INTERSPEECH 2021
30 August – 3 September, 2021, Brno, Czechia

Transformer Based End-to-End Mispronunciation Detection and Diagnosis

Minglin Wu[1,3], *Kun Li*[2], *Wai-Kim Leung*[1,3], *Helen Meng*[1,3]

[1]Human-Computer Communications Laboratory,
Department of System Engineering and Engineering Management,
The Chinese University of Hong Kong, HKSAR, China
[2]SpeechX Limited, China
[3]Centre for Perceptual and Interactive Intelligence (CPII) Limited, HKSAR, China

minglinwu@link.cuhk.edu.hk, kli@speechx.cn,
wkleung@se.cuhk.edu.hk, hmmeng@se.cuhk.edu.hk

Abstract

This paper introduces two Transformer-based architectures for Mispronunciation Detection and Diagnosis (MDD). The first Transformer architecture (T-1) is a standard setup with an encoder, a decoder, a projection part and the Cross Entropy (CE) loss. T-1 takes in Mel-Frequency Cepstral Coefficients (MFCC) as input. The second architecture (T-2) is based on wav2vec 2.0, a pretraining framework. T-2 is composed of a CNN feature encoder, several Transformer blocks capturing contextual speech representations, a projection part and the Connectionist Temporal Classification (CTC) loss. Unlike T-1, T-2 takes in raw audio data as input. Both models are trained in an end-to-end manner. Experiments are conducted on the CU-CHLOE corpus, where T-1 achieves a Phone Error Rate (PER) of 8.69% and F-measure of 77.23%; and T-2 achieves a PER of 5.97% and F-measure of 80.98%. Both models significantly outperform the previously proposed AGPM and CNN-RNN-CTC models, with PERs at 11.1% and 12.1% respectively, and F-measures at 72.61% and 74.65% respectively.

Index Terms: Mispronunciation Detection and Diagnosis (MDD), Transformer, encoder-decoder, wav2vec 2.0, CNN feature encoder

1. Introduction

Learning a new language offers many benefits, such as personal enjoyment, facilitating cross-cultural communication, increasing one's confidence and employability, etc. Computer-assisted Pronunciation Training (CAPT) offers learners an accessible and economical way to learn and practice new languages. Mispronunciation detection and diagnosis (MDD) is an essential part in CAPT, which aims to identify pronunciations that need practicing and provide pinpointed feedback to support learning. However, MDD needs to analyze accented speech, where there is a diversity of influences from different primary languages, speaker characteristics, vocabularies, domain contexts, etc., and there is generally a lack of accepted speech data. Hence MDD remains a challenging task.

Previous research in MDD may be grouped into three categories: (1) Approaches that aim at scoring pronunciations [1–3]; (2) Approaches that model with phonological rules [4–14]; and (3) Approaches that aim to use free-phone recognition to find mispronounced sounds [15–17]. The first type of approaches that aim at scoring pronunciations are based on different confidence measures, such as likelihoods and likelihood ratios [18–20]. Goodness of Pronunciation (GOP) [1] is used to detect mispronunciations with low scores. The limitation of this approach is the inability to diagnose the detected mispronunciations. The second type of approaches that model with phonological rules include the generation of extended recognition networks (ERN) [5], which provides extra decoding paths that cover both the correct pronunciation and possible mispronunciations. However, it is very difficult, if not impossible, to ensure exhasutive coverage of all possible mispronunciation patterns. More importantly, the model cannot detect mispronunications that are absent from the ERN. In order to address the limtations mentioned above, the third type of approaches introduce deep neural networks (DNNs) to perform the task of free-phone recognition, which can technically cover all possible mispronunciation patterns. By aligning the canonical phone sequences, human-annotated phone sequences and recognized phone sequences, mispronunciation detection and diagnosis can be derived simultaneously. Based on multi-distribution DNN, [16] presented an acoustic-graphemic-phonemic model (AGPM) that takes in acoustic features, force-aligned graphemes and canonical transcriptions as input. Also, a phone-state transition model is trained for the AGPM for decoding. Based on the CU-CHLOE corpus [6], AGPM achieves a PER of 11.1% and a F-measure of 72.6%, outperforming the results of previous ERN-based approaches. Further development presented an end-to-end architecture referred as the CNN-RNN-CTC model [15], which takes in acoustic features as input and seeks to avoid fragile forced-alignments that may affect performance. Convolutional Neural Networks (CNNs) are used to capture high-level acoustic features which are then fed to the Recurrent Neural Network (RNN) to generate the final outputs. CTC loss is chosen for end-to-end training. On CU-CHLOE test set, the PER of CNN-RNN-CTC model is 12.1% and the F-measure is 74.65%.

