INCORPORATING UNCERTAINTY FROM SPEAKER EMBEDDING ESTIMATION TO SPEAKER VERIFICATION

Qiongqiong Wang, Kong Aik Lee, Tianchi Liu

Institute for Infocomm Research (I^2R), $A^\star$STAR, Singapore

ABSTRACT

Speech utterances recorded under differing conditions exhibit varying degrees of confidence in their embedding estimates, i.e., uncertainty, even if they are extracted using the same neural network. This paper aims to incorporate the uncertainty estimate produced in the xi-vector network front-end with a probabilistic linear discriminant analysis (PLDA) back-end scoring for speaker verification. To achieve this we derive a posterior covariance matrix, which measures the uncertainty, from the frame-wise precisions to the embedding space. We propose a log-likelihood ratio function for the PLDA scoring with the uncertainty propagation. We also propose to replace the length normalization pre-processing technique with a length scaling technique for the application of uncertainty propagation in the back-end. Experimental results on the VoxCeleb-1, SITW test sets as well as a domain-mismatched CNCeleb1-E set show the effectiveness of the proposed techniques with 14.5%–41.3% EER reductions and 4.6%–25.3% minDCF reductions.

***Index Terms*—** speaker embeddings, PLDA, speaker verification, uncertainty, xi-vector

1. INTRODUCTION

Automatic speaker verification (ASV) is the process to verify whether a given speech utterance is from a specific speaker or not. Recent ASV systems have benefited from deep learning by replacing individual components [1, 2, 3], as well as the entire pipeline in an end-to-end manner [4, 5]. Among these, it has been shown to be the most viable and effective to use DNNs in the front-end for discriminative speaker embedding extractions. Speaker embeddings are fixed-length continuous-value representations of input sequences that contain acoustic feature vectors living in large, complex spaces. Using it simplifies ASV task because in the embedding space, probability distributions and geometric concepts are easily applicable. Therefore, speaker embeddings and a scoring back-end are often used together in ASV frameworks [6, 7, 8, 9, 10].

Substantial amount of works have reported on network architectures that produce embedding vectors with better speaker representations [1, 11, 12, 13]. Unlike the generative i-vector extraction model in which a posterior mean and precision are calculated simultaneously [14], deep speaker embedding networks mostly only produce a points estimate without a guarantee or a measure of its precision. Other factors not related to a speaker's voice, however, do affect the estimation of the embedding vector representing the speaker [15]. As we know, speech utterances exhibit both extrinsic variability due to the background noises, channel distortions, as well as intrinsic variability including the physiological nature of the vocal apparatus and psychological states. It also has been pointed out that shorter speech utterances produce less reliable embeddings and result in poor ASV performance [16]. Therefore, it is essential to measure the precision, or equivalently, the uncertainty of the embeddings.

Few work has been done about the uncertainty of deep speaker embeddings. For speaker diarization for which long recordings are typically cut into very short segments for clustering, a neural network has been used to predict the uncertainty of an x-vector [17], using the output of the statistics pooling layer of the original x-vector extractor. Then they are both input to an agglomerative hierarchical clustering (AHC) algorithm. Xi-vector network [12] with a posterior inference pooling has been proposed to integrate the Bayesian formulation of a linear Gaussian model to speaker-embedding neural networks. The precisions, however, are only used in the calculation for posterior mean vectors from which xi-vectors are obtained and discarded afterwards.

Probabilistic linear discriminant analysis (PLDA) is a promising back-end for deep speaker embeddings [18]. The ability to handle uncertainty has been the cornerstone in the successful use of PLDA generative models [14, 19, 20]. Thus, it is natural to propagate the uncertainty seen in the front-end to PLDA. In this paper, we aim to incorporate the uncertainty that are calculated along the xi-vector extraction into the back-end PLDA scoring. Our contributions are: i) we derive the formulation for the PLDA scoring with uncertainty propagation and ii) propose to use a length scaling technique to replace the length normalization to enable its application to an uncertainty propagated back-end.

The paper is organized as follows. Section 2 presents the PLDA with uncertainty propagation. Section 3 derives the uncertainty in xi-vector space and presents a length scaling technique. Section 4 describes our experimental setup, results, and analyses. Section 5 summarizes our work.

