

CoCA-MDD: A Coupled Cross-Attention based Framework for Streaming Mispronunciation Detection and Diagnosis

Nianzu Zheng, Liqun Deng, Wenyong Huang, Yu Ting Yeung,
Baohua Xu, Yuanyuan Guo, Yasheng Wang, Xiao Chen, Xin Jiang, Qun Liu

Huawei Noah’s Ark Lab, Shenzhen, China

zhengnianzu@huawei.com, dengliqun.deng@huawei.com

Abstract

Mispronunciation detection and diagnosis (MDD) is a popular research focus in computer-aided pronunciation training (CAPT) systems. End-to-end (e2e) approaches are becoming dominant in MDD. However an e2e MDD model usually requires entire speech utterances as input context, which leads to significant time latency especially for long paragraphs. We propose a streaming e2e MDD model called CoCA-MDD. We utilize conv-transformer structure to encode input speech in a streaming manner. A coupled cross-attention (CoCA) mechanism is proposed to integrate frame-level acoustic features with encoded reference linguistic features. CoCA also enables our model to perform mispronunciation classification with whole utterances. The proposed model allows system fusion between the streaming output and mispronunciation classification output for further performance enhancement. We evaluate CoCA-MDD on publicly available corpora. CoCA-MDD achieves F1 scores of 57.03% and 60.78% for streaming and fusion modes respectively on L2-ARCTIC. For phone-level pronunciation scoring, CoCA-MDD achieves 0.58 Pearson correlation coefficient (PCC) value on SpeechOcean762.

Index Terms: mispronunciation detection and diagnosis, coupled cross-attention, streaming end-to-end model

1. Introduction

Mispronunciation detection and diagnosis (MDD) is a key technology in computer-assisted pronunciation training (CAPT) systems [1]. In recent years, the emergence of end-to-end (e2e) neural models promises potential performance improvement, leading to new research interests for e2e MDD models [2, 3, 4, 5, 6] from both research and commercial communities.

Different MDD frameworks are illustrated in Fig. 1. Traditional goodness-of-pronunciation (GOP) [7, 8, 9] based methods (a) usually require a long pipeline of multiple steps. These steps such as forced-alignment and phonological rule matching are prone to error accumulation, leading to inferior MDD performance. Automatic Speech Recognition (ASR) based approaches (b) [10, 11] utilize acoustic encoder to convert input speech into recognized phones. Mispronunciation is detected by aligning recognized phones with phonetic transcription converted from reference text. The authors in [6] apply a convolution neural network and recurrent neural network (CNN-RNN) as acoustic encoder and Connectionist Temporal Classification (CTC) criterion as training objective. Performance of ASR based methods improves significantly compared with the traditional methods. Recent studies suggest that injecting text information during acoustic modeling (c) improves MDD performance [5, 2, 12, 3]. SED-MDD [5] extends the work in [6] by applying text encoder to encode linguistic features.

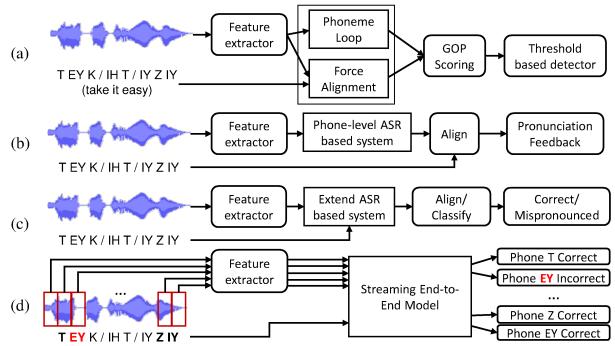


Figure 1: An overview of MDD frameworks. (a) GOP based method, (b) ASR based model, (c) Extended ASR based model, (d) Our streaming framework.

The encoded linguistic features are merged with acoustic features through attention mechanism for further context modeling. The authors in [2] apply a transformer network to implement the similar text-encoding strategy. Their results demonstrate that utilization of prior text prompt improves detection performance. Self-supervised learning (SSL) methods which learn context representation from unlabeled data are explored in [13, 14]. SSL allows the same or better detection performance with fewer labelled training data.

