ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 978-1-7281-7605-5/20/$31.00 ©2021 IEEE | DOI: 10.1109/ICASSP39728.2021.9414292

AISPEECH-SJTU ACCENT IDENTIFICATION SYSTEM FOR THE ACCENTED ENGLISH SPEECH RECOGNITION CHALLENGE

†*Houjun Huang*[1,2], †*Xu Xiang*[1], *Yexin Yang*[2], *Rao Ma*[2], ✉*Yanmin Qian*[2]

[1]AISpeech Ltd, Suzhou China
[2]MoE Key Lab of Artificial Intelligence, AI Institute SpeechLab, Department of Computer Science
and Engineering Shanghai Jiao Tong University, Shanghai,China
{houjun.huang, xu.xiang}@aispeech.com, {yangyexin, rm1031, yanminqian}@sjtu.edu.cn

ABSTRACT

This paper describes the AISpeech-SJTU system for the accent identification track of the Interspeech-2020 Accented English Speech Recognition Challenge. In this challenge track, only 160-hour accented English data collected from 8 countries and the auxiliary Librispeech dataset are provided for training. To build an accurate and robust accent identification system, we explore the whole system pipeline in detail. First, we introduce the ASR based phone posteriorgram (PPG) feature to accent identification and verify its efficacy. Then, a novel TTS based approach is carefully designed to augment the very limited accent training data for the first time. Finally, we propose the test time augmentation and embedding fusion schemes to further improve the system performance. Our final system is ranked first in the challenge and outperforms all the other participants by a large margin. The submitted system achieves 83.63% average accuracy on the challenge evaluation data, ahead of the others by more than 10% in absolute terms.

***Index Terms*—** Accent identification, phone posteriorgram, PPG, TTS based data augmentation, test time augmentation

1. INTRODUCTION

An accent is a manner of pronunciation peculiar to a particular individual, location, or nation. It may be influenced by the speaker's locality, education attainment or first language. For the pervasiveness of accents, accent identification is widely utilized in robust speech recognition, speaker recognition, language identification and forensic applications.

Earlier studies in accent identification mainly focus on combining linguistic theory with statistical analysis. Piat *et al.*, used a statistical approach based on prosodic parameters and found that the duration and energy are promising parameters for correct identification [1]. Berkling *et al.*, leveraged the structure of English syllable to improve the accent identification [2]. Chen *et al.*, proposed a Gaussian mixture model (GMM) based method for Mandarin accent identification [3]. Recently, deep neural network based approaches have emerged in this field. Weninger *et al.*, made use of bidirectional Long Short-Term Memory (bLSTM) networks to model longer-term acoustic context [4]. Jiao *et al.*, proposed a system utilizing long and short term features in parallel using DNNs and RNNs [5].

In this work, we describe our accent identification system for the Interspeech-2020 Accented English Speech Recognition Challenge (AESRC) [6] in detail. Since accent training data is rather limited, it is critical to effectively make use of the auxiliary Librispeech dataset. In contrast to the previous studies, we have three distinct contributions. First, we introduce the ASR based PPGs as the discriminative features for accent identification. Second, we propose a novel TTS based approach to synthesize the accent data, which provides richer speaker, channel and text variability for training the accent classifier. Third, we develop the test time augmentation and the hierarchical multi-embedding joint model for improving system performance.

This paper is arranged as follows. Section 2 gives an in-depth description of the framework of our system. Section 3 presents our experiments with different settings. Section 4 concludes our paper.

2. SYSTEM DESCRIPTION

In this section, we depict our system for the accent identification challenge, which is shown in Figure 1.

First, we leverage both the accent training data and the Librispeech data to train an ASR model with conventional data augmentation. The PPG features are extracted from the ASR model and employed to train the accent identification(AID) model. Then, we propose a novel TTS based data augmentation to augment the accent data for training an accurate and robust accent classifier. Finally, we introduce the test time augmentation scheme to improve system performance on the test data. Moreover, we further boost the system performance with the hierarchical multi-embedding joint model.

