Highly Accurate Mandarin Tone Classification In The Absence of Pitch Information

Neville Ryant[1], Malcolm Slaney[2], Mark Liberman[1], Elizabeth Shriberg[3], and Jiahong Yuan[1]

[1]Linguistic Data Consortium, Philadelphia, PA, USA
[2]Microsoft Research, Mountain View, CA, USA
[3]SRI International, Menlo Park, CA, USA

nryant@gmail.com, malcolm@ieee.org, markyliberman@gmail.com
ees@icsi.berkeley.edu, jiahong.yuan@gmail.com

Abstract

A deep neural network (DNN) classifier based only on 40 mel-frequency cepstral coefficients (MFCCs) achieved 29.99% frame error rate (FER) and 16.86% segment error rate (SER) in recognizing five tonal categories in Mandarin Chinese broadcast news. With the addition of subband autocorrelation change detection (SACD) pitch-class features [1], the classifier scored 27.58% FER and 15.56% SER. These results are substantially better than the best previously reported results on broadcast news tone classification [2] and are also better than a human listener achieved in categorizing test stimuli created by amplitude- and frequency-modulating complex tones to match the extracted F_0 and amplitude parameters [3]. The same DNN architecture scored substantially worse when trained and tested with SACD pitch-class parameters alone: 39.22% FER and 24.89% SER. RAPT F_0 estimates are worse yet: 44.37% FER and 27.28% SER. The 40 MFCC parameters do not encode F_0 in any obvious way and attempts to predict SACD or other pitch features from them work badly. These surprising results raise difficult questions for theories of Chinese tone.

Index Terms: speech recognition, Mandarin, tone modeling, deep neural networks

1. Introduction

Typically, Chinese speech-recognition systems have included tonal features in order to improve performance in the integrated task of recognizing tonally-specified segments [4, 5, 6, 7]. More recently, there has been increased interest in the more specific problem of automated recognition of tonal categories alone in continuous speech [8, 2, 9, 10]. For instance, Pui-Fung [8] uses decision trees and a segmental representation based on the fitting of polynomials to the F_0 contour to achieve 27.8% segment error rate (SER). Lei [2] achieves 23.8% SER using MLPs and contextual information. Most recently, Kalinli [10] achieved 21% SER, albeit for command-and-control utterances, with the incorporation of biologically inspired auditory features.

Of the above papers, all save Kalinli perform explicit pitch tracking (though even Kalinli includes parameters that are probably an excellent proxy for F_0 slope). However, pitch is notoriously hard to accurately estimate even in cases where it is not inherently ambiguous [11]. Moreover, for the task of interest, tone classification, absolute pitch is not itself even particularly relevant but, rather, changes in pitch over an interval of time. Such being the case, it has been suggested that it is more appropriate to estimate pitch changes directly [1]. Using subband autocorrelation change detection (SACD) features, Slaney achieves superior performance for 4-way tone classification on a corpus of Mandarin phone speech with the SACD features providing relative reductions in error ranging from 10% for clean materials to 17% for speech corrupted by white noise.

Yet more recent work demonstrated successful Mandarin tone classification for broadcast news materials in the absence of any explicit pitch-related information whatsoever [3]. Using a deep neural network (DNN) based classifier and an input representation consisting of 21 consecutive frames of 40 mel frequency cepstral coefficients, our previous work achieves an SER of 16.62%, a 7.04% absolute reduction relative to a baseline system incorporating explicit, but perhaps errorful, F_0 information.

Jointly the findings of Slaney and Ryant suggest that at least for some tone languages, highly accurate tone classification is possible in the absence of explicit pitch tracking; indeed, that tone is not "just" about F_0. In this paper we extend this work and directly compare the efficacy of these features for Mandarin tone classification using the same training/test sets and machine learning infrastructure. We also consider possible explanations for why the MFCC frontend is so successful.

2. Data and evaluation

Testing and training sets were constructed using the 1997 Mandarin Broadcast News Speech corpus [12][1]. We extracted all "utterances" (the between-pause units that are time-stamped in the transcripts) from the corpus and manually excluded those containing background noise or music. Utterances from speakers whose names were not tagged in the corpus or from speakers with accented speech were also excluded. In total 7,849 utterances from 20 speakers were selected. From these we randomly selected 50 utterances from each of six speakers to compose a test set, with the remaining 7,549 utterances reserved for training. The 300 test utterances were manually labeled and segmented into initials and finals by a native Mandarin speaker. Tones were marked on the finals, including Tone1 through Tone4, and Tone0 for the neutral tone. The total number of utterances, segments, and hours of speech are detailed in Table 1.