Current MDD systems require reasonable amount of computation. These systems usually start to return MDD feedback only after a user finishes recording of the entire utterance. Without additional engineering efforts, such as operating in blocks, there is considerable amount of delay in returning MDD feedback when the recording is in paragraph-length. For an ideal CAPT system [15], learners would expect a system to response immediately once they start speaking. The immediate feedback helps the learners to improve sustained attention for the rest of paragraph. A modification for this purpose is to apply streaming architecture to acoustic encoder in e2e models (b) or (c). However, streaming models usually lead to degraded ASR results and hence lower MDD accuracy due to lack of future context.

We propose a streaming e2e MDD framework named CoCA-MDD as illustrated in Fig. 1(d) to alleviate the problem. Motivated by the success of streaming ASR [16, 17, 18], we apply conv-transformer blocks [17] as streaming acoustic encoder to encode input speech. To inject given text prompt to the e2e model, we propose a coupled cross-attention (CoCA) mechanism. CoCA allows flexible alignment between input text prompt and streaming acoustic features. We further close the performance gap between streaming and offline MDD by applying system fusion when entire speech utterance is available.

Phone-level pronunciation scoring is another useful metric for pronunciation assessment of language learners. We show

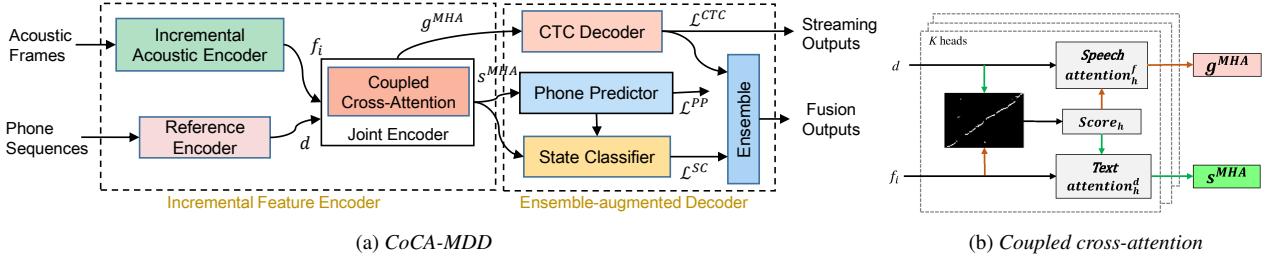


Figure 2: (a) Illustration of the proposed CoCA-MDD, which consists of an incremental feature encoder and an ensemble-augmented decoder. (b) Structure of Coupled cross-attention. The upper part is speech attention, which is input to CTC decoder. The text attention is shown in lower part and is used in phone predictor and state classifier.

that CoCA-MDD is convertible to pronunciation scoring task given the manually annotated scores.

This paper is organized as follows. We first describe our proposed CoCA-MDD in the next section. We present the experimental setup and evaluation results in Section 3. Finally we conclude this work in Section 4.

2. Proposed Method

The detail of CoCA-MDD is shown in Fig. 2. CoCA-MDD is composed of an incremental feature encoder and an ensemble-augmented decoder, which are described as follows.

2.1. Incremental Feature Encoder (IFE)

The incremental feature encoder (IFE) contains three parts, i.e., an incremental acoustic encoder (IAE), a reference encoder (RE), and a joint encoder (JE). We use the conv-transformer based encoder as described in [17] to implement our IAE module. Instead of low frame rate of 80 ms with 3 conv-transformer blocks [17], IAE operates at 40 ms frame rate with 2 blocks for better speech granularity. Benefited from interleaved transformers and convolution layers, IAE processes acoustic features frame-by-frame with a small look-ahead window (60 ms for CoCA-MDD) for modeling future context. The RE module takes reference phone strings converted from given text prompt by an open-source grapheme-to-phoneme (G2P) [19] as input. We construct the RE module with a network of two bidirectional transformer layers. As the text prompt is known beforehand, the output of RE module is pre-computed before audio recording. JE consists of a coupled cross-attention (CoCA) layer. JE takes linguistic features output by RE and acoustic features from IAE as inputs. As shown in Fig. 2(b), CoCA consists of 2 multi-head cross-attentions with shared score map. CoCA is formulated as,

$$score_h = \text{softmax}\left(\frac{f_i W_h^Q (d W_h^K)^T}{\sqrt{m}}\right) \quad (1)$$

