Speaker Turn Aware Similarity Scoring for Diarization of Speech-Based Cognitive Assessments

Sean Shensheng Xu*†, Man-Wai Mak†, Ka Ho Wong‡, Helen Meng‡, and Timothy C.Y. Kwok§¶
* School of Biomedical Engineering, Health Science Center, Shenzhen University
† Department of Electronic and Information Engineering, The Hong Kong Polytechnic University
‡ Department of Systems Engineering and Engineering Management, The Chinese University of Hong Kong
§ Department of Medicine and Therapeutics, The Chinese University of Hong Kong
¶ Jockey Club Centre for Osteoporosis Care and Control, The Chinese University of Hong Kong

***Abstract*—This paper proposes two enhancements to the conventional speaker diarization methods for speech-based Montreal cognitive assessments (MoCA). The enhancements address the technical challenges of MoCA recordings on two fronts. First, multi-scale channel interdependence speaker embedding is used as the front-end speaker representation for overcoming the acoustic mismatch caused by far-field microphones. Specifically, a squeeze-and-excitation (SE) unit and channel-dependent attention are added to Res2Net blocks for multi-scale feature aggregation. Second, a sequence comparison approach with a holistic view of the whole conversation is applied to measure the similarity of short speech segments in the conversation, which results in a speaker-turn aware scoring matrix for the subsequent clustering step. Evaluations on an interactive dialog dataset for MoCA show that the proposed enhancements lead to a diarization system that outperforms the conventional x-vector/PLDA systems under language-, age-, and microphone mismatch scenarios. The results also show that the speaker-turn timestamps can be hypothesized, suggesting that the proposed enhancements are amendable to datasets without speaker timestamp information.**

I. Introduction

Cognitive tests are tools for evaluating the cognitive capabilities of humans. Montreal cognitive assessments (MoCA) [1] is a widely used test for detecting mild cognitive impairment (MCI) and Alzheimer's disease (AD) in older adult. Studies have found that the irregularities due to MCI and AD will appear in patients' speech [2]. Because a MoCA session involves the spoken dialogs between an assessor and a patient, it is essential to perform speaker diarization to extract the utterances spoken by the patients as a first step towards the efficient analysis of the patient's speech.

Speaker diarization is the process of partitioning an input audio into homogeneous segments according to the speaker identities. It answers the question of who spoke when. In general, the diarization process consists of the following steps. First, a voice activity detector (VAD) is applied to remove non-speech parts from the input audio. Next, speech regions are uniformly partitioned into short overlapping segments. After that, the segments are mapped into a fixed-dimensional feature space by a speaker embedding network such as the x-vector network [3], [4]. Then, a similarity matrix is produced by computing the probabilistic linear discriminant analysis (PLDA) scores [5], [6] between each pair of segments. Finally, agglomerative hierarchical clustering (AHC) is applied to the similarity matrix to obtain the diarizaion results.

The MoCA recordings present special challenges to speaker diarization. Conventionally, researchers of speaker embeddings focused on long utterances (over 5s). However, the MoCA tests consist mainly of short utterances in interactive dialogs. It is difficult to extract sufficient information for discriminating speakers. This problem is exacerbated by the fact that the interactive dialogs have backchannel cues and frequent changes in speaker turns, which lead to a high probability of missing the speaker change points. An essential requirement of MoCA tests is that the recording devices should not disturb or affect the patient during a recording session. Ideally, the patient should not know the existence of the devices. Therefore, in practice, MoCA sessions use far-field microphones for recording. But this will cause microphone mismatch issues because diarization systems are typically trained on speech recorded by close-talking microphones. The mismatch calls for a more robust speaker embedding method that is less sensitive to the microphone types.

Agglomerative hierarchical clustering [7] is one of the most widely used clustering approach to speaker diarization. Bayesian information criterion is usually used to estimate which couple of clusters should be merged at each agglomerative iteration. This leads to a high computational cost when the number of data points increases. Also, the performance of AHC heavily depends on the choice of the distance metric [8]. In contrast, spectral clustering (SC) [7] does not require a statistical metric to determine whether two clusters should be merged. Previous researches have applied SC to infer speaker clusters and achieved good performance [9], [10], specifically in speaker diarization task [11].