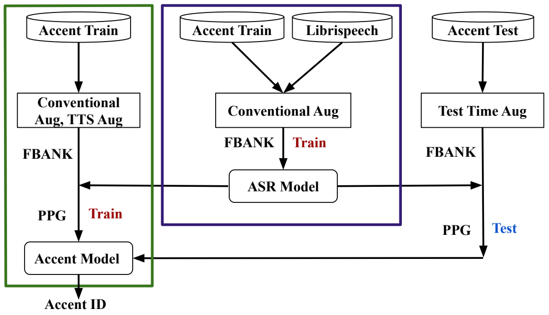

Fig. 1. Our system diagram for the challenge. Firstly, an ASR model is trained on the pooled data. Then, the PPG features derived from the ASR model are prepared to train an AID model. Finally, The AID model makes predictions on the test data using PPG features.

† :These authors contributed equally to this work. Yanmin Qian is the corresponding author. This work was supported by the National Key R&D Program of China (No. 2018YFB1004602) and the China NSFC project (No. 62071288)

2.1. ASR based PPG feature extraction

While MFCC and FBANK features are popular in speech related tasks, we wouldn't build our AID system on them directly as the following reasons. First, AID system built on MFCC or FBANK features directly could not take advantage of data-sets without accent labels Second, as these features are low level and not task-oriented, they may contain some nuisance attributes like speaker or text specific, which makes the learning of accent related representation harder, especially when the amount of the training data is limited.

To address these issues, we adopt the phone posteriorgram (PPG) feature that has been successfully applied for cross-lingual voice conversion [7] and cross-accent voice conversion [8] to train the accent classifier. PPG is a time-versus-class vector that represents the posterior probabilities of phonetic classes for a specific time frame. In this work, we first train a speaker independent (SI) automatic speech recognition (ASR) model with both the accent training data and the Librispeech data and then extract the PPG features with the SI-ASR model. With this process, the resulting PPG features have the speaker independent property which helps improve the robustness of the system.

To train a robust ASR model for PPG feature extraction, we employ two kinds of conventional data augmentation that have been widely used in automatic speech recognition tasks. The first one augments original data with additive noise and reverberation. For additive noise, the music, noise and speech part in the MUSAN dataset [9] are used. For reverberation, the room impulse responses (RIRs) and the simulated RIRs described in Kaldi's [10] VoxCeleb recipe are used. The second one is based on warping the signal in the time domain. We randomly change the tempo of the audio signal while ensuring that the pitch and spectral envelope of the signal unchanged. The *tempo* effect of the SoX tool was used to achieve such speech rate perturbation.

2.2. TTS based data augmentation

Since only 160 hours of accent data are provided, it is too limited to train an accurate and robust accent identification model. Therefore, we develop a novel TTS based data augmentation approach that is specially designed for synthesizing accent data.

Using the generated high-quality artificial speech as the augmented data has been successfully facilitated in ASR systems [11]. Recent advances in speech synthesis (text-to-speech, TTS) allow unsupervised modeling of prosody and speaker variations, which give the power to synthesize the same texts with diverse speaking styles. Moreover, the development of TTS models has made synthesized speech indistinguishable from human speech. In this work, we choose FastSpeech [12] as our synthesizer and LPCNet [13] as the vocoder.

FastSpeech is a transformer-based model which generates the entire sequence in a non-autoregressive manner. The implementation of FastSpeech is based on the ESPnet toolkit [14]. We use the default settings as in [12]. Instead of applying the knowledge distillation process, we train the FastSpeech model from scratch only using the extracted features. The synthesizer converts input text to spectrogram. The size of the input vocabulary is 41, including English phonemes, a pause break token, and a sentence boundary token. Additionally, we augment the decoder with a five-layer post-net [15], which slightly enhance model performance. LPCNet is a variant of WaveRNN that combines linear prediction with recurrent neural networks, greatly promoting the synthesis efficiency [13]. Our implementation is based on [13].

First, we train a TDNN x-vector speaker model [16] with the pooled data consists of the accent training data and the auxiliary Librispeech data. The x-vectors and the phoneme representation extracted from the pooled data are then used to train the FastSpeech synthesizer. In order to better capture the characteristics of each accent, we create 8 accent specified synthesizer by finetuning the FastSpeech Model on each 20-hour accent data respectively. Finally, on the clean subset of the overall training data, we train two LPCNet vocoders for the male and female speakers respectively.

For each speaker from the accent training data, 30 utterances are grouped to calculate the speaker specific statistics. We use these statistics with randomly selected reference texts to synthesize data with the previously trained accent specific synthesizers. With this process, the generated speech can preserve the speaker's speaking style while adopting new accents.

