



Mispronunciation Detection without Nonnative Training Data

Ann Lee, James Glass

MIT Computer Science and Artificial Intelligence Laboratory,
Cambridge, Massachusetts 02139, USA

{annlee, glass}@mit.edu

Abstract

Conventional mispronunciation detection systems that have the capability of providing corrective feedback typically require a set of common error patterns that are known beforehand, obtained either by consulting with experts, or from a human-annotated nonnative corpus. In this paper, we propose a mispronunciation detection framework that does not rely on nonnative training data. We first discover an individual learner’s possible pronunciation error patterns by analyzing the acoustic similarities across their utterances. With the discovered error candidates, we iteratively compute forced alignments and decode learner-specific context-dependent error patterns in a greedy manner. We evaluate the framework on a Chinese University of Hong Kong (CUHK) corpus containing both Cantonese and Mandarin speakers reading English. Experimental results show that the proposed framework effectively detects mispronunciations and also has a good ability to prioritize feedback.

Index Terms: Computer-Assisted Pronunciation Training (CAPT), Gaussian mixture model (GMM), Extended Recognition Network (ERN)

1. Introduction

With increasing globalization, there has been a rapid growth in the quantity of people with various native language (L1) backgrounds learning a second language (L2). Computer-assisted pronunciation training (CAPT) systems have gained popularity due to the flexibility they provide for empowering students to practice speaking skills at their own pace. With automatic speech recognition (ASR) technology, CAPT systems are able to provide automatic pronunciation assessment and corrective feedback to the students [1, 2, 3].

Initial research on mispronunciation detection started with likelihood-based scoring [4, 5]. While computing hidden Markov model (HMM)-based log-likelihood scores or log-posterior scores has the advantage of being L1-independent, it does not have the ability to provide corrective feedback on the type of errors that were made. To tackle this problem, prior work focused on specific phone pairs that are known to be problematic, and extracted acoustic phonetic features for classifier training [6, 7]. Under the supervised framework, exact pronunciation error types are part of the system output, and thus the pedagogical value of the system can be enhanced. Other work took a general approach in which possible error types are incorporated into the lexicon during recognition [8, 9, 10, 11]. These extended recognition networks (ERNs) have the advantage that the errors and the error types are detected together, and thus can be used for the system to provide diagnostic feedback.

While the approaches above have the benefit of being able to identify the error types, there exists the limitation that the common error patterns for a given L2, or an L1-L2 pair, have

to be known. The pronunciation error patterns that a system focuses on are typically extracted by either comparing human transcriptions (surface pronunciations) and canonical pronunciations from a lexicon (underlying pronunciations) [10, 11, 12, 13] or consulting with language teachers [9, 12]. As a result, the system’s assessment ability is constrained by the training data or experts’ input. To make matters worse, the data collection process is costly and has scalability issues, as it is impractical to collect data for every L1-L2 pair. This makes it hard to tailor a system to meet every student’s need. In contrast, it is relatively easy for native speakers to identify patterns that deviate from the norm without being trained on nonnative examples.

Our previous work [14] has shown that the pronunciation variation of an underlying phoneme on the learner-level is much lower than that on the corpus-level. Furthermore, more than 99% of the time, an individual learner pronounces an underlying phone in the same way if the triphone contexts remain the same. These findings indicate that we can effectively constrain the search space of possible pronunciation errors by focusing on one single learner and one particular triphone context at a time.

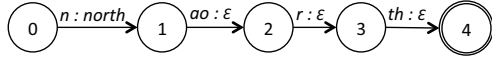
In this work, we propose a novel framework for mispronunciation detection based on our previous findings. The proposed framework is script-based, which provides texts for learners to read, and is built with a recognizer that is trained on native speech. We focus on phonemic pronunciation errors and attempt to discover an individual learner’s common error patterns by exploiting the acoustic similarities between speech segments produced by the learner. This procedure reduces the search space from the size of the whole phoneme inventory to a small set of error candidates that is learner-specific. In order to impose the constraint where an individual learner pronounces a triphone in only one way, instead of running one-pass forced alignment on each single utterance, we propose to run forced alignment on all utterances from a learner in an iterative and greedy manner using ERNs that are built based on the discovered error patterns.

The rest of this paper is organized as follows. The following section provides background on ERNs. Section 3 describes our system in detail. In Section 4, experimental results will demonstrate how the proposed system can effectively detect mispronunciations without requiring nonnative data for training. Section 5 concludes with a discussion of potential future work.

