한국음향학회지 제35권 제6호 pp. 501~509 (2016)
The Journal of the Acoustical Society of Korea Vol.35, No.6 (2016)
http://dx.doi.org/10.7776/ASK.2016.35.6.501

pISSN : 1225-4428
eISSN : 2287-3775

짧은 음성을 대상으로 하는 화자 확인을 위한 심층 신경망

Deep neural networks for speaker verification with short speech utterances

양일호, 허희수, 윤성현, 유하진†

(IL-Ho Yang, Hee-Soo Heo, Sung-Hyun Yoon, and Ha-Jin Yu†)

서울시립대학교 컴퓨터과학부

(Received September 1, 2016; accepted November 25, 2016)

초 록: 본 논문에서는 짧은 테스트 발성에 대한 화자 확인 성능을 개선하는 방법을 제안한다. 테스트 발성의 길이가 짧을 경우 i-벡터/확률적 선형판별분석 기반 화자 확인 시스템의 성능이 하락한다. 제안한 방법은 짧은 발성으로부터 추출한 특징 벡터를 심층 신경망으로 변환하여 발성 길이에 따른 변이를 보상한다. 이 때, 학습시의 출력 레이블에 따라 세 종류의 심층 신경망 이용 방법을 제안한다. 각 신경망은 입력 받은 짧은 발성 특징에 대한 출력 결과와 원래의 긴 발성으로부터 추출한 특징과의 차이를 줄이도록 학습한다. NIST (National Institute of Standards Technology, 미국) 2008 SRE(Speaker Recognition Evaluation) 코퍼스의 short 2-10 s 조건 하에서 제안한 방법의 성능을 평가한다. 실험 결과 부류 내 분산 정규화 및 선형 판별 분석을 이용하는 기존 방법에 비해 최소 검출 비용이 감소하는 것을 확인하였다. 또한 짧은 발성 분산 정규화 기반 방법과도 성능을 비교하였다.

핵심용어: 화자 확인, 짧은 테스트 발성, 심층 신경망, 주성분분석, I-벡터, 확률적 선형판별분석, 짧은 발성 분산 정규화

ABSTRACT: We propose a method to improve the robustness of speaker verification on short test utterances. The accuracy of the state-of-the-art i-vector/probabilistic linear discriminant analysis systems can be degraded when testing utterance durations are short. The proposed method compensates for utterance variations of short test feature vectors using deep neural networks. We design three different types of DNN (Deep Neural Network) structures which are trained with different target output vectors. Each DNN is trained to minimize the discrepancy between the feed-forwarded output of a given short utterance feature and its original long utterance feature. We use short 2-10 s condition of the NIST (National Institute of Standards Technology, U.S.) 2008 SRE (Speaker Recognition Evaluation) corpus to evaluate the method. The experimental results show that the proposed method reduces the minimum detection cost relative to the baseline system.

Keywords: Speaker verification, Short test utterance, Deep neural network, Principal component analysis, i-vector, Probabilistic linear discriminant analysis, Short utterance variance normalization

PACS numbers: 43.60.Lq, 43.60.Bf

I. 서 론

화자 확인 분야의 최신 기술인 i-벡터 확률적 선형 판별분석(Probabilistic Linear Discriminant Analysis, PLDA) 시스템[1]은 충분히 긴 학습 및 테스트 발성이 주어졌을 때 높은 정확도를 보인다. 그러나 화자 확인을 적용하는 실제 환경에서는 충분히 긴 테스트 발성을 획득하기 어려울 수 있으며, 이로 인하여 화자 확인 정확도가 급격히 하락할 수 있다. Kanagasundaram *et al.*[2]은 테스트 발성 길이가 줄어듦에 따라 i-벡터 PLDA 시스템의 정확도가 하락함을 보였다.

