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

# Dynamic Multi-scale Convolution for Dialect Identification

*Tianlong Kong*[1], *Shouyi Yin*[1†], *Dawei Zhang*[2], *Wang Geng*[2], *Xin Wang*[2], *Dandan Song*[1], *Jinwen Huang*[2], *Huiyu Shi*[1], *Xiaorui Wang*[2]

[1]Institute of Microelectronics, Tsinghua University
[2]Kwai, Beijing, P.R. China
yinsy@tsinghua.edu.cn

## Abstract

Time Delay Neural Networks (TDNN)-based methods are widely used in dialect identification. However, in previous work with TDNN application, subtle variant is being neglected in different feature scales. To address this issue, we propose a new architecture, named dynamic multi-scale convolution, which consists of dynamic kernel convolution, local multi-scale learning, and global multi-scale pooling. Dynamic kernel convolution captures features between short-term and long-term context adaptively. Local multi-scale learning, which represents multi-scale features at a granular level, is able to increase the range of receptive fields for convolution operation. Besides, global multi-scale pooling is applied to aggregate features from different bottleneck layers in order to collect information from multiple aspects. The proposed architecture significantly outperforms state-of-the-art system on the AP20-OLR-dialect-task of oriental language recognition (OLR) challenge 2020, with the best average cost performance ($C_{avg}$) of 0.067 and the best equal error rate (EER) of 6.52%. Compared with the known best results, our method achieves 9% of $C_{avg}$ and 45% of EER relative improvement, respectively. Furthermore, the parameters of proposed model are 91% fewer than the best known model.

**Index Terms**: dialect identification, dynamic kernel convolution, local multi-scale learning, global multi-scale pooling.

## 1. Introduction

Dialect identification refers to the identification of dialect categories from utterances, which is usually presented at the front-end of speech processing systems, such as automatic speech recognition (ASR), multilingual translation systems, targeted advertising, and biometric authentication [1, 2, 3, 4, 5]. In recent years, due to the raising amount of dialect relating campaigns and participants [6, 7, 8, 9, 10], the increasing interest in spoken or written dialect identification among them is witnessable.

Over the past four years, x-vector [11] is still the mainstream method for dialect identification. Recently, with the flourishing of DNN model, significant architecture improvements in the frame-level layers are springing up, including TDNN [12, 13], extended TDNN (E-TDNN) [14], factorized TDNN (F-TDNN) [15]. ResNet [16] shows exciting performance in speaker verification and language identification. Inspired by the dense connection in DenseNet [17], D-TDNN [18] network is adopted to improve accuracy in the field of speaker verification. Compared with the above TDNN-based networks, D-TDNN reduce the number of parameters and further improve the accuracy in speaker verification significantly.

However, the difference in dialects is very subtle that distinguishing features of dialects could be local or global in one utterance. Therefore, it is necessary to construct a multi-scale neural network to extract the distinguishing features for dialects.

In this paper, we propose a novel model for dialect identification, named dynamic multi-scale convolution, for the purpose of further improving the performance and reducing the number of parameters by leveraging dynamic kernel convolution, local multi-scale learning and global multi-scale pooling method. By introducing the methods of dynamic kernel convolution, the features are captured between short-term and long-term context adaptively. Specifically, local multi-scale learning represents multi-scale features at a granular level, which is applied to increase the range of receptive fields for convolution operation. In addition, the reduction of convolutional filter numbers contributes to a great decrement in model parameters. Besides, global multi-scale pooling is applied to aggregate features from different bottleneck layers for collecting information from multiple aspects.

The contributions of our work are summarized as follows:

- Introducing dynamic kernel convolution into dialect recognition for the first time.
- Local multi-scale learning, which represents multi-scale features at a granular level.
- Global multi-scale pooling, which aggregate features from multiple aspects.

The rest of this paper is organized as follows. In Section 2 the related works are reviewed, and we describe the dynamic multi-scale convolution architecture in Section 3. The experimental settings are presented in Section 4, and the experimental results are shown and analysed in Section 5. Finally, Section 6 concludes this paper.