Recently, there has been increasing use of the Transformer [21] in automatic speech recognition (ASR). A typical Transformer-based encoder-decoder ASR model is presented in [22], which takes in the Mel-filterbank coefficients and fundamental frequency features as input. A Transformer-based pre-training architecture referred as wave2vec2.0 is presented in [23], which fully utilizes a large amount of unlabeled (but easily acquired) audio data. The wav2vec 2.0 takes in raw audio data as input. A convolutional feature encoder is designed to extract acoustic features and a contrastive task is performed to identify representations of masked speech from distracting negative samples. Performance improvements shown in [22, 23] motivates the current exploration in using Transformers for MDD.

In this paper, we introduce two Transformer-based models for MDD. The first model, T-1, basically follows the architec-

http://dx.doi.org/10.21437/Interspeech.2021-1467

ture of the vanilla Transformer. The second model, T-2, is based on wav2vec 2.0. Details about the two models will be presented in Section 2. Section 3 describes the experiments, results and analyses. Conclusions will be drawn in Section 4.

2. Models

As mentioned earlier, the first model, T-1 follows an encoder-decoder architecture, and the second model, T-2 utilizes CTC loss to fulfill end-to-end training.

2.1. T-1 Architecture

As shown in Figure 1, the model T-1 is composed of 4 parts: encoder, decoder, projection and loss. The input to the encoder is MFCC features, together with the corresponding sinusoidal positional embeddings. The two transformer blocks on the left (see Figure 1, 2X Transformer Blocks) use the combined input features to generate the encoder output. The input to the decoder is right-shifted outputs of the whole network, together with the corresponding sinusoidal positional embeddings. The encoder and decoder are linked using 2 multi-head attention blocks. The projection part projects the outputs of decoder to the desired number of logits, for example, the total number of distinct phones used. Cross Entropy loss is adopted in T-1. The architecture of the Transformer block is illustrated in Figure 2.

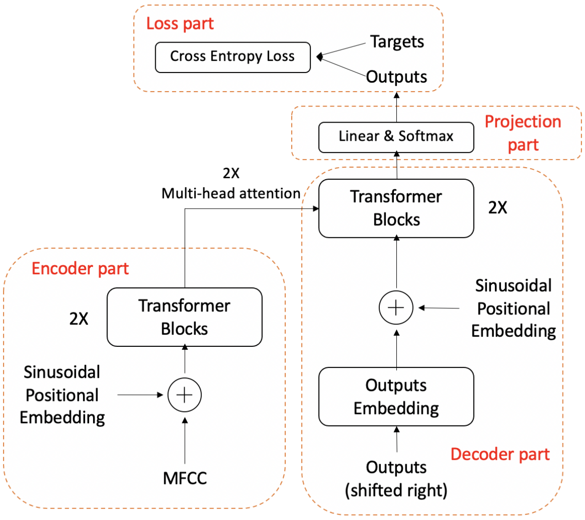

Figure 1: *The architecture of model T-1*

2.1.1. T-2 Architecture

The model T-2 is based on wav2vec 2.0. The architecture of the Transformer blocks remains unchanged (see Figure 2). The T-2 architecture is designed to fine-tune the well-trained wav2vec 2.0 to perform the MDD task, as illustrated in Figure 3. T-2 takes in raw audio data as input. The CNN feature encoder serves as a feature extractor, outputting latent speech representations Z. Then Z are fed into Transformer blocks to generate contextual representations of speech (see C in Figure 3). In this process, the positional embeddings required are generated by a 1-D convolutional layer. The contextual speech representations are projected and Softmax is applied to generate the probabilities of the occurrences of phones. CTC loss can be applied to fine-tune the whole model, which also has well-trained initial weights in the CNN feature encoder and Transformer blocks.