2. SPEAKER VERIFICATION SCORING WITH UNCERTAINTY

Speaker verification can be accomplished by calculating the similarity between the two speaker embeddings corresponding to an enrollment and a test utterance. PLDA is a supervised parametric scoring method which is widely used in speaker recognition [21, 22]. The propagation of speaker embeddings' uncertainty in PLDA has been addressed previously. I-vector posterior uncertainty, which is defined by the generative i-vector extraction model, is propagated to PLDA for the quality effect analysis caused by duration and phonetic variability [23, 24]. In [17] probabilistic embeddings, which consist of x-vectors and precisions, have been proposed to work with PLDA . The explicit form of the PLDA scoring log-likelihood ratio function, however, has not been presented.

2.1. Uncertainty propagation in PLDA scoring (UP-PLDA)

Let ϕ_r be a D-dimensional speaker embedding vector of a speech utterance r with a precision $\mathbf{\Phi}_{\mathrm{U},r}^{-1}$, which measures the uncertainty

in embedding extraction process. We assume that the vector ϕ is generated from a linear Gaussian model [25]

$$\phi_r = \mu + \mathbf{F}\mathbf{y} + \mathbf{U}_r\mathbf{x} + \sigma \tag{1}$$

with two latent variables: $\mathbf{y} \in \mathbb{R}^{d_y}$ and as the speaker variable

$$\mathbf{y} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \tag{2}$$

and $\mathbf{x} \in \mathbb{R}^{d_x}$ as the variable to model the statistical noise in embedding extraction process

$$\mathbf{x} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \tag{3}$$

The vector $\mu \in \mathbb{R}^D$ represents the global mean. The matrices $\mathbf{F} \in \mathbb{R}^{D\times d}$ is the speaker loading matrix, $\mathbf{U}_r$ is the lower-triangular Cholesky decomposition of the covariance $\mathbf{\Phi}_{\mathrm{U},r}$, which represents the uncertainty of the vector ϕ_r, and σ models the residual variances

$$\sigma \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}) \tag{4}$$

Integrating out the latent variables, we arrive at the following marginal density

$$p(\phi_r) = \mathcal{N}\left(\phi|\mu, \mathbf{\Phi}_{\mathrm{B}} + \mathbf{\Phi}_{\mathrm{W},r}\right) \tag{5}$$

where $\mathbf{\Phi}_{\mathrm{B}}$ is the between-speaker covariance matrix

$$\mathbf{\Phi}_{\mathrm{B}} = \mathbf{F}\mathbf{F}^{\mathsf{T}} \tag{6}$$

and $\mathbf{\Phi}_{\mathrm{W},r}$ is an utterance-dependent within-speaker covariance matrix

$$\mathbf{\Phi}_{\mathrm{W},r} = \mathbf{\Sigma} + \mathbf{\Phi}_{\mathrm{U},r} \tag{7}$$

The total covariance correspondent to the utterance r is the summation of the two covariance matrices

$$\mathbf{\Phi}_{\mathrm{tot},r} = \mathbf{\Phi}_{\mathrm{B}} + \mathbf{\Phi}_{\mathrm{W},r} \tag{8}$$

and thus, is also utterance dependent.

In the testing phase, the log-likelihood ratio (LLR) between the enrollment (ϕ_{e}) and test (ϕ_{t}) embeddings is used to score how likely they are from the same speaker

$$s(\phi_{\mathrm{e}}, \phi_{\mathrm{t}}) = \log \frac{p(\phi_{\mathrm{e}}, \phi_{\mathrm{t}})}{p(\phi_{\mathrm{e}})p(\phi_{\mathrm{t}})} = \log \frac{\mathcal{N}(\phi_{\mathrm{t}}|\mathbf{M}_{\mathrm{cond}}, \mathbf{\Sigma}_{\mathrm{cond}})}{\mathcal{N}(\phi_{\mathrm{t}}|\mathbf{M}_0, \mathbf{\Sigma}_0)} \tag{9}$$

