INTERSPEECH 2016
September 8–12, 2016, San Francisco, USA

Tone Classification in Mandarin Chinese using Convolutional Neural Networks

Charles Chen[1], *Razvan Bunescu*[1], *Li Xu*[2], *Chang Liu*[1]

[1]Electrical Engineering and Computer Science, Ohio University, USA
[2]Communication Sciences and Disorders, Ohio University, USA

lc971015,bunescu,xul,liuc@ohio.edu

Abstract

In tone languages, different tone patterns of the same syllable may convey different meanings. Tone perception is important for sentence recognition in noise conditions, especially for children with cochlear implants (CI). We propose a method that fully automates tone classification of syllables in Mandarin Chinese. Our model takes as input the raw tone data and uses convolutional neural networks to classify syllables into one of the four tones in Mandarin. When evaluated on syllables recorded from normal-hearing children, our method achieves substantially higher accuracy compared with previous tone classification techniques based on manually edited F_0. The new approach is also more efficient, as it does not require manual checking of F_0. The new tone classification system could have significant clinical applications in the speech evaluation of the hearing impaired population.

Index Terms: tone classification, Mandarin Chinese, feature learning, convolutional neural networks

1. Introduction

In tone languages, such as Mandarin Chinese, variations of tone patterns (i.e., the fundamental frequency (F_0) and its harmonics) of each syllable convey lexical meaning. In Mandarin Chinese, there are four different patterns (thus four tones): (1) flat and high, (2) rising, (3) low and dipping, and (4) falling. Figure 1 shows the time waveforms (top), spectrograms (middle) and Mel-frequency cepstral coefficients (MFCCs, bottom) of four tones of the Mandarin Chinese syllable /yi/. Tones 1 through 4 associated with the syllable /yi/ may mean (1) "one", (2) "aunt", (3) "chair", and (4) "art", respectively. It has long been known that F_0 contours even in non-tonal languages, such as English, play a significant role in reducing the deleterious effects of noise, especially those of competing speech [1–3]. A couple of recent reports [4, 5] have shown the extent to which tone information can help circumvent the deleterious effects of noise in Mandarin Chinese sentence recognition. With appropriate tones, sentence recognition accuracy in steady-state noise at 0 dB signal-to-noise ratio was nearly perfect but reduced to about 70% when the tone information was removed or disrupted [5].

There is a critical need for larger-scale, more in-depth research on tone perception and production in children with CI in real-world situations (see [6] for a review). Such research requires detailed analysis of tone accuracy in the speech production of the children. Current computer-based tone classification tools used in the lab require manual checking of F_0 to improve classification accuracy and are therefore inefficient. In particular, environment noise can make the F_0 extraction unreliable. The accuracy of speech recognition has been improved in recent years using context-based optimization, as seen in commercial products such as Apple Siri, Google Now, and Microsoft Cortana. Tone classification is a sub-problem of the overall speech recognition problem and high accuracy of automatic tone classification of monosyllabic words without contextual speech is still a unique problem that has not been solved yet. It is therefore important to develop automatic tone classification tools that are robust and accurate in the presence of noise. This is the focus of our present research. In this study, we use unsupervised feature learning techniques to automate the tone classification task. We employ denoising and sparse autoencoders to learn feature kernels that are then used as filters in a convolution layer. The resulting feature maps are down-sampled through max-pooling and used as input to a softmax classifier. Experimental evaluations on a children speech dataset show that the best performing CNN model obtains a 95.5% tone classification accuracy, which is significantly higher when compared with previous models based on manually edited F_0 [7, 8].

