Interspeech 2018
2-6 September 2018, Hyderabad

Unsupervised Discovery of Non-native Phonetic Patterns in L2 English Speech for Mispronunciation Detection and Diagnosis

Xu Li[1], *Shaoguang Mao*[2], *Xixin Wu*[1], *Kun Li*[3], *Xunying Liu*[1], *Helen Meng*[1,2]

[1]Department of Systems Engineering and Engineering Management,
The Chinese University of Hong Kong
[2]Tsinghua-CUHK Joint Research Center for Media Sciences, Technologies and Systems,
Graduate School at Shenzhen, Tsinghua University
[3]SpeechX Limited, Shenzhen

{xuli, wuxx, xyliu, hmmeng}@se.cuhk.edu.hk, msg16@mails.tsinghua.edu.cn, kli@speechx.cn

Abstract

Second language (L2) speech is often annotated with the native phoneme categories. However, we often observe that an L2 speech segment generally deviates from a canonical phoneme, and sometimes it is very difficult for linguists to annotate with any canonical phoneme label. We refer to these segments as non-native phonetic patterns. Existing approaches to mispronunciation detection and diagnosis (MDD) focus mainly on canonical mispronunciations, i.e. one canonical phoneme is substituted for another, aside from those deleted or inserted. To better represent L2 speech, this work explores non-native phonetic patterns (NN-PPs) of each native phoneme by an unsupervised approach. We apply an optimized k-means algorithm to cluster state-based phonemic posterior-grams, which are generated with a deep neural network. Then, to discover the NN-PPs related to each native phoneme, we perform forced alignment to divide L2 speech into segments grouped by native phonemes. We use the cluster sequences within segments derived from clustering results to represent different phonetic patterns of each native phoneme. Finally, we apply Cluster Sequence Analysis to discover each phoneme's potential NN-PPs. We verified experimentally that NN-PPs can extend the native phoneme categories to better describe L2 speech, which can enrich the existing approaches to MDD for better performance.

Index Terms: mispronunciation detection and diagnosis, non-native phonetic patterns, unsupervised clustering, sequence analysis, phonemic posterior-grams

1. Introduction

Our increasingly globalized world desires a growing demand for support in language acquisition and hence, the Computer-Aided Pronunciation Training (CAPT) systems. The core of CAPT, Mispronunciation Detection and Diagnosis (MDD), which aims at detecting mispronunciations occurring in second language (L2) speech and giving effective and corrective feedback.

Previous approaches [1–14] based on pronunciation scoring are popular and involve many types of confidence measures as pronunciation scores [1, 2, 4–9]. These approaches perform reasonably well on detection, but do not support diagnosis. Other methods, such as Extended Recognized Networks (ERNs) [15–17] and Acoustic-Phonemic Model (APM) [15] also perform well. ERNs incorporate manually designed or data-derived phonological rules to generate possible phoneme paths in the word pronunciation. These rules include not only the canonical phonemic path, but also common mispronunciations. The APM maps input features with both acoustic information and phoneme context information into phone-state posterior-grams for better performance on MDD tasks.

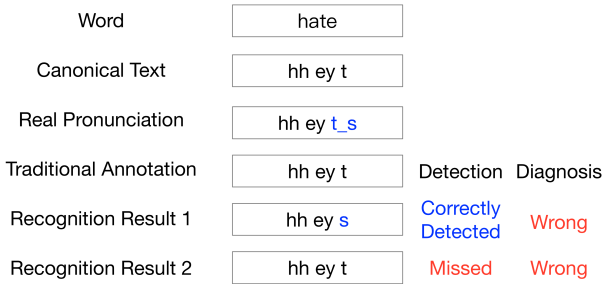

Figure 1: *An example for how non-native mispronunciations are wrongly treated in traditional MDD*

These existing approaches mostly model L2 speech with native phoneme units, but non-native speech often exhibits non-native deviations. For example, L2 English speech uttered by native Cantonese speakers often shows that the phoneme /t/ may be mispronounced as a sound that resembles both /t/ and /s/ (A sample audio of this case is provided for your reference). Fig. 1 shows an example where the canonical annotation for "hate" should be /hh ey t/, but with a non-native segment that resembles both /t/ and /s/, it may be recognized as either /hh ey s/ (Result 1), which enables mispronunciation detection but inaccurate diagnosis. Alternatively, if it is recognized as /hh ey t/ (Result 2), it will fail in both mispronunciation detection and diagnosis.