This work aims to enhance our age-invariant diarization system [12] for speech-based cognitive assessments. A speaker embedding extractor, CE-Res2Net [13], is used to produce multi-scale channel interdependence speaker embeddings as front-end representations. Instead of PLDA, a long short-term memory (LSTM) scoring model [14] trained on the sequential information across short speech segments is applied for similarity measure. The resulting diarization system was applied to a MoCA dataset comprising 469 older adults, including healthy individuals and patients with mild to major neurocognitive disorders (NCDs). It was found that the en-

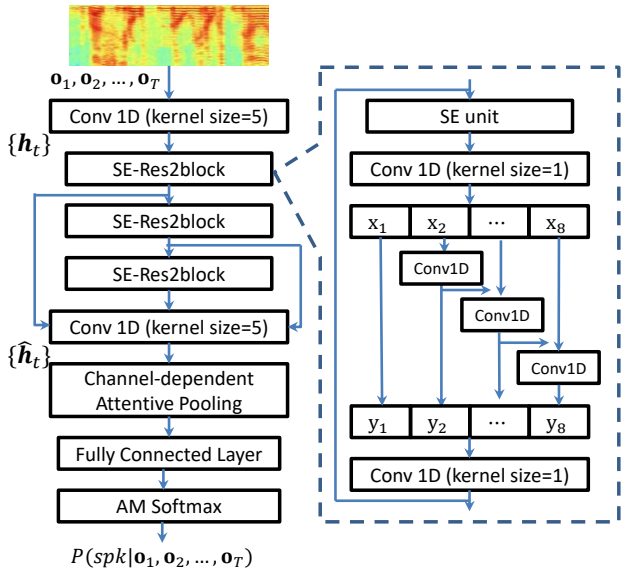

Fig. 1. Structure of the CE-Res2Net and the SE-based Res2block. $\{\boldsymbol{h}_t\}_{t=1}^{T}$ denote the frame-level features. T is the utterance length. $\hat{\boldsymbol{h}}_t$ denotes the last frame-level convolutional layer's output.

hanced embedding can overcome the acoustic mismatch due to the far-field microphones and that the LSTM model can leverage the ground-truth speaker-turn information in training data or the hypothesized timestamps in the MoCA data.

II. Diarization System Overview

A. Kaldi X-Vector Networks

X-vector is a speaker embedding approach based on deep neural networks (DNN), which has demonstrated good performance in both speaker recognition [3] and speaker diarization [4]. In Kaldi x-vector networks [15], MFCCs are extracted and fed to time-delay layers [16] for frame-level processing. Then, a statistics pooling layer aggregates over the frame-level representations at the last time-delay layer into a segment-level representation, followed by two fully connected layers and a softmax layer to output the posterior probabilities of speakers. The penultimate layer's outputs form the speaker embeddings called x-vectors.

B. PLDA Scoring and AHC

The x-vector/PLDA/AHC framework has been widely used in speaker diarization systems [8], [12], [17]. The AHC is an unsupervised clustering and merging method. We performed PLDA scoring on all pairs of segments (x-vectors) for each recordings. The PLDA scores were then used as input to the AHC algorithm for classifying speech segments by speaker identities. In this work, the baseline systems were conducted based on this framework.

III. Proposed Diarization System

A. Channel-interdependence Enhanced Res2Net

The channel-interdependence enhanced Res2Net (CE-Res2Net) [13] was designed for tackling the problems of environmental noise and reverberation distortion in far-field speaker verification. Because the same problems exist in MoCA recordings, in this work, we applied CE-Res2Net for speaker embedding. The configuration of the CE-Res2Net is shown in Fig. 1. The squeeze-and-excitation (SE) unit [18] is placed before the convolutional operations of the Res2block, which rescales the channel activations and facilitates the convolutional operations to learn multi-scale features.