The probability density functions (pdf) are assumed to be conditioned on the model parameters and on ϕ_{e}. The predictive distribution in the numerator evaluated at ϕ_{t} with mean and covariance equal to

$$\begin{aligned} \mathbf{M}_{\mathrm{cond}} &= \mu + \mathbf{F}\mathbf{M}_{\mathrm{e}} \\ \mathbf{\Sigma}_{\mathrm{cond}} &= \mathbf{F}\mathbf{L}_{\mathrm{e}}^{-1}\mathbf{F}^{\mathsf{T}} + \mathbf{\Phi}_{\mathrm{W,t}} \end{aligned} \tag{10}$$

which consists of estimating speaker vector posterior expectation $\mathbf{M}_{\mathrm{e}}$ and precision $\mathbf{L}_{\mathrm{e}}$

$$\begin{aligned} \mathbf{M}_{\mathrm{e}} &= \mathbf{L}_{\mathrm{e}}^{-1}\mathbf{F}^{\mathsf{T}}\mathbf{\Phi}_{\mathrm{W,e}}^{-1}\phi_{\mathrm{e}} \\ \mathbf{L}_{\mathrm{e}} &= \mathbf{I} + \mathbf{F}^{\mathsf{T}}\mathbf{\Phi}_{\mathrm{W,e}}^{-1}\mathbf{F} \end{aligned} \tag{11}$$

With the relations shown in (6-7) and Woodbury identity for inversion of matrices [26], we use (11) in (10) and obtain another form using the two covariance matrices

$$\begin{aligned} \mathbf{M}_{\mathrm{cond}} &= \mu + \mathbf{\Phi}_{\mathrm{B}}\mathbf{\Phi}_{\mathrm{tot,e}}\phi_{\mathrm{e}} \\ \mathbf{\Sigma}_{\mathrm{cond}} &= \mathbf{\Phi}_{\mathrm{tot,t}} - \mathbf{\Phi}_{\mathrm{B}}\mathbf{\Phi}_{\mathrm{tot,e}}^{-1}\mathbf{\Phi}_{\mathrm{B}} \end{aligned} \tag{12}$$

The denominator is also a normal pdf evaluated at ϕ_{t} with parameters

$$\begin{aligned} \mathbf{M}_0 &= \mu \\ \mathbf{\Sigma}_0 &= \mathbf{\Phi}_{\mathrm{tot,t}} \end{aligned} \tag{13}$$

The posterior mean and covariance are set to their priors.

When uncertainty of speaker embeddings is not considered, i.e. the covariance $\mathbf{\Phi}_{\mathrm{U},r}$ is assumed to be zero, (1) becomes the traditional PLDA, and all embeddings share the same between-, within-, and total-speaker covariance matrices. In the log-likelihood function (9), the parameters of the predictive distribution in the numerator at ϕ_t are

$$\begin{aligned} \mathbf{M}_{\mathrm{cond}} &= \mu + \mathbf{\Phi}_{\mathrm{B}}\mathbf{\Phi}_{\mathrm{tot}}\phi_{\mathrm{e}} \\ \mathbf{\Sigma}_{\mathrm{cond}} &= \mathbf{\Phi}_{\mathrm{tot}} - \mathbf{\Phi}_{\mathrm{B}}\mathbf{\Phi}_{\mathrm{tot}}^{-1}\mathbf{\Phi}_{\mathrm{B}} \end{aligned} \tag{14}$$

and in the denominator $\mathbf{M}_0 = \mu$ and $\mathbf{\Sigma}_0 = \mathbf{\Phi}_{\mathrm{tot}}$.

Therefore, PLDA scoring LLR function with or without the uncertainty propagation has the same form. The difference is that with the uncertainty propagation, LLR has the within-speaker covariance that is dependent on the individual recording. They are adapted with an increase of uncertainty of the enrollment or test xi-vector estimate as shown in (12-13).

3. DERIVE EMBEDDING UNCERTAINTY FROM NETWORK

3.1. Posterior inference in speaker embedding networks

Xi-vector [12] was proposed to include a posterior inference pooling that integrates the Bayesian formulation of linear Gaussian model to speaker-embedding neural networks. It assumes that a linear Gaussian model is responsible for generating representations and characterizes the frame uncertainty with a covariance matrix associated with each estimate

$$\mathbf{z}_t = \mathbf{h} + \epsilon_t \tag{15}$$

where $\mathbf{h}$ is a latent speaker variable with the prior mean vector μ_p and covariance matrix $\mathbf{L}_p^{-1}$