	Hours	Utterances	Segments	TBUs
Train	6.05	7,549	196,330	96,697
Test	0.22	300	7,189	3,464

Table 1: Train/test set composition. TBU = tone-bearing unit, defined as the syllable final.

System performance is measured in two ways. As an initial evaluation of the quality of the representation learned by the network, we consider its frame error rate (FER), defined as the percentage of frames incorrectly classified by the DNN. Our primary metric, however, is segment error rate (SER), defined as the percentage of TBUs incorrectly classified.

3. System description

We propose attacking the problem of explicit tone classification as follows:

1) Train a DNN to classify each frame of speech into one of six tone classes: Tone0, Tone1, Tone2, Tone3, Tone4, No-tone.

2) Compute "tonal features" for each segment, defined as the mean of the outputs of the DNN over all frames contained within that segment. These are similar to Chao's articulatory features [9].

3) Use these "tonal features", along with segment duration and contextual features, to classify the tone-bearing units (TBUs).

3.1. Features

We train four separate tone-classification systems using different feature frontends:

1. *RAPT F_0 estimate*
   Our first feature consists of F_0 as estimated by peaks in the normalized cross-correlation function using RAPT [13] as implemented in ESPS's *get_f0* with the following parameters: wind_dur=0.01, min_f0=60, max_f0=650.

2. *SAaC F_0 estimate*
   A second F_0 estimate is computed using the SAaC system [14]. A correlogram is constructed by running the signal through an auditory filterbank and calculating the autocorrelation for each channel. The size of this representation is reduced using PCA and the retained principal components serve as input to an MLP that classifies frames into one of 70 pitch classes (67 classes spanning 60-400 Hz on a logarithmic axis plus additional classes corresponding to unvoiced, out of range low, and out of range high). Viterbi decoding of the MLP outputs produces a smoothed pitch track.

3. *SACD*
   We also consider SACD features [1]. As with SAaC, an MLP is trained to classify frames into one of 70 pitch classes on the basis of the principal components of a correlogram. The MLP-derived pitch class probabilities are smoothed across frames using a 5-frame moving average window and cross-correlation between adjacent frames calculated for a range of lags. The final SACD features consist of the cross-correlation values corresponding to lags from -2 to 2.

4. *MFCC*
   Forty mel frequency cepstral coefficients (MFCCs) were extracted using the following analysis parameters: i) 0.97 pre-emphasis factor; ii) 25 ms Hamming window; iii) 1024-point DFT; iv) 40 filter mel-scale filterbank[2].

In addition to systems trained using the RAPT, SAaC, SACD, and MFCC features individually, we also consider each combination of the MFCC features and the pitch-related featuress. All features, including the F_0 estimates, were computed every 10 ms and normalized to have 0 mean and unit variance on a per-utterance basis[3].

3.2. Network training

For each feature combination a DNN was trained [16] to classify frames of the signal as one of the six targets.

[1]The specific dataset used in these experiments will be published by the LDC and meanwhile is available from the authors by request.

[2]Our MFCCs may be reproduced using *melfcc* [15] with the following parameter values: wintime=0.025, hoptime=0.010, nbands=40, numcep=40, lifterexp=-22, sumpower=0, minfreq=0, maxfreq=8000, dcttype=3.

[3]We examined other normalization schemes, including one in which F_0 normalization was restricted to voiced segments, but this choice had negligible impact on the final accuracy.

Input to the DNN consisted of a high-dimensional feature vector derived by concatenating the extracted features for all frames in a 21-frame context window (10-1-10). Training targets were derived by forced alignment of the HUB-4 training utterances using an HMM-based forced aligner built on the training utterances with the CALLHOME Mandarin Chinese Lexicon [17] and HTK. The aligner employed explicit phone boundary models [18] and achieved 93.1% agreement within 20 ms compared to manual segmentation on the test set. Additionally, we checked 100 training utterances on the tone labels automatically generated by the aligner. Among the 1,252 syllables in the 100 utterances, 15 syllables had a wrong tone, an error rate of 1.2%[4].

The full network topology consisted of: i) the input layer; ii) 4 hidden layers, each consisting of 2000 rectified linear units (ReLUs) [19]; iii) an output layer consisting of 6 softmax units. The network was trained for 60 epochs (each epoch consisting of 250,000 examples) using stochastic gradient descent with a minibatch size of 128, 20% dropout [20] in the input layer, 30% dropout in the hidden layers, and a cross-entropy objective. Learning rate was kept constant within epochs and followed the schedule $\eta(n) = \eta(0)\frac{500}{n+500}$, where $\eta(0) = 0.5$, while momentum was kept constant at 0.5 throughout training. No L_2 weight decay was used, but the incoming weight vector at each hidden unit was constrained to have a maximum L_2-norm of 3.