In conventional speaker embedding, a self-attentive pooling layer [19], [20] assigns a weight e_t for each frame-level vector $\boldsymbol{h}_t \in \mathbb{R}^C$, where C is the number of channels in the last convolutional layer. The weights e_t's are the output of a trainable network whose input is $\boldsymbol{h}_t$'s. However, this kind of mechanism assumes that all channels are of equal importance. To explore the importance of individual channels, the CE-Res2Net uses channel-dependent attentive pooling [21] to compute a scalar score $e_{t,c}$ for each channel and each frame-level vector $\hat{\boldsymbol{h}}_t$ at the last convolutional layer's output. Therefore, in Fig. 1, given $\hat{\boldsymbol{h}}_t$, the attention network computes

$$e_{t,c} = \boldsymbol{v}_c^{\mathsf{T}} f(\boldsymbol{W}\hat{\boldsymbol{h}}_t), \quad c = 1, \ldots, C, \tag{1}$$

where $\boldsymbol{v}_c$ and $\boldsymbol{W}$ are trainable parameters and $f()$ is a non-linear function such as ReLU. $e_{t,c}$ is then normalized across time by a softmax function:

$$w_{t,c} = \frac{\exp(e_{t,c})}{\sum_{\tau=1}^{T} \exp(e_{\tau,c})}, \quad c = 1, \ldots, C. \tag{2}$$

Given a set of channel-dependent weights $w_{t,c}$, the weighted average of channel c can be obtained:

$$\hat{\mu}_c = \sum_{t=1}^{T} w_{t,c}\hat{h}_{t,c}. \tag{3}$$

The weighted mean vector is $\hat{\boldsymbol{\mu}} = [\hat{\mu}_1, \hat{\mu}_2, \ldots, \hat{\mu}_C]^{\mathsf{T}}$. Similar to the self-attentive pooling, the elements of the weighted standard deviation vector $\hat{\boldsymbol{\sigma}} = [\hat{\sigma}_1, \hat{\sigma}_2, \ldots, \hat{\sigma}_C]^{\mathsf{T}}$ can be computed as follows:

$$\hat{\sigma}_c = \sqrt{\frac{1}{T}\sum_{t=1}^{T} w_{t,c}\hat{h}_{t,c}^2 - \hat{\mu}_c^2}, \quad c = 1, \ldots, C. \tag{4}$$

By concatenating the weighted mean vector $\hat{\boldsymbol{\mu}}$ and the weighted standard deviation vector $\hat{\boldsymbol{\sigma}}$, the output of the channel-dependent attention pooling is obtained.

B. LSTM-Based Similarity Measurement

Although PLDA scoring is a widely used method for quantifying the similarity between short speech segments (typically 1.5s) in speaker diarization systems, each PLDA score is based on the speaker embeddings of two short segments only, ignoring the remaining segments in a conversation. Because of the nature of conversations, each speaker will likely produce

a consecutive sequence of short segments in a conversation, i.e., neighboring segments have a higher chance of being produced by the same speaker. Therefore, instead of treating the segments independently, as in PLDA scoring, we should have a more holistic view of the segments. This notion leads to the LSTM-based scoring in [14], which aims to capture the sequential information across the segments.

Given a conversation, we obtain a sequence of speaker embeddings $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_t, \ldots, \mathbf{x}_T\}$, where $\mathbf{x}_t$ represents the t-th segment's embedding and T is the number of segments. Each embedding, say $\mathbf{x}_t$, is concatenated with all the other embeddings to form a vector sequence of double dimension:

$$\mathcal{X}_t = \left\{ \begin{bmatrix} \mathbf{x}_t \\ \mathbf{x}_1 \end{bmatrix}, \ldots, \begin{bmatrix} \mathbf{x}_t \\ \mathbf{x}_t \end{bmatrix}, \ldots, \begin{bmatrix} \mathbf{x}_t \\ \mathbf{x}_T \end{bmatrix} \right\}. \tag{5}$$