2.3. Hierarchical multi-embedding joint model

In our final submission, we use a hierarchical multi-embedding joint model to predict the accent label, which is shown in Figure 2. The structure of the TDNN sub-model is the same as the one in [16], but with 2× size, as we find it gives higher accuracy on the development data. The RES2SETDNN sub-model is developed in a way similar to [17], by introducing the Res2Net [18] type convolution and the squeeze-and-excitation [19] module to the original TDNN structure. However, we do not include the residual connection in our model, for there is no improvement in performance.

We train the joint model in a progressive way as follows. First, we pretrain each accent identification model independently using an additive angular margin softmax loss [20, 21]. Then, for each model the embedding extraction part is fixed. Finally, we train a linear regression classifier based on the concatenated embeddings extracted from each sub-model.

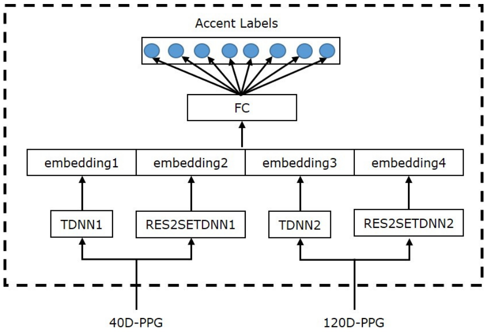

Fig. 2. The hierarchical multi-embedding joint model includes the TDNN and RES2SETDNN sub-models and accepts two set of features. *40D-PPG* denotes the 40-dimensional PPG features, *120D-PPG* denotes the 40-dimensional PPG features and its first and second order difference and *FC* denotes the fully connected linear layer.

2.4. Test time augmentation

Test time augmentation is a common trick in image classification tasks to improve the test accuracy [22, 23]. Instead of predicting the label of test image itself, the model takes multiple augmented versions of the test image as input, and the predicting results are then aggregated to give the final result. In our work, similar test time augmentation for speech data is adopted. We use the *tempo* effect

Table 1. This table shows the identification accuracy (%) of two model architectures on the development set: TDNN and RES2SETDNN. For each architecture, two kinds of feature FBANK and PPG and the corresponding data augmentation strategies are listed. Here "CONV Aug" means conventional data augmentation, "TTS Aug" means TTS based data augmentation, "TTA" means test-time data augmentation, and "Delta" means we use the original PPG feature and its first and second difference for training.

ID	Model	Configuration	Accuracy (%)								
			US	UK	CHN	IND	JPN	KR	PT	RU	Avg.
1	TDNN	FBANK	36.54	90.13	54.32	92.61	51.34	35.94	73.08	49.44	60.24
2		+CONV Aug	53.86	87.92	71.39	88.58	58.00	58.30	79.15	52.97	68.74
3		+TTS Aug	54.70	90.51	71.67	96.34	65.32	64.27	85.71	61.76	73.66
4		PPG	78.89	92.28	81.20	98.71	76.01	83.61	84.47	76.79	83.74
5		+CONV Aug	77.07	91.84	86.78	99.39	77.96	87.11	86.88	78.65	85.57
6		+TTS Aug	80.43	93.04	94.48	99.54	81.72	90.33	90.90	87.19	89.74
7		+TTA	80.79	92.98	94.87	99.70	83.94	88.55	91.96	89.29	90.32
8		+Delta	81.63	92.86	94.20	99.70	84.07	87.11	93.19	87.68	90.10
9	RES2SETDNN	FBANK	50.00	82.86	45.23	85.83	46.51	32.58	70.24	48.39	57.32
10		+CONV Aug	65.78	88.61	66.59	84.23	60.28	46.78	83.35	54.64	68.73
11		+TTS Aug	68.44	92.35	75.07	94.82	68.48	61.32	90.66	64.42	76.85
12		PPG	82.12	91.71	81.99	98.93	77.96	82.44	86.94	76.11	84.51
13		+CONV Aug	81.56	92.17	89.68	99.31	79.44	88.54	88.74	83.04	87.71
14		+TTS Aug	79.19	93.23	94.25	99.31	81.45	90.47	93.25	87.25	89.86
15		+TTA	80.86	93.17	94.70	99.77	84.95	86.76	93.32	87.19	90.15
16		+Delta	81.49	93.30	94.59	99.62	86.16	88.41	92.14	88.49	90.56
17	challenge baseline		60.2	93.9	67.0	97.0	73.2	55.6	85.5	75.7	76.1
18	our final system		82.68	93.36	95.37	99.77	84.88	88.96	93.69	89.85	91.13

of the SoX tool to do data augmentation by changing the speech rates. The augmented versions of the test file are then appended to the original test file, and we test our model on the resulting file.