3.3. Segment-level classification

Segment-level classification decisions were made using a single-layer neural network trained to assign tone classes to the TBUs. Input features consisted of the tonal features of the segment, duration (in seconds) of the segment (as determined by the forced alignment boundaries), and tonal features and durations of the two immediately preceding and two immediately following segments. The neural network contained a single hidden layer of 128 ReLUs and was trained for 1,000 epochs (epoch=100,000 instances) using stochastic gradient descent with minibatch size of 512, 30% hidden layer dropout, a decaying learning rate beginning at 1, and a constant momentum of 0.9. The incoming weight vector at each hidden unit was constrained to have a maximum L_2-norm of 1.

4. Results

FERs and SERs for the trained systems are shown in Table 2. Because silences and other unvoiced regions are relatively easy to recognize in material of this kind and, therefore, a FER that includes such regions will depend on the amount of silence that is included in the test set, we depict not only overall FER, but also FER excluding frames that do not correspond to a tone bearing unit in the gold standard segmentation. Three results are immediately apparent. One, in accord with earlier findings [1], the SACD features are more informative than either RAPT or SAaC-derived F_0 estimates. Indeed, FER on TBUs for the the system trained on SACD features is 39.22% and SER 24.89%, which represents relative error reductions of 11.61% and 8.76% respectively from the figures achieved by the system using RAPT F_0 estimates. Two, replicating our earlier findings [3], the system trained only using the MFCC frontend trounces the systems trained using only pitch related features, reducing TBU FER by 23.53% and SER by 32.36% relative to the system trained using SACD. Three, while inclusion of F_0 alongside MFCCs fails to improve (hurts actually) performance, adding SACD features does appear to help, resulting in relative reductions of 8.04% for FER on TBUs and 8.35% for SER. This result suggests that, whatever information is contained in the MFCCs, it is complementary to that contained in the SACD features.

	Frame Error Rate (FER)			
	Overall	TBUs	Tones 1−4	SER
RAPT	29.09	44.37	42.05	27.28
SAaC	32.39	49.64	47.55	28.67
SACD	25.25	39.22	37.05	24.89
MFCC	18.88	29.99	29.35	16.86
MFCC+RAPT	18.70	29.98	29.43	17.47
MFCC+SAaC	18.79	29.38	28.75	17.52
MFCC+SACD	17.57	27.58	27.00	15.56

Table 2: Frame error rates and segment error rates (%) on test set for DNNs trained using various combinations of the feature frontends.

5. General discussion

The success of MFCCs with a context window of many frames of MFCCs is, at first glance, perplexing: how does a representation in which information about pitch should be eradicated, or at least substantially blurred, do so well at predicting tones on segments, a task that is supposedly entirely about pitch? One possible explanation for our performance is that the DNN system is actually somehow implicitly learning to do overall phone recognition with the tone recognition merely a byproduct. While perfect (toneless) phone recognition is implausible[5], we do put the idea to the test by comparing FER and SER of the MFCC-trained system with the FER and SER of an oracle

[4]These errors are primarily due to application of third tone sandhi across word boundaries.

[5]When we trained a DNN with the same topology and hyperparameters used for the tone classification experiments to predict the (toneless) phone categories using the MFCC features as input, final frame error rate on the test set came to 21.2%, suggesting that, in the absence of a language-model or other higher-level information, we would be unlikely to do better than 80% accuracy at predicting phones, much less 100%.

with perfect knowledge of the pinyin of each initial/final, but no other information about tone at all (Table 3). An oracle making maximum-likelihood guesses given perfect phone knowledge produces 51.96% SER compared to 16.86% for the MFCC only system.

Perhaps more context helps? Just in case, we also consider the performance of a second oracle, which predicts the tone class of each segment using perfect (toneless) knowledge of the phone class of the preceding, current, and following segments and a neural network with a single hidden layer of 128 rectified linear units (depicted as Oracle (tri) in the table). Inclusion of this additional context does improve performance markedly, bringing TBU FER down to 21.27% and SER to 20.76%, suggesting that in the unlikely event of perfect recognition of a span of three pinyin initials or finals, reasonably good tone recognition is possible, even in the absence of a language model. However, these error rates remain substantially higher than what the DNN is achieving, suggesting some other mechanism is at work.

	Frame Error Rate (FER)			
	Overall	TBUs	Tones 1−4	SER
Oracle (mono)	28.28	52.14	53.79	51.96
Oracle (tri)	11.54	21.27	21.84	20.76

Table 3: Frame error rates and segment error rates (%) on test set for two oracle systems.