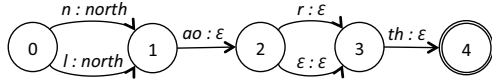
2. Extended recognition network (ERN)

In a finite state transducer (FST) based recognizer, the lexicon is represented as an FST that maps phoneme sequences to words. Fig. 1(a) shows an example of the FST representation of the word “north” in the lexicon. To deal with mispronunciations in nonnative speech, the FST can be enhanced by allowing multiple arcs corresponding to possible phoneme variations to be added, and thus form an ERN (see Fig. 1(b)). Running recog-

10.21437/Interspeech.2015-232



(a) An FST of the canonical pronunciation of the word “north”



(b) An FST representing an ERN of the word “north,” including one substitution error ($n \rightarrow l$) and one deletion error ($r \rightarrow \epsilon$)

Figure 1: An example of an ERN of the word “north” (ϵ denotes an empty string in FST input/output).

nition or forced alignment with the expanded FST will result in output phoneme sequences that may be different from the canonical pronunciations. For instance, if the decoded phoneme sequence from Fig. 1(b) is *ll ao r thl*, we can conclude that a substitution error has occurred ($n \rightarrow l$).

3. System Design

Fig. 2 shows the flowchart of the proposed system.

3.1. Selective speaker adaptation

Since the acoustic model of the recognizer is trained on native data, model adaptation can yield significant recognition accuracy improvements on nonnative speech [15, 16]. In our case, we perform Maximum A Posteriori (MAP) adaptation using the learner’s input utterances. One problem of adapting with all available material from the learner is that the model can easily be over adapted to mispronunciations. As a result, a selective speaker adaptation scheme is proposed. We compute goodness of pronunciation (GOP) score [4], which is the duration normalized absolute difference between the log-likelihood of a phoneme segment from forced alignment and the log-likelihood score from phoneme recognition within that segment. The larger the difference is, the more likely that the segment is mispronounced. Only segments whose GOP score is below a threshold are used for adaptation. In Section 4, we will show how adjusting this threshold affects system performance.

3.2. Error candidate selection

Forced alignment with the adapted acoustic model produces a set of underlying phoneme segments. Assuming that there are N phonemes in the phoneme inventory, for each underlying phoneme segment, there are $O(N)$ possible pronunciations. The goal of error candidate selection is to select a subset of phonemes that represent possible errors for each segment.

We propose to identify phoneme confusion pairs by exploiting the acoustic similarities between speech segments. Intuitively, if segments of underlying phoneme α are very close to segments of underlying phoneme β , it is likely that there are substitution errors ($\alpha \rightarrow \beta$, $\beta \rightarrow \alpha$, or $\alpha \rightarrow \gamma$ and $\beta \rightarrow \gamma$, where γ is another phoneme which is close to both), insertion errors (e.g. $\alpha \rightarrow \alpha\beta$), or deletion errors (e.g. $\alpha \rightarrow \epsilon$ and α has right or left context of β). We take into account both global distance between phone classes, as well as local distance between phone segments to determine the confusion pairs.

3.2.1. Global candidate selection

We approach the problem of finding major error patterns by examining how close each pair of phoneme classes is. Each underlying phoneme class can be modeled by a Gaussian mixture

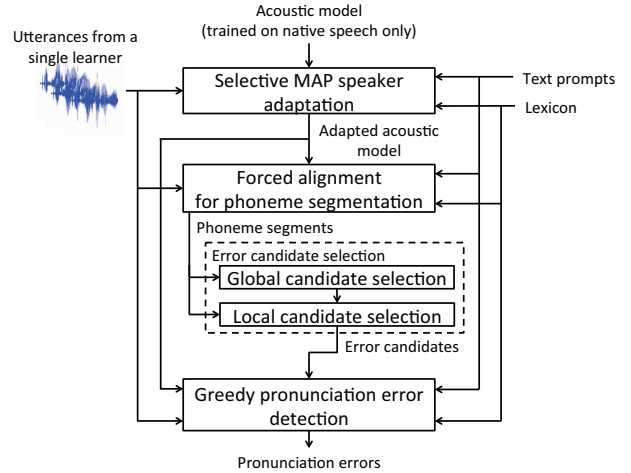


Figure 2: System flowchart. The system gathers a number of utterances from a single learner. After speaker adaptation, a two-stage process is carried out to determine possible context-dependent pronunciation error candidates. With iterations of forced alignment based on the candidates, the final output error patterns are selected in a greedy manner.