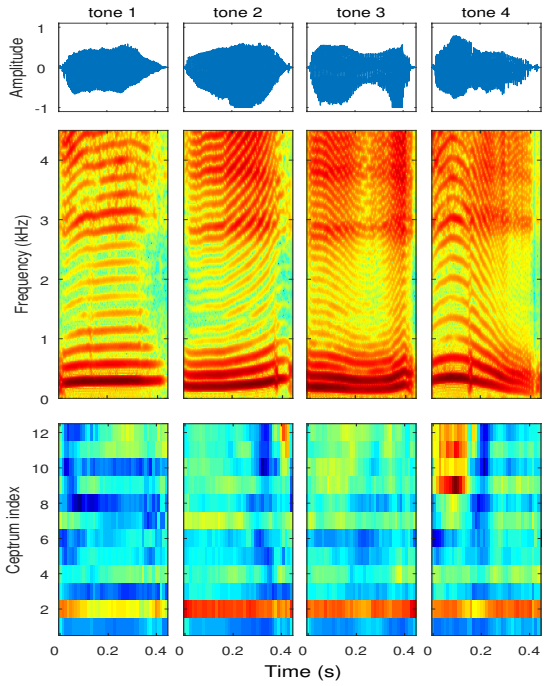

Figure 1: Four tones of Chinese syllable /yi/. Top: Time waveforms. Middle: Spectrograms. Bottom: MFCCs.

http://dx.doi.org/10.21437/Interspeech.2016-528

2. The Tone Classification Model

The overall pipeline architecture of our model is shown in Figure 2. The input waveform data is first preprocessed and transformed into MFCC vectors (Section 2.1). The MFCC vectors serve as input to a convolution layer using features that are pretrained with denoising autoencoders (Section 2.2). After convolution and pooling, the sound features are optionally concatenated with pooled MFCCs and used as input to a softmax layer that is trained to compute a probability for each tone category (Section 2.3).

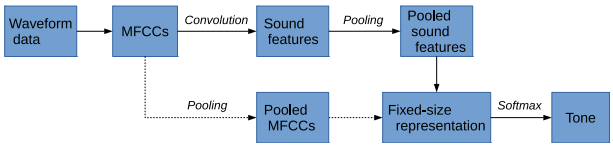

Figure 2: Tone classification pipeline. Top: The convolution and pooling layers extract a fixed-size set of sound features from MFCC input, as shown in more detail in Figure 3. Bottom: Raw MFCCs are pooled into a fixed-size vector of MFCC features.

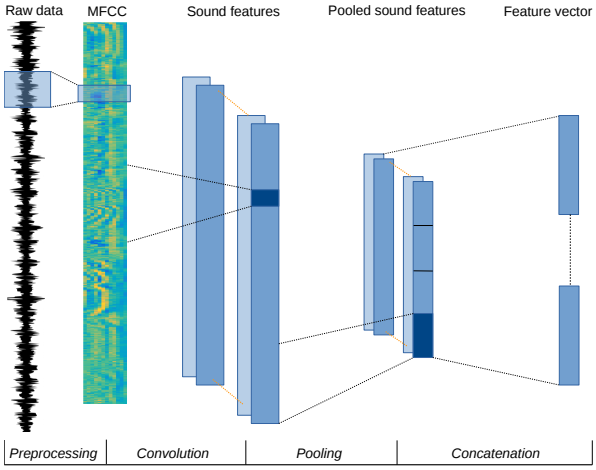

Figure 3: The CNN Architecture. *Preprocessing*: Convert the variable-length waveforms into MFCCs. *Convolution*: convolve the kernels learned by dAE on the input MFCCs, output K feature maps. *Pooling*: down-sample (pool) the K feature maps to produce K reduced D dimensional feature maps. *Concatenation*: concatenate the K reduced feature maps to form a (K * D) dimensional input for the output softmax layer.

2.1. Preprocessing

We use Mel-Frequency Cepstral Coefficients (MFCCs) as input. Each example is divided into 25 ms segments, with an overlap of 10 ms among segments. For each segment, 13 cepstral coefficients are extracted. Due to the varying duration of the input syllables, the numbers of MFCCs are different. Typically, each monosyllable contains 50 to 90 MFCCs. All the waveforms in the tone dataset are converted to MFCCs and ZCA whitened before being used as input in the CNN model.