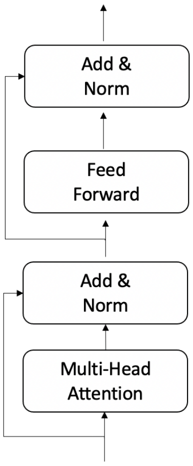

Figure 2: *The architecture of the Transformer blocks in T-1 and T-2*

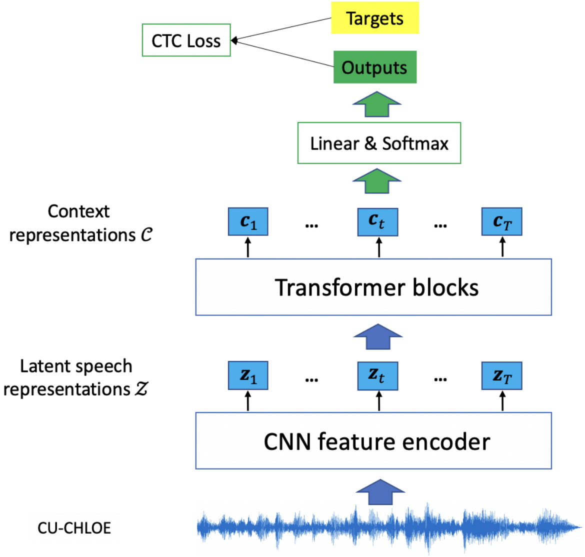
CTC Loss
Targets
Outputs
Linear & Softmax
Context representations $\mathcal{C}$
c_1 ... c_t ... c_T
Transformer blocks
Latent speech representations $\mathcal{Z}$
z_1 ... z_t ... z_T
CNN feature encoder
CU-CHLOE

Figure 3: *The architecture of model T-2*

3. Experiments

3.1. Datasets

CU-CHLOE is a corpus within the Chinese University of Hong Kong, aiming to enhance English learning by the students, many of whom are native Cantonese and Putonghua speakers. 100 Cantonese speakers (50 males and 50 females) and 110 Putonghua speakers (60 males and 50 females) contributed to the dataset, forming 34.6 hours of speech data. There are 18139 utterances recorded according to the following 4 kinds of prompts.

- 6 utterances chosen from the AESOP's fable, "The North Wind and the Sun".

- 20 utterances designed to cover common English phones.
- 10 utterances composed of confusable words.
- 50 utterances composed of minimal pairs.

CU-CHLOE is sampled at a rate of 16K Hz and is transcribed by trained linguists.

Besides CU-CHLOE, both models T-1 and T-2 also use other datasets for training. T-1 puts together all the data from TIMIT [24] and training set of CU-CHLOE to form a combined training dataset. Model T-2 only uses the training set of CU-CHLOE for fine-tuning. The pre-trained parameters of model T-2 are from the 960-hour Librispeech [25]. The pretraining phase did not utilize the transcriptions of Librispeech.

3.2. Model Training

3.2.1. Model T-1 setup

In the encoder, the inputs are MFCCs plus sinusoidal positional embeddings. A 13-dimensional MFCC is extracted every 10ms, covering 25ms of audio data (in a frame). To make use of the contextual information in the input, the MFCCs of two frames to the left and two frames to the right are concatenated with those of the central frame, forming a 65-dimensional feature vector. The sinusoidal positional embedding size is also 65. Within 2 Transformer blocks, the number of heads is 4, the model dimension is 32 and the feedforward dimension is 256.