$$\mathbf{h} \sim \mathcal{N}(\mu_{\mathrm{p}}, \mathbf{L}_{\mathrm{p}}^{-1}) \tag{16}$$

and ϵ_t represents uncertainty for the frame t

$$\epsilon_t \sim \mathcal{N}(\mathbf{0}, \mathbf{L}_t^{-1}) \tag{17}$$

Given the input sequence and uncertainty estimate, posterior distribution of latent variable $\mathbf{h}$ is also Gaussian

$$p(\mathbf{h}|\mathbf{z}_1, ..., \mathbf{z}_T, \mathbf{L}_1^{-1}, ...\mathbf{L}_T^{-1}) = \mathcal{N}(\mathbf{h}|\phi_{\mathrm{s}}, \mathbf{L}_{\mathrm{s}}^{-1}) \tag{18}$$

with a posterior mean ϕ_{s}

$$\phi_{\mathrm{s}} = \mathbf{L}_{\mathrm{s}}^{-1}(\sum_{t=1}^{T}\mathbf{L}_t\mathbf{z}_t + \mathbf{L}_{\mathrm{p}}\mathbf{z}_{\mathrm{p}}) \tag{19}$$

and a precision matrix $\mathbf{L}_{\mathrm{s}}$

$$\mathbf{L}_{\mathrm{s}} = \sum_{t=1}^{T}\mathbf{L}_t + \mathbf{L}_{\mathrm{p}} \tag{20}$$

Let $\mathbf{A}_t = \mathbf{L}_{\mathrm{s}}^{-1}\mathbf{L}_t$ for $t = 0, 1, ...T$, and the index $t = 0$ represents the prior such that $\mathbf{z}_0 = \mu_{\mathrm{p}}$ and $\mathbf{L}_0 = \mathbf{L}_{\mathrm{p}}$, then the posteriors can be written

$$\phi_{\mathrm{s}} = \sum_{t=1}^{T}\mathbf{A}_t\mathbf{z}_t \qquad\qquad \mathbf{L}_{\mathrm{s}} = \sum_{t=1}^{T}\mathbf{L}_t \tag{21}$$

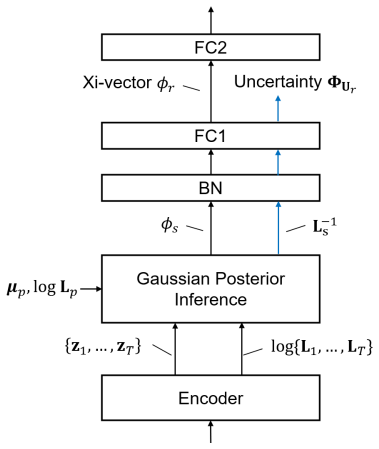

Fig. 1. *xi-vector network. Blue arrows are only for testing phase*

After the posterior mean ϕ_s is obtained in the Gaussian posterior inference layer (see Fig 1), it is followed by a batch normalization layer (BN) and a fully connected layer (FC1) from which xi-vectors are extracted.

3.2. Derive xi-vector uncertainty

The posterior precision $\mathbf{L}_s$ can be considered a measure of the uncertainty associated with the point estimate of latent variable $\mathbf{h}$. It is used in the mean estimation in (19) so as to extract xi-vector indirectly. It is, however, possible to propagate it to the space of xi-vector to represent its uncertainty associated with the embedding estimator besides a point estimate. In this way, the embedding is extended to be a distribution with the mean and variance $(\phi_r, \mathbf{\Phi}_{\mathrm{U},r})$. To obtain this, we propagate the posterior covariance $\mathbf{L}_s^{-1}$ through the same layers accordingly (see Fig 1). The details are shown in Algorithm 1.