2.2. Pretraining Sound Features

After the waveforms are preprocessed into MFCC vectors, we use denoising autoencoders [9] to learn K features kernels. First, a large number (N) of segments are extracted uniformly at random from the unlabeled dataset, using a receptive field of size W. A typical value for N is 100,000. W is a hyper-parameter to be tuned using the validation set. In order to sample each MFCC vector with the same probability, an input waveform is first sampled with a probability that is proportional to its length (i.e. the number of MFCCs), and then W consecutive MFCCs are randomly sampled from the MFCC representation of the selected waveform. This procedure is repeated N times. The resulting N samples are then subjected to a corruption process, where the level of artificial noise is controlled by the corruption parameter C. The N samples are partially corrupted by setting C percentage of the input values in the W MFCC vectors to be zero, as described in [9]. The N noisy samples are then fed into a denoising autoencoder (dAE) that learns the K feature kernels in the hidden layer by training to reproduce the uncorrupted samples in its output layer. K is also a hyper-parameter to be tuned on the validation data.

2.3. The CNN Architecture

Once the feature kernels are learned, they are used as filters in the convolution layer of a CNN, as shown in Figure 3 (corresponding to the top processing path in the pipeline from Figure 2). Instead of fully connecting the neurons between two adjacent layers, CNNs [10] enforce a local connectivity between the neurons in the adjacent layers, forcing the feature kernels to capture spatially-local correlations, regardless of their position in the input. Moreover, by sharing weights among the receptive fields across the entire example, CNN has less parameters to learn, largely expediting the learning procedure.

In the CNN architecture from Figure 3, the MFCCs obtained from the input waveform are convolved with the K kernels, using a stride of 1 between two adjacent convolution windows. The resulting K feature maps are passed to a max pooling layer, where they are down-sampled into K reduced D dimensional feature maps. This is achieved by partitioning the input of the pooling layer into D non-overlapping segments, and computing a max over the feature values in each segment. Thus, the pooling layer produces fixed-size feature maps for all the examples, regardless of the number of MFCCs in the original input. For example, if an example contains L MFCCs and an MFCC vector has 13 cepstral coefficients, then the input to the convolution layer has a size of 13 * L. Since the L MFCCs are convolved with the K kernels (each with input size 13 * W) using a stride of 1, this results in a matrix with K rows and L - W + 1 columns. During pooling, the matrix is divided into D sub-matrices, each of which has K rows and (L - W + 1) / D columns. The maximum value for each row is extracted from each sub-matrix, to produce a vector with the size K. The resulting D vectors, each with the size K, were concatenated to form the fixed-size K * D input for the softmax layer. Optionally, pooling is also performed on the MFCCs directly, as shown at the bottom of Figure 2, resulting in an additional fixed-size 13 * D input vector for the softmax layer. The softmax layer had 4 output classes, corresponding to the 4 tones in Mandarin Chinese.

3. Experimental Evaluation

The CNN architecture was trained using gradient descent and the feature kernels were fined-tuned by backpropagating the gradient through the pooling and convolution layers [11]. We

used the "early stop" strategy to determine when to stop the fine-tuning procedure. We implemented our method in Matlab and performed the experiments on a 12-CPU machine with 16GB memory. For autoencoders, we used 0.035 as the desired average activation of the hidden units, 5 as the sparsity penalty parameter and 0.003 as the regularization parameter. For softmax, we used 0.0002 for the weight of regularization term. The maximum number of iterations in the stochastic gradient descent was 2000. These are default values that have been used elsewhere[1] and were not tuned. The hyper-parameters that were tuned were as follows: number of pretraining samples N = 150,000; receptive field size W = 10; number of feature kernels K = 200; and the pooling size D = 4.

In the present study, we used a dataset of Mandarin Chinese Children Speech (MC-CS) syllables. This dataset contains spoken syllables recorded from 125 normal-hearing children (66 boys and 59 girls) with ages ranging from 3 to 10 years old. Each child produced 36 monosyllabic words in Mandarin Chinese. The 36 words were common Chinese words at the level of vocabulary of young children [12]. The production was elicited with pictures pertaining to the meaning of the words. The average duration of the four tones across all 125 children were 572, 606, 670, and 477 milliseconds, respectively. We partitioned the tone dataset into 10 equal sized folds based on children, each of which contained the examples recorded from 12 or 13 children. Each child appeared in exactly one fold. We then used 10-fold cross validation to evaluate our method, as follows: 1) select one fold for testing; 2) select one fold for validation; 3) use the remaining 8 folds for training. This procedure was repeated 10 times, each time selecting a different fold for testing. Although hyper-parameters were tuned on randomly selected validation folds, their values were always similar if not the same with the values listed above. For pretraining the K feature kernels, the input to the dAE was obtained by sampling from the training and validation folds, using the sampling procedure described in Section 2.2, with a corruption level C = 10%. Therefore, there were totally 10 sets of convolution kernels (filters), one set per test fold.