$$attention_h^f = score_h \cdot (d W_h^V) \quad (2)$$

$$g^{MHA} = \text{concat}_{h \in H}(attention_h^f) W_f^O \quad (3)$$

$$attention_h^d = \text{softmax}(score_h^T) \cdot (f W_h^{QV}) \quad (4)$$

$$s^{MHA} = \text{concat}_{h \in H}(attention_h^d) W_d^O \quad (5)$$

where h is denoted as the index of attention heads with a total number of H heads. m is the dimension of attention head. W_h^K , W_h^V are projection matrices of RE outputs d . W_h^Q is projection matrix of incremental speech feature f_i . After finishing speech recording, W_h^{QV} is projection matrix of accumulated acoustic

features f . W_f^O and W_d^O are output weights of speech attention and text attention respectively. Concatenation operator is denoted as $\text{concat}(\cdot)$.

The speech attention $attention_h^f$ shown in upper part of Fig. 2(b) takes incremental speech feature f_i as query for related linguistic features from d as indicated by red arrows. The text attention $attention_h^d$ computes linguistic features by integrating related acoustic features. The two attentions are conditioned on the same matching score $score_h$, which is computed online with currently processed acoustic frames. The speech attention output g^{MHA} is computed frame-by-frame as the whole linguistic context d is known as prior information. The text attention output s^{MHA} is computed with the entire utterance. The on-the-fly generation of g^{MHA} allows streaming phone recognition. The full-context output s^{MHA} performs final sentence-based mispronunciation detection.

2.2. Ensemble-augmented Decoder (EAD)

A CTC decoder, a phone predictor (PP) and a state classifier (SC) are included in ensemble-augmented decoder (EAD). The CTC decoder consists of a unidirectional transformer and one-layer fully connected feed-forward network of 512 hidden units to predict phones from g^{MHA} . Pronunciation mistakes are obtained by aligning recognized phones with given phone reference using Needleman–Wunsch algorithm [20].

Both PP and SC take s^{MHA} as input. PP predicts phone labels. SC classifies whether the phones are pronounced correctly. Each of PP and SC consists of a stack of two bidirectional transformer layers followed by a multi-layer perception (MLP). PP uses a softmax layer for final phone prediction. SC takes a logistic layer for output. SC further takes the output of second transformer of PP as intermediate representation. We observe that adding the intermediate representation to s^{MHA} improves the performance of SC. Since s^{MHA} requires full speech context, both PP and SC do not support streaming output and require entire speech utterances as input.

2.3. Training Objective

The training objective of CoCA-MDD consists of three aspects: CTC-loss \mathcal{L}_{CTC} [21] for the CTC decoder, cross-entropy loss \mathcal{L}_{PP} for PP and binary cross-entropy loss \mathcal{L}_{SC} for SC. The three decoder are jointly trained in multi-task learning. The training objective \mathcal{L}_{CTC} is negative log-likelihood on speech-to-text data $\{x_i, y_i\}$ from data S as the loss function.

$$\mathcal{L}_{CTC} = -\mathbb{E}_{(x,y) \in S} \log P(y|x) \quad (6)$$

where x and y is speech sequence and ground-truth pronounced phonetic transcription. The \mathcal{L}_{PP} and \mathcal{L}_{SC} are defined as,

$$\mathcal{L}_{PP} = -\frac{1}{M} \sum_{t=1}^M \alpha_t p_t \log q_t \quad (7)$$

$$\mathcal{L}_{SC} = -\frac{1}{M} \sum_{t=1}^M \alpha_t \left[e_t \log s_t + (1 - e_t) \log(1 - s_t) \right] \quad (8)$$

where M is the length of the reference phones and α_t is defined as,

$$\alpha_t = \begin{cases} \alpha & \text{if } e_t = 1 \\ 1 & \text{otherwise} \end{cases} \quad (9)$$

We introduce α to alleviate class imbalance problem in which there are fewer mispronounced pairs in training data. The targets p_t of PP and e_t of SC are obtained by aligning ground-truth pronounced phones y of speech with the given phone sequence input to RE, where $e_t = 1$ means that the reference phone is labelled as mispronunciation. The predicted phones from PP are denoted as q_t . The probability predicted by SC is denoted as s_t . These losses are combined for the objective of model training as following,

$$\mathcal{L} = \mathcal{L}_{CTC} + \beta \mathcal{L}_{SC} + \gamma \mathcal{L}_{PP} \quad (10)$$

where hyper-parameters α , β , and γ are set to be 5, 1 and 0.5 based on simple grid search in our experiments.

