INTERSPEECH 2023
20-24 August 2023, Dublin, Ireland

Automatic Deep Neural Network-Based Segmental Pronunciation Error Detection of L2 English Speech (L1 Bengali)

Puja Bharati, Sabyasachi Chandra, Shayamal Kumar Das Mandal

Indian Institute of Technology Kharagpur 721302, India

pujabharati@iitkgp.ac.in, sabyasachichandra@iitkgp.ac.in, sdasmandal@cet.iitkgp.ac.in

Abstract

In the last few decades, English has become a popular language as it helps us to communicate with the global world. A large population of English learners find it challenging to achieve an 'acceptable' and 'intelligible' pronunciation. To overcome these issues, various computer-assisted pronunciation training tools are designed where automatic pronunciation error detection (APED) is a core component of the system. Most of the works of APED are based on European English speech, but there is no such work reported for Bengali English speech. This paper proposes a system for pronunciation error detection of L2 English speech (L1 Bengali) at the phoneme/segmental level using a hybrid convolutional neural network and long short-term memory modules with CTC loss. Experiments are done based on newly created L2 English speaker (L1 Bengali) speech data. The results demonstrate that the proposed system outperforms the goodness of pronunciation-based methods by 15% in terms of F1 score using fbank.

Index Terms: Phoneme detection, automatic pronunciation error detection, convolutional neural network, computer-assisted pronunciation training, long short-term memory.

1. Introduction

As globalization increases, there is rising in the population of English learners. Since there is such a high demand for learning English as a second or foreign language, there is a scarcity of qualified English teachers. Due to this scarcity, technology-based learning emerged, and applications for computer-assisted language learning (CALL) [1, 2] can enhance current learning materials and provide the student with special advantages in terms of accessibility, anxiety reduction, and personalised training. Computer-assisted pronunciation training (CAPT) [3], which is a subset of CALL, is an effective language learning tool.

For the learner of second languages (L2), CAPT systems offer options for independent language study. It can enhance instructors' lesson plans, provide tailored feedback, and lessen the teacher shortage issue. The pronunciation error detection module is a crucial part of CAPT systems since it helps to identify specific mispronunciation segments and provides diagnosis feedback at the phone level.

Bengali is one of the most spoken languages (ranking seventh) in the world, with nearly 300 million total speakers [4]. The Bengali language arose from eastern Middle Indo-Aryan dialects of Magadhi Prakrit and Pali. It is the official state language of the Eastern Indian state of West Bengal and the national language of Bangladesh [5]. As phonological patterns or phoneme set of Bangla is different from English [6, 7], it becomes difficult for a Bengali speaker to speak English as fluent as a native speaker [8].

There has been a lot of study on pronunciation error detection [9, 10, 11], and these techniques may be divided into two groups. The first is pronunciation evaluation using confidence metrics such as phone duration, phone posterior probability scores, and segment duration scores that were first suggested for automated speech recognition (ASR). The goodness of pronunciation (GOP) scores [10] are calculated using log-posterior probability based on force alignment. This approach and its well-known variations are now very well-liked and provide promising results for the identification of pronunciation errors. However, this type of approach is not only unable to address pronunciation insertion mistakes but also fails to provide learners with a more thorough diagnostic.

The second group of techniques seeks to evaluate the specifics of pronunciation errors, offering diagnosis input on particular problems, including phone substitutions, deletions, and insertions. A well-known method in this field is the extended recognition network (ERN) [11, 12] method, which adds phonological rules to the ASR decoding network and may immediately provide diagnosis feedback based on the comparison between an ASR output and the related phoneme sequences. These ERNs created using custom rules, or data-driven algorithms have the benefit of detecting mistakes and their types (insertion, deletion and substitution) simultaneously. On the other hand, finding and incorporating adequate phonological rules into the decoding network for all native and non-native language pairings could be difficult. Furthermore, excessive phonological limitations would lower ASR precision, which would have a negative impact on automatic pronunciation error detector performance.