## 2. Related Works

In this section, we briefly review the related works including the densely connected time delay neural network-statistics-and-selection (D-TDNN-SS) [18] and Res2Net [19] model. D-TDNN-SS can significantly improve performance of speaker verification in a small number of parameters. Res2Net is proposed and applied in residual block, improving the performance of ResNet network, significantly.

### 2.1. DTDNN-SS

[18] proposed a model named D-TDNN in speaker verification, by adopting bottleneck layers and dense connectivity. D-TDNN comprises fewer parameters than existing TDNN-based models. Furthermore, statistics-and-selection (SS) is proposed in [18]

† Corresponding author

http://dx.doi.org/10.21437/Interspeech.2021-56

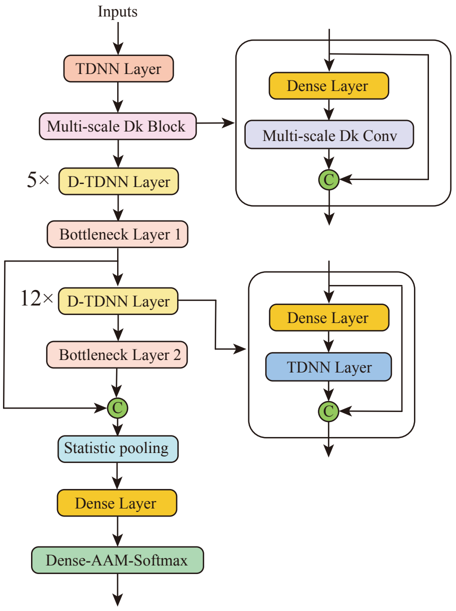

Figure 1: *Dynamic multi-scale convolution architecture. In this figure, "Multi-scale Dk Block" denotes global and local multi-scale dynamic kernel convolution block, "Multi-scale Dk Conv" denotes local multi-scale dynamic kernel convolution operation. Green "C" icon denotes the operation "concat".*

to fuse short-term and long-term context from multiple TDNN branches [20].

### 2.2. Res2Net

Different from the existing methods that represent the multi-scale features in a layer-wise manner, [19] propose a novel block for CNNs, named Res2Net, by constructing hierarchical residual-like connections within each single residual block. The Res2Net represents multi-scale features at a granular level and increases the range of receptive fields for each network layer.

## 3. Dynamic Multi-scale Convolution for Dialect Identification

In this section, we describe the proposed dynamic multi-scale convolution method for dialect identification. The complete architecture is depicted in Figure 1. It is noted that batch normalization (BN) and ReLU activation are employed but omitted in the figure.

The D-TDNN [18] network is adopted as the basic skeleton. We modify the first D-TDNN layer to a dynamic multi-scale convolution block, which represents local multi-scale features at a granular level and increases the range of receptive fields for convolution operation. Besides, Global multi-scale pooling aggregates different bottleneck layer features in order to collect information from multiple aspects.

### 3.1. Dynamic Kernel Convolution

The dynamic kernel convolution (Dk Conv) is a dynamic channel selection mechanism based on softmax attention. Specifically, the structure of selection mechanism in dynamic kernel convolution is illustrated as followed: high order statistic pooling (HOSP) - dense layer - dense layer - softmax, where HOSP is aimed to collect channel information from spatial dimension. The subsequent layers are aimed to assess the importance of

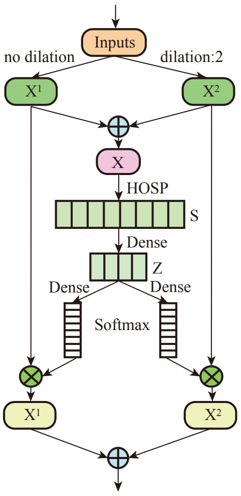

Figure 2: *Dynamic kernel convolution (Dk Conv) module. In this figure, ⊗ denotes element-wise multiplication. ⊕ denotes element-wise addition.*

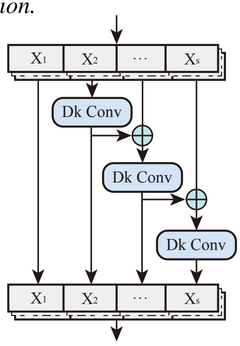

Figure 3: *Local multi-scale learning. In this figure, "Dk Conv" denotes dynamic kernel convolution operation, ⊕ denotes element-wise addition.*

different branches.