2.4. MDD System Fusion

We further employ a system fusion scheme to combine the predictions for final MDD decision. The ensemble is based on the following rule. For each phone in the reference, when the corresponding SC output is classified as mispronunciation, we view the phone as mispronounced even if the result from the CTC decoder agrees with the reference. The final MDD decision is overridden with *serr*. Otherwise, the final MDD decision always follows the output of the CTC decoder. We apply this rule to improve recall rate from streaming MDD decision. An example is given in Table 1. In this example, SC probability $s_t > 0.5$ is considered as mispronounced. Reference phones are phones converted from the text prompt by G2P. Pronounced phones are phones actually pronounced in the recorded speech. In this example, the word “she” is not presented in the reference. SC does not output any probability. The word is pronounced in the speech and is recognized by the CTC decoder. Thus the corresponding phones appear in the fusion result. The SC probability on the phone “EH” of the word “bed” is greater than 0.5. The phone is considered as mispronounced. The fusion MDD decision becomes *serr*. SC sometimes make mistakes. The phone “T” in the word “went” is mis-classified as mispronounced, leading to false rejection. The last phone “D” in the word “bed” is missed by the CTC decoder. SC probability is smaller than 0.5. The final MDD decision follows the CTC decoder according to the rule. The missed error is not recovered.

3. Experiments

3.1. Experimental Setup

Dataset We conduct our experiments on 3 publicly available corpora, TIMIT [22], L2-ARCTIC [23], and SpeechOcean762 [24]. TIMIT contains recordings of 630 speakers of 8 US English dialects. All the recordings are well-labeled with 61-phone transcription. L2-ARCTIC is a non-native English corpus. We perform our experiments with version 5 of the corpus.

Table 1: The ensemble scheme of CTC decoder and SC module for CoCA-MDD inference

Sentence	(she)	went			to		bed			
Reference Phones	- -	W	EH	N	T	T	UW	B	EH	D
Pronounced Phones	SH IY	W	EH	N	T	T	UW	B	EY	D
CTC decoder (streaming)	SH IY	W	EH	N	T	T	UW	B	EH	-
SC probability s_t	- -	0.0	0.0	0.0	0.63	0.0	0.4	0.0	0.92	0.44
CoCA-MDD (fusion)	SH IY	W	EH	N	serr	T	UW	B	serr	-

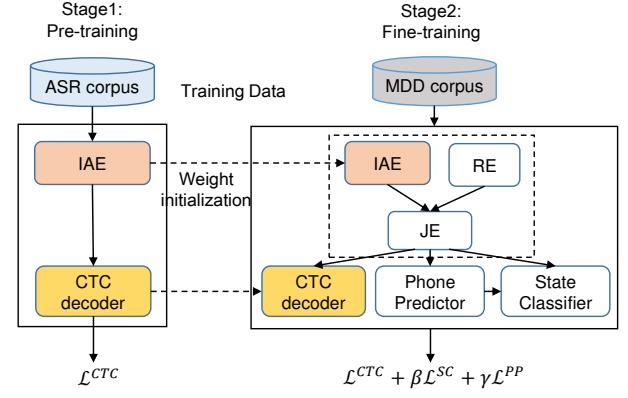


Figure 3: The two-stage training scheme in CoCA-MDD.

The corpus consists of recordings from 24 speakers (12 males and 12 females) of six different first languages (Hindi, Korean, Mandarin, Spanish, Arabic and Vietnamese). Phone-level transcription is available with a 48-phone set, plus an additional *err* symbol for unclear pronunciation. We follow the data processing pipeline of [12, 13]. We first convert the transcriptions of the two corpora into 39-phone set according to CMUDict [25]. Then we take the recordings of six speakers (NJS, TLV, TNI, TXHC, YKMK, ZHAA) from L2-ARCTIC as test set. The remaining 18 speakers of L2-ARCTIC and TIMIT are used as training set. There are 7.2 h speech data for training and 0.88 h L2-ARCTIC data for testing in the MDD task. We further evaluate phone-level pronunciation scoring with SpeechOcean762 [24]. The corpus consists of 5000 English utterances from 250 non-native speakers, with two sets of 2500 utterances for training and testing respectively. The corpus provides manually annotated phone-level accuracy by 5 experts in range of [0, 1, 2], where 0 is incorrect or missed and 2 is correct. We normalize the scores into [0, 0.5, 1].