To exploit the temporal information in $\mathcal{X}_t$, it is fed to a Bi-LSTM network [22] to produce the output:

$$\begin{aligned} \mathbf{S}_t &= [S_{t1}, \ldots, S_{tt}, \ldots, S_{tT}] \\ &= f_{\text{LSTM}} \left(\begin{bmatrix} \mathbf{x}_t \\ \mathbf{x}_1 \end{bmatrix}, \ldots, \begin{bmatrix} \mathbf{x}_t \\ \mathbf{x}_t \end{bmatrix}, \ldots, \begin{bmatrix} \mathbf{x}_t \\ \mathbf{x}_T \end{bmatrix} \right). \end{aligned} \tag{6}$$

The vectors $\mathbf{S}_t$, $t = 1, \ldots, T$, are then stacked row-wise to form a scoring matrix $\mathbf{S}$. By using $\mathcal{X}_t$ in Eq. 5 as the input to the LSTM, each LSTM score in $\mathbf{S}_t$ depends not only on two embeddings but also on all other embeddings and the sequential information in the concatenated vectors.

The basic idea of the method is to learn a reference similarity matrix, comprising blocks of ones and zeros. A '1' in the (i, j) entry of the reference matrix means that the i-th and j-th segments are produced by the same speaker; otherwise, it is a '0'. The matrix is formed from the speaker labels and timestamps of who spoke when in the training data, which is used as the labels for training the LSTM network.

In [14], a K-fold cross validation was applied to the Callhome dataset because timestamped speaker labels are avaliable in Callhome. LSTM scoring can leverage the timestamp information about who spoke when in the training data. The diarization performance and the impact of utilizing both speaker labels and timestamp information are revealed in Section V.

C. Spectral Clustering

Spectral clustering (SC) can be viewed as graph cuts [7]. The basic idea is to use the spectrum (eigenvalues) of an affinity matrix to perform dimension reduction. The general process of spectral clustering consists of three steps. First, a similarity graph based on all data points is constructed. Second, the data points are embedded on a low-dimensional space (spectral embedding), using the eigenvectors of the graph Laplacian. Third, a classical clustering algorithm (e.g., K-means) is applied to partition the embeddings.

Specifically, given a scoring matrix $\mathbf{S} \in \mathbb{R}^{n \times n}$ with elements $S_{ij} \geq 0$ and $S_{ii} = 0 \ \forall \ i$, we consider S_{ij} as the weight of the edge between nodes i and j in an undirected graph. Then, we compute a Laplacian matrix $\mathbf{L} = \mathbf{D} - \mathbf{S}$ and perform the following normalization:

$$\mathbf{L}_{\text{norm}} = \mathbf{D}^{-\frac{1}{2}} \mathbf{L} \mathbf{D}^{-\frac{1}{2}}, \tag{7}$$

where $\mathbf{D}$ is a diagonal matrix with $D_{ii} = \sum_j S_{ij}$. Next, we select the number of clusters k and take the k smallest eigenvalues $\lambda_1, \ldots, \lambda_k$ and their corresponding eigenvectors $\mathbf{u}_1, \ldots, \mathbf{u}_k$ from $\mathbf{L}_{\text{norm}}$ to form a matrix $\mathbf{U} = [\mathbf{u}_1, \ldots, \mathbf{u}_k] \in \mathbb{R}^{n \times k}$ using $\mathbf{u}_1, \ldots, \mathbf{u}_k$ as columns. Finally, we apply the K-means algorithm to cluster row vectors $\mathbf{y}_1, \ldots, \mathbf{y}_n$ in $\mathbf{U}$ to form k classes, where $\mathbf{y}_i \in$ class k indicates that segment i belongs to speaker k.

Fig. 2. Collection of JCCOCC MoCA Cantonese Speech Corpus.

IV. Experimental Setup

A. MoCA Cantonese Speech Corpus

The JCCOCC Montreal Cognitive Assessment (MoCA) Cantonese Speech corpus was collected by the CUHK Jockey Club Centre for Osteoporosis Care and Control. In the corpus, a MoCA test was conducted for each participants. There are 469 participants (both genders), each having an interactive spoken dialog session with an assessor with an average duration of 26 minutes. The participants cover an age range of 72–100. The recordings were captured in a quiet office by two smartphones (iPhone 6 and Samsung Galaxy S6) placing at a distance from the participant, as shown in Fig. 2. All of the 469 conversations were used in this work.