model (GMM). On the basis of the idea that each component in a GMM is likely to capture one type of surface pronunciation [17], the process can be described as follows:

1. Train an N_i -component GMM for each phoneme class p_i using frames of segments with underlying label p_i .
2. Compute the global distance between every pair of phoneme classes. Let g_k^i be the k -th component in the GMM of class p_i . The global distance between phoneme classes p_i and p_j can be computed as

$$D_G(i, j) = \min_{\substack{n_i = \{1, 2, \dots, N_i\} \\ n_j = \{1, 2, \dots, N_j\}}} BD(g_{n_i}^i, g_{n_j}^j), \quad (1)$$

where $BD(\cdot)$ is the Bhattacharyya distance between two multivariate Gaussian distributions, which has been used in building phonology structure [18, 19]. In other words, the distance between two phoneme classes is determined by their closest components.

At the end of this stage, for every phoneme class p_i , there will be a set of global error candidates, $C_G^i = \{p_j | j \neq i, D_G(i, j) \leq \tau_G\}$, where τ_G is a threshold on the distance.

3.2.2. Local candidate selection

The set C_G^i can be viewed as error patterns for each monophone, i.e. no phoneme context information is considered. Previous work has demonstrated that modeling context-dependent error patterns achieves better performance than modeling context-independent error patterns [10]. In this stage, we further compute distance between every pair of phoneme segments to refine C_G^i into a set of local error candidates that is triphone-specific. We average the normalized MFCCs at three regions within each phoneme segment: 0%-30%(start), 30%-70%(middle), 70%-100%(end), and concatenate the three averaged MFCCs to form a single vector for each segment. The Euclidean distance is then computed between every pair of vectors.

The local error candidates of a segment come from the intersection of the global error candidate set and the underlying

pronunciations of its nearest neighbors (segments whose distance to it is $\leq \tau_L$, where τ_L is another threshold). We compute distances across all utterances, gather local candidates of segments under the same triphone context, and form a set of error candidates for every triphone pattern.

3.3. Greedy pronunciation error detection

With the set of error candidates for each triphone pattern, a *candidate_list* consisting of substitution, insertion and deletion error patterns with respect to each triphone can be generated. For example, if a triphone pattern $\alpha\beta\gamma$ has δ as an error candidate, we consider the possibility of β being substituted with δ under the triphone context, and δ being inserted next to β (either before or after). Deletion error is considered for every triphone.

ERNs can be built by incorporating error patterns in the *candidate_list*. In order to enforce the constraint that only one pronunciation rule for a triphone should be selected from the *candidate_list*, we propose to run forced alignment iteratively to decode surface pronunciations and select error patterns in a greedy manner. Given the *candidate_list* as input, the algorithm works as follows:

0. Initialize *error_list* as an empty list. Start iterating with the current best score set as the likelihood score from forced alignment with a canonical lexicon.
1. In each iteration, run multiple forced alignments. At each alignment, incorporate only one error pattern from the *candidate_list*, together with those already in the *error_list*, into the lexicon to build the ERN.
2. Pick the error pattern from the *candidate_list* that produces the maximum likelihood score in decoding.
3. If the score improves upon the current best score, move the pattern to the *error_list* and update the current best score. For the rest of the error patterns, those with scores worse than the current best score are removed from the *candidate_list*.
4. If the score is worse than the current best score, or the *candidate_list* becomes empty after updating, the process is completed.

In the end, the algorithm outputs the *error_list*, an ordered list of learner-specific context-dependent error patterns, which is also the final output of the system.

4. Experiments

4.1. Corpus

The Chinese University Chinese Learners of English (CU-CHLOE) corpus consists of two parts: 100 Cantonese speakers, including 50 males and 50 females, and 111 Mandarin speakers, including 61 males and 50 females, both reading a set of specially-designed English scripts [9]. The scripts range from minimal pairs, confusable words, phonemic sounds, and TIMIT scripts to the story “*the north wind and the sun.*” In this work, we focus on the scripts from the story, and all the utterances are fully transcribed by an expert.