Attention configuration All multi-head attentions in transformer layers and CoCA layer consist of 6 heads for each attention. The dimension of each head is 64. The sizes of input and feed-forward layers are 384 and 1536 respectively.

Data augmentation There are fewer mispronounced examples in training data. The CTC decoder, PP and SC are prone to overfitting with examples of correct pronunciation, hence biasing towards the reference phone input from RE. We augment the phone reference by randomly inserting, deleting or substituting phones to simulate mispronunciation.

Model training We consider the model architecture of CoCA-MDD as an extension of a CTC-based phone-level acoustic model (AM), with an addition of PP and SC. We first perform supervised training to the mentioned CTC-based AM. As shown in Fig.3, both CoCA-MDD and the AM share the same IAE and CTC decoder configurations. Rather than training the entire CoCA-MDD from scratch, we find that initializing IAE

Table 2: MDD results of proposed CoCA-MDD. True rejection is further analysed with correct diagnosis and error diagnosis.

Models	Correct pronunciation		Mispronunciation			F1(%)	PER(%)		
	True Acceptance	False Rejection	False Acceptance	True Rejection					
				Correct Diag.	Error Diag.				
w2v2.0-XLSR+TIMIT [13]	94.30%(24273)	5.70%(1467)	41.80%(1783)	70.72%(1756)	29.28%(727)	60.44%	16.20%		
CoCA-MDD (streaming)	95.34%(24517)	4.66%(1197)	48.99%(2102)	80.95%(1772)	19.05%(417)	57.03%	11.84%		

Table 3: Performance of CoCA-MDD on Recall, Precision and F1 metrics.

Models	Recall(%)	Precision(%)	F1(%)
GOP [26]	35.42	52.88	42.42
CTC-ATT [27]	46.57	70.28	56.02
CNN-RNN-CTC+VC [12]	56.04	56.12	56.08
w2v2.0-XLSR+TIMIT [13]	58.20	62.86	60.44
CoCA-MDD (streaming)	51.01	64.65	57.03
CoCA-MDD (fusion)	67.49	55.29	60.78

and CTC decoder with the weights from the trained AM improves training stability. The IAE probably learns discriminative acoustic representations during AM training, reducing phone recognition error in this stage. We refer CTC-based AM training to as pre-training and CoCA-MDD training to as fine-tuning. We pre-train the model with TIMIT and L2-ARCTIC in Stage 1. For the MDD task, we fine-tune with L2-ARCTIC. For phone-level pronunciation scoring, we fine-tune with SpeechOcean762 in Stage 2. Note that for MDD task, we label the SC target $e_t = 1$ for mispronunciation. For pronunciation scoring, e_t follows the normalized manually annotated scores.

Performance Evaluation For MDD task, we follow the evaluation metrics adopted in [12, 13, 6, 2]. For correct pronunciation, true acceptance (TA) indicates correct predictions, while false rejection (FR) denotes failure to accept correct pronunciation. In mispronunciation, false acceptance (FA) indicates the models mis-classified mispronounced phones as correct, while true rejection (TR) indicates that the models detect the mispronounced phones successfully. The F-Measure (F1) is calculated as $2 \times precision \times recall / (precision + recall)$, where $precision$ is $TR / (FR + TR)$ and $recall$ is $TR / (TR + FA)$ respectively. The performance of phone recognition is another key factor for MDD. We use phone error rate (PER) as the metric. For phone-level pronunciation scoring, the performance is measured with Pearson Correlation Coefficient (PCC) between predicted scores and reference scores.

3.2. Evaluation on Phone Recognition

Accurate phone decoding is helpful to MDD problem. The PER results are shown in the last column of Table 2. CoCA-MDD (streaming) corresponds to the output from the CTC decoder, which achieves PER of 11.84%. The result demonstrates the effectiveness of the text-acoustic modeling with CoCA.