In the decoder, the output embedding size is 32, which is the same as the corresponding sinusoidal positional embedding. Within the 2 Transformer blocks and 2 encoder-decoder multi-head attention modules, the number of heads is 2, the model dimension is 32 and the feedforward dimension is 256.

The batch size in training is 8. The optimizer is Adam [26] and the learning rate is fixed at 1e-3.

3.2.2. Model T-2 setup

The input of T-2 is raw audio data with a sampling rate of 16kHz. There are 7 1-dimensional convolutional layers in the CNN feature encoder, with kernel sizes [10, 3, 3, 3, 3, 2, 2], strides [5, 2, 2, 2, 2, 2, 2] and channel number 512. By this configuration, the CNN feature encoder outputs a latent representation corresponding to 25ms of raw speech data every 20ms. Another 1-dimensional convolutional layer capturing positional relationships has a kernel size of 128, stride of 1 and padding of 64. The model dimension of all 12 Transformer blocks is 768, the multi-head size is 8 and the feedforward dimension is 3,072.

2 GPUs are used for fine-tuning, with the maximum number of tokens per GPU being 3,200,000, representing 200s of speech. Changes in the learning rate is composed of three stages. First, we warm up the learning rate until 10% of the set training steps. Second, we hold the learning rate constant at 1e-4 for next 40% of the set training steps. Third, we linearly decay the learning rate.

Every time we update the parameters of the model, we regard it as an update. The set number of updates is 20,000. During fine-tuning, the parameters of the CNN feature encoder are always fixed and those of transformer blocks are fixed during the first 10,000 updates.

3.3. Evaluation

Both models T-1 and T-2 are evaluated on CU-CHLOE test set, based on phone recognition and performance of MDD.

Table 1: *Performances of different approaches*

Methods	**Accuracy**	**Correction Rate**
AGPM	88.92%	91.87%
CNN-RNN-CTC	87.93%	90.08%
T-1	91.31%	93.41%
T-2	**94.03%**	**95.08%**

3.3.1. Performance in phone recognition

Evaluation is based on two criteria: Accuracy and Correct Rate, which are shown in Equations (1) and (2) respectively:

$$Accuracy = \frac{N - S - D - I}{N} \tag{1}$$

$$Correct\ rate = \frac{N - S - D}{N} \tag{2}$$

Where S, D and I represent the number of substitutions, deletions and insertions found in the recognized phone sequences. N is the total number of labels annotated. Table 1 shows the performances of the different approaches.

The phone accuracy achieved by model T-1 is 91.31%. This outperforms AGPM and CNN-RNN-CTC respectively by 2.69% and 3.84% relative. Similarly, phone accuracy achieved by model T-2 is 94.03% and this outperforms AGPM and CNN-RNN-CTC respectively by 5.75% and 6.94% relative. In terms of Correct Rate, model T-1 achieved 93.41% and model T-2 achieved 95.08%, both outperforming AGPM and CNN-RNN-CTC. The improved performance values from models T-1 and T-2 in free-phone recognition of L2 speech offers good potential for MDD.

3.3.2. Performance in MDD

In terms of MDD, we follow the measures developed in [10]. For mispronunciation detection, we consider True Acceptance (TA), True Rejection (TR), False Rejection (FR), and False Acceptance (FA). TA indicates that both the human-transcribed phone and the recognized phone are the same as the canonical pronunciation. TR indicates that neither the transcribed phone nor the recognized phone is identical to the canonical pronunciation. FR indicates that transcribed phone is the same as the canonical pronunciation, but the recognized phone is different. FA indicates that the recognized phone is the same as the canonical phone, but the transcribed phone is different. The False Rejection Rate (FRR) and False Acceptance Rate (FAR) are calculated using Equations (3) and (4) respectively.

$$FRR = \frac{FR}{TA + FR} \tag{3}$$

$$FAR = \frac{FA}{FA + TR} \tag{4}$$

For mispronunciation diagnosis, we consider Correct Diagnosis (CD) and Diagnosis Error (DE). CD refers to the situation where with TR (true rejection), the human-transcribed phone is identical to the recognized phone. DE refers to the situation where with TR, the transcribed phone is not identical to recognized phone. The Diagnosis Error Rate (DER) is calculated using Equation (5).