Recently, applying a deep neural network (DNN) [13] to a CALL system has received much attention because of advancements in deep learning for ASR [14, 15]. DNN acoustic models outperformed the first two categories (GOP and ERN based methods) in the field of ASR. Phone recognition supports a CAPT tool in giving student-specific feedback at the phone level. But for non-native speech, however, extremely accurate phone identification is difficult [16]. This is because of interference of the learner's first language or mother tongue to the second language. Different region learners have a different accent. Also, detecting the acoustic-phonetic features more suitable for pronunciation error detection is still an open research area.

This paper presents L2 English (L1 Bengali) speaker speech pronunciation error detection. The contributions of this work are (i) to design a simple pipeline structure for pronunciation error detection which comprises a feature extractor, phone recognizer and pronunciation error detector, (ii) to compare the performance of various acoustic features for error detection, and (iii) to evaluate the system using our created dataset of Bengali

10.21437/Interspeech.2023-1481

speakers' English speech.

The remainder of the paper is divided into the following sections. Section 2 provides a brief overview of the proposed system design. Experiments and their results are demonstrated in section 3. Finally, this paper is concluded with some discussion about future potential work in section 4.

2. Proposed system design

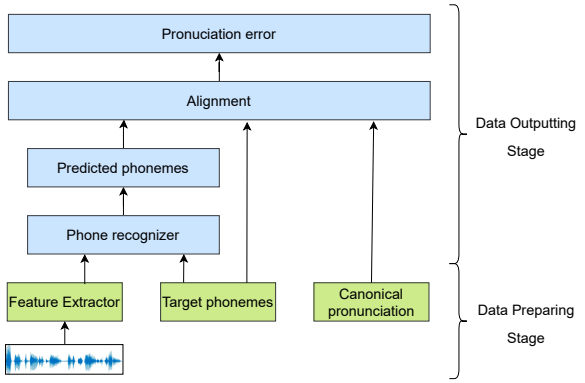

Figure 1: *Functional block diagram of pronunciation error detection system*

For pronunciation error detection, the input of the system is acoustic features extracted from the speech waveform such as spectrogram, PLP, MFCC and fbank and target text(or human annotated text) at the phone level. The phone recogniser consists of hybrid convolutional neural network(CNN) - Long short-term memory (LSTM) modules with connectionist temporal classification (CTC) loss inspired by CNN-RNN-CTC [17]. It predicts the possible phones as an output. The predicted phones are then compared with canonical (native-English) pronunciation using a pronunciation error detector. It gives us a predicted pronunciation error. The ground truth of pronunciation error or actual pronunciation error is acquired by comparing human-annotated text and canonical pronunciation text at the phone level. System performance can be measured from predicted and actual pronunciation errors. This process is depicted in figure 1.

2.1. Feature extractor

Features are extracted from speech utterances. Four types of features such as spectrogram, MFCC, PLP and fbank are extracted. This is done to know which features are performing better. The extraction of different features is done by Kaldi toolkit [18].

2.2. Phone recognizer

This block predicts the probability of phones from features. It consists of hybrid CNN-LSTM modules with CTC loss as shown in figure 2. First, it is trained with speech features with their respective text at the phone level. Then, it is tested with new speech utterances and predicts the phones.

2.2.1. CNN

Convolutional neural network (CNN) [19] has been used in image processing and gives a better performance compared to other algorithms. Recently, CNN has given promising results