To solve this problem, our previous work [18] discovered an Extended Phoneme Set in L2 speech (L2-EPS) by using unsupervised clustering algorithm based on phoneme-based features. It gives a more complete description on pronunciation patterns in L2 speech, some of which are ignored by the native phoneme set. However, there are still some phonetic patterns that cannot be captured well by that approach. Because the phoneme-based phonemic posterior-grams (PPGs) used as clustering features in [18] cannot provide the state information within a segment, which is also important in reflecting the phonetic patterns.

This work focuses on state-based features and explores the non-native phonetic patterns (NN-PPs) for each phoneme, which can capture more complete phonetic patterns in L2 speech. In this approach, we cluster speech frames based on state-level features, and then generate a cluster sequence repre-

10.21437/Interspeech.2018-2027

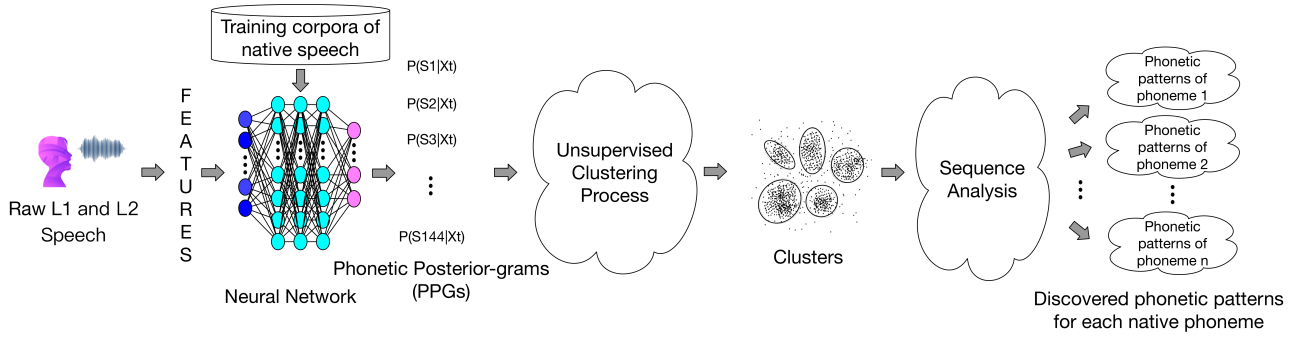

Figure 2: *Illustration of the proposed approach*

sentation of each phoneme segment. We focus on one phoneme at a time, and perform sequence analysis in a statistical way to obtain the NN-PPs of that phoneme. Finally, experiments are designed to verify the discovered NN-PPs. This work has the following contributions: (1) It proposes a framework to discover NN-PPs of each native phoneme to achieve better coverage of phonetic patterns in L2 speech. (2) The extended acoustic-phonemic coverage can be further used by the existing approaches to improve MDD performance.

2. Approach

The proposed approach is depicted in Fig. 2. To represent the articulation of speech sounds in a speaker-normalized space, we use PPGs [19–23] as features to cluster both L1 and L2 speech frames. Firstly, we extract features from the raw speech audio, and pass them through deep neural networks trained with native (L1) speech corpus to generate state-based Phonemic Posterior-grams (PPGs) (represented as a probability vector in Fig. 2, where each entry $P(S_i|X_t)$ denotes the posterior probability of observing state S_i given the input feature X_t) . Here we avoid using L2 speech to train the model, with the consideration that annotations of L2 speech are not accurate due to the existence of NN-PPs. Next, we apply an unsupervised clustering on the PPGs to generate different state-level clusters. Each cluster represents a state-based phonetic pattern. We then derive a cluster sequence for representing each phoneme segment. Finally, we perform Cluster Sequence Analysis (CSA) for each native phoneme to generate possible related phonetic patterns.

2.1. Neural Network Model Generating Phonemic Posterior-grams

To reduce the influence of speaker-dependent information, we introduce a deep neural network to map the acoustic features, i.e. Mel-Frequency Cepstral Coefficients (MFCC) of the raw speech audio, into a phonetic space. Since the size of L1 speech corpus (as the training set of the network) is not big, a traditional deep neural network (DNN) can sufficiently handle this work.

2.2. Unsupervised Clustering Process

To capture the variation of state-level phonetic patterns in L2 speech, we perform unsupervised clustering on the state-level PPGs.