B. Evaluation Data

Among the 469 MoCA recordings, 256 (named MoCA-256) have been manually transcribed, and they were used for evaluating the performance of different diarization systems. The total duration of the evaluation data is 103.5 hours, of which the speech duration of the assessors and the participants are 33.8 hours and 18.6 hours, respectively.

C. Training Data for Speaker Embedding Networks

In the experiments, the x-vector extractors and the CE-Res2Net were trained on the National Institute of Standards and Technology (NIST) Speaker Recognition Evaluations (SREs) and the Switchboard (SWB) datasets, including SRE 2004, 2005, 2006, 2008, SWB2 Phases 1, 2 and 3, SWB

TABLE I
SOURCE OF DATA FOR TRAINING THE X-VECTOR EXTRACTORS (KALDI) AND CE-RES2NET.

Data Source	#Speakers	#Hours	#Utterances
SRE 2004–2008 SWB and augmentation	4,979	2,789	62,151 (clean) 184,533 (aug.)

TABLE II
DIARIZATION PERFORMANCE OF THE BASELINE SYSTEMS (BASED ON MOCA-256). THE SRE DATA WAS USED TO TRAIN THE PLDA MODELS.

System Architecture	Performance Metrics (%)			
	DER	MS	FA	SE
Kaldi x-vectors + PLDA + AHC	7.37	2.7	1.8	2.9
CE-Res2Net + PLDA + AHC	6.86	2.7	1.8	2.4

Cellular1, and SWB Cellular2. To obtain robust embeddings for diarization, we followed the data augmentation procedure in the Kaldi recipe and roughly doubled the size of the original clean data, i.e., using the room impulse responses (RIR) [23] and the MUSAN datasets [24] to create room reverberation and additive noise, respectively. Note that short utterances (number of frames less than 400) and speakers with less than 8 utterances were excluded. The statistics of the data for training the speaker embedding networks are shown in Table I. We followed the Kaldi's Callhome recipe[1] to train the speaker embedding networks.

D. Training Data for Similarity Measurement Models

To investigate the performance of LSTM scoring, in addition to the 256 transcribed recordings, we also utilized the remaining unlabeled[2] 213 MoCA recordings (called MoCA-213) as in-domain data to train the scoring models. Because information of speaker-turn timestamps is required for training the LSTM scoring models, we hypothesized the timestamped speaker labels of MoCA-213 in our experiments. In addition, we also used the Callhome portion of NIST SRE 2000 as out-of-domain data for training. Callhome[3] is a widely used telephone speech dataset containing 500 sessions with a total duration of 18 hours. The number of speakers per session varies from 2 to 7. Note that timestamped speaker labels are available in Callhome. Therefore, we used it to train both the LSTM and PLDA models for performance comparison.

To train the LSTM models, we partitioned a long conversation into 300-second blocks (i.e., $T = 400$ in Eq. 6) and created a $T \times T$ reference matrix for each block. The partitioning is to ensure enough temporal information in the blocks without excessive burden on computation resources. During scoring, the same partitioning was applied to the test conversations.

[1]https://github.com/kaldi-asr/kaldi/tree/master/egs/callhome_diarization/v2

[2]In this work, the term "unlabeled" means there is no annotation, such as speaker labels and timestamps of who spoke when in the dataset, i.e., it only has audio files.

[3]2000 NIST Speaker Recognition Evaluation (LDC2001S97), Disk-8.

E. Experimental Settings

For SRE and SWB data, we used Kaldi's energy-based voice activity detection (VAD) to remove silence regions. For the JCCOCC MoCA data, we used the ASpIRE speech activity detector (SAD).[4] The reason for using two different VADs is that SRE and SWB contain clean telephone conversations. The signal-to-noise ratios are very high, and Kaldi VAD can do a good job. On the other hand, the interactive dialogs in JCCOCC MoCA were collected by smartphones placing far away from the patients, causing lower signal-to-noise ratios. As a result, a DNN-based VAD that is more robust to noise was used for silence removal.