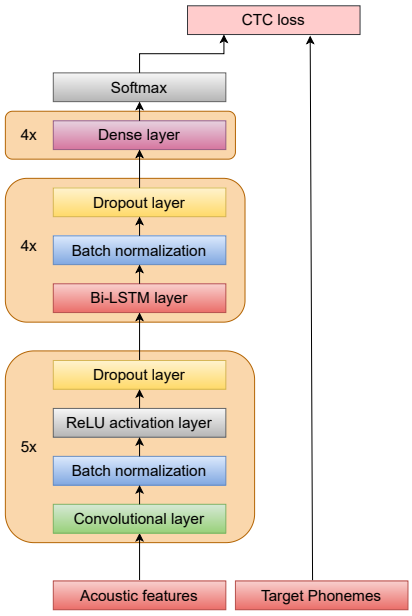

Figure 2: *Schematic diagram of phone recognizer*

in the domain of speech processing. It takes out spatial information and discovers local information hierarchically. This network primarily consists of the convolutional, pooling, and activation layers. Suppose there are 2D vectors of speech features H and 2D filter(kernel) F, then convolution operation:

$$G = H * F \tag{1}$$

$$G[i,j] = \sum_{u=-\infty}^{\infty} \sum_{v=-\infty}^{\infty} H[u,v] \cdot F[i-u, j-v] \tag{2}$$

2.2.2. LSTM

LSTM [20] is a feed-forward neural network which is good with sequence processing. It is a higher version of RNN, which overcomes the drawback of vanishing gradient. Unlike RNN, it can process longer sequences of data. It extracts temporal information from the speech data. Suppose x_τ be the input sequence, then

$$f_\tau = \phi_g\left(\Omega_f \times x_\tau + \Psi_f \times \lambda_{\tau-1} + \beta_f\right) \tag{3}$$

$$i_\tau = \phi_g\left(\Omega_i \times x_\tau + \Psi_i \times \lambda_{\tau-1} + \beta_i\right) \tag{4}$$

$$o_\tau = \phi_g\left(\Omega_o \times x_\tau + \Psi_o \times \lambda_{\tau-1} + \beta_o\right) \tag{5}$$

$$c'_\tau = \phi_g\left(\Omega_c \times x_\tau + \Psi_c \times \lambda_{\tau-1} + \beta_c\right) \tag{6}$$

$$c_\tau = f_\tau \cdot c_{\tau-1} + \tau \cdot c'_\tau \tag{7}$$

$$\lambda_\tau = o_\tau \cdot \phi_c\left(c_\tau\right) \tag{8}$$

Where, f_τ stands for the forget gate, i_τ for the input gate, o_τ means the output gate, c_τ stands for the cell state, λ_t for the hidden state, ϕ_g is sigmoid function, ϕ_c indicates tanh function, Ω_x and β_x be weight of the input and biases for the respective gate x, Ψ_x stands for weight of the hidden layer of respective gate x and, $(\cdot)$ be the element wise multiplication.

2.2.3. CTC

CTC [21] helps align the sequences where aligning is difficult. For example, aligning each phone to its respective feature vec-

tors. Here, it calculates loss between time series vectors and target phones. For this, different path combinations are searched to find the most likely phone sequences. Then sum over all possible paths that generate the same phone sequence.

$$L_{CTC} = -\log P(\mathbf{S}|\mathbf{X}) \tag{9}$$

Where, $\mathbf{S}$ = The ground truth of the word sequence, and $\mathbf{X}$ = acoustic frames.

$$P(\mathbf{S}|\mathbf{X}) = \sum_{\mathbf{c} \in A(\mathbf{S})} P(\mathbf{C}|\mathbf{X})$$

where, $\mathbf{c} \in A(\mathbf{S})$ is sum of all possible path.

$$P(\mathbf{C}|\mathbf{X}) = \prod_{t=1}^{T} y\left(c_t, t\right)$$

is the joint probability of a path.

2.3. Pronunciation error detector

This module aligns all three phone sequences, predicted phonemes, human-annotated phonemes and canonical phonemes illustrated in figure 3. Predicted phonemes are predicted by the phone recognizer, human-annotated/target phonemes are the actual phonemes that an L2 English learner (Bengali speaker) pronounces, and canonical phonemes are those how native English speaker would utter those phonemes. Aligning is done using the Needleman-Wunsch algorithm [22]. The alignment process is such that it calculates the smallest possible number of edit operations, such as insertion, substitution, and deletion, to change from one sequence to other. After aligning, the system performance is measured.