In order to comparatively evaluate the effectiveness of the features generated by our method and the features used in the previous methods by Xu et al. [7, 8], we experimented with the following models:

1. **ANN-(F_0)**: this is the approach from [7], which used neural networks trained on manually edited F_0.
2. **Softmax-(raw-MFCCs)**: this is a softmax classifier using directly the pooled raw MFCCs as features (lower branch in Figure 2).
3. **CNN-(dAE-MFCCs)**: this is our proposed CNN architecture employing features pretrained on MFCC vectors using a denoising autoencoder.
4. **CNN-(dAE-MFCCs+F_0)**: same as (3) above, but also adding the manually edited F_0 as input features for the softmax classifier.
5. **CNN-(raw+dAE-MFCCs)**: same as (3) above, but also using the pooled raw MFCCs as features for the softmax classifier.
6. **CNN-(raw+sAE-MFCCs)**: same as (5) above, but using a sparse autoencoder (sAE) instead of the dAE.

In Table 1, we report the average and standard deviation of the accuracy for these models over the 10 test folds. All models use softmax in the output layer. The results show that the proposed

[1] http://deeplearning.stanford.edu/tutorial/

Table 1: *Accuracy mean (μ) and stdandard deviation (σ) for the six models.*

Model	**Accuracy $\mu(\sigma)$ %**
ANN-(F_0) [7, 8]	88.35(4.06)
Softmax-(raw-MFCCs)	81.63 (6.04)
CNN-(dAE-MFCCs)	**95.53** (4.18)
CNN-(dAE-MFCCs+F_0)	95.39 (3.57)
CNN-(raw+sAE-MFCCs)	95.42 (3.80)
CNN-(raw+dAE-MFCCs)	95.40 (4.10)

CNN method with pretrained features outperforms substantially both the simple softmax using pooled raw MFCCs and the previous method of Xu et al. [7, 8] that uses manually edited F_0. The improvement in performance over the ANN-(F_0) model is statistically significant (p-value > 0.01 in a one-tailed T test). The results also show that:

- Adding the manually edited F_0 features to the best performing CNN model does not improve the performance. This indicates that the features learned by the dAE-CNN combination effectively subsume the manually edited F_0.
- Adding the pooled raw MFCCs to the best performing CNN model does not improve the performance. It actually seems to hurt performance, which may be due to overfitting caused by the larger number of parameters.
- Features learned with the sparse and denoising autoencoders lead to similar performance.

Furthermore, we evaluated the model CNN-(raw+dAE-MFCCs) using features that are pretrained under different levels of noise (ϵ) in the denoising autoencoder. The results in Table 2 show that increasing the level of noise during feature learning hurts the final performance of the CNN.

Table 2: Left: *Confusion matrix for the best performing model CNN-(dAE-MFCCs)*; Right: *Accuracy mean (μ) and standard deviation (σ) for the model CNN-(raw+dAE-MFCCs), using features pretrained under different noise levels (ϵ).*

Confusion Matrix				
Tone	T_1	T_2	T_3	T_4
T_1	1061	18	8	27
T_2	16	1038	41	5
T_3	7	34	1037	11
T_4	18	12	18	1060

ϵ	μ (σ), in %
0.1	95.51 (3.97)
0.2	95.35 (4.08)
0.3	95.40 (4.10)
0.4	95.43 (4.09)
0.5	95.15 (4.10)