3.3. Embedding length scaling (LS)

Length normalization (LN) together with whitening is a popular pre-processing technique to reinforce speaker embeddings to be more Gaussian distributed [27]. It would be hard, however, to apply it to posterior covariance matrices because of its non-linearity. Therefore, we refine LN into an embedding length scaling (LS) technique

$$\phi_{\mathrm{LS}} = \left(\frac{d}{\phi^\mathsf{T} \mathbf{\Sigma} \phi} \right)^{\frac{1}{2}} \phi \tag{22}$$

With the multiplication with $d^{\frac{1}{2}}$ in the numerator to preserve the scaling of the original embedding space, we only need to apply LS to the enrollment and test embeddings in evaluations, not to PLDA training embeddings. Note that when a total covariance matrix from the training data is used as $\mathbf{\Sigma}$, LS would be equivalent to LN but without retraining PLDA models. When the uncertainty of the embeddings is available, it can be propagated to LS (UP-LS) as well by including the embedding posterior covariance

$$\mathbf{\Sigma} = \mathbf{\Phi}_{\mathrm{tot},r} \tag{23}$$

Algorithm 1: Derive xi-vector uncertainty from a network

Input Testing speech acoustic features $A = \{a_1, ..., a_T\}$ and xi-vector network parameters
Output xi-vector point estimate ϕ_r and uncertainty covariance $\mathbf{\Phi}_{\mathrm{U},r}$
Obtain frame-wise point estimate and its uncertainty
$\{\mathbf{z}_1, ..., \mathbf{z}_T, \mathbf{L}_1^{-1}, ...\mathbf{L}_T^{-1}\} = f_{\mathrm{enc}}(a_1, ..., a_T)$
Obtain posterior mean vector and precision
$\{\phi_\mathrm{s}, \mathbf{L}_\mathrm{s}\} = f_{\mathrm{pool}}(\mathbf{z}_1, ..., \mathbf{z}_T, \mathbf{L}_1^{-1}, ...\mathbf{L}_T^{-1})$
Calculate xi-vector point estimate and posterior using BN layer parameters $(\mu_{\mathrm{BN}}, \sigma_{\mathrm{BN}})$ and FC1 layer parameters $(\mathbf{W}, b)$
$\phi_r = \mathbf{W}((\phi_\mathrm{s} - \mu_{\mathrm{BN}}) \circ q_{\mathrm{BN}}) + b$
$\mathbf{\Phi}_{\mathrm{U},r} = \mathbf{W}({q_{\mathrm{BN}}}^\mathsf{T} \mathbf{L}_\mathrm{s}^{-1} q_{\mathrm{BN}}) \mathbf{W}^\mathsf{T}$
where $q_{\mathrm{BN},i} = 1/\sigma_{\mathrm{BN},i}$, $\circ$ is element-wise multiplication.

4. EXPERIMENTS

4.1. Experimental settings

The experiments were conducted on the VoxCeleb [28], the Speaker in the Wild (SITW) core-core *eval* [29], and the CNCeleb1 dataset [30]. For VoxCeleb1, we exploited the original test set Vox1-O and the hard test set Vox1-H. The front-end networks were trained using the original segments of VoxCeleb2 dataset [31] with augmentations following the settings in [18]. The same training dataset without augmentation was used to train back-ends after all VoxCeleb2 segments belonging to the same session were concatenated. For SITW and CNCeleb evaluations, SITW core-core *dev* set and CNCeleb1-T dataset were used, respectively, as the development sets for the mean normalization.

We used x-vector and xi-vector networks with an ECAPA-TDNN [11] backbone and optimized them with AAM-Softmax cross-entropy loss [32]. We used their standard forms. In the x-vector network, both weighted means and standard deviations from an attentive statistics pooling layer were propagated, while in xi-vector network only the posterior mean from the Gaussian posterior inference layer was propagated. Both types of embeddings have 192 dimensions. The posterior covariance matrices in the xi-vector network were assumed diagonal [12], and the prior mean and covariance were initialized as $(0, I)$. In the testing phase , we obtained the corresponding posterior covariance following Algorithm 1.

For the back-end, the advantage of using the diagonalized within-speaker covariance matrix in PLDA has been proved [18]. We further diagonalized the between-speaker covariance matrix for the computational efficiency. We used the total covariance of the training data in LS for PLDA back-ends and that plus testing utterance's posterior covariance in UP-LS for UP-PLDA back-end. Raw embeddings were used in PLDA training in the systems where LS was applied to testing data. We used the SpeechBrain open-source toolkit [33] for the front-end implementations and embedding extractions. The input of the neural networks were 80-dimensional filter-bank features. Results are reported in terms of equal error rate (EER) and the minimum normalized detection cost function (MinDCF) at $P_{\mathrm{target}} = 10^{-2}$ and $C_{\mathrm{FA}} = C_{\mathrm{Miss}} = 1$.