3.3. Evaluation on MDD

We report the MDD results in Table 2. We also include the results from a recent non-streaming MDD system fine-tuned from SSL models [13] as reference. The proposed CoCA-MDD (streaming) achieves true acceptance and false rejection rates of 95.34% and 4.66% respectively. The results indicate that our model is able to well recognize correct pronunciation. For mispronunciation, our model also achieves reasonable diagnosis accuracy of 80.95% for true rejection. The trade-off of CoCA-

Table 4: Performance of CoCA-MDD on phone-level pronunciation scoring

Models	PCC
GOP-NN + SVR [24]	0.45
CoCA-MDD	0.58

MDD (streaming) is high false acceptance rate. We suspect that under streaming configuration, the CTC decoder is able to predict correct phones due to inherit phone language model in IAE and biased linguistic cues from RE. The high false acceptable rate is alleviated by our fusion scheme. As shown in Table 3, CoCA-MDD (fusion) improves the recall rate from 51.01% to 67.49%, and achieves F1-score of 60.78%. However, there is a trade-off with lower precision rate (64.65% → 55.29%), with more false rejection after system fusion. The trade-off should be adjustable with the threshold of SC probability.

We further list the performance of four other non-streaming MDD approaches in Table 3. The statistics in Table 3 suggest that CoCA-MDD (streaming) is among the same performance level as other non-streaming approaches. Note that although all the approaches test on L2-ARCTIC, direct comparison may not be appropriate as different systems are tuned to widely different operating points.

3.4. Phone-level Pronunciation Scoring

Our baseline model is a neural network based goodness of pronunciation method (GOP-NN) [28]. GOP-NN requires a deep neural network to extract GOP-based features. A support vector regressor (SVR) is trained using GOP-based features to predict phone-level scores. System details and results of SpeechOcean762 is published in [24].

CoCA-MDD directly utilizes SC output probability as phone-level pronunciation scores. The results of pronunciation scoring is shown in Table 4. CoCA-MDD achieves PCC of 0.58, which is significantly better than the baseline. Note that CoCA-MDD for pronunciation scoring and MDD can be trained end-to-end at the same fine-tuning stage. CoCA-MDD demonstrates the potential of integrating manually annotated pronunciation scores for MDD system training.

4. Conclusion

To our best knowledge, our work is the first attempt to apply an e2e streaming neural model for mispronunciation detection and diagnosis problem. Our proposed CoCA-MDD employs conv-transformer network for stream acoustic data processing, a coupled cross-attention to fully integrate the linguistic and acoustic cues, and a system fusion of CTC decoder and state classifier trained in multi-task learning to improve the MDD accuracy. The result meets the state-of-the-art counterparts. We further apply CoCA-MDD for phone-level pronunciation scoring and it achieves significant performance improvement from the baseline. As a future work, we are interested in detecting non-categorical error or distortion.