4.2. Experimental setting

Table 1 shows the division of the corpus for our experiments. All waveforms are transformed into 39-dimensional MFCCs every 10-ms, including first and second order derivatives. Cepstral mean normalization (CMN) is done on a per speaker basis. The GMM-HMM-based recognizer for forced alignment

Table 1: *Division of the corpus for experiments*

L1	Speakers	# instances
Training (for baselines)		
Cantonese	25 males, 25 females	19,218
Mandarin	25 males, 25 females	19,173
Testing		
Cantonese	25 males, 25 females	19,227
Mandarin	36 males, 25 females	23,361

has a monophone acoustic model trained on the TIMIT training set [20] using the Kaldi toolkit [21].

Two settings of the proposed framework are tested. The first setting runs the full system (*error candidates + greedy*), while the second setting skips the error candidate selection step and runs greedy pronunciation error detection through the whole phoneme inventory space (*greedy*). All the GMMs trained in error candidate selection have one shared diagonal covariance per phoneme class, with at most three components, depending on the number of frames of the phone class. As the GMMs are randomly initialized, we repeat the candidate selection process 10 times and take the intersection of the results as the final candidate set. The thresholds τ_G and τ_L are empirically chosen so that on average each triphone pattern has five candidates.

Both unsupervised and supervised baselines are implemented. For the unsupervised baseline, we run phoneme recognition, compare the output with the lexicon, and detect mispronunciation when there is mismatch between the two (*phone-rec*). It is unsupervised since no nonnative training data is required. In the supervised baseline, we compile error patterns from the training data of the same L1, build an ERN, and run one-pass forced alignment. Both context dependent (*supervised (c-d)*) and context independent (*supervised (c-ind)*) error patterns are considered. We evaluate the performance on both phoneme-level and word-level. Different thresholds on GOP scores for selective speaker adaptation are examined. It is adjusted so that in each scenario, 0%, 30%, 50%, 80%, 90% or 100% of the frames are used for adaptation, respectively.

4.3. Phoneme-level evaluation

For phoneme-level evaluation, three metrics are computed: *i*) false rejection rate (FRR): the ratio between the number of correct phonemes that are misidentified as being mispronounced and the total number of correct phonemes, *ii*) false acceptance rate (FAR): the ratio between the number of incorrect segments that are accepted by the system as correct and the number of all the incorrect phonemes, and *iii*) diagnostic error rate (DER): the percentage of the correctly detected pronunciation errors that have incorrect diagnostic feedback. Fig. 3 shows the results of FRR and FAR, and Fig. 4 shows the results of DER on the Cantonese and Mandarin test sets, respectively. In general, using more frames for adaptation makes the acoustic model fit better to the learner’s speech, including correct and incorrect pronunciations, and thus results in lower FRR and higher FAR.

Running the full system performs better than running greedy decoding only. The slight improvement in FRR and FAR and the average 8.7% absolute improvement in DER indicates that the error candidate selection process removes lots of noise and produces a reasonable candidate set for decoding. In fact, the error candidate selection process reduces the search space by more than 80%, which greatly decreases the system’s running time. However, running only error candidate selection with one-pass forced alignment deteriorates the performance.

On the Cantonese test set, the proposed full system has

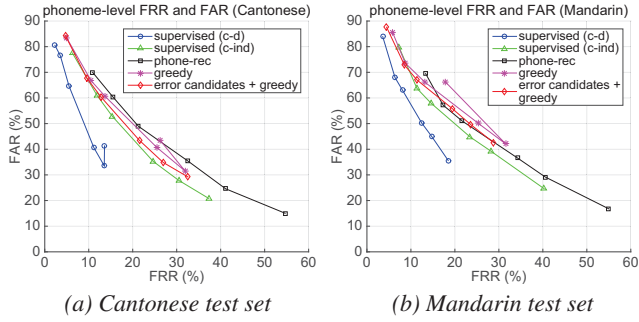


Figure 3: False rejection rate (FRR) and false acceptance rate (FAR) of three settings of the proposed framework, one unsupervised baseline from phoneme recognition, and two supervised baselines described in Sec. 4.2.