The dynamic kernel convolution is depicted in Figure 2. Multi-branch extension [21] of convolution captures features between short-term and long-term context adaptively. In proposed model, two branches are used, one of which exploits the methods of no dilation convolution, named $h_d^1$, while the other exploits the methods of the same kernel but dilation convolution, with dilation factor of 2, named $h_d^2$. $h_d^2$ is equivalent to a conventional convolution, which convolutional kernel is different from $h_d^1$. First, we combine the information from different convolution by summing up the features extracted from different branches:

$$X = h_d^1 + h_d^2. \tag{1}$$

Then the HOSP layer collects the mean, standard deviation, skewness and kurtosis information of the sum $X$ for each channel, named $\mu \in R^C$, $\sigma \in R^C$, $s \in R^C$, and $k \in R^C$.

Given the concatenation of $\mu$, $\sigma$, s, and k, we are able to obtain the softmax-weight after 2 dense layers by the following equation:

$$s_i = \tau({W_i}^T(\mathrm{V}^T[\mu;\sigma;s;k] + b) + n_i), \tag{2}$$

where $V \in R^{4C \times C/r}$ and $W_i \in R^{C/r \times C}$ are weight metrics, $b \in R^{C/r}$ and $n_i \in R^C$ are bias items. $\tau$ is the softmax activation function.

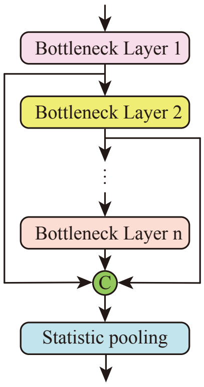

Figure 4: *Global multi-scale pooling. In this figure, Green "C" icon denotes the operation "concat".*

Finally, the dynamic representation $h_{out}$ indicates the element-wise dot production sum:

$$h_{out} = \sum_{i=1}^{2} s_i \otimes h_d^i. \tag{3}$$

### 3.2. Local Multi-Scale Learning

Inspired by the residual connection within ResNet layer in Res2Net [19, 22], the local multi-scale learning is adopted to improve representation performance within convolution. Local multi-scale learning refers to the multiple available receptive fields at a more granular level. As shown in Figure 3, we evenly split the feature into $s$ feature subsets, denoted by $X_i$, where $i \in [1, 2, \cdots, s]$.

A group of filters firstly extract features from the corresponding feature subsets. Output features of the previous group are then sent to the next group of filters along with another group of input features:

$$\begin{aligned} &Out_1 = X_1, \\ &Out_2 = F(X_2), \\ &Out_3 = F(Out_2 + X_3), \\ &\vdots \\ &Out_i = F(Out_{i-1} + X_i), \\ &\vdots \\ &Out_s = F(Out_{s-1} + X_s), \end{aligned} \tag{4}$$

where $F$ denotes the operation of Dk Conv. In Multi-scale Dk Block, the number of Dk Conv filters is divided by $s$. The number of operation $F$ filters is the same as D-TDNN layer. After that, we can get the concatenation of $Out_i$ as the module's output:

$$Out = [Out_1; \cdots; Out_i; \cdots; Out_s]. \tag{5}$$

Finally, after these feature subsets are processed, features from all groups are concatenated and sent to the next operation to fuse information altogether. By introducing the hyperparameter $s$, local multi-scale learning, which represents multi-scale features at a granular level, is shown effective to increase the range of receptive fields for convolution operation. Besides, with the reduction of convolutional filter numbers, the number of parameters decline significantly.

### 3.3. Global Multi-scale Pooling

Previous works [23, 24] concluded that feature aggregation at different layers can improve the accuracy of speaker embedding models in speaker verification task. The bottleneck feature is a kind of high-level information integration. Therefore, we aggregate different bottleneck features in channel dimension and send them to statistic pooling layer for the purpose of enhancing the dialect classification capabilities. The structure of global multi-scale pooling method is shown in Figure 4.