4.2. Experimental results

We first compare LS and LN pre-processing techniques in both x-vector and xi-vector PLDA systems. As shown in Tab. 1, the uses of LN and LS give almost the same results, and they are consistently

Table 1. Performance of two ASV systems: ECAPA-TDNN x-vector and xi-vectors with PLDA on the four test sets: Vox1-O, Vox1-H, SITW core-core eval, and CNCeleb1-E. Two pre-processing techniques LN and LS are investigated. Results are shown as EER(%)/minDCF

	Vox1-O			Vox1-H			SITW			CNCeleb		
	-	LN	LS	-	LN	LS	-	LN	LS	-	LN	LS
x	1.44/0.184	1.23/0.173	1.23/0.173	2.68/0.248	2.46/0.235	2.46/0.236	1.80/0.180	1.67/0.172	1.69/0.172	17.38/0.675	13.04/0.625	13.00/0.626
xi	1.49/0.166	1.23/0.143	1.23/0.141	2.76/0.247	2.44/0.236	2.43/0.235	1.86/0.175	1.65/0.170	1.65/0.170	17.32/0.675	12.64/0.629	12.63/0.630

Table 2. Performance of xi-vectors with PLDA and UP-PLDA. LS and UP-LS pre-processing techniques are evaluated. SITW* and CNCeleb* are the evaluations when mean adaptation is not used in centralization. Results are shown as EER/minCprimary

Back-end	Vox1-O	Vox1-H	SITW	CNCeleb	SITW*	CNCeleb*
PLDA	1.49/0.166	2.76/0.247	1.86/0.175	17.32/0.675	1.91/0.178	17.99/0.689
LS PLDA	1.23/0.141	2.43/0.235	1.65/0.170	12.63/0.630	1.77/0.168	15.04/0.676
UP-PLDA	**0.99**/0.129	**2.13**/0.231	1.77/**0.167**	10.81/0.652	1.80/0.170	13.18/0.714
UP-LS UP-PLDA	1.01/**0.124**	2.18/**0.224**	**1.59/0.167**	**10.16/0.608**	**1.59/0.166**	**11.57/0.686**

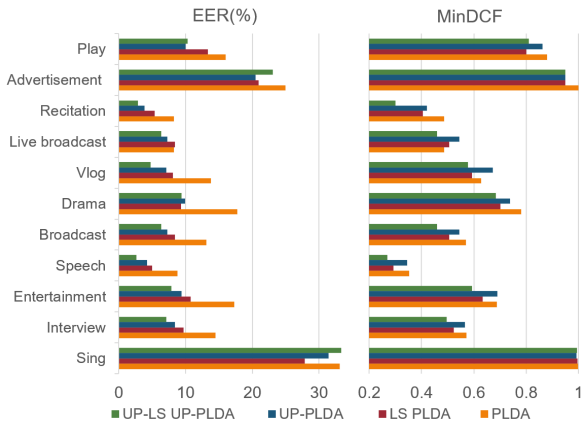

Fig. 2. *Evaluation of length scaling (LS) and uncertainty propagation (UP) in different genres in CNCeleb1-E set*

better than those without any pre-processing. It proves the equivalency between LN and LS when using the total covariance matrix as mentioned in 3.3. Thus, we next apply LS in the back-end of PLDA with uncertainty. The advantage of xi-vectors is not obvious over x-vectors. We argue that features with different time-scales derived from the dense connection in ECAPA-TDNN may cause confusion to the xi-vector posterior inference process.

Next we evaluate the uncertainty propagation in LS pre-processing and in PLDA, referred as UP-LS and UP-PLDA, respectively. Table 2 shows a significant improvement due to the explicit use of xi-vector uncertainty in UP-PLDA scoring (line 3) over the system without it (line 1). Such observations are consistent in all four evaluation sets. It indicates that the explicit use of uncertainty in the back-end is more effective than its implicit use in xi-vector estimation. For the Vox1 and SITW evaluations, UP-PLDA yields a greater improvement in both EER and minDCF than that LS does, while for CNCeleb evaluation, UP-PLDA gives a better EER but a slight increase in minDCF. Further application of UP-LS pre-processing to xi-vectors, for UP-PLDA back-end, improved ASV performance in the SITW and CNCeleb evaluation sets in which certain mismatches exist in domains. Overall, the use of LS pre-processing and uncertainty propagation in PLDA together achieved the best performance, with 14.5%–41.3% and 4.6%–25.3% reductions, respectively, in EER and minDCF. For the SITW and CNCeleb sets, we also show in Tab 2 the performance using the mean of the training data for the embedding centralization. The comparison shows a clear advantage of using the mean adaptation in domain mismatched evaluations.