$$DER = \frac{DE}{CD + DE} \tag{5}$$

Table 2: *Performance in MDD using different approaches*

Methods	FRR	FAR	DER	Precision	Recall	F-measure	Detection Accuracy	Diagnosis Accuracy
AGPM	**4.57%**	30.53%	13.49%	76.05%	69.47%	72.61%	90.94%	86.51%
CNN-RNN-CTC	8.66%	18.85%	16.76%	69.06%	81.15%	74.62%	89.38%	83.24%
T-1	11.46%	**1.28%**	**8.61%**	63.43%	**98.72%**	77.23%	90.25%	**91.39%**
T-2	4.75%	19.32%	9.95%	**81.27%**	80.68%	**80.98%**	**92.28%**	90.05%

In addition, Precision, Recall and F-measure are also used to evaluate the performance of mispronunciation detection of a system. Detection Accuracy and Diagnosis Accuracy are calculated as well. Corresponding equations are shown in Equation (6)-(10).

$$Precision = \frac{TR}{TR + FR} \tag{6}$$

$$Recall = \frac{TR}{TR + FA} \tag{7}$$

$$F - measure = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{8}$$

$$Detection\ Accuracy = \frac{TA + TR}{TA + FR + FA + TR} \tag{9}$$

$$DiagnosisAccuracy = \frac{CD}{CD + DE} \tag{10}$$

Performances in MDD using the different approaches are shown in Table 2. Recall that the calculation of FRR (false rejection rate) lies within the scope of correct pronunciations, which means the phone is pronounced without much accent. The low FRR of 4.75% of T-2 indicates that T-2 works well in recognizing correct pronunciations after reaping benefits from pre-training using large-scale data. Analogously, the low FAR of 1.28% of T-1 indicates that T-1 is good at modeling mispronunciations. The F-measure model of T-1 is 77.23% and that of model T-2 is 80.98%. T-1 outperforms AGPM and CNN-RNN-CTC respectively by 6.36% and 3.50% in F-measure. T-2 outperforms AGPM and CNN-RNN-CTC respectively by 11.53% and 8.52% in F-measure.

When it comes to mispronunciation diagnosis, T-1 achieved a Diagnosis Accuracy of 91.39%, outperforming AGPM and CNN-RNN-CTC respectively by 5.64% and 9.79%. T-2 achieved a Diagnosis Accuracy of 90.05%, outperforming AGPM and CNN-RNN-CTC respectively by 4.09% and 8.18%. Thus, both T-1 and T-2 can provide more pinpointed feedback for English learners.

4. Conclusions

This paper proposes two Transformer-based models for the task of mispronunciation detection and diagnosis (MDD) to support computer-aided pronunciation training. The architectures and training schemes of both models are straightforward. The first model, T-1, follows the standard encoder-decoder architecture, takes in MFCC as input and adopts the Cross Entropy loss. The second model, T-2, is based on the wav2vec 2.0, takes in raw audio data and adopts CTC loss. Both Transformer-based models significantly outperform our previous (most competitive) AGPM and CNN-RNN-CTC models, both in terms of free-phone recognition performance and mispronunciation detection and diagnosis performance. Future work will incorporate additional information, such as phonological contexts, for further performance improvements.

5. Acknowledgements

The grant of this work is partially from the HKSAR Government Research Grants Council General Research Fund (project number 14207315). Also, this work is partially supported by the Centre for Perceptual and Interactive Intelligence (CPII) Ltd under the Innovation and Technology Fund.

6. References

[1] S. M. Witt and S. Young, "Phone-level pronunciation scoring and assessment for interactive language learning," *Speech Commun.*, vol. 30, pp. 95–108, 2000.

[2] W. ping 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 Commun.*, vol. 67, pp. 154–166, 2015.