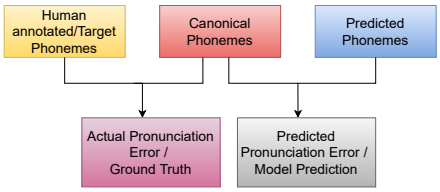

Figure 3: *Hierarchical evaluation structure*

3. Experiments and discussions

3.1. Experimental Corpus

For this experiment, English speech spoken by Bengali speakers was recorded as there is no reported available dataset for Bengali speakers' English speech. Stimuli/text were selected according to contrastive analysis between Bengali and English phonemes set [23, 24]. As per IPA notation of English consonant phonemes compared to Bengali consonant phonemes, it is observed that English consonant phonemes /p/, /t/, /k/, /b/, /d/, /g/, /ʧ/, /ʤ/, /m/, /ŋ/, /s/, /ʃ/, /h/ are identical and the consonant phonemes /n/, /l/, /r/ are similar and consonant phonemes /f/, /v/, /ɵ/, /ð/, /z/, /ʒ/ are new to L1 Bengali speakers [23]. In case of vowel phonemes, based on height of the tongue (F1), position of the tongue (F2), rounded of lips (F3), there are differences between English and Bengali vowel phonemes [4]. Bengali speakers face difficulties in pronouncing similar and new phonemes. So, words, sentences and passages of recordings were designed in such a way that the occurrence of similar and new phonemes were more [25]. Twenty four participants (12 male and 12 female L1 Bengali speaker) were selected for the experiment. The speech was recorded using a recording app which shows the word or sentence on the screen one by one.

All the collected speech data were annotated automatically using canonical pronunciation with the help of Montreal forced aligner [26] as shown in Figure 4. To make the gold standard dataset, some portions of each speaker's speech are annotated manually by a linguistics expert using Praat software [27]. These speech utterances were divided into subgroups for training, validation, and testing. The TIMIT corpus [28] and a small fraction of the VCTK (modified it to phone level) corpus [29] were utilised to build an appropriate amount of native (L1) English speech data that was used to bootstrap the training of this model. Table 1 summarises some key statistics of these voice datasets. Canonical phoneme sets were defined based on the CMU pronunciation dictionary [30].

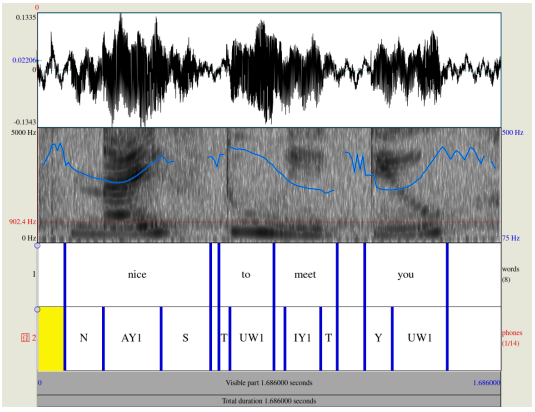

Figure 4: *A TextGrid after automatic annotation of Bengali speaker speech using canonical pronuncaition*

Table 1: *Statistics of speech corpus used in the experiment*

Speech corpus	Subsets	Speaker	Utterances
Modified VCTK	Train	50	1000
TIMIT	Train	630	6300
Bengali speech dataset	Train	12	1800
	Validation	6	897
	Test	6	900

3.2. Data pre-processing

TIMIT corpus were already annotated at the phone level. But VCTK corpus was at the word level. So, to convert from word sequences to phone sequences, g2pE [31] is applied. Then, speech-to-phone alignment was done with Montreal Forced Aligner (MFA) [26] and phone-level time stamps are obtained. After that, training, validation, and testing sets are separated, and all four features sets are extracted using the Kaldi toolkit.