The CNCeleb evaluations give large values in EER and minDCF in all the systems (see Tab 1 and Tab 2) due to its severe conditions. We next investigate the UP-PLDA and LS effects in each genre, as shown in Fig 2. Despite the difference in language and speaking styles between CNCeleb set and the training data VoxCeleb, the observations of the improvement due to the use of UP-PLDA and LS pre-processing, in most of the genres, are consistent with the overall results in Tab 2.

5. SUMMARY

This paper has revisited uncertainty propagation in PLDA. Based on the xi-vector framework, we derive a posterior covariance matrix from frame-wise precisions, to measures the uncertainty of speaker embeddings. We propose a log-likelihood ratio function for the PLDA scoring with the propagation of embedding uncertainty. At last, we propose to replace the length normalization pre-processing technique with a length scaling technique for the application of uncertainty propagation in the back-end. Experimental results on the VoxCeleb-1, SITW core-core eval sets as well as the domain-mismatched CNCeleb1 set show the effectiveness of the two techniques with 14.5%–41.3% and 4.6%–25.3% reductions, respectively, in EER and minDCF. In future, we will investigate the networks with different backbones and the use of uncertainty in domain adaptation.

6. ACKNOWLEDGEMENTS

This project is supported by the Agency for Science, Technology and Research (A*STAR), Singapore, through its Council Research Fund (Project No. CR-2021-005).

7. REFERENCES

[1] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust DNN embeddings for speaker recognition," in *Proc. IEEE ICASSP*, 2018, pp. 5329–5333.

[2] Y. Tang, G. Ding, J. Huang, X. He, and B. Zhou, "Deep speaker embedding learning with multi-level pooling for text-independent speaker verification," in *Proc. IEEE ICASSP*, 2019, pp. 6116–6120.

[3] J. Chien and C. Hsu, "Variational manifold learning for speaker recognition," in *Proc. IEEE ICASSP*, 2017, pp. 4935–4939.

[4] C. Li, X. Ma, B. Jiang, X. Li, X. Zhang, X. Liu, Y. Cao, A. Kannan, and Z. Zhu, "Deep speaker: an end-to-end neural speaker embedding system," in *arXiv:1705.02304*, 2017.

[5] J. Rohdin, A. Silnova, M. Diez, O. Plchot, P. Matejka, L. Burget, and O. Glembek, "End-to-end DNN based text-independent speaker recognition for long and short utterances," in *Computer Speech & Language*, 2020, pp. 22–35.

[6] K. A. Lee, V. Hautamaki, T. Kinnunen, H. Yamamoto, K. Okabe, et al., "I4U submission to NIST SRE 2018: Leveraging from a decade of shared experiences," in *Proc. Interspeech*, 2019, pp. 1497–1501.

[7] P. Matejka, O. Plchot, O. Glembek, L. Burget, J. Rohdin, et al., "13 years of speaker recognition research at BUT, with longitudinal analysis of NIST SRE," in *Computer Speech & Language*, 2020, vol. 63, p. 101035.

[8] J. Villalba, N. Chen, D. Snyder, D. Garcia-Romero amd A. McCree, et al., "State-of-the-art speaker recognition with neural network embeddings in NIST SRE18 and Speakers in the Wild evaluations," in *Computer Speech & Language*, 2020, vol. 60, p. 101026.

[9] A. Nagrani, J. S. Chung, W. Xie, and A. Zisserman, "Voxceleb: Largescale speaker verification in the wild," in *Computer Speech & Language*, 2020, vol. 60, p. 101027.

[10] K. A. Lee, H. Yamamoto, K. Okabe, Q. Wang, L. Guo, et al., "NEC-TT system for mixed-bandwidth and multi-domain speaker recognition," in *Computer Speech & Language*, 2020, vol. 61, p. 101033.