Alternately, it may be the case that the DNN is making use of a multitude of other, non-pitch, phonetic dimensions, which jointly are predictive of tone class. Acoustic analysis of Mandarin syllables suggests that duration [21, 22], temporal envelope [22], and formant structure [23] differ for different lexical tones of the same syllable. Moreover, it is well established that, though impaired relative to clean speech, native speakers are able to identify tone in both real [24, 25] and synthetic [26, 22] whispered speech at well above chance levels. In light of these findings the thesis tying the DNN's performance to efficient use of non-pitch information represented in the MFCCs is plausible.

Finally, it should be considered that the DNN may be recovering F_0 information from the MFCC parameters, either in terms of the actual pitch track or some other form. Conventional wisdom suggests that MFCC and its ilk are good for speech recognition because they represent the rough shape of the spectrum, but without the pitch information[6]. Nevertheless, as a test of the hypothesis that F_0 is being extracted, we trained a DNN to predict the SAaC/SACD pitch classes using the MFCC features as input. While this network was able to achieve a frame error rate of 29.48% on the test set for pitch-class prediction, error analysis reveals that this is principally because the network is very good at making voicing distinctions as opposed to actually successfully determining pitch in voiced segments (unvoiced frame error: 1.83%; voiced frame error: 45.3%).

However, this does not rule out the other possibility: that the network is pulling out some other, pitch-related information from the MFCC representation that has predictive power. To explore this idea, we performed a simple comparison experiment by synthesizing a large number of static vowels with random pitches centered at 120 Hz (male) and 200 Hz (female) using Praat [29]. Figure 1 shows the relative contrast between three different vowels (/a/, /ɪ/, /u/), and the same vowel at two pitches separated by 2 semitones. As we vary the number of cepstral coefficients between 1 and 40, the MFCC representation does a better job of capturing their differences, as reflected in the Euclidean distance. For these highly stylized vowels (fixed pitch, no noise, no coarticulation) the female pitch change leads to longer distances, suggesting that the pitch change is reflected in the MFCC coefficients at least for widely spaced harmonics. Interestingly, this difference only shows up when the number of cepstral coefficients is more than 20. This difference might allow a classifier to more easily notice the two classes. Yet, we were not able to see any significant difference in the performance of the DNN network when we looked at male vs. female speakers.

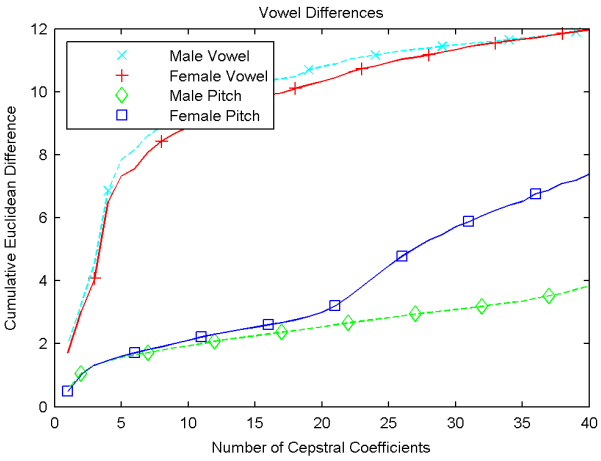

Figure 1: The average distance between two vowels, as a function of the size of the cepstral vector. The top two curves are for two different vowels at the same pitch. The bottom two curves are for the *same* vowel at two pitches that differ by 2 semitones. The synthetic vowels have an average pitch of 120 Hz for the male examples, and 200 Hz for the female examples, in line with the Mandarin database used in this paper.

Most probably, all three hypotheses are true to an extent and the DNN is using all three sources of information jointly to make its final predictions.

[6]Though see also [27, 28], who report success in predicting F0 in English read speech using a standard 23 channel, 13 cepstral coefficient MFCC representation.

6. References

[1] M. Slaney, E. Shriberg, and J.-T. Huang, "Pitch-gesture modeling using subband autocorrelation change detection," in *INTERSPEECH*, 2013, pp. 1911–1915.

[2] X. Lei, M.-H. Siu, M.-Y. Hwang, M. Ostendorf, and T. Lee, "Improved tone modeling for Mandarin broadcast news speech recognition," in *INTERSPEECH*, 2006.

[3] N. Ryant, J. Yuan, and M. Liberman, "Mandarin tone classification without pitch tracking," in *Proceedings of ICASSP*, 2014.