To reduce the effect of noise, we first perform n-best filtering on state-level PPGs. It is done by preserving the first n largest values and setting the remaining to zero in each PPG vector. It has been shown that filtering can improve clustering performance [24, 25] by decreasing the influence of noise data. And next, we perform random initialization for k-means clustering and select the best result (in terms of a clustering metric) in ten independent experiments.

2.3. Cluster Sequence Analysis

We apply CSA within segments annotated as a native phoneme to statistically figure out its phonetic patterns.

2.3.1. Preprocessing

By forced alignment, we divide speech data into segments grouped by native phonemes. For each segment, we use a cluster sequence to represent it according to the clustering result. Then, we get a set of cluster sequences for each native phoneme. We perform CSA algorithm on the set of sequences related to each native phoneme to discover its phonetic patterns.

2.3.2. CSA Algorithm

The main process of CSA is illustrated in Fig. 3. It consists of 3 steps.

The first step is Sequence Filtering and Representation. It is operated individually on each sequence of the input set. Given a sequence, we firstly split it into small pieces by cluster ID. We then remove short pieces, the length of which is below a filtered threshold t (here we set t=2), from the sequence. We call this new sequence Filtered Cluster Sequence (FCS). Next, representation is performed on each FCS. This process only keeps a single cluster ID to represent each continuous piece of that cluster. Thereby, we get the Single Filtered Cluster Sequence (SFCS) in this step. Four sample sequences in Fig. 4 show the related FCS and SFCS derived from different Raw Cluster Sequences (RCS). Sequence Filtering is to reduce the influence of noise in RCS, which is caused by wrongly classified frames. And Sequence Representation is to extract the information of transition patterns among different clusters. We believe this information the most important in terms of reflecting phonetic patterns.

The second step is Types Selection. Following the previous step, we get a set of SFCS corresponding to a native phoneme. We then calculate the frequency for each SFCS type within this set. The dominating types with the frequency above a threshold c (to reach certain statistical threshold, here we set c=0.05) are kept for further processing. Thereby, we get a set of dominating types in this step. The motivation behind this is that we actually only care about the phonetic patterns (including both canonical or non-native ones) that frequently appear in L2 speech.

The final step is Types Merging. In this step, we traverse

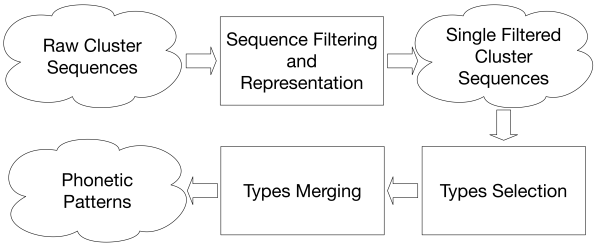

Figure 3: *The Flow Diagram of Cluster Sequence Analysis*

the set of types derived from Step 2. If a type is a sub-sequence of another, we merge the first into the second one. We then get a set of Merged SFCS (MSFCS). Any type in this set represents a phonetic pattern of the given native phoneme. Table 1 shows SFCS and MSFCS frequency result according to the samples in Fig. 4. Since Type "c a" is a sub-sequence of Type "a c a", we merge the first into the second one. The phonetic deviations between two types where one is a sub-sequence of another are much smaller than those between types consisting of different clusters or different cluster orders. Based on this, we merge similar phonetic types to generate distinct phonetic patterns.

Table 1: *"SFCS" and "MSFCS" frequency results derived from the samples shown in Fig. 4*

"SFCS" Types	Frequency	"MSFCS" Types	Frequency
aca	2/4	*aca*	3/4
ca	1/4	*bc*	1/4
bc	1/4		

2.3.3. *Apply CSA in exploring phonetic patterns*

We apply CSA algorithm to both L1 and L2 speech, and get two phonetic pattern sets for each native phoneme. We regard the one derived from L1 data as a reference of canonical phonetic patterns (C-PPs). The patterns not belonging to C-PPs derived from L2 data are regarded as NN-PPs. Since the deviation of a phoneme's C-PPs is subtle, we only retain the pattern with the highest frequency as the canonical phonetic pattern for each phoneme.