A sliding window of 1.5s with 0.75s shift was used to extract the embeddings in the speech regions of each conversation. Speech regions less than 1.5s were ignored. For each segment (or embedding), we computed a sequence of 23-dimensional MFCCs using a sliding window of 25ms with a frameshift of 10ms; the MFCCs were then presented to the speaker embedding network to extract a speaker embedding vector.

We followed the configuration of CE-Res2Net described in [13]. 192-dimensional speaker embeddings were extracted from the affine layer's output after the statistics pooling layer. In addition, the LSTM-based scoring model in our experiments consists of two Bi-LSTM layers (384–384), followed by two dense layers (64–1). Each Bi-LSTM layer has 384 nodes including 192 forward nodes and 192 backward nodes. The first dense layer has 64 nodes with ReLU activation. The output layer has one node with sigmoid activation, which gives similarity scores between 0 and 1.

In general, a stopping threshold is needed in the clustering algorithms. However, because the number of speakers per recording is known, such stopping threshold is not needed in our case.

F. Performance Metrics

We reported the diarization error rate (DER) [25] of different systems, which is a common performance metric for speaker diarization. DER is the sum of the duration of missed speech (MS), false alarm (FA), and speaker error (SE) divided by the total duration:

$$\mathrm{DER} = \frac{\mathrm{Dur(MS)} \ + \ \mathrm{Dur(FA)} \ + \ \mathrm{Dur(SE)}}{\mathrm{Total\ Duration\ of\ Reference\ Speech}}. \qquad (8)$$

In accordance with other studies [4], [14], [26], we allowed a non-scoring collar of 0.25s around the reference segment boundaries and ignored the overlapped segments. Because MS and FA are caused by VAD errors, we may use SE to compare performance if the same VAD was used for all systems.

V. EXPERIMENTAL RESULTS

First, we constructed two baseline systems based on the evaluation set (MoCA-256), i.e., using Kaldi x-vector networks and CE-Res2Net for embedding extraction and PLDA for similarity measures. We used SRE data (without augmentation)

[4]https://kaldi-asr.org/models/m4

TABLE III
DIARIZATION PERFORMANCE ACHIEVED BY DIFFERENT SIMILARITY MEASURES BASED ON DIFFERENT TRAINING DATA AND LABEL INFO.

Similarity Measurement				Clustering	Performance Metrics (%)			
Case	Model	Training Data	Label Info	Algorithm	DER	MS	FA	SC
1	PLDA	Callhome	Speaker labels (Ground truth)	AHC	7.72	2.7	1.8	3.3
2	LSTM	Callhome	Timestamped speaker labels (Ground truth)	AHC	6.89	2.7	1.8	2.4
3	LSTM	Callhome	Timestamped speaker labels (Ground truth)	SC	6.60	2.7	1.8	2.1
4	LSTM	MoCA-256 (5-fold)	Timestamped speaker labels (Ground truth)	AHC	5.95	2.7	1.8	1.5
5	LSTM	MoCA-256 (5-fold)	Timestamped speaker labels (Ground truth)	SC	5.75	2.7	1.8	1.3
6	LSTM	MoCA-213	Timestamped speaker labels (Hypothesized)	AHC	6.23	2.7	1.8	1.8
7	LSTM	MoCA-213	Timestamped speaker labels (Hypothesized)	SC	6.12	2.7	1.8	1.7

for training the PLDA models. Table II shows the diarization performance of the baseline systems. The results based on the evaluation set (MoCA-256) show that CE-Res2Net can produce better embeddings and achieve a lower DER. Therefore, we only used CE-Res2Net for embedding extraction in subsequent experiments (see Table III).