We redefine frame-level features $h_t$, integrating different bottleneck layer features $h_t^{bi} (i = 1, \cdots, n)$ in channel dimension, where n is the number of bottleneck layers.

$$h_t = [h_t^{b1}, \cdots h_t^{bi}, \cdots, h_t^{bn}]. \tag{6}$$

Then global multi-scale pooling layer calculates the mean vector $\mu$ as well as the second-order statistics in the form of the standard deviation vector $\sigma$ over frame-level features $h_t(t = 1, \cdots, T)$.

$$\mu = \frac{1}{T} \sum_{t=1}^{T} h_t. \tag{7}$$

$$\sigma = \sqrt{\frac{1}{T} \sum_{t=1}^{T} h_t \odot h_t - \mu \odot \mu}. \tag{8}$$

In experiments, two bottleneck layers are used for global multi-scale pooling. Feature representation in terms of global multi-scale pooling, producing higher dialect discriminability to utterance-level features.

## 4. Experimental setup

### 4.1. Datasets

In oriental language recognition (OLR) challenge 2020, additional training materials are forbidden to participants, and the permitted resources are several specified data sets [10]. We leverage AP17-OL3, AP17-OLR-test, AP18-OLR-test, AP19-OLR-dev, AP19-OLR-test, and AP20-OLR-dialect as our training set for dialect task. All of the training materials are collected from the permitted resources, which include sixteen languages. The detailed information of the combined dataset is shown in Table 1.

Table 1: *Dataset for training and evaluation.*

| # | Training | Validation | Test |
|---|---|---|---|
| languages | 16 | 16 | 6 |
| utterances | 194,814 | 3168 | 11399 |

### 4.2. Implementation Detail

Before training, we employ six types of data augmentation, including speed perturbation, the open-source SoX (tempo up, tempo down), the Kaldi recipe [14] in bubble, noise, reverb and music disturbance, increasing the amount and diversity of training data.

All features are extracted from 16kHz audio data. The 64-dimensional MFCC features are used in our system. Besides, additional 3-dimensional pitch features are appended to acoustic features. All features have frame-lengths of 25ms, frame-shifts of 10ms, and mean normalization over a sliding window of up to 3 seconds. No voice activity detection is applied. The feature engineering is executed using the Kaldi [27] platform.

Table 2: *Comparison for the top system. In this table, Team Royal Flush and Phonexia are the No.2 and No.1 in the leaderboard OLR challenge 2020, respectively. The proposed system achieves 9% of $C_{avg}$ and 45% of EER relative improvement, respectively.*

| Team | Parameters (M) | Model | $C_{avg}$ | EER(%) |
|---|---|---|---|---|
| Royal Flush[25] | 41 | Transformer-12L | 0.0871 | 11.97 |
| Phonexia[26] | 33.1 | ResNet18 | 0.0738 | 11.97 |
| ours | 2.9 | Global and Local Multi-scale Dk Conv | 0.0670 | 6.52 |

Table 3: *Results on AP20-OLR-dialect-task.*

| Model | Parameters (M) | Loss Function | $C_{avg}$ | EER(%) |
|---|---|---|---|---|
| D-TDNN | 3.3 | Softmax | 0.0798 | 8.02 |
| D-TDNN | 3.3 | AAM-Softmax | 0.0766 | 7.63 |
| Dynamic kernel Convolution (Dk Conv) | 3.4 | AAM-Softmax | 0.0758 | 7.50 |
| Local Multi-scale Dk Conv | 2.5 | AAM-Softmax | 0.0699 | 6.93 |
| Global and Local Multi-scale Dk Conv | 2.9 | AAM-Softmax | 0.0670 | 6.52 |

Our model is implemented with Tensorflow toolkits [28]. At every training step, we sample 16 languages. For each language, a segment with 200 to 400 frames is sliced from the utterances. Additive Angular Margin Softmax (AAM-Softmax) [29] loss is used to train the baseline system. $L_2$ regularization is applied to all layers in the network to prevent overfitting. We select stochastic gradient descent (SGD) as the optimizer and the initial learning rate is set to 0.01. The loss converges after the learning rate goes down below $10^{-6}$, resulting to around 1.20M training steps.