3.3. Model Architecture

With a window size of 30 ms and a time step of 10 ms, features were extracted. The model was started with a convolutional layer followed by batch normalization, ReLu and a dropout layer which repeats five times. Then the output feature vectors of CNN were fed to four stacked Bidirectional LSTM lay-

ers (380 hidden units in each layer), batch normalization and dropout layers. After that, four dense layers were employed with a softmax layer as shown in Figure 2. The CTC loss, batch size 20, and learning rate 0.001 were used to train the phone recognizer. It is trained for 500 epochs. The system configuration used for training was 'Intel® Core™ i7-10700 CPU @ 2.90GHz × 16' desktop with NVIDIA T600 GPU.

3.4. Performance Evaluation

The performance of phone recognizer was measured by phone error rate (PER).

$$\text{Phone Error Rate} = 1 - \text{Accuracy} \tag{10}$$

$$\text{and, Accuracy} = \frac{N - S - D - I}{N} \tag{11}$$

where,I stands for insertions, D for deletions, S for substitutions, and N for the total number of phone sequences.

The hierarchical evaluation structure is used for pronunciation error detection as shown in Figure 3, and Table 2 illustrates the related confusion matrix. In order to assess the effectiveness of pronunciation error detection, the values of several metrics are computed, such as precision, recall, and the F-1 measure (the harmonic mean of the recall and precision), based on the statistics gathered from the four test circumstances [32]. The metrics for pronunciation error detection of the proposed system are calculated as follows:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{12}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{13}$$

$$F_1 - \text{measure} = \frac{2\,(\text{Precision} \times \text{Recall})}{\text{Precision} + \text{Recall}} \tag{14}$$

Where, TP indicates True Positive, FP stands for False Positive,
FN for False Negative.

Table 2: *Confusion matrix of pronunciation error detection*

Total Conditions		Ground Truth	
		Actual Positive	Actual Negative
Modal Prediction	Predicted Positive	True Positive (TP)	False Positive (FP)
	Predicted Negative	False Negative (FN)	True Negative (TN)

3.5. Experimental results

First, phone recognition error is analysed using the phone error rate (PER) for the proposed system trained with hand-crafted acoustic features (spectrogram, mfcc, plp and fbank). In this experiment, a test subset of speech corpus is used. Table 3 provide the summary of PER of the findings. It is clear from the experiment that fbank features outperform other acoustic feature sets for our dataset.

With regard to pronunciation error detection, performances for spectrogram, MFCC, PLP, and fbank are compared. The goodness of pronunciation (GOP) technique is also provided, which is based on automatic speech recognition (ASR) using a hybrid deep neural network and hidden Markov model (DNN-HMM) model and pronunciation scoring-based approach. GOP's DNN component especially consists of a 4-layer time delay neural network (TDNN) [33]. The proposed model can detect identical and new phonemes' pronunciation easily but get confused with similar phonemes' pronunciation. Table 4 presents comparable outcomes. As shown, the GOP-based approach cannot compete with hybrid-based CNN-LSTM-based solutions, which have at least 15% gains in the F1 measures. This suggests that phone recognition technology can improve the effectiveness of pronunciation error detection. It is also seen from this experiment that fbank as an input feature performs better than the other features for the Bengali dataset.

Table 3: *Comparison of phone error rate*

Input Features	% PER
Spectrogram	28.23
MFCC	27.92
PLP	27.52
FBANK	26.37

Table 4: *Results of the proposed technique and the GOP-based method for pronunciation error detection*

Model	Mispronunciation Detection		
	% Recall	% Precision	% F1
GOP	54.26	25.36	34.57
PR-SPECTROGRAM	67.02	41.43	51.21
PR-MFCC	65.32	43.68	52.35
PR-PLP	64.24	44.21	52.37
PR-FBANK	62.46	46.32	53.19