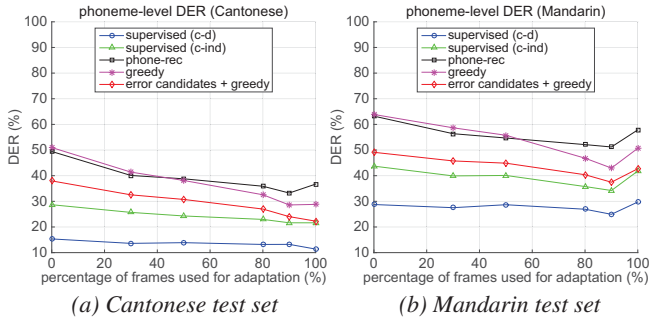


Figure 4: Diagnostic error rate (DER) of different test scenarios with respect to various degrees of speaker adaptation. Refer to Sec. 4.2 for the explanation of each system setting and baseline.

an average 3.5% absolute improvement on FRR under the same FAR, and 9.9% absolute improvement on DER over the phoneme recognition baseline. On the Mandarin test set, while the performance in FRR and FAR are similar, there is 12.5% absolute improvement in DER. These improvements come from the greedy decoding process. Forcing phonemes under the same triphone context to have the same surface pronunciation has the effect of voting, which performs better than making decisions individually due to the characteristics of L2 speech.

To analyze the gap between the performance of the proposed framework and the context-dependent supervised baseline, we run greedy pronunciation error detection using the error patterns compiled from training data. The result has small improvement in FRR and FAR over the supervised system. This indicates that the gap is due to the quality of the candidate set. We examine the coverage of the candidate set and find that 67.2% of the ground truth pronunciation error patterns are covered in the supervised system, while on average 49.5% of the error patterns are covered in the error candidate set. While increasing τ_G and τ_L can increase the coverage, the benefit from the candidate selection process will gradually disappear. To improve the process, analysis on a finer level can be incorporated in distance computation, e.g. alignment-based features from frame-wise dynamic time warping [22].

4.4. Word-level evaluation

In terms of pedagogical value, CAPT systems need not return all the errors at once, which may discourage a learner [23, 24]. Instead, the system should be able to prioritize its feedback, and the precision of the feedback is crucial. We believe word-level feedback is a good unit for learners to start focusing on prac-

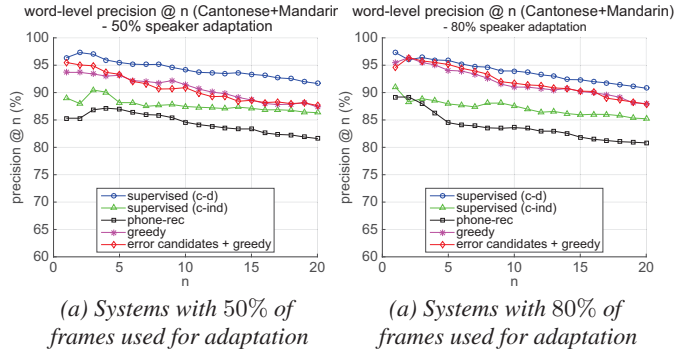


Figure 5: Precision @ n ($n = 1$ to 20) of the word-level feedback provided by each test scenario. Refer to Sec. 4.2 for the explanation of each system setting and baseline.

ticing. As a result, we examine a system’s ability to prioritize its word-level feedback by computing precision at n , which is the precision of the top n results in a ranked list. For the baselines, words with more phoneme errors and errors with higher GOP scores have higher priority, while the rank in the proposed framework is based on the order in the *error_list*.

Due to space constraints, only results from using 50% and 80% of frames for speaker adaptation are shown in Fig. 5. The relative order in performance is similar across all adaptation scenarios. Phoneme recognition performs the worst because of its high FRR. The proposed framework consistently performs better than a context-independent supervised system. As the degree of adaptation increases, the gap between the proposed system and the context-dependent supervised system decreases.

One advantage of the proposed framework is that the *error_list* is an ordered list of error patterns. To generate n words for feedback, it does not have to run until the *candidate_list* becomes empty. Since the pattern chosen in each iteration maximizes the improvement in overall likelihood score, it is usually related to phonemes with longer duration or more frequent occurrences. Therefore, the feedback provided by the system are not only precise, but also reflect more substantial errors.

5. Conclusion and future work

In this paper, we have presented a mispronunciation detection framework that does not require expert input, or nonnative training data. The proposed framework exploits acoustic similarities between segments from an individual learner’s utterances to discover possible pronunciation error candidates, and imposes triphone context constraints to decode mispronunciations. Treating each learner individually not only has the benefit of removing speaker variations in speech, but also echoes the concept of personalization in CAPT. Experimental results have shown that the proposed system outperforms a phoneme recognition framework, which also does not require any nonnative training data, while there is indeed room for improvement compared with a context-dependent supervised system.