3. Experiments

3.1. Speech Corpus

Our experiments are based on two speech corpora: (1) the TIMIT data set as the L1 speech corpus; and (2) the Chinese University-Chinese Learners of English (CU-CHLOE) data set [26] as the L2 speech corpus. We use labeled data of 400 speakers in the L1 corpus as the training set and 63 speakers in the development set to train the DNN model for generating the posterior-grams as explained in Section 2.1. The rest of L1 corpus is used as the test set. Also, the labeled L2 corpus is used as the test set only. Finally, only the test set (consisting of both L1 and L2 data) is used for unsupervised clustering and further CSA in exploring phonetic patterns.

3.2. Experimental Setup

We implement clustering experiments with different configurations for comparison: (1) The K value in k-means algorithm

Figure 4: *An example of generating intermediate sequences in sample segments of a native phoneme*

Table 2: *Results of clustering measured using DBI (see Section 3.3)*

Features	MFCCs	PPGs from DNN
k=111	2.21	1.88
k=123	2.20	1.84
k=135	2.19	1.94
k=147	2.17	1.79
k=159	2.19	1.76
k=171	2.18	1.75
k=**174**	2.19	**1.68**
k=183	2.21	1.85
k=195	2.21	1.86
k=207	2.22	1.93
k=219	2.22	1.94
k=231	2.20	1.90

is set from 111 to 234 with step-length being 3. Since there are 144 traditional phoneme states in a mono phone system, we set the K value around it for results comparison. (2) Frame-level features used for clustering includes MFCC and state-level PPGs derived from DNN. (3) The n value in n-best filtering is empirically set to 3. All clustering processes are randomly initialized.

Based on experimentation, we choose the configuration of 4 hidden layers with 1024 units per layer and tanh as activation function for the DNN. We combine 11 frames (5 before, 1 current and 5 after) of MFCC as acoustic features to fed in the DNN. MFCCs are extracted by using 25-ms Hamming window and 10-ms frame shift.

3.3. Clustering Results Evaluation

We reference the Davies Bouldin Index (DBI) [27] (see Equation 1), which is widely used in clustering performance evaluation. It is defined as a function of the ratio of the within cluster scatter, to the between cluster separation. A lower value means that the clustering is better.

$$DBI = \frac{1}{N}\sum_{i=1}^{N} max_{j\neq i}\frac{S_i + S_j}{d_{i,j}} \tag{1}$$

where N is the number of clusters and S_i, $d_{i,j}$ are defined as following:

$$S_i = \frac{1}{|C_i|}(\sum_{X\in C_i} \|X - Z_i\|) \tag{2}$$

$$d_{i,j} = \|Z_i - Z_j\| \tag{3}$$

where Z_i is the centroid of cluster C_i, $|C_i|$ is the size of cluster C_i, $d_{i,j}$ is the distance between Z_i and Z_j (Manhattan distance). As shown in Table 2, we compare the state-based PPGs and MFCCs as clustering features by the DBI metric. To save space, the number of clusters are shown with a step being 12 to

Table 3: *The deviation between Canonical and Non-native Phonetic Patterns of some phonemes*

	ae	aw	ax	eh	ey	f	ix	iy	jh	t
Same	26.5%	35.8%	25.3%	29.0%	40.7%	44.4%	32.7%	42.1%	34.6%	34.6%
Different	66.7%	58.0%	69.8%	62.3%	53.1%	49.4%	61.1%	49.9%	61.7%	57.4%
Not Sure	6.8%	6.2%	4.9%	8.6%	6.2%	6.2%	6.2%	8.0%	3.7%	8.0%

illustrate the variation trend among different number of clusters. According to the results, we choose the clustering results with state-based PPGs extracted from DNN and k=174 for further experimental analysis.

3.4. Perceptual Tests On Discovered Non-native Phonetic Patterns

To verify that the discovered NN-PPs of each native phoneme are indeed different from canonical patterns, we designed a set of perceptual tests.

The perceptual tests are based on the pronunciation similarity comparison between different pronunciation patterns of each given native phoneme. To prepare our perceptual test set, we randomly select 10 audio files from each pattern for every native phoneme. And then, for every two patterns of each native phoneme, 5 comparison tests are conducted. For each test, we randomly select one audio from each pattern respectively. Our perceptual tests are conducted on the crowd sourcing platform of Amazon Mechanical Turk. The listeners are required to have background knowledge on English pronunciation and all of them are from countries where English is the official language. There are totally 50 listeners involved in this test. Subjects are asked "Whether the phonetic patterns in the two audios are the same or not" with 3 options being provided: 1) Yes; 2) No; 3) Not Sure.