Various approaches have been tried in previous research on tone recognition [7, 8, 13–18]. Besides the ANN-based method from [7, 8], close to our method are also the neural network approaches from [17, 18]. Lei et al. [17, 19] train a single-hidden-layer neural network using features such as the pitch contour and duration of the final part or the whole syllable, and the F_0 for the initial part of the syllable. The contour and F_0 features are sampled into a fixed number of points. The neural models are evaluated on a dataset that contains syllables (15 frames or longer) from one show in Mandarin Broadcast News speech, henceforth called MC-BN. The tone classification accuracy is shown in the first row of results in Table 3 in two settings: syllables classified in isolation (no context) vs.

syllables classified using co-articulation features (in context). Kalinli [18] proposed a biologically inspired neural model in which a set of multi-scale features is extracted from the sound spectrum. The features are transformed into a fixed-vector representation using mean pooling and PCA, and then provided as input to a neural network with one hidden layer. Like Lei et al. [17,19], Kalinli evaluates the role of co-articulation information, for various context sizes. The evaluation dataset contains continuous Mandarin Chinese speech (referred to as MC-CC) in the form of 7,513 command-and-control utterances from 8 female and 8 male speakers. The corresponding tone classification accuracy is shown in the second row of results in Table 3, for both the context-dependent and context-independent settings. The results obtained by the two previous approaches indicate that co-articulation features improve the tone classification accuracy, which makes the CNN system's performance in the context-independent setting even more impressive. We were unable to obtain the MC-CC [18] (due to legal restrictions) or MC-BN [17, 19] datasets. Therefore, while the results Table 3 appear to indicate that our CNN model compares favorably with the two previous neural approaches, it should be noted that these systems have been evaluated on different datasets, with different characteristics. In particular, the MC-CS dataset used to evaluate our models does not contain continuous speech. In terms of the label distribution, our MC-CS dataset is balanced and thus more difficult for a majority baseline classifier, which obtains 25% accuracy on MC-CS, versus 39.6% on MC-CC. Furthermore, speakers (125 children) that appear in testing are never used during training in our MC-CS dataset, whereas the evaluation in [18] randomly assigns syllables to training and test folds. Thus, the same speakers (8 female and 8 male) appear in both the training and testing folds of the MC-CC dataset, which is likely to make the tone classification task from [18] easier.

Table 3: *Tone classification results from two related models and our CNN-based approach.*

System	Dataset	In context	No context
Lei et al. [17]	MC-BN	76.2%	74.4%
Kalinli [18]	MC-CC	79.0%	72.8%
Our CNN model	MC-CS	–	**95.5%**

A notable differences between our CNN model and the neural models from [17, 18] is in the type of features used in the network input layer. The features in the CNN model are learned automatically from unlabeled speech data, whereas the features in [18] are manually engineered. Also, while the F_0 features were observed in [17] to improve the tone classification performance, they did provide any improvement when added to the set of automatically learned features used in our approach.

4. Related Work

Besides the methods that were already considered in Section 3 [7,8,17–19], there have been a number of related approaches for tone recognition. Kertkeidkachorn et al. [20] use Hidden Conditional Random Field (HCRF) to perform the tone classification in Thai. Both isolated word and continuous speech are used in the experimental evaluation. Hu et al. [21], extend their previous system YAAPT [22], adding spectral temporal features. YAAPT implements a feed-forward neural network with two hidden layers and 4 output nodes. The Shanghai region data from the Regional Accented Speech Corpus (RASC863) [23] is used in the evaluation. Similarly, Wu et al. [24] train a neural network with two hidden layers on the RASC863 dataset [23], using manually engineered features. The overall highest tone classification accuracy reported in [24] is approximately 76% using a combination of spectral and temporal features such as pitch contours (F_0), Discrete Cosine Transform Coefficients (DCTCs), and Discrete Cosine Series Coefficients (DCSCs). All of these methods [20–22, 24] use F_0-based features. In contrast, our approach does not rely on F_0, instead using features automatically learned from a MFCC-based representation.