To improve diarization performance, we replaced the conventional PLDA backend with LSTM scoring. We employed in-domain (e.g., MoCA) and out-of-domain (e.g., Callhome) data for training the models. Moreover, the in-domain data with hypothesized labels were utilized. We also applied different clustering algorithms (e.g., AHC and SC) for comparisons. The diarization performance is given in Table III. In Case 4 and Case 5, the LSTM models were trained using the labeled in-domain data. Specifically, 5-fold cross-validation was conducted to estimate the performance, i.e., the evaluation set (MoCA-256) was randomly partitioned into five equal-sized subsets. A subset was retained as the test data while the remaining four subsets were used for training the LSTM model. The procedure was repeated five times, and each subset was used once as the test data. After that, the 5-fold test results were combined to calculate the DER. In contrast, the LSTM models in Case 2 and Case 3 were trained using the labeled out-of-domain data (Callhome). In Case 6 and Case 7, the unlabeled in-domain data (MoCA-213) were used to train the LSTM models. Note that, the corresponding labels (i.e., timestamped speaker labels) of MoCA-213 were hypothesized by the baseline system (CE-Res2Net + PLDA) in Table II. Therefore, the training in Case 6 and Case 7 is semi-supervised.

The results based on the evaluation set (MoCA-256) demonstrate that spectral clustering outperforms the AHC in all cases. The in-domain 5-fold cross validation with ground-truth labels in Case 5 achieves the lowest DER. In Case 1, the lack of training data in Callhome may cause poorer performance than the baseline (Table II). Case 2 and Case 3 achieve performance comparable with the baseline even with less training data, which demonstrate the benefit of using the timestamp information in Callhome. We utilized unlabeled in-domain data to train the LSTM models in Case 6 and Case 7, and both DERs are lower than the baseline in Table II, demonstrating the effectiveness of learning representations from in-domain data. Note that, the unlabeled in-domain data (MoCA-213) cannot be used to train the PLDA model because we cannot be sure the same speaker exists in another MoCA recording.

VI. CONCLUSIONS

In this paper, we propose a speaker diarization system for speech-based MoCA recordings. The system incorporates a CE-Res2Net embedding extractor and an LSTM-based scoring model. To obtain better speaker embeddings, the CE-Res2Net exploits the interdependence between the channels in the last convolutional layer. The LSTM model, which learns the speaker turn patterns in MoCA recordings, substitutes the conventional PLDA for similarity measures. Experimental results based on MoCA data show that by leveraging both speaker labels and timestamp, the LSTM scoring model trained on in-domain or out-of-domain data performs better than the PLDA model. While LSTM scoring requires the timestamp information about who spoke when in the training data, results show that the LSTM model can tolerate some errors in the timestamps, suggesting that this scoring approach can leverage unlabeled training data via hypothesizing the timestamp information.

ACKNOWLEDGMENT

This work was in part supported by Research Grands Council of Hong Kong, Theme-based Research Scheme (Ref.: T45-407/19-N).

REFERENCES

[1] Z. S. Nasreddine, N. A. Phillips, V. Bédirian, S. Charbonneau, V. Whitehead, I. Collin, J. L. Cummings, and H. Chertkow, "The Montreal cognitive assessment, MoCA: A brief screening tool for mild cognitive impairment," *Journal of the American Geriatrics Society*, vol. 53, no. 4, pp. 695–699, 2005.