Table 3 shows the comparison results between the canonical pattern and one of the most distinctive NN-PPs for some phonemes. According to the results, we can figure out the phonetic deviation between C-PPs and NN-PPs of some phonemes is obvious, such as "ae", "ax" and "eh". These phonemes can be easily mispronounced due to the substitution by similar phonemes in their mother language. But for some phonemes, such as "f" and "iy", the phonetic deviation between C-PPs and NN-PPs is subtle so that the proportion of choices "Same" and "Different" is very close. Especially, for the phoneme "t" mentioned at the beginning, the proportion of three choices is 34.6%, 57.4% and 8.0% respectively. For the patterns obviously different from canonical patterns, linguists annotate the segments of them as a native phoneme, while from the results, there are indeed some deviations. Therefore, these patterns belong to the NN-PPs of that native phoneme.

Table 4 shows the mean and standard deviation (std) statistical results of perceptual tests between canonical and non-native patterns among all native phonemes. From the results, we can figure out that the proportion of option "No" is higher than that of option "Yes" on average sense. This make sense because there is supposed to be some difference between our explored NN-PPs and C-PPs. And also, even though there are some phonemes with subtle deviation between NN-PPs and C-PPs, the standard deviation of each option is still not high. This verifies that our approach of exploring non-native pronunciation patterns make sense.

Table 4: *The statistical results of perceptual tests among all native phonemes*

	Yes	No	Not Sure
mean	37.6%	55.9%	6.5%
std	0.109	0.104	0.015

4. Conclusion

This work aims to find non-native phonetic patterns within each native phoneme to better describe L2 English speech. We first apply k-means clustering on state-based PPGs to generate different state phonetic patterns. Then, Cluster Sequences Analysis is applied to explore potential phonetic patterns within each native phoneme. According to the results, it is confirmed that these non-native phonetic patterns are different from the canonical ones. Besides, this approach works well on L2 English speech spoken by speakers of Cantonese. While during the approach, there is no specific limitations on native language of the speakers, it can generalize to L2 English produced by speakers of other L1s (such as Mandarin). But we have not done experiments to verify this yet.

How to describe these discovered NN-PPs and incorporate them with the existing approach to better solve the MDD problem will be conducted in the future.

5. Acknowledgements

This project is partially supported by a grant from the HKSAR RGC General Reseach Fund (Project no. 14207315).

References

[1] L. Neumeyer, H. Franco, M. Weintraub, and P. Price, "Automatic text-independent pronunciation scoring of foreign language student speech," in *Spoken Language, 1996. ICSLP 96. Proceedings., Fourth International Conference on*, vol. 3. IEEE, 1996, pp. 1457–1460.

[2] H. Franco, L. Neumeyer, Y. Kim, and O. Ronen, "Automatic pronunciation scoring for language instruction," in *Acoustics, Speech, and Signal Processing, 1997. ICASSP-97., 1997 IEEE International Conference on*, vol. 2. IEEE, 1997, pp. 1471–1474.

[3] C.-H. Jo, T. Kawahara, S. Doshita, and M. Dantsuji, "Automatic pronunciation error detection and guidance for foreign language learning," in *Fifth International Conference on Spoken Language Processing*, 1998.

[4] H. Franco, L. Neumeyer, M. Ramos, and H. Bratt, "Automatic detection of phone-level mispronunciation for language learning," in *Sixth European Conference on Speech Communication and Technology*, 1999.

[5] S. M. Witt and S. J. Young, "Phone-level pronunciation scoring and assessment for interactive language learn-

ing," *Speech communication*, vol. 30, no. 2-3, pp. 95–108, 2000.

[6] W. Menzel, D. Herron, P. Bonaventura, and R. Morton, "Automatic detection and correction of non-native english pronunciations," *Proceedings of InSTILL 2000*, pp. 49–56, 2000.

[7] S. Seneff, C. Wang, and J. Zhang, "Spoken conversational interaction for language learning," in *InSTIL/ICALL Symposium 2004*, 2004.

[8] J. Zheng, C. Huang, M. Chu, F. K. Soong, and W.-p. Ye, "Generalized segment posterior probability for automatic mandarin pronunciation evaluation," in *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, vol. 4. IEEE, 2007, pp. IV–201.