3. EXPERIMENTS

A detailed comparison of our systems are presented in this section. Kaldi is used for FBANK feature extraction and PyTorch [24] is utilized to train the neural network models and PPG feature extraction. All experiment results on the development set are shown in Table 1, where 8 accented English are denoted by their corresponding country codes respectively: **US** (the USA), **UK** (the United Kingdom), **CHN** (China), **IND** (India), **JPN** (Japan), **KR** (Korea), **PT** (Portugal) and **RU** (Russia).

3.1. Baseline systems

Our baseline systems are trained on the original 160-hour accent training data, with 40-dimensional FBANK feature. In Table 1, the baseline system for TDNN and RES2SETDNN are denoted by ID 1 and 9 respectively.

3.2. Conventional data augmentation

We then apply the conventional data augmentation to the accent training data. First, the speech rate of each utterance in the training set is randomly changed to 0.8×, 0.9×, 1.1× or 1.2×, which increase the amount of the data to 320 hours. Then, by augmentation with additive noise and reverberation, we extend the training data to 1600 hours. The TDNN and RES2SETDNN models trained on the extended training data are shown in Table 1 with ID 2 and ID 10 respectively. Compared with the baseline systems, on average accuracy the conventional data augmentation can give an absolute improvement of 8.50% and 11.41% respectively. Besides, the improvement is consistent across all 8 accents.

3.3. TTS based data augmentation

In addition to the conventional data augmentation, we further apply the TTS based data augmentation approach described in Section 2.2 on the 1600-hour augmented data to generate 4800-hour synthesized data.

To check the correlation between the synthesized speech and its corresponding accent, we test the synthesized data using system ID 2 and 10 in Table 1 which are trained on the conventional augmented data. From Table 2, we can find that the accent identification accuracy on the synthesized data is comparable to that of the development data.

Table 2. Accuracy (%) on synthesized data

Accent	System ID	
	2	10
US	59.64	60.50
UK	90.73	86.04
CHN	70.93	64.08
IND	67.13	70.70
JPN	63.42	53.73
KR	67.48	66.91
PT	86.44	81.77
RU	46.27	45.04
Avg.	69.08	66.16

We train system ID 3 and 11 with the combined 6400-hour data. As shown in Table 1, the average accuracy is largely improved again, also, the improvement across the 8 accents is consistent. This verifies the effectiveness of our proposed TTS based data augmentation.

Table 3. Final submission results on the accent identification evaluation set of the top 4 teams in the rank list

Ranking	Team	Accuracy (%)								
		US	UK	CHN	IND	JPN	KR	PT	RU	Avg.
1	S2 (ours)	65.64	94.77	87.6	97.11	81.49	83.43	79.66	85.25	83.63
2	E2	52.55	93.11	60.65	90.41	68.43	79.12	76.47	65.83	72.39
3	Z2	37.64	90.87	72.79	92.32	61.65	78.90	69.93	63.19	69.63
4	F	33.55	89.60	62.75	89.37	69.73	83.09	77.46	62.26	69.59
*	challenge baseline	40.2	89.4	57.4	88.4	62.4	53.8	63.6	63.8	64.9

3.4. PPG features versus FBANK features

In this section, we compare the systems trained with PPG features or FBANK features. The SI-ASR model for PPG feature extraction is prepared with the same setting in [25]. Similar to the training data augmentation schemes for systems trained with FBANK features, we train the TDNN and RES2SETDNN models on the original data and two sets of augmented data respectively. The system trained on PPG features are denoted by ID 4, 5, 6 and 12, 13, 14 in Figure 1.

According to the reported numbers of the average accuracy in Table 1, the system ID 4 (or 12) beats the system ID 1 (or 9) by a large margin, though they are trained on the original data only. This suggests that, with the speaker independent property, PPG features make the discrimination of accent much easier. In addition, comparing system ID 5, 6 (or 13, 14) to ID 4 (or 12), we can find the system performance can be improved when applying data augmentation.

In this work, the SI-ASR system used as extractor for PPG features is trained over all available data (accented+librispeech). When only the accented data are used for training the SI-ASR system, PPG features achieves about 3% performance improvement compared to the FBANK features on average accuracy, which is small then using all data.