I-벡터 PLDA 시스템에서 짧은 발성 혹은 발성 길

†**Corresponding author:** Ha-Jin Yu (hjyu@uos.ac.kr)
School of computer Science College of Engineering, University of Seoul 163 Siripdae-ro, Dongdaemun-gu, Seoul 02504, Republic of Korea
(Tel: 82-2-6490-2448, Fax: 82-2-6490-2444)

이 차이로 인한 정확도 하락을 완화하기 위해 다양한 연구들이 이루어졌다. Sarkar *et al.*[3]은 화자 모델 학습 발성과 테스트 발성의 길이가 다른 상황에서 화자 확인 정확도를 높이기 위해 짧은 발성과 긴 발성을 모두 활용하여 i-벡터 기반 시스템의 통계적 파라미터를 학습하는 방법을 제안하였다. Kenny *et al.*[4]은 발성 길이의 변이를 다루기 위해 i-벡터 추출 단계에서 PLDA 적용 단계로 전파되는 불확실성을 정량화하였다. Mandasari *et al.*[5]은 발성 길이를 고려한 음질평가함수를 이용하여 스코어 캘리브레이션을 수행하였다. Kanagasundaram *et al.*[6]은 학습 및 테스트 발성 모두가 짧은 상황에서의 화자 확인 정확도를 개선하기 위해 짧은 발성 분산 정규화(Short Utterance Variance Normalization, SUVN)를 제안하였다.

최근 딥러닝은 기계학습 분야에서 각광받고 있는 기술이다. 이를 화자 확인 시스템에 적용한 사례는 다음과 같다. Variani *et al.*[7]은 기존의 화자 특징인 i-벡터 대신 심층 신경망(Deep Neural Network, DNN)을 이용하여 추출한 d-벡터를 새로운 특징으로 사용하였다. 이 때 DNN은 개발세트상의 각 화자를 식별하도록 학습되었다. Lei *et al.*[8]은 가우시안 혼합 모델(Gaussian Mixture Model, GMM) 대신 DNN 음소-상태 인식기를 이용하여 i-벡터 추출에 필요한 통계치를 계산하였다.

본 연구에서는 짧은 테스트 발성으로 인한 정확도 하락을 방지하기 위해, 원 테스트 발성으로부터 추출한 특징 벡터를 DNN으로 변환하는 방법을 제안한다(단, 학습 발성은 충분히 길다고 간주). 발성 길이가 줄어들수록 화자 확인 정확도가 하락하므로, 반대로 충분히 긴 발성으로부터 추출한 i-벡터 기반 특징을 이상적인 화자 특징인 "긴 발성 특징"이라 가정한다. 이러한 가정 하에 짧은 발성으로부터 추출한 특징인 "짧은 발성 특징"을 긴 발성 특징에 가까워지도록 보상하는 것이 제안한 방법의 핵심 아이디어이다. 제안한 방법은 DNN을 이용하여 입력 받은 기존의 화자 특징 벡터를 동일한 차원의 다른 벡터로 변환한다. 이 때, DNN의 입력은 짧은 발성의 특징 벡터이고, 출력은 보상된 특징 벡터이다. DNN 학습을 위해 개발세트의 모든 발성으로부터 임의의 10초 길이 세그먼트를 취하고, 이들로부터 각각 긴 발성 특징과 짧은 발성 특징을 추출한다. 이렇게 추출한 10초 길이의 짧은 발성을 특징을 DNN에 입력하여 변환한 출력 특징과 실제 긴 발성 특징 간의 차이를 줄이도록 DNN을 학습한다. 이 때, DNN 학습 데이터의 정답 출력 레이블을 달리한 세 종류의 DNN 구조를 제안한다.

제안한 방법을 평가하기 위하여 NIST(National Institute of Standards Technology, U.S.) 2008 SRE(Speaker Recognition Evaluation) 코퍼스[9] 중 short 2-10 s 조건의 남성 화자 발성을 사용한다. 기존의 다른 특징 보상 방법인 부류 내 분산 정규화(Within-Class Covariance Normalization, WCCN) 및 SUVN과 성능을 비교한다.