The D-TDNN layer of our system is the same as the layer presented in [18]. For all layers, we select 64 as the number of filters. The hyperparameter $s$ is set to 4. The context of the first TDNN layer and the last 6 D-TDNN layers is set to $[t-5:t+5]$, and the other layers is $[t-3:t+3]$. For dynamic kernel convolution, the kernel is set to 3.

### 4.3. Evaluation protocol

The system is evaluated on the AP20-OLR-dialect-task. Performance is measured by providing the $C_{avg}$, which is defined as the average of the pair-wise performance of test languages, given $P_{target}$ = 0.5 as the prior probability of the target language and the Equal Error Rate (EER). A concise ablation study is used to gain a deeper understanding on the effects of each proposed improvement to the overall performance.

## 5. Result and Analysis

The dialect identification is evaluated and ranked in OLR challenge 2020.

### 5.1. Horizontal Result Comparison

From Table 2, we can observe that the dynamic multi-scale convolution method performs noticeably better than other teams' model while operating in the same dialect identification task. Compared with OLR Challenge 2020 leaderboard No.1, our model can achieve 9.2% $C_{avg}$ and 45% EER relative improvement with only 2.9M parameters, respectively.

### 5.2. Result on AP20-OLR-dialect-task

Table 3 shows the performance of our ablation study on AP20-OLR-dialect-task. The softmax-output scores of Hokkienese, Sichuanese, and Shanghainese are analysed in the evaluation. All proposed models outperform the state-of-the-art system submitted to OLR challenge 2020 in EER. Noticeably, the proposed dynamic multi-scale convolution method achieves the best performance in all metrics including $C_{avg}$, which suggests that the proposed method is effective for dialect identification.

Results demonstrate that the model with AAM-Softmax is potential to achieve better accuracy than those with softmax loss function. Comparing D-TDNN based method, we discover the operation of dynamic kernel convolution is helpful for dialect identification. Local multi-scale dynamic kernel convolution combines multi-scale learning with dynamic kernel convolution, further improving the performance and reducing 36% parameters with only 2.5 million by introducing multi-scale learning. Besides, Local multi-scale learning within the convolution is effective in reducing the number of parameters by the hyperparameter $s$. Global and local multi-scale dynamic kernel convolution denotes a variant of local multi-scale DK Conv which adopts global multi-scale pooling. Comparing the results of global and local multi-scale dynamic kernel convolution with local multi-scale Dk Conv result, it can be seen that global multi-scale pooling is excellent to improve the accuracy of dialect identification.

## 6. Conclusions

In this paper, we propose a novel dynamic multi-scale convolution model for dialect identification, which introduces dynamic kernel convolution, local multi-scale learning, and global multi-scale pooling.

We evaluate the proposed method with D-TDNN baseline system and carry out the comparison with other submitted systems. Experiments are conducted on OLR challenge AP20-OLR-dialect-task. Significantly, the dynamic multi-scale convolution model achieves the best $C_{avg}$ of 0.0670 and EER of 6.52% in OLR challenge 2020 AP20-OLR-dialect-task. Compared with the known best results, our method achieves 9% of $C_{avg}$ and 45% of EER relative improvement, respectively. Furthermore, the parameters of the proposed model are 91% fewer than the best known model.

## 7. Acknowledgements

This work was supported in part by the China Major S&T Project (2018ZX01028101-004), the National Key R&D Project(2018YFB2202600), the NSFC (61774094 and U19B2041) and the Beijing S&T Project (Z191100007519016).

# 8. References

[1] H. Li, B. Ma, and K. A. Lee, "Spoken language recognition: from fundamentals to practice," *Proceedings of the IEEE*, vol. 101, no. 5, pp. 1136–1159, 2013.

[2] F. Biadsy, "Automatic dialect and accent recognition and its application to speech recognition," Ph.D. dissertation, Columbia University, 2011.

[3] A. Waibel, P. Geutner, L. M. Tomokiyo, T. Schultz, and M. Woszczyna, "Multilinguality in speech and spoken language systems," *Proceedings of the IEEE*, vol. 88, no. 8, pp. 1297–1313, 2000.

[4] P. L. Patrick, M. S. Schmid, and K. Zwaan, "Language analysis for determination of origin," *The Oxford handbook of language and law*, 2012.