3.5. Test-time data augmentation

Table 1 shows, with the test-time data argumentation described in Section 2.4, the performance has a slight lift from system ID 6 to 7 (or ID 9 to 10) .

3.6. Delta PPG features

We train systems with 120-dimensional delta PPG features (the original feature and its first and second difference) on the combined 6400 hours training data. We find the performance has a small drop on the TDNN model (system ID 7 and 8), but rises a little on the RES2SETDNN model (system ID 15 and 16).

3.7. Final system

In our final system, following the design in Section 2.3, we first initialize the embedding extraction part with four systems (ID 7, 8, 15 and 16) in Table 1 and fix the parameters. Then we train the fully connected layer for accent classification. The final prediction is given by the 8-way classification hierarchical multi-embedding joint model on the test time augmented wave file. As shown in Table 1, our final system (ID 18) significantly outperforms the challenge baseline system (ID 17), reaching a remarkable 15.03% accuracy improvement on the development set.

On the challenge evaluation data, Table 3 shows the rankings of the leading submissions for this challenge. Our system ranks first in the challenge with an average accuracy of 83.63% on the evaluation set, ahead of the second team by 11.23%. Moreover, when comparing the system performance shown in Table 1, our final system achieves much smaller performance gap between the development data and the evaluation data than the challenge baseline.

To better understand our system performance on each accent, we visualize the accent embeddings of the test data using t-SNE [26]. As shown in Figure 3, the cluster of the **IND** utterances is compact and far from the other clusters, while the cluster of the **US** utterances has many overlaps with the other clusters. This partially explains the differences in identification accuracy.

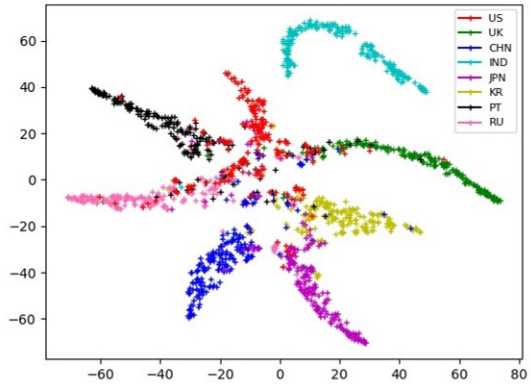

Fig. 3. Accent embeddings of 8 accents in test set. 200 accent embeddings for every accent are chosen in the test set.

4. CONCLUSIONS

In this paper, we describe our submitted system for the Interspeech-2020 Accented English Speech Recognition Challenge (AESRC). Several novel approaches are developed to improve the robustness of our accent identification system. To the best of our knowledge, it is the first time phone posteriorgram feature has been introduced to accent classification, which brings an improvement of more than 15% compared to the regular FBANK feature. To train a robust system from such limited data, we adopt TTS based data augmentation to synthesize additional accented training data, improving the system performance by 2.15%∼4.17%. Test-time data augmentation and hierarchical multi-embedding joint model training are employed for further boosting the system performance. Based on these approaches, our final system achieves an average accent identification accuracy of 83.63% on the AESRC evaluation set, ranking first among all the participants. We find that the evaluation set is more challenging than the development set, which leads to more than 7% performance degradation. In the future, we will focus on analyzing and narrowing this performance gap.

5. REFERENCES

[1] Marina Piat, Dominique Fohr, and Irina Illina, "Foreign accent identification based on prosodic parameters," in *Ninth Annual Conference of the International Speech Communication Association*, 2008.

[2] Kay Berkling, Marc A Zissman, Julie Vonwiller, and Christopher Cleirigh, "Improving accent identification through knowledge of english syllable structure," in *Fifth International Conference on Spoken Language Processing*, 1998.

[3] Too Chen, Chao Huang, Eric Chang, and Jingehan Wang, "Automatic accent identification using gaussian mixture models," in *IEEE Workshop on Automatic Speech Recognition and Understanding, 2001. ASRU'01.* IEEE, 2001, pp. 343–346.

[4] Felix Weninger, Yang Sun, Junho Park, Daniel Willett, and Puming Zhan, "Deep learning based mandarin accent identification for accent robust asr.," in *INTERSPEECH*, 2019, pp. 510–514.

[5] Yishan Jiao, Ming Tu, Visar Berisha, and Julie M Liss, "Accent identification by combining deep neural networks and recurrent neural networks trained on long and short term features.," in *Interspeech*, 2016, pp. 2388–2392.

