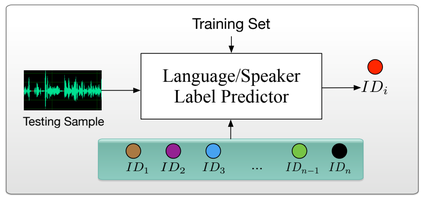
Analysis of Length Normalization in End-to-End Speaker Verification System

Weicheng Cai[2], *Jinkun Chen*[2] and *Ming Li*[1]

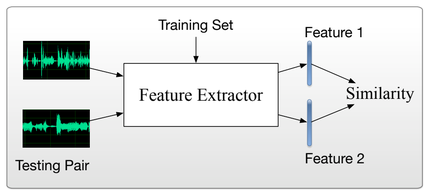
1. Data Science Research Center, Duke Kunshan University, Kunshan, China
2. School of Electronics and Information Technology, Sun Yat-sen University, Guangzhou, China

ming.li369@dukekunshan.edu.cn

Introduction



(a) Closed-set identification



(b) Open-set verification

Open-set Speaker Verification:
(determines whether a pair of utterances belongs to the same person)

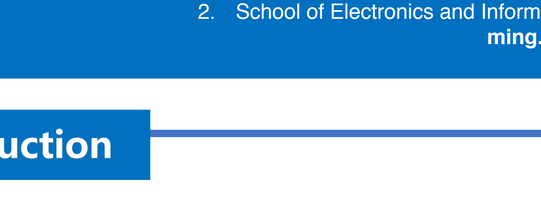
- Speaker identities in testing set are usually disjoint from the ones in training set, which makes the speaker verification more challenging yet closer to practice.
- Since it is impossible to classify testing utterances to known identities in training set, we need to map speakers to a discriminative feature space.
- In this scenario, open-set speaker verification is essentially a metric learning problem, where the key is to learn discriminative large-margin features.

Traditional Length Normalization

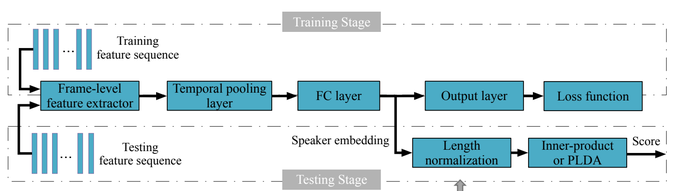
Length normalization on i-vector has been the de facto standard before back-end modeling.

For open-set SV task, cosine similarity or length normalization followed by probabilistic linear discriminant analysis (PLDA) scoring modeling is widely used to get the final pairwise scores.

The cosine similarity is a similarity measure which is independent of magnitude, it can be seen as the length-normalized version of inner-product of two vectors..



Length normalization in classical i-vector approach



Once speaker embeddings (such as x-vectors) are extracted, just the same as in i-vector approach, cosine similarity or length normalization followed by PLDA is commonly adopted to get the final pairwise scores.

Length normalization in typical speaker embedding approach

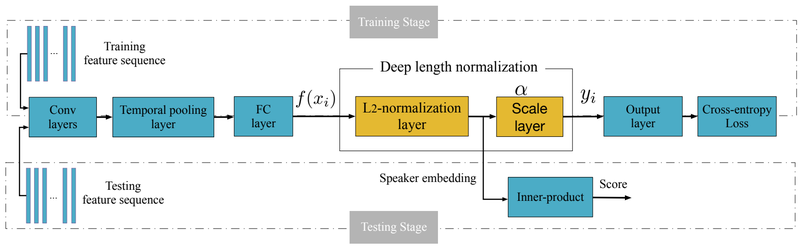
Deep Length Normalization

Motivation:

Is it possible to learn the deep speaker embeddings being length-normalized in an end-to-end manner within common classification network ?

We add a length normalization layer followed by a scale layer before the output layer of the common classification network.

$$\boldsymbol{y}_i = \alpha \times \frac{f(\boldsymbol{x_i})}{\|f(\boldsymbol{x_i})\|_2}$$



- In total, only a single scalar parameter α is introduced, and it can be inherently trained with other components of the network together.
- This scale parameter α has a crucial impact on the performance since it determines the radius of the length-normalized hyperspace.
- The network could have stronger $L2$-constraint on the small radius hyperspace with smaller α, but faces the risk of not convergent.