[11] B. Desplanques, J. Thienpondt, and K. Demuynck, "ECAPA-TDNN: Emphasized channel attention, propagation and aggregation in TDNN based speaker verification," in *Proc. Interspeech*, 2020, pp. 3830–3834.

[12] K. A. Lee, Q. Wang, and T. Koshinaka, "Xi-vector embedding for speaker recognition," in *IEEE Signal Processing Letters*, 2021, pp. 1385–1389.

[13] T. Liu, R. K. Das, K. A. Lee, and H. Li, "MFA: TDNN with multi-scale frequency-channel attention for text-independent speaker verification with short utterances," in *Proc. IEEE ICASSP*, 2022, pp. 7517–7521.

[14] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," in *Proc. IEEE ICASSP*, 2011, pp. 788–798.

[15] K. Takahashi and T. Murakami, "A measure of information gained through biometric systems," in *Image and Vision Computing*, 2014, vol. 32, pp. 1194–1203.

[16] H. Zeinali, K. A. Lee, J. Alam, and L. Burget, "SdSV challenge 2020: Large-scale evaluation of short-duration speaker verification," in *Proc. Interspeech*, 2020, pp. 731–735.

[17] A. Silnova, N. Brummer, J. Rohdin, T. Stafylakis, and Lukas Burget, "Probabilistic embeddings for speaker diarization," in *Proc. Odyssey*, 2020, pp. 24–31.

[18] Q. Wang, K. A. Lee, and T. Liu, "Scoring of large-margin embeddings for speaker verification: Cosine or PLDA?," in *Proc. Interspeech*, 2022, pp. 600–604.

[19] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted gaussian mixture models," in *Digital Signal Processing*, 2000, pp. 19–41.

[20] P. Kenny, P. Ouellet, N. Dehak, V. Gupta, and P. Dumouchel, "A study of inter-speaker variability in speaker verification," in *IEEE Trans. Audio, Speech and Language Processing*, 2008.

[21] P. Kenny, "Bayesian speaker verification with heavy-tailed priors," in *Proc. Odyssey*, 2010, pp. 14–21.

[22] K. A. Lee, Q. Wang, and T. Koshinaka, "The CORAL+ algorithm for unsupervised domain adaptation of PLDA," in *Proc. IEEE ICASSP*, 2019, pp. 5821–5825.

[23] P. Kenny, T. Stafylakis, P. Ouellet, M. J. Alam, and P. Dumouchel, "PLDA for speaker verification with utterances of arbitrary duration," in *IEEE ICASSP*, 2013, pp. 7649–7653.

[24] S. Cumani, O. Plchot, and P. Laface, "On the use of i–vector posterior distributions in probabilistic linear discriminant analysis," in *IEEE Trans. Audio, Speech and Language Processing*, 2014, pp. 846–857.

[25] C. Bishop, *Pattern recognition and machine learning*, Springer, 2006.

[26] G. H. Golub and C. F. Van Loan, "Matrix computations," in *The Johns Hopkins University Press*, 1996.

[27] D. Garcia-Romero and C. Espy-Wilson, "Analysis of i-vector length normalization in speaker recognition systems," in *Proc. Interspeech*, 2011, pp. 249–252.

[28] A. Nagrani, J. Chung, and A. Zisserman, "VoxCeleb: a large-scale speaker identification dataset," in *Proc. Interspeech*, 2017, pp. 2616–2620.

[29] M. McLaren, L. Ferrer, D. Castan, and A. Lawson, "The speakers in the wild (SITW) speaker recognition database," in *Proc. Interspeech*, 2016, pp. 818–822.

[30] Y. Fan, J. Kang, L. Li, K. Li, H. Chen, S. Cheng, P. Zhang, Z. Zhou, Y. Cai, and D. Wang, "CN-CELEB: a challenging chinese speaker recognition dataset," in *IEEE ICASSP*, 2020, pp. 7604–7608.

[31] J. Chung, A. Nagrani, and A. Zisserman, "VoxCeleb2: Deep speaker recognition," in *Proc. Interspeech*, 2018, pp. 1086–1090.

[32] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 4690–4699.

[33] M. Ravanelli, T. Parcollet, P. Plantinga, A. Rouhe, S. Cornell, et al., "SpeechBrain: A general-purpose speech toolkit," in *arXiv:2106.04624*, 2021.