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

[4] A. M. Harrison, W. Lau, H. Meng, and L. Wang, "Improving mispronunciation detection and diagnosis of learners' speech with context-sensitive phonological rules based on language transfer," in *INTERSPEECH*, 2008.

[5] A. M. Harrison, W. Lo, X. Qian, and H. Meng, "Implementation of an extended recognition network for mispronunciation detection and diagnosis in computer-assisted pronunciation training," in *SLaTE*, 2009.

[6] H. Meng, Y. Lo, L. Wang, and W. Lau, "Deriving salient learners' mispronunciations from cross-language phonological comparisons," *2007 IEEE Workshop on Automatic Speech Recognition & Understanding (ASRU)*, pp. 437–442, 2007.

[7] W. Lo, S. Zhang, and H. Meng, "Automatic derivation of phonological rules for mispronunciation detection in a computer-assisted pronunciation training system," in *INTERSPEECH*, 2010.

[8] W. K. Lo, A. M. Harrison, H. Meng, and L. Wang, "Decision fusion for improving mispronunciation detection using language transfer knowledge and phoneme-dependent pronunciation scoring," in *2008 6th International Symposium on Chinese Spoken Language Processing*, 2008, pp. 1–4.

[9] H. Meng, "Developing speech recognition and synthesis technologies to support computer-aided pronunciation training for Chinese learners of English," in *Proceedings of the 23rd Pacific Asia Conference on Language, Information and Computation, Volume 1*, Dec. 2009.

[10] X. Qian, H. Meng, and F. Soong, "Capturing l2 segmental mispronunciations with joint-sequence models in computer-aided pronunciation training (capt)," *2010 7th International Symposium on Chinese Spoken Language Processing*, pp. 84–88, 2010.

[11] X. Qian, H. Meng, and F. Soong, “On mispronunciation lexicon generation using joint-sequence multigrams in computer-aided pronunciation training (capt),” in *INTERSPEECH*, 2011.

[12] X. Qian, H. Meng, and F. Soong, “The use of dbn-hmms for mispronunciation detection and diagnosis in l2 english to support computer-aided pronunciation training,” in *INTERSPEECH*, 2012.

[13] X. Qian, F. Soong, and H. Meng, “Discriminatively trained acoustic model for improving mispronunciation detection and diagnosis in computer aided pronunciation training (capt),” 2010.

[14] L. Wang, X. Feng, and H. Meng, “Automatic generation and pruning of phonetic mispronunciations to support computer-aided pronunciation training,” in *INTERSPEECH*, 2008.

[15] W. 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)*, 2019, pp. 8132–8136.

[16] 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.

[17] 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 *INTERSPEECH*, 2020.

[18] H. Franco, L. Neumeyer, Yoon Kim, and O. Ronen, “Automatic pronunciation scoring for language instruction,” in *1997 IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 2, 1997, pp. 1471–1474 vol.2.

[19] S. Wei, G. Hu, Y. Hu, and R. Wang, “A new method for mispronunciation detection using support vector machine based on pronunciation space models,” *Speech Commun.*, vol. 51, pp. 896–905, 2009.

[20] M. Nicolao, A. Beeston, and T. Hain, “Automatic assessment of english learner pronunciation using discriminative classifiers,” *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5351–5355, 2015.

[21] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, “Attention is all you need,” in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30. Curran Associates, Inc., 2017. [Online]. Available: https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf

[22] A. Mohamed, D. Okhonko, and L. Zettlemoyer, “Transformers with convolutional context for asr,” 2020.

[23] A. Baevski, H. Zhou, A. rahman Mohamed, and M. Auli, “wav2vec 2.0: A framework for self-supervised learning of speech representations,” *ArXiv*, vol. abs/2006.11477, 2020.

[24] V. Zue, S. Seneff, and J. R. Glass, “Speech database development at mit: Timit and beyond,” *Speech Commun.*, vol. 9, pp. 351–356, 1990.

[25] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, “Librispeech: An asr corpus based on public domain audio books,” in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 5206–5210.

[26] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *CoRR*, vol. abs/1412.6980, 2015.