본 논문의 구성은 다음과 같다. II장에서는 i-벡터/PLDA 화자 확인 시스템에 대해 소개하고, III장에서 제안한 방법을 설명한다. IV장에서 실험 및 결과를 보이고, V장에서 결론을 맺는다.

II. 기존의 화자 확인 시스템

본 연구에서는 베이스라인 성능 평가를 위해 화자 확인 분야의 최신 기술인 i-벡터/확률적 선형판별분석

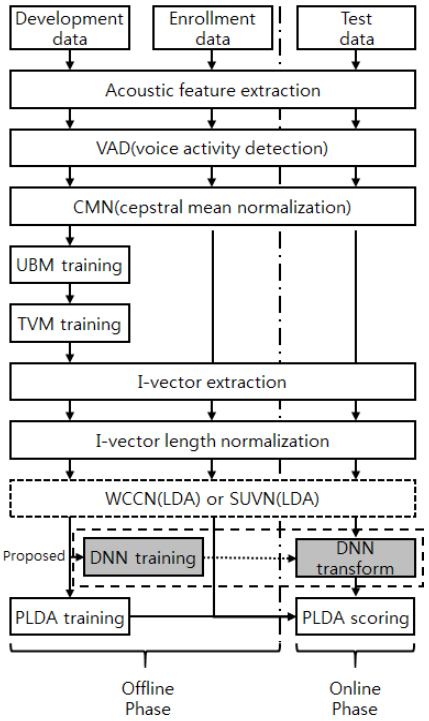

Fig. 1. Overall flow of the baseline and the proposed system.

시스템을 이용한다. I-벡터/PLDA 시스템의 전체 구조는 Fig. 1과 같다.

각 발성은 음향 특징 추출, 켑스트럼 평균 정규화(Cepstral Mean Normalization, CMN), i-벡터 추출, 길이 정규화를 거쳐 하나의 특징 벡터로 표현된다.

I-벡터[10]는 다음과 같이 표현할 수 있다.

$$M = \boldsymbol{m} + T\boldsymbol{w}. \tag{1}$$

M은 화자 및 채널에 종속적인 GMM 수퍼벡터이고, $\boldsymbol{m}$은 화자 및 채널에 독립적인 배경 화자 모델(Universal Background Model, UBM) 수퍼벡터, T는 전체 변이 행렬(Total Variability Matrix, TVM)이라 불리는 낮은 랭크의 요인 부하 행렬이다. $\boldsymbol{w}$는 표준 정규 분포 $N(\mathbf{0}, I)$를 따르는 랜덤 벡터이며, 이를 i-벡터라 부른다.

I-벡터는 화자 정보뿐만 아니라 세션 정보도 함께 포함하고 있으므로, 길이 정규화[10] 이후 세션이나 발성 내용 등에 따른 변이를 보상하는 방법을 널리 사용한다. 이를 위해, 선형 판별 분석(Linear Discriminant Analysis, LDA)으로 길이 정규화된 i-벡터를 차원 축소한 뒤 다음과 같이 WCCN 혹은 SUVN을 적용할 수 있다. 이후 이렇게 보상한 특징들에 대해 PLDA[1]를 적용한다.

배경 화자 모델, TVM, PLDA등은 별도의 개발세트를 이용하여 학습한다.

2.1 WCCN(LDA)

I-벡터 추출 후 길이 정규화, LDA, WCCN[11]이 순차적으로 적용될 수 있다. LDA 변환 행렬을 A라 하면, 부류 내 분산 행렬 W는 다음과 같이 구할 수 있다.

$$W = \frac{1}{N}\sum_{n=1}^{N}\frac{1}{u_n}\sum_{i=1}^{u_n}\left[A^T\left(\boldsymbol{w}_{n,i} - \overline{\boldsymbol{w}_n}\right)\right], \left[A^T\left(\boldsymbol{w}_{n,i} - \overline{\boldsymbol{w}_n}\right)\right]^T, \tag{2}$$

여기서 N은 개발세트의 화자 수, u_n은 n번째 화자의 발성 수, $\boldsymbol{w}_{n,i}$는 n번째 화자의 i번째 발성으로부터 추출한 i-벡터, $\overline{\boldsymbol{w}}_n$은 n번째 화자의 평균 i-벡터이다.