[5] A. H. Poorjam, R. Saeidi, T. Kinnunen, and V. Hautamäki, "Incorporating uncertainty as a quality measure in i-vector based language recognition." in *Odyssey*, 2016, pp. 74–80.

[6] D. Wang, L. Li, D. Tang, and Q. Chen, "Ap16-ol7: A multilingual database for oriental languages and a language recognition baseline," in *2016 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*. IEEE, 2016, pp. 1–5.

[7] Z. Tang, D. Wang, Y. Chen, and Q. Chen, "Ap17-olr challenge: Data, plan, and baseline," in *2017 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE, 2017, pp. 749–753.

[8] Z. Tang, D. Wang, and Q. Chen, "Ap18-olr challenge: Three tasks and their baselines," in *2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, 2018, pp. 596–600.

[9] Z. Tang, D. Wang, and L. Song, "Ap19-olr challenge: Three tasks and their baselines," in *2019 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE, 2019, pp. 1917–1921.

[10] Z. Li, M. Zhao, Q. Hong, L. Li, Z. Tang, D. Wang, L. Song, and C. Yang, "Ap20-olr challenge: Three tasks and their baselines," in *2020 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE, 2020, pp. 550–555.

[11] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. 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.

[12] 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.

[13] D. Snyder, D. Garcia-Romero, and D. Povey, "Time delay deep neural network-based universal background models for speaker recognition," in *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*. IEEE, 2015, pp. 92–97.

[14] D. Snyder, D. Garcia-Romero, G. Sell, A. McCree, D. Povey, and S. Khudanpur, "Speaker recognition for multi-speaker conversations using x-vectors," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 5796–5800.

[15] D. Povey, G. Cheng, Y. Wang, K. Li, H. Xu, M. Yarmohammadi, and S. Khudanpur, "Semi-orthogonal low-rank matrix factorization for deep neural networks." in *Interspeech*, 2018, pp. 3743–3747.

[16] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[17] Y. Zhu and S. Newsam, "Densenet for dense flow," in *2017 IEEE international conference on image processing (ICIP)*. IEEE, 2017, pp. 790–794.

[18] Y.-Q. Yu and W.-J. Li, "Densely connected time delay neural network for speaker verification," *Proc. Interspeech 2020*, pp. 921–925, 2020.

[19] S. Gao, M.-M. Cheng, K. Zhao, X.-Y. Zhang, M.-H. Yang, and P. H. Torr, "Res2net: A new multi-scale backbone architecture," *IEEE transactions on pattern analysis and machine intelligence*, 2019.

[20] X. Li, W. Wang, X. Hu, and J. Yang, "Selective kernel networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 510–519.

[21] X. Zhao, H. Zhang, G. Zhu, F. You, S. Kuang, and L. Sun, "A multi-branch 3d convolutional neural network for eeg-based motor imagery classification," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 27, no. 10, pp. 2164–2177, 2019.

[22] B. Desplanques, J. Thienpondt, and K. Demuynck, "Ecapa-tdnn: Emphasized channel attention, propagation and aggregation in tdnn based speaker verification," in *Interspeech2020*, 2020, pp. 1–5.

[23] Z. Gao, Y. Song, I. McLoughlin, P. Li, Y. Jiang, and L.-R. Dai, "Improving aggregation and loss function for better embedding learning in end-to-end speaker verification system." in *INTERSPEECH*, 2019, pp. 361–365.

[24] A. Hajavi and A. Etemad, "A deep neural network for short-segment speaker recognition," *Proc. Interspeech 2019*, pp. 2878–2882, 2019.

[25] D. Wang, S. Ye, and X. Hu, "The royal flush system for ap20-olr challenge."

[26] M. Klco, O. Novotnỳ, J. Profant, and J. Slavıcek, "Phonexia sro submission to olr 2020."

[27] 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.

[28] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard *et al.*, "Tensorflow: A system for large-scale machine learning," in *12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16)*, 2016, pp. 265–283.

[29] Y. Liu, L. He, and J. Liu, "Large margin softmax loss for speaker verification," *Proc. Interspeech 2019*, pp. 2873–2877, 2019.