In recent years, Deep Neural Networks (DNN) [25, 26] have been shown to be highly effective in acoustic modeling in speech recognition [27]. Ryant et al. [28] use a DNN for frame-level 5-tone classification and a single-layer neural network at segment (syllable) level. The segment-level models are trained to classify syllables from the 1997 Mandarin Broadcast News Speech corpus [29], using co-articulation features. When provided with only raw MFCCs as input, the method obtains an error rate of 16.86%. In [30], a DNN model is trained to classify each frame of speech into one of six classes: five tones and one no-tone. Each frame is represented by a 40-d MFCC vector. Input to the DNN is formed by concatenating the MFCCs for the frames around the center frame. The tone-bearing units (TBUs) are classified based on "tonal features", segment duration and contextual features. Experiments without context segments result in a segment error rate (SER) of 17.73%. However, contextual frames (frames before and after the center frames) are still used to classify each frame within the segment, whereas our method does not rely on any contextual information.

Convolutional Neural Networks (CNN) [10] are a type of biologically-inspired neural network that were initially designed to emulate the animal visual processing system. Lee et al. [31] used convolutional belief network to classify audio data and obtained good performance in audio classification tasks. Abdel-Hamid et al. [32] applied CNNs to phone recognition and reduced the error rate by 6%-10% compared with DNNs.

5. Conclusions and Future Work

Tone perception is important for tone languages, particularly in noisy listening conditions. Children with cochlear implants have significant deficits in tone development, consequently there is an urgent need to establish an automatic way to assess tone-production accuracy in these children. We propose a method that fully automates tone classification of syllables in Mandarin Chinese. Our model takes as input the raw tone data and uses convolutional neural networks to classify syllables into one of the four tones in Mandarin. When evaluated on syllables recorded from normal-hearing children, our method achieves substantially higher accuracy compared with previous tone classification techniques based on manually edited F_0. The new tone classification system could have significant clinical applications in the speech evaluation of the hearing impaired population. In future work, we plan to evaluate the CNN-based approach on continuous speech, thus enabling the learning and use of co-articulation features, which have been observed to improve accuracy in previous research [17, 18].

6. Acknowledgments

We would like to thank the anonymous reviewers for their helpful remarks. This study was supported in part by the NIH NIDCD Grant No. R15-DC014587.

7. References

[1] J. S. Laures and G. Weismer, "The effects of a flattened fundamental frequency on intelligibility at the sentence level," *Journal of Speech, Language and Hearing Research*, vol. 42, no. 5, pp. 1148–1156, 1999.

[2] C. Binns and J. F. Culling, "The role of fundamental frequency contours in the perception of speech against interfering speech," *The Journal of the Acoustical Society of America*, vol. 122, no. 3, pp. 1765–1776, 2007.

[3] S. E. Miller, R. S. Schlauch, and P. J. Watson, "The effects of fundamental frequency contour manipulations on speech intelligibility in background noise," *The Journal of the Acoustical Society of America*, vol. 128, no. 1, pp. 435–443, 2010.

[4] J. Wang, H. Shu, L. Zhang, Z. Liu, and Y. Zhang, "The roles of fundamental frequency contours and sentence context in Mandarin Chinese speech intelligibility," *The Journal of the Acoustical Society of America*, vol. 134, no. 1, pp. EL91–EL97, 2013.

[5] F. Chen, L. L. Wong, and Y. Hu, "Effects of lexical tone contour on Mandarin sentence intelligibility," *Journal of Speech, Language and Hearing Research*, vol. 57, no. 1, pp. 338–345, 2014.

[6] L. Xu and N. Zhou, "Tonal languages and cochlear implants," in *Auditory Prostheses*. Springer, 2011, pp. 341–364.

[7] L. Xu, X. Chen, N. Zhou, Y. Li, X. Zhao, and D. Han, "Recognition of lexical tone production of children with an artificial neural network," *Acta Oto-laryngologica*, vol. 127, no. 4, pp. 365–369, 2007.

[8] N. Zhou, W. Zhang, C.-Y. Lee, and L. Xu, "Lexical tone recognition with an artificial neural network," *Ear and Hearing*, vol. 29, no. 3, p. 326, 2008.

[9] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proceedings of the 25th International Conference on Machine Learning*. ACM, 2008, pp. 1096–1103.