4. Conclusions and future works

In this paper, a system pipeline is proposed for detecting non-native English (native Bangla) speech phoneme pronunciation errors with respect to native English, especially insertion, deletion and substitution of phones (segmental errors). This pipeline includes a feature extractor, phoneme recognizer and error detector. Different types of features are extracted, such as spectrogram, perceptual linear prediction (PLP), fbank, and mel-frequency cepstral coefficients (MFCC) for the input of the phone recognizer. Phoneme recognizer include convolutional neural networks and long short-term memory modules which recognize phones of speech. Then the error detector compares output phones with the canonical and target phones and calculates pronunciation errors. All extracted features are compared with each other. The proposed system performed better when fbank is used as feature sets. Finally, a non-native English (L1 Bengali) CAPT prototype system has been developed that detect segmental (phoneme level) pronunciation errors and helps English as second language instructors as well as learners in real-world teaching and learning circumstances. In future, this system will be tested with different non-native English and a system can also be designed for detecting stress and intonation (supra-segmental) errors for L2 English speakers.

5. Acknowledgements

This work is partially supported by the grant from Science and Engineering Research Board (SERB), Government of India.

6. References

[1] H. Hao, J. Wang, and H. Abudureyimu, "Maximum f1-score discriminative training for automatic mispronunciation detection in computer-assisted language learning." in *INTERSPEECH*, 2012, pp. 815–818.

[2] D. Luo, Y. Qiao, N. Minematsu, Y. Yamauchi, and K. Hirose, "Regularized-mllr speaker adaptation for computer-assisted language learning system," in *Eleventh annual conference of the international speech communication association*, 2010.

[3] A. Raux and T. Kawahara, "Automatic intelligibility assessment and diagnosis of critical pronunciation errors for computer-assisted pronunciation learning." in *INTERSPEECH*, 2002.

[4] S. N. Saha, "Segmental and suprasegmental aspects of non-native (l2) english by native (l1) bengali speakers compared to native (l1) english: A study," Ph.D. dissertation, IIT, Kharagpur, 2016.

[5] M. P. Lewis, G. F. Simons, and C. D. Fennig, "Ethnologue: languages of ecuador," *SIL International, Dallas*, 2015.

[6] T. Mostafa, "A contrastive analysis between bangla and english phonology: Some pedagogical recommendations," 2013.

[7] B. Barman, "A contrastive analysis of english and bangla phonemics," *Dhaka University Journal of Linguistics*, vol. 2, no. 4, pp. 19–42, 2009.

[8] D. D. Miller, "Production of bangla stops by native english speakers learning bangla: an acoustic analysis," 2008.

[9] T. Pongkittiphan, N. Minematsu, T. Makino, and K. Hirose, "Improvement of intelligiblity prediction of spoken word in japanese accented english using phonetic pronunciation distance and word confusability," *Proc. O-COCOSDA*, pp. 276–281, 2014.

[10] S. Sudhakara, M. K. Ramanathi, C. Yarra, and P. K. Ghosh, "An improved goodness of pronunciation (gop) measure for pronunciation evaluation with dnn-hmm system considering hmm transition probabilities." in *INTERSPEECH*, 2019, pp. 954–958.

[11] 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.

[12] 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.

[13] S. P. Rath, D. Povey, K. Veselỳ, and J. Cernockỳ, "Improved feature processing for deep neural networks." in *Interspeech*, 2013, pp. 109–113.

[14] Q. B. Nguyen, B. Q. Dam, M. H. Le *et al.*, "Development of a vietnamese speech recognition system for viettel call center," in *2017 20th Conference of the Oriental Chapter of the International Coordinating Committee on Speech Databases and Speech I/O Systems and Assessment (O-COCOSDA)*. IEEE, 2017, pp. 1–5.

[15] Y. Shinohara, "Adversarial multi-task learning of deep neural networks for robust speech recognition." in *Interspeech*. San Francisco, CA, USA, 2016, pp. 2369–2372.