5. References

- [1] P. M. Rogerson-Revell, “Computer-assisted pronunciation training (CAPT): Current issues and future directions,” *RELC Journal*, vol. 52, no. 1, pp. 189–205, 2021.
- [2] Z. Zhang, Y. Wang, and J. Yang, “Text-conditioned transformer for automatic pronunciation error detection,” *Speech Communication*, vol. 130, pp. 55–63, 2021.
- [3] B. Lin and L. Wang, “Attention-based multi-encoder automatic pronunciation assessment,” in *2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 7743–7747.
- [4] M. Wu, K. Li, W.-K. Leung, and H. Meng, “Transformer based end-to-end mispronunciation detection and diagnosis,” in *Proc. Interspeech 2021*, pp. 3954–3958, 2021.
- [5] Y. Feng, G. Fu, Q. Chen, and K. Chen, “SED-MDD: Towards sentence dependent end-to-end mispronunciation detection and diagnosis,” in *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 3492–3496.
- [6] W.-K. Leung, X. Liu, and H. Meng, “CNN-RNN-CTC based end-to-end mispronunciation detection and diagnosis,” in *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 8132–8136.
- [7] S. M. Witt and S. J. Young, “Phone-level pronunciation scoring and assessment for interactive language learning,” *Speech communication*, vol. 30, no. 2-3, pp. 95–108, 2000.
- [8] H. Huang, H. Xu, Y. Hu, and G. Zhou, “A transfer learning approach to goodness of pronunciation based automatic mispronunciation detection,” *The Journal of the Acoustical Society of America*, vol. 142, no. 5, pp. 3165–3177, 2017.
- [9] W. Dong and Y. Xie, “Normalization of GOP for Chinese mispronunciation detection,” in *2019 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, 2019, pp. 1004–1008.
- [10] W. Hu, Y. Qian, F. Soong, and Y. Wang, “Improved mispronunciation detection with deep neural network trained acoustic models and transfer learning based logistic regression classifiers,” *Speech Communication*, vol. 67, pp. 154–166, 2015.
- [11] M. Maqsood, H. A. Habib, S. Anwar, M. A. Ghazanfar, and T. Nawaz, “A comparative study of classifier based mispronunciation detection system for confusing,” *Nucleus*, vol. 54, pp. 114–120, 2017.
- [12] K. Fu, J. Lin, D. Ke, Y. Xie, J. Zhang, and B. Lin, “A full text-dependent end to end mispronunciation detection and diagnosis with easy data augmentation techniques,” 2021, arXiv:2104.08428.
- [13] L. Peng, K. Fu, B. Lin, D. Ke, and J. Zhan, “A study on fine-tuning wav2vec2.0 model for the task of mispronunciation detection and diagnosis,” in *Proc. Interspeech 2021*, pp. 4448–4452, 2021.
- [14] L. Yang, K. Fu, J. Zhang, and T. Shinozaki, “Pronunciation erroneous tendency detection with language adversarial represent learning,” in *Proc. Interspeech 2020*, 2020, pp. 3042–3046.
- [15] W. Menzel, D. Herron, R. Morton, D. Pezzotta, P. Bonaventura, and P. Howarth, “Interactive pronunciation training,” *ReCALL*, vol. 13, pp. 67 – 78, 2001.
- [16] Y. He, T. Sainath, R. Prabhavalkar, I. McGraw, R. Alvarez, D. Zhao, D. Rybach, A. Kannan, Y. Wu, R. Pang, Q. Liang, D. Bhatia, Y. Shangguan, B. Li, G. Pundak, K. Sim, T. Bagby, S.-Y. Chang, K. Rao, and A. Gruenstein, “Streaming end-to-end speech recognition for mobile devices,” in *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 6381–6385.
- [17] W. Huang, W. Hu, Y. T. Yeung, and X. Chen, “Conv-Transformer Transducer: Low latency, low frame rate, streamable end-to-end speech recognition,” in *Proc. Interspeech 2020*, 2020, pp. 5001–5005.
- [18] B. Li, S.-Y. Chang, T. Sainath, R. Pang, Y. He, T. Strohman, and Y. Wu, “Towards fast and accurate streaming end-to-end ASR,” in *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 6069–6073.
- [19] “g2pE: A simple Python module for English grapheme to phoneme conversion,” <https://github.com/Kyubyong/g2p>, [Online; accessed 21 March 2022].
- [20] S. B. Needleman and C. D. Wunsch, “A general method applicable to the search for similarities in the amino acid sequence of two proteins,” *Journal of molecular biology*, vol. 48, no. 3, pp. 443–453, 1970.
- [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, p. 369–376.
- [22] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, and D. S. Pallett, “DARPA TIMIT acoustic-phonetic continuous speech corpus CD-ROM,” *NASA STI/Recon technical report N*, vol. 93, p. 27403, 1993.
- [23] G. Zhao, S. Sonsaat, A. Silpachai, I. Lucic, E. Chukharev-Hudilainen, J. Levis, and R. Gutierrez-Osuna, “L2-ARCTIC: A non-native English speech corpus,” in *Proc. Interspeech 2018*, 2018, pp. 2783–2787.
- [24] J. Zhang, Z. Zhang, Y. Wang, Z. Yan, Q. Song, Y. Huang, K. Li, D. Povey, and Y. Wang, “speechocean762: An open-source non-native English speech corpus for pronunciation assessment,” in *Proc. Interspeech 2021*, 2021, pp. 3710–3714.
- [25] “The CMU pronouncing dictionary,” <https://github.com/cmusphinx/cmudict>, [Online; accessed 21 March 2022].
- [26] B.-C. Yan and B. Chen, “End-to-end mispronunciation detection and diagnosis from raw waveforms,” 2021, arXiv:2103.03023.
- [27] B.-C. Yan, M.-C. Wu, H.-T. Hung, and B. Chen, “An end-to-end mispronunciation detection system for L2 English speech leveraging novel anti-phone modeling,” in *Proc. Interspeech 2020*, 2020, pp. 3032–3036.
- [28] W. Hu, Y. Qian, F. K. Soong, and Y. Wang, “Improved mispronunciation detection with deep neural network trained acoustic models and transfer learning based logistic regression classifiers,” *Speech Communication*, vol. 67, pp. 154–166, 2015.