[2] A. Konig, A. Satt, and A. S. *et al.*, "Use of speech analyses within a mobile application for the assessment of cognitive impairment in elderly people," *Current Alzheimer Research*, vol. 15, no. 2, pp. 120–129, 2018.
[3] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust DNN embeddings for speaker recognition," in *Proc. ICASSP 2018*, pp. 5329–5333.
[4] D. Garcia-Romero, D. Snyder, G. Sell, D. Povey, and A. McCree, "Speaker diarization using deep neural network embeddings," in *Proc. ICASSP 2017*, pp. 4930–4934.
[5] S. J. D. Prince and J. H. Elder, "Probabilistic linear discriminant analysis for inferences about identity," in *Proc. ICCV 2007*, pp. 1–8.
[6] M. W. Mak and J. T. Chien, "Machine Learning for Speaker Recognition." Cambridge University Press, 2020.
[7] U. V. Luxburg, "A tutorial on spectral clustering," *Statistics and Computing*, vol. 17, no. 4, pp. 395–416, 2007.
[8] K. J. Han, S. Kim, and S. S. Narayanan, "Strategies to improve the robustness of agglomerative hierarchical clustering under data source variation for speaker diarization," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, no. 8, pp. 1590–1601, 2008.
[9] H. Ning, W. Xu, Y. Chi, Y. Gong, and T. S. Huang, "Incremental spectral clustering by efficiently updating the eigen-system," *Pattern Recognition*, vol. 43, no. 1, pp. 113–127, 2010.
[10] K. Iso, "Speaker clustering using vector quantization and spectral clustering," in *Proc. ICASSP 2010*, pp. 4986–4989.
[11] H. Ning, M. Liu, H. Tang, and T. S. Huang, "A spectral clustering approach to speaker diarization," in *Proc. INTERSPEECH 2006*, pp. 2178–2181.
[12] S. S. Xu, M. W. Mak, K. H. Wong, H. Meng, and T. C. Y. Kwok, "Age-invariant speaker embedding for diarization of cognitive assessments," in *Proc. ISCSLP 2021*.
[13] L. Zhao and M. W. Mak, "Channel interdependence enhanced speaker embeddings for far-field speaker verification," in *Proc. ISCSLP 2021*.
[14] Q. Lin, R. Yin, M. Li, H, Bredin, and C. Barras, "LSTM based similarity measurement with spectral clustering for speaker diarization," in *Proc. INTERSPEECH 2019*, pp. 366–370.
[15] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, and M. H. *et al.*, "The Kaldi speech recognition toolkit," in *Proceedings of the Automatic Speech Recognition and Understanding Workshop*, 2011.
[16] D. Snyder, D. Garcia-Romero, and S. K. D. Povey, "Deep neural network embeddings for text-independent speaker verification," in *Proc. INTERSPEECH 2017*, pp. 999–1002.
[17] M. Diez, L. Burget, S. Wang, J. Rohdin, and J. Černocký, "Bayesian HMM based x-vector clustering for speaker diarization," in *Proc. INTERSPEECH 2019*, pp. 346–350.
[18] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE CVPR 2018*, pp. 7132–7141.
[19] K. Okabe, T. Koshinaka, and K. Shinoda, "Attentive statistics pooling for deep speaker embedding," *arXiv:1803.10963*, 2018.
[20] Y. Zhu, T. Ko, D. Snyder, B. Mak, and D. Povey, "Self-attentive speaker embeddings for text-independent speaker verification," in *Proc. INTERSPEECH 2018*, pp. 3573–3577.
[21] B. Desplanques, J. Thienpondt, and K. Demuynck, "Ecapa-tdnn: Emphasized channel attention, propagation and aggregation in TDNN based speaker verification," *arXiv:2005.07143*, 2020.
[22] Z. Huang, W. Xu, and K. Yu, "Bidirectional lstm-crf models for sequence tagging," *arXiv:1508.01991*, 2015.
[23] T. Ko, V. Peddinti, D. Povey, M. Seltzer, and S. Khudanpur, "A study on data augmentation of reverberant speech for robust speech recognition," in *Proc. ICASSP 2017*, pp. 5220–5224.
[24] D. Snyder, G. Chen, and D. Povey, "Musan: A music, speech, and noise corpus," *arXiv:1510.08484v1*, 2015.
[25] J. G. Fiscus, J. Ajot, M. Michel, and J. S. Garofolo, "The rich transcription 2006 spring meeting recognition evaluation," in *International Workshop on Machine Learning for Multimodal Interaction*, 1980, pp. 309–322.
[26] X. Anguera, S. Bozonnet, N. Evans, C. Fredouille, G. Friedland, and O. Vinyals, "Speaker diarization: A review of recent research," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 2, pp. 356–370, 2012.