[16] S. Sultana, M. S. Rahman, and M. Z. Iqbal, "Recent advancement in speech recognition for bangla: A survey," *Int. J. Adv. Comput. Sci. Appl*, vol. 12, no. 3, pp. 546–552, 2021.

[17] W.-K. Leung, X. Liu, and H. Meng, "Cnn-rnn-ctc based end-to-end mispronunciation detection and diagnosis," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 8132–8136.

[18] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz *et al.*, "The kaldi speech recognition toolkit," in *IEEE 2011 workshop on automatic speech recognition and understanding*, no. CONF. IEEE Signal Processing Society, 2011.

[19] K. O'Shea and R. Nash, "An introduction to convolutional neural networks," *arXiv preprint arXiv:1511.08458*, 2015.

[20] A. Graves, "Long short-term memory," *Supervised sequence labelling with recurrent neural networks*, pp. 37–45, 2012.

[21] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proceedings of the 23rd international conference on Machine learning*, 2006, pp. 369–376.

[22] V. Likic, "The needleman-wunsch algorithm for sequence alignment," *Lecture given at the 7th Melbourne Bioinformatics Course, Bi021 Molecular Science and Biotechnology Institute, University of Melbourne*, pp. 1–46, 2008.

[23] S. N. Saha and S. K. D. Mandal, "Phonetic and phonological interference of english pronunciation by native bengali (l1-bengali, l2-english) speakers," in *2014 17th Oriental Chapter of the International Committee for the Co-ordination and Standardization of Speech Databases and Assessment Techniques (COCOSDA)*. IEEE, 2014, pp. 1–6.

[24] S. Kibria, M. S. Rahman, M. R. Selim, and M. Z. Iqbal, "Acoustic analysis of the speakers' variability for regional accent-affected pronunciation in bangladeshi bangla: a study on sylheti accent," *IEEE Access*, vol. 8, pp. 35 200–35 221, 2020.

[25] F. Alam, S. Habib, D. A. Sultana, and M. Khan, "Development of annotated bangla speech corpora," 2010.

[26] M. McAuliffe, M. Socolof, S. Mihuc, M. Wagner, and M. Sonderegger, "Montreal forced aligner: Trainable text-speech alignment using kaldi." in *Interspeech*, vol. 2017, 2017, pp. 498–502.

[27] P. Boersma, "Praat: doing phonetics by computer," *http://www.praat. org/*, 2007.

[28] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, and D. S. Pallett, "Darpa timit acoustic-phonetic continous speech corpus cd-rom. nist speech disc 1-1.1," *NASA STI/Recon technical report n*, vol. 93, p. 27403, 1993.

[29] J. Yamagishi, C. Veaux, K. MacDonald *et al.*, "Cstr vctk corpus: English multi-speaker corpus for cstr voice cloning toolkit (version 0.92)," 2019.

[30] J. Kominek and A. W. Black, "The cmu arctic speech databases," in *Fifth ISCA workshop on speech synthesis*, 2004.

[31] C. Hansakunbuntheung and S. Thatphithakkul, "Unsupervised graphoneme alignment evaluation for grapheme-to-phoneme conversion on complex asian-language orthographies," in *2016 Conference of The Oriental Chapter of International Committee for Coordination and Standardization of Speech Databases and Assessment Techniques (O-COCOSDA)*. IEEE, 2016, pp. 194–199.

[32] S. S. S. Yee and K. M. Soe, "Myanmar dialogue act recognition using bi-lstm rnn," in *2020 23rd Conference of the Oriental COCOSDA International Committee for the Co-ordination and Standardisation of Speech Databases and Assessment Techniques (O-COCOSDA)*. IEEE, 2020, pp. 89–93.

[33] V. Peddinti, D. Povey, and S. Khudanpur, "A time delay neural network architecture for efficient modeling of long temporal contexts," in *Sixteenth annual conference of the international speech communication association*, 2015.