector along with its delta and double delta components.

The experimental result shows a character error rate of 7.32% when no special tone processing is performed in the acoustic model. When the final syllable model set is expanded to include tones, the error rate drops to 6.43% (error rate reduction of 12.2%). When pitch information and the larger final syllable set are used in combination, the error rate is 6.03% (cumulative error rate reduction of 17.6%). In the future, we intend to better incorporate tonal contextual information such as the identity of the previous tone and the following tone to further improve accuracy.

6. ACKNOWLEDGEMENT

We thank our colleagues A. Acero, H. Hon, X. Huang, M. Hwang, and S. Meredith from Microsoft Research for their suggestions. We thank M. Li, Z. Chen, and J. Gao for providing the language model.

7. REFERENCES

[1] Lee L. S., et. al, "Golden Mandarin (I)—A Real Time Mandarin Speech Dictation Machine for Chinese Language with Very Large Vocabulary", *IEEE Trans. on Speech and Audio Processing*, Vol. 1, NO. 2, pp 158-179, April 1993.

[2] 吴宗济主编，"现代汉语语音概要"，华语教学出版社，1992，北京.

[3] Chen C. J., et. al., "New Methods in Continuous Mandarin Recognition", *Proc. Eurospeech 97*, Volume 3, pages 1543-1546.

[4] Huang X., Acero A, Alleva F., Hwang M. Y., Jiang L., and Mahajan, M., "Microsoft Windows highly intelligent speech recognizer: Whisper", *Proc. ICASSP 95*, Volume 1, pages 93-96.

[5] Hon H. W., Ju Y. C., and Otani K., "Japanese Large-Vocabulary Continuous Speech Recognition System Based on Microsoft Whisper." *Proc. ICSLP 98*.

[6] Rabiner L.R., et. al, "A Comparative performance Study of Several Pitch Detection Algorithms.", *IEEE Trans. on Acoustic, Speech, and Signal Processing*, Vol. ASSP-24, pp 399-418, Oct. 1976.

[7] Talkin D., "A robust algorithm for pitch tracking (RAPT)," in *Speech Coding and Synthesis*, W. B. Kleign and K. K. Paliwal, eds., Elsevier Science, Amsterdam, pp. 495-518, 1995.

[8] Gao J., et. al., "A Unified Approach to Statistical Language Modeling for Chinese", *Proc. ICASSP 2000*, Volume III, pp. 1703-1706.

[9] Liu F. H., Picheny M., Srinivasa P., Monkowski M., and Chen J, "Speech Recognition on Mandarin Call Home: A Large-Vocabulary, Conversational, and Telephone Speech Corpus," *Proc. ICASSP 96*, Volume 1, pp. 157-160.