Experimental Results and Discussion

Network Setup

- The model is trained with a mini-batch size of 128, using typical stochastic gradient descent with momentum 0.9 and weight decay 1e-4.
- The learning rate is set to 0.1, 0.01, 0.001 and is switched when the training loss plateaus.
- For each training step, an integer L within [300,800] interval is randomly generated, and each data in the mini-batch is cropped or extended to L frames.
- After model training finished, the 128-dimensional speaker embeddings are extracted after the penultimate layer of neural network.

Table 1: *Baseline end-to-end system architecture*

Layer	Output size	Downsample	Channels	Blocks
Conv1	$64 \times L$	False	16	-
Res1	$64 \times L$	False	16	3
Res2	$32 \times \frac{L}{2}$	True	32	4
Res3	$16 \times \frac{L}{4}$	True	64	6
Res4	$8 \times \frac{L}{8}$	True	128	3
Average pool	128	-	-	-
FC (embedding)	128	-	-	-
Output	speaker categories	-	-	-

Table 3: *Verification performance on VoxCeleb1 for various scale parameter α (lower is better)*

System Description	minDCF10^{-2}	minDCF10^{-3}	$EER(\%)$
Deep embedding baseline	0.553	0.713	5.48
fixed $\alpha = 1$	0.922	0.967	10.18
fixed $\alpha = 4$	0.601	0.828	6.36
fixed $\alpha = 8$	0.515	0.687	5.49
fixed $\alpha = 12$	**0.475**	**0.586**	**5.01**
fixed $\alpha = 16$	0.499	0.596	5.32
fixed $\alpha = 20$	0.503	0.637	5.46
fixed $\alpha = 24$	0.502	0.638	5.54
fixed $\alpha = 28$	0.497	0.640	5.52
trained $\alpha = 26.1$	0.486	0.599	5.60

Experiments on different α

- We first investigate the setting of scale parameter α. For those systems in Table 3 and Fig. 4, the cosine similarity or equivalently L2-normalized inner-product is adopted to measure the similarities between speaker embeddings.
- From Fig. 4, we can observe the proposed L2-normalized deep embedding system achieves the best minDCF of 0.475, 0.586 and EER of 5.01%, which outperforms the baseline system significantly.
- The performance is poor when α is too small and stable with α is higher. The best α in our experiment is 12.

(a) Performance in terms of minDCF10^{-2}

(b) Performance in terms of minDCF10^{-3}

Figure 4: *Performance tendency curve considering various α*

Table 2: *Voxceleb1 open-set verification task performance, in comparing the effect of our introduced deep length normalization strategy and traditional extra length normalization step (lower is better)*

System Description	Deep L_2-norm	Traditional L_2-norm	Similarity Metric	minDCF10^{-2}	minDCF10^{-3}	$EER(\%)$
i-vector + inner-product	N/A	✗	inner-product	0.736	0.800	13.80
i-vector + cosine	N/A	✓	inner-product	0.681	0.771	13.80
i-vector + PLDA	N/A	✗	PLDA	0.488	0.639	5.48
i-vector + L_2-norm + PLDA	N/A	✓	PLDA	0.484	0.627	5.41
Deep embedding + inner-product	✗	✗	inner-product	0.758	0.888	7.42
Deep embedding+ cosine	✗	✓	inner-product	0.553	0.713	5.48
Deep embedding+ PLDA	✗	✗	PLDA	0.524	0.739	5.90
Deep embedding + L_2-norm + PLDA	✗	✓	PLDA	0.545	0.733	5.21
L_2-normalized deep embedding + inner-product	✓	✗	inner-product	**0.475**	**0.586**	5.01
L_2-normalized deep embedding + PLDA	✓	✗	PLDA	0.525	0.694	**4.74**

We further compare the effect of deep length normalization strategy and traditional extra length normalization in the whole SV pipeline. The results are shown in Table 2.

- No matter in i-vector or baseline deep speaker embedding systems, extra length normalization step followed by PLDA scoring achieves the best performance.
- When it turns into L2-normalized deep speaker embedding systems, since the speaker embeddings extracted from neural network have already been normalized to unit length, we need no more extra length normalization step.
- In testing stage, a simple inner-product achieves the best performance, even slightly better than the PLDA scoring result. It might be the reason that our L2-normalized speaker embedding is highly optimized, which could incompatible with the objective function introduced by PLDA.