WCCN(LDA) 변환된 i-벡터는 다음과 같이 계산한다.

$$\hat{\boldsymbol{w}}_{WCCN[LDA]} = B^T A^T \boldsymbol{w}, \tag{3}$$

이 때 B는 $BB^T = W^{-1}$에 대한 촐레스키 분해로 구할 수 있다.

2.2 SUVN(LDA)

짧은 발성을 처리하는 경우 WCCN 대신 SUVN[6]을 이용할 수도 있다. Kanagasundaram *et al*.[6]은 학습 및 테스트 발성이 모두 짧은 경우에 대해 SUVN의 성능을 평가하였으나, 본 연구에서는 SUVN이 테스트 발성만 짧은 경우에도 적용 가능할 것이라 기대하였다. 짧은 발성 분산은 다음과 같이 계산한다.

$$S_{SUV(LDA)}^{-1} = \frac{1}{N}\sum_{n=1}^{N}\left[A^T\left(\boldsymbol{w}_n^l - \boldsymbol{w}_n^s\right)\right]\left[A^T\left(\boldsymbol{w}_n^l - \boldsymbol{w}_n^s\right)\right]^T. \tag{4}$$

$\boldsymbol{w}_n^l$과 $\boldsymbol{w}_n^s$는 각각 긴 원본 발성과, 이로부터 일부 세그먼트를 취해 얻은 짧은 발성으로부터 추출한 i-벡터들이다. SUVN(LDA) 변환된 i-벡터는 다음과 같이 계산한다.

$$\hat{\boldsymbol{w}}_{SUVN(LDA)} = DA^T\boldsymbol{w}, \tag{5}$$

이 때 D는 $DD^T = S_{SUV(LDA)}^{-1}$에 대한 촐레스키 분해로 구할 수 있다.

III. 제안한 화자 확인 시스템

제안한 방법은 i-벡터/PLDA 기반 시스템에 적용 가능한 DNN기반 짧은 발성 보상 기법이다. 이는 짧은 테스트 발성으로부터 추출한 짧은 발성 특징 벡터를 가상의 긴 발성에서 추출한 적절한 긴 발성 특징 벡터로 변환한다. 이 때, 새로운 부공간을 찾는 주성분분석(Principal Component Analysis, PCA)이나 LDA와는 달리, 본래의 전체 변이 공간상의 벡터로 변환한다. d차원의 입력 특징 벡터 $\boldsymbol{w}_{in}$에 대한 DNN 변환 특

징을 다음과 같이 정의한다.

$$\hat{\boldsymbol{w}}_{proposed} = (\Phi \circ f_{\boldsymbol{\theta}})(\boldsymbol{w}_{in}), \quad (6)$$

여기서 $f_{\boldsymbol{\theta}}$는 파라미터 $\boldsymbol{\theta}$를 갖는 DNN, Φ는 DNN 출력을 새로운 특징 벡터로 변환하는 함수이다. 입력 특징 벡터 $\boldsymbol{w}_{in}$은 길이 정규화된 i-벡터거나 WCCN(LDA) 혹은 SUVN(LDA) 변환된 i-벡터가 될 수 있다.