[4] C. J. Chen, R. A. Gopinath, M. D. Monkowski, M. A. Picheny, and K. Shen, "New methods in continuous Mandarin speech recognition," in *Eurospeech*, 1997.

[5] E. Chang, J.-L. Zhou, S. Di, C. Huang, and K.-F. Lee, "Large vocabulary Mandarin speech recognition with different approaches in modeling tones," in *INTERSPEECH*, 2000, pp. 983–986.

[6] H. C.-H. Huang and F. Seide, "Pitch tracking and tone features for Mandarin speech recognition," in *Proceedings of ICASSP*, 2000, pp. 1523–1526.

[7] R. Sinha, M. Gales, D. Kim, X. Liu, K. Sim, and P. Woodland, "The CU-HTK Mandarin broadcast news transcription system," in *Proceedings of ICASSP*, 2006.

[8] W. Pui-Fung and S. Man-Hung, "Decision tree based tone modeling for Chinese speech recognition," in *Proceedings of ICASSP*, 2004, pp. 905–908.

[9] H. Chao, Z. Yang, and W. Liu, "Improved tone modeling by exploiting articulatory features for Mandarin speech recognition," in *Proceedings of ICASSP*, 2012, pp. 4741–4744.

[10] O. Kalinli, "Tone and pitch accent classification using auditory attention cues," in *Proceedings of ICASSP*, 2011, pp. 5208–5211.

[11] R. N. Shepard, "Circularity in judgments of relative pitch," *The Journal of the Acoustical Society of America*, vol. 36, no. 12, pp. 2346–2353, 1964.

[12] S. Huang, J. Liu, X. Wu, L. Wu, Y. Yan, and Z. Qin, *1997 Mandarin Broadcast News Speech (HUB4-NE)*. Linguistic Data Consortium, 1998.

[13] D. Talkin, "A robust algorithm for pitch tracking (RAPT)," in *Speech coding and synthesis*. New York: Elsevier, 1995, pp. 495–518.

[14] B. S. Lee and D. Ellis, "Noise robust pitch tracking by subband autocorrelation classification," in *INTERSPEECH*, 2012.

[15] D. P. W. Ellis, "PLP and RASTA (and MFCC, and inversion) in Matlab," 2005, online web resource. [Online]. Available: http://www.ee.columbia.edu/∼dpwe/resources/matlab/rastamat/

[16] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

[17] S. Huang, X. Bian, G. Wu, and C. McLemore, *CALLHOME Mandarin Chinese Lexicon*. Linguistic Data Consortium, 1997.

[18] J. Yuan, N. Ryant, M. Liberman, A. Stolcke, V. Mitra, and W. Wang, "Automatic phonetic segmentation using boundary models," in *INTERSPEECH*, 2013, pp. 2306–2310.

[19] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Procedings of ICML*, 2010, pp. 807–814.

[20] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," *arXiv preprint arXiv:1207.0580*, 2012.

[21] C.-y. Tseng, *An acoustic phonetic study on tones in Mandarin Chinese*. Brown University, 1981.

[22] Q.-J. Fu and F.-G. Zeng, "Identification of temporal envelope cues in Chinese tone recognition," *Asia Pacific Journal of Speech Language and Hearing*, vol. 5, no. 1, pp. 45–58, 2000.

[23] Y.-Y. Kong and F.-G. Zeng, "Temporal and spectral cues in Mandarin tone recognition," *The Journal of the Acoustical Society of America*, vol. 120, no. 5, pp. 2830–2840, 2006.

[24] M. K. Jensen, "Recognition of word tones in whispered speech," *Word*, vol. 14, no. 2-3, pp. 187–96, 1958.

[25] M. Gao, "Tones in whispered chinese: articulatory features and perceptual cues," Ph.D. dissertation, University of Victoria, 2002.

[26] D. H. Whalen and Y. Xu, "Information for Mandarin tones in the amplitude contour and in brief segments," *Phonetica*, vol. 49, no. 1, pp. 25–47, 1992.

[27] B. Milner and X. Shao, "Prediction of fundamental frequency and voicing from mel-frequency cepstral coefficients for unconstrained speech reconstruction," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 15, no. 1, pp. 24–33, 2007.

[28] J. Darch, B. Milner, and S. Vaseghi, "Analysis and prediction of acoustic speech features from mel-frequency cepstral coefficients in distributed speech recognition architectures," *The Journal of the Acoustical Society of America*, vol. 124, no. 6, pp. 3989–4000, 2008.

[29] P. Boersma and D. Weenink, "Praat speech processing software," *Institute of Phonetics Sciences of the University of Amsterdam*. [Online]. Available: http://www. praat. org