[10] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural Computation*, vol. 1, no. 4, pp. 541–551, Dec. 1989.

[11] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, "Natural language processing (almost) from scratch," *The Journal of Machine Learning Research*, vol. 12, pp. 2493–2537, 2011.

[12] D. Han, B. Liu, N. Zhou, X. Chen, Y. Kong, H. Liu, Y. Zheng, and L. Xu, "Lexical tone perception with hiresolution and hiresolution 120 sound-processing strategies in pediatric Mandarin-speaking cochlear implant users," *Ear and Hearing*, vol. 30, no. 2, p. 169, 2009.

[13] P.-F. Wong and M.-H. Siu, "Decision tree based tone modeling for Chinese speech recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'04)*, vol. 1. IEEE, 2004, pp. I–905.

[14] S.-H. Chen and Y.-R. Wang, "Tone recognition of continuous Mandarin speech based on neural networks," *IEEE Transactions on Speech and Audio Processing*, vol. 3, no. 2, pp. 146–150, 1995.

[15] Y. Qian, "Use of tone information in Cantonese LVCSR based on generalized character posterior probability decoding," Ph.D. dissertation, The Chinese University of Hong Kong (People's Republic of China), 2005.

[16] G. Peng and W. S.-Y. Wang, "Tone recognition of continuous Cantonese speech based on Support Vector Machines," *Speech Communication*, vol. 45, no. 1, pp. 49–62, 2005.

[17] 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.

[18] O. Kalinli, "Tone and pitch accent classification using auditory attention cues," in *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2011, pp. 5208–5211.

[19] X. Lei, "Modeling lexical tones for Mandarin large vocabulary continuous speech recognition," Ph.D. dissertation, University of Washington, 2006.

[20] N. Kertkeidkachorn, P. Punyabukkana, and A. Suchato, "A hidden Conditional Random Field-based approach for Thai tone classification," *Engineering Journal*, vol. 18, no. 3, pp. 99–122, 2014.

[21] H. Hu, S. A. Zahorian, P. Guzewich, and J. Wu, "Acoustic features for robust classification of Mandarin tones." in *INTERSPEECH*, 2014, pp. 1352–1356.

[22] S. A. Zahorian and H. Hu, "A spectral/temporal method for robust fundamental frequency tracking," *The Journal of the Acoustical Society of America*, vol. 123, no. 6, pp. 4559–4571, 2008.

[23] A. Li, Z. Yin, T. Wang, Q. Fang, and F. Hu, "RASC863-A Chinese speech corpus with four regional accents," *International Conference on Speech and Language Technology & Oriental-COCOSDA 2004 (ICSLT-O-COCOSDA 2004)*, 2004.

[24] J. Wu, S. A. Zahorian, and H. Hu, "Tone recognition for continuous accented Mandarin Chinese," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2013, pp. 7180–7183.

[25] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.

[26] Y. Bengio, "Learning deep architectures for AI," *Foundations and Trends® in Machine Learning*, vol. 2, no. 1, pp. 1–127, 2009.

[27] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-R. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath *et al.*, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, 2012.

[28] N. Ryant, M. Slaney, M. Liberman, E. Shriberg, and J. Yuan, "Highly accurate Mandarin tone classification in the absence of pitch information," in *Proceedings of Speech Prosody*, vol. 7, 2014.

[29] L. D. Consortium *et al.*, "1997 Mandarin Broadcast News Speech (HUB4-NE)."

[30] N. Ryant, J. Yuan, and M. Liberman, "Mandarin tone classification without pitch tracking," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2014, pp. 4868–4872.

[31] H. Lee, P. Pham, Y. Largman, and A. Y. Ng, "Unsupervised feature learning for audio classification using convolutional deep belief networks," in *Advances in Neural Information Processing Systems*, 2009, pp. 1096–1104.

[32] O. Abdel-Hamid, A.-R. Mohamed, H. Jiang, L. Deng, G. Penn, and D. Yu, "Convolutional neural networks for speech recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 10, pp. 1533–1545, 2014.