DNN의 앞먹임 결과 $\boldsymbol{o}_{dnn}$은 t개 출력 노드의 활성값들을 원소로 하는 t차원 벡터이다.

$$\boldsymbol{o}_{dnn} = f_{\boldsymbol{\theta}}(\boldsymbol{w}_{\in}). \quad (7)$$

변환 함수 $\Phi(\bullet)$는 DNN 출력 벡터 $\boldsymbol{o}_{dnn}$을 최종적인 DNN 변환 특징 벡터 $\hat{\boldsymbol{w}}_{proposed}$로 변환한다.

$$\hat{\boldsymbol{w}}_{proposed} = \Phi(\boldsymbol{o}_{dnn}). \quad (8)$$

본 연구에서는 각각 2048개의 ReLU (Rectified Linear Unit, ReLU)[12]들로 구성된 다섯 개의 은닉층, 선형 출력 노드들로 구성된 출력층을 가진 DNN을 학습한다 (Fig. 2).

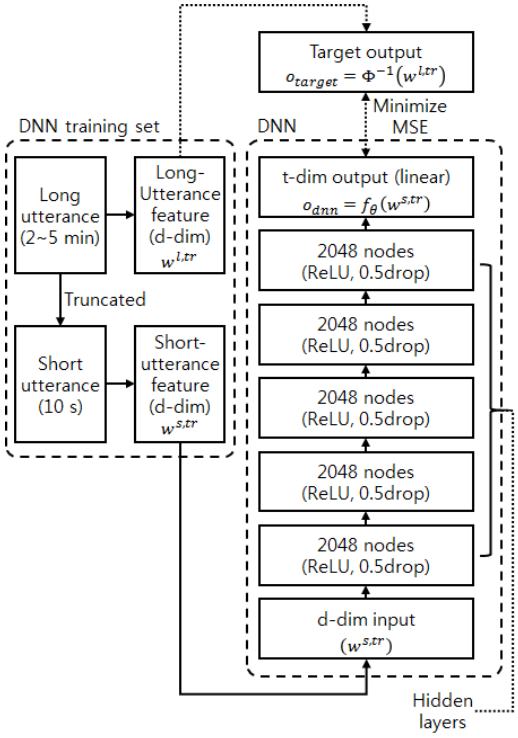

Fig. 2. DNN training for short utterance compensation.

DNN 학습을 위해 DNN 학습세트의 모든 발성들로부터 각각 10초 길이의 세그먼트를 랜덤하게 취한다. 이 때, 원본 발성(긴 발성)의 특징을 $\boldsymbol{w}^{l,tr}$, 10초 길이 세그먼트(짧은 발성)의 특징을 $\boldsymbol{w}^{s,tr}$라 하고, DNN의 입력이 $I^{in}(\boldsymbol{w}^{s,tr})$일 때의 목표 출력 레이블을 다음과 같이 가정한다.

$$\boldsymbol{o}_{target} = \Phi^{-1}(\boldsymbol{w}^{l,tr}), \quad (9)$$

여기서 $\Phi^{-1}(\bullet)$는 $\Phi(\bullet)$의 역함수이다($\Phi(\boldsymbol{x}) = \boldsymbol{y} \Leftrightarrow \Phi^{-1}(\boldsymbol{y}) = \boldsymbol{x}$). 즉, 제안한 DNN은 모든 DNN 학습세트의 각 발성에 대한 $\boldsymbol{o}_{dnn}$과 $\boldsymbol{o}_{target}$의 차이를 줄이도록 학습된다.

본 연구에서는 변환 함수 $\Phi(\bullet)$를 달리하는 세 가지 버전의 DNN을 제안한다.

3.1 DNN 사상

제안한 DNN의 첫 번째 버전(DNN 사상)은 입력된 짧은 발성 특징 벡터를 가상의 긴 발성 특징 벡터로 직접 사상한다. 이 경우, DNN의 출력 벡터를 바로 새로운 특징 벡터로 이용하므로, 별도의 변환 함수는 필요하지 않다. 즉, DNN 사상의 변환 함수 $\Phi_{DProj}(\bullet)$는 다음과 같다.

$$\hat{\boldsymbol{w}}_{proposed} = \Phi_{DProj}(\boldsymbol{o}_{dnn}) = \boldsymbol{o}_{dnn}, \quad (10)$$

이 때 정답 출력은 다음과 같다.

$$\boldsymbol{o}_{target} = \Phi^{-1}_{DProj}(\boldsymbol{w}^{l,tr}) = \boldsymbol{w}^{l,tr}. \quad (11)$$

하나의 입력 특징에 대한 DNN 출력은 i-벡터 공간상의 한 점이다[Fig. 3(a)]. 따라서 DNN의 입력과 출력의 차원은 $t = d$로 동일하다. 하지만 이 방법은 DNN 파라미터 학습시 탐색 공간이 매우 넓은데 비해 주어진 정보가 적으므로 좋은 결과를 얻기 어려울 수 있다.