In theory, the proposed framework is L1-independent and can be portable to any L2 as long as there is a speech recognizer available. In the future, we would like to run experiments on a larger variety of L1-L2 pairs. Also, we plan to design an interface for user studies.

6. Acknowledgements

This project is supported by Quanta Computers, Inc. The authors would like to thank Helen Meng for the CUHK corpus, and David Harwath for help with the Kaldi recognizer.

7. References

- [1] A. Neri, C. Cucchiari, H. Strik, and L. Boves, "The pedagogy-technology interface in computer assisted pronunciation training," *Computer assisted language learning*, 2002.
- [2] M. Eskenazi, "An overview of spoken language technology for education," *Speech Communication*, 2009.
- [3] S. M. Witt, "Automatic error detection in pronunciation training: where we are and where we need to go," in *Proc. ISADEPT*, 2012.
- [4] S. Witt and S. Young, "Phone-level pronunciation scoring and assessment for interactive language learning," *Speech Communication*, 2000.
- [5] H. Franco, L. Neumeyer, M. Ramos, and H. Bratt, "Automatic detection of phone-level mispronunciation for language learning," in *Proc. Eurospeech*, 1999.
- [6] H. Strik, K. Truong, F. De Wet, and C. Cucchiari, "Comparing different approaches for automatic pronunciation error detection," *Speech Communication*, 2009.
- [7] I. Amdal, M. H. Johnsen, and E. Versvik, "Automatic evaluation of quantity contrast in nonnative Norwegian speech," in *Proc. SLaTE*, 2009.
- [8] J. Kim, C. Wang, M. Peabody, and S. Seneff, "An interactive English pronunciation dictionary for Korean learners," in *Proc. Interspeech*, 2004.
- [9] H. Meng, Y. Lo, L. Wang, and W. Lau, "Deriving salient learners' mispronunciations from cross-language phonological comparisons," in *Proc. ASRU*, 2007.
- [10] A. M. Harrison, W. Y. Lau, H. M. Meng, and L. Wang, "Improving mispronunciation detection and diagnosis of learners' speech with context-sensitive phonological rules based on language transfer," in *Proc. Interspeech*, 2008.
- [11] W. K. Lo, S. Zhang, and H. Meng, "Automatic derivation of phonological rules for mispronunciation detection in a computer-assisted pronunciation training system," in *Proc. Interspeech*, 2010.
- [12] C. Cucchiari, H. Van den Heuvel, E. Sanders, and H. Strik, "Error selection for ASR-based English pronunciation training in "my pronunciation coach"," in *Proc. Interspeech*, 2011.
- [13] H. Hong, S. Kim, and M. Chung, "A corpus-based analysis of Korean segments produced by Japanese learners," in *Proc. SLaTE*, 2013.
- [14] A. Lee and J. Glass, "Context-dependent pronunciation error pattern discovery with limited annotation," in *Proc. Interspeech*, 2014.
- [15] Z. Wang, T. Schultz, and A. Waibel, "Comparison of acoustic model adaptation techniques on non-native speech," in *Proc. ICASSP*, 2003.
- [16] H. Ye and S. Young, "Improving the speech recognition performance of beginners in spoken conversational interaction for language learning," in *INTERSPEECH*, 2005.
- [17] Y.-B. Wang and L.-S. Lee, "Toward unsupervised discovery of pronunciation error patterns using universal phoneme posteriorgram for computer-assisted language learning," in *Proc. ICASSP*, 2013.
- [18] B. Mak and E. Barnard, "Phone clustering using the Bhat-tacharyya distance," in *Proc. ICSLP*, 1996.
- [19] N. Minematsu, "Mathematical evidence of the acoustic universal structure," in *Proc. ICASSP*, 2005.
- [20] J. Garofolo, L. Lamel, W. Fisher, J. Fiscus, D. Pallett, N. Dahlgren, and V. Zue, "TIMIT acoustic-phonetic continuous speech corpus," *Linguistic Data Consortium*, 1993.
- [21] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The Kaldi speech recognition toolkit," in *Proc. ASRU*, 2011.
- [22] A. Lee and J. Glass, "A comparison-based approach to mispronunciation detection," in *Proc. SLT*, 2012.
- [23] R. Ellis, "Corrective feedback and teacher development," *L2 Journal*, 2009.
- [24] H. Wang, X. Qian, and H. Meng, "Predicting gradation of L2 english mispronunciations using crowdsourced ratings and phonological rules," *Proc. SLaTE 2013*, 2013.