[6] Xian Shi, Fan Yu, Yizhou Lu, Qiangze Feng, Daliang Wang, Yanmin Qian, and Lei Xie, "The accented english speech recognition challenge 2020: open datasets, tracks, baselines, results and methods," 2020.

[7] Lifa Sun, Hao Wang, Shiyin Kang, Kun Li, and Helen M Meng, "Personalized, cross-lingual tts using phonetic posteriorgrams," in *INTERSPEECH*, 2016, pp. 322–326.

[8] Guanlong Zhao, Shaojin Ding, and Ricardo Gutierrez-Osuna, "Foreign accent conversion by synthesizing speech from phonetic posteriorgrams," in *INTERSPEECH*, 2019, pp. 2843–2847.

[9] David Snyder, Guoguo Chen, and Daniel Povey, "Musan: A music, speech, and noise corpus," *arXiv preprint arXiv:1510.08484*, 2015.

[10] Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlicek, Yanmin Qian, Petr Schwarz, et al., "The kaldi speech recognition toolkit," in *IEEE 2011 workshop on automatic speech recognition and understanding*. IEEE Signal Processing Society, 2011, number CONF.

[11] Guangzhi Sun, Yu Zhang, Ron J Weiss, Yuan Cao, Heiga Zen, Andrew Rosenberg, Bhuvana Ramabhadran, and Yonghui Wu, "Generating diverse and natural text-to-speech samples using a quantized fine-grained vae and autoregressive prosody prior," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6699–6703.

[12] Yi Ren, Yangjun Ruan, Xu Tan, Tao Qin, Sheng Zhao, Zhou Zhao, and Tie-Yan Liu, "Fastspeech: Fast, robust and controllable text to speech," in *Advances in Neural Information Processing Systems*, 2019, pp. 3171–3180.

[13] Jean-Marc Valin and Jan Skoglund, "Lpcnet: Improving neural speech synthesis through linear prediction," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 5891–5895.

[14] Shinji Watanabe, Takaaki Hori, Shigeki Karita, Tomoki Hayashi, Jiro Nishitoba, Yuya Unno, Nelson Enrique Yalta Soplin, Jahn Heymann, Matthew Wiesner, Nanxin Chen, et al., "Espnet: End-to-end speech processing toolkit," *arXiv preprint arXiv:1804.00015*, 2018.

[15] Jonathan Shen, Ruoming Pang, Ron J Weiss, Mike Schuster, Navdeep Jaitly, Zongheng Yang, Zhifeng Chen, Yu Zhang, Yuxuan Wang, Rj Skerrv-Ryan, et al., "Natural tts synthesis by conditioning wavenet on mel spectrogram predictions," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 4779–4783.

[16] David Snyder, Daniel Garcia-Romero, Gregory Sell, Daniel Povey, and Sanjeev Khudanpur, "X-vectors: Robust dnn embeddings for speaker recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5329–5333.

[17] Brecht Desplanques, Jenthe Thienpondt, and Kris Demuynck, "Ecapa-tdnn: Emphasized channel attention, propagation and aggregation in tdnn based speaker verification," *arXiv preprint arXiv:2005.07143*, 2020.

[18] Shanghua Gao, Ming-Ming Cheng, Kai Zhao, Xin-Yu Zhang, Ming-Hsuan Yang, and Philip HS Torr, "Res2net: A new multi-scale backbone architecture," *IEEE transactions on pattern analysis and machine intelligence*, 2019.

[19] Jie Hu, Li Shen, and Gang Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7132–7141.

[20] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4690–4699.

[21] Xu Xiang, Shuai Wang, Houjun Huang, Yanmin Qian, and Kai Yu, "Margin matters: Towards more discriminative deep neural network embeddings for speaker recognition," in *2019 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE, 2019, pp. 1652–1656.

[22] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[23] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[24] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al., "Pytorch: An imperative style, high-performance deep learning library," in *Advances in neural information processing systems*, 2019, pp. 8026–8037.

[25] Tan Tian, Lu Yizhou, Ma Rao, Zhu Sen, Guo Jiaqi, and Yanmin Qian, "Aispeech-sjtu asr system for the accented english speech recognition challenge," IEEE, 2020.

[26] G. E. Hinton, "Visualizing high-dimensional data using t-sne," *Journal of Machine Learning Research*, vol. 9, no. 2, pp. 2579–2605, 2008.