[9] F. Zhang, C. Huang, F. K. Soong, M. Chu, and R. Wang, "Automatic mispronunciation detection for mandarin," in *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*. IEEE, 2008, pp. 5077–5080.

[10] K. Truong, A. Neri, C. Cucchiarini, and H. Strik, "Automatic pronunciation error detection: an acoustic-phonetic approach," in *InSTIL/ICALL Symposium 2004*, 2004.

[11] H. Strik, K. Truong, F. De Wet, and C. Cucchiarini, "Comparing different approaches for automatic pronunciation error detection," *Speech communication*, vol. 51, no. 10, pp. 845–852, 2009.

[12] A. Lee and J. R. Glass, "Context-dependent pronunciation error pattern discovery with limited annotations," in *Fifteenth Annual Conference of the International Speech Communication Association*, 2014.

[13] X. Qian, H. Meng, and F. Soong, "A two-pass framework of mispronunciation detection and diagnosis for computer-aided pronunciation training," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 6, pp. 1020–1028, 2016.

[14] K. Li, X. Qian, and H. Meng, "Mispronunciation detection and diagnosis in l2 english speech using multidistribution deep neural networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 1, pp. 193–207, 2017.

[15] A. M. Harrison, W.-K. Lo, X.-j. Qian, and H. Meng, "Implementation of an extended recognition network for mispronunciation detection and diagnosis in computer-assisted pronunciation training," in *International Workshop on Speech and Language Technology in Education*, 2009.

[16] W.-K. Lo, S. Zhang, and H. Meng, "Automatic derivation of phonological rules for mispronunciation detection in a computer-assisted pronunciation training system," in *Eleventh Annual Conference of the International Speech Communication Association*, 2010.

[17] X. Qian, F. K. Soong, and H. Meng, "Discriminative acoustic model for improving mispronunciation detection and diagnosis in computer-aided pronunciation training (capt)," in *Eleventh Annual Conference of the International Speech Communication Association*, 2010.

[18] S. Mao, X. Li, K. Li, Z. Wu, X. Liu, and H. Meng, "Unsupervised discovery of an extended phoneme set in l2 english speech for mispronunciation detection and diagnosis," in *Acoustics, Speech and Signal Processing (ICASSP), 2018 IEEE International Conference on*. IEEE, 2018.

[19] Y.-B. Wang and L.-s. Lee, "Supervised detection and unsupervised discovery of pronunciation error patterns for computer-assisted language learning," *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 23, no. 3, pp. 564–579, 2015.

[20] L. Sun, K. Li, H. Wang, S. Kang, and H. Meng, "Phonetic posteriorgrams for many-to-one voice conversion without parallel data training," in *Multimedia and Expo (ICME), 2016 IEEE International Conference on*. IEEE, 2016, pp. 1–6.

[21] A. Lee and J. Glass, "A comparison-based approach to mispronunciation detection," in *Spoken Language Technology Workshop (SLT), 2012 IEEE*. IEEE, 2012, pp. 382–387.

[22] Y.-B. Wang and L.-S. Lee, "Toward unsupervised discovery of pronunciation error patterns using universal phoneme posteriorgram for computer-assisted language learning," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013, pp. 8232–8236.

[23] A. Lee, Y. Zhang, and J. Glass, "Mispronunciation detection via dynamic time warping on deep belief network-based posteriorgrams," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013, pp. 8227–8231.

[24] A. Lee, N. F. Chen, and J. Glass, "Personalized mispronunciation detection and diagnosis based on unsupervised error pattern discovery," in *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*. IEEE, 2016, pp. 6145–6149.

[25] Y.-L. Boureau, J. Ponce, and Y. LeCun, "A theoretical analysis of feature pooling in visual recognition," in *Proceedings of the 27th international conference on machine learning (ICML-10)*, 2010, pp. 111–118.

[26] H. Meng, W.-K. Lo, A. M. Harrison, P. Lee, K.-H. Wong, W.-K. Leung, and F. Meng, "Development of automatic speech recognition and synthesis technologies to support chinese learners of english: The cuhk experience," *Proc. APSIPA ASC*, pp. 811–820, 2010.

[27] D. L. Davies and D. W. Bouldin, "A cluster separation measure," *IEEE transactions on pattern analysis and machine intelligence*, no. 2, pp. 224–227, 1979.