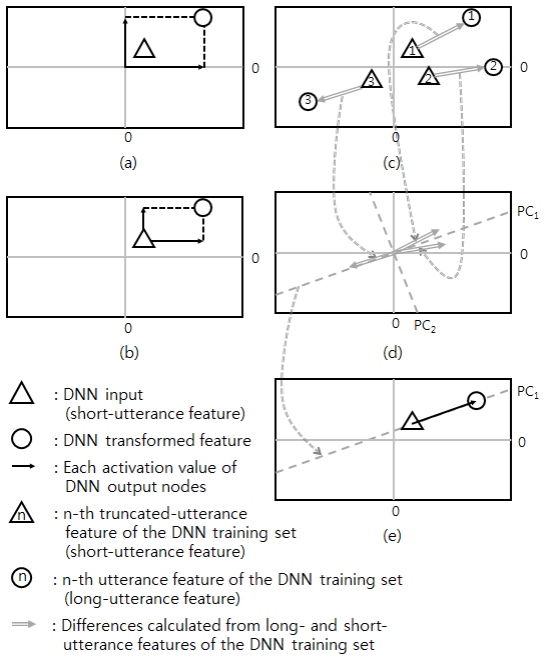

Fig. 3. Two-dimensional examples based on the following translation functions: (a) DNN projection, (b) DNN translation, (c) Calculation of difference vectors for PCs, (d) Estimating PCs, (e) DNN-PCA translation.

3.2 DNN 평행이동

제안한 DNN 구조의 두 번째 버전은 주어진 짧은 발성 특징 벡터 $\boldsymbol{w}_{in}$에 DNN 출력을 더하여 가상의 긴 발성 특징 벡터로 변환한다. 따라서 변환 함수 및 그에 따른 수식들은 다음과 같다.

$$\hat{\boldsymbol{w}}_{proposed} = \Phi_{DTrans}(\boldsymbol{o}_{dnn}) = \boldsymbol{w}_{in} + \boldsymbol{o}_{dnn}, \quad (12)$$

$$\boldsymbol{o}_{target} = \Phi^{-1}_{DTrans}(\boldsymbol{w}^{l,tr}) = \boldsymbol{w}^{l,tr} - \boldsymbol{w}^{s,tr}. \quad (13)$$

DNN 평행이동에서는 주어진 짧은 발성 특징 벡터로부터의 평행이동으로 적절한 가상의 긴 발성 특징 벡터를 표현할 수 있다고 간주한다. DNN 사상에 비해 이동 시작점을 약간의 정보로서 더 제공한 것이므로 보다 쉽게 DNN을 학습할 수 있을 것으로 기대하였다[Fig. 3(b)]. DNN 사상에서는 DNN 출력을 그대로 특징으로 사용하므로, 때때로 미지의 입력에 대해 전혀 엉뚱한 보상 결과를 얻을 수 있다. 예를 들어, DNN 출력의 크기가 작을 경우 새로운 특징은 i-벡터 공간 상의 원점 근방에 위치하게 된다. DNN 평행이동에서는 본래의 특징 벡터에 DNN의 출력을 가산하므로, 미지의 입력에 대해 DNN 출력의 크기가 작아지는 상황에서도 최소한 본래의 특징과 유사한 성능을 낼 수 있을 것으로 기대하였다. 이 경우 역시, DNN의 입출력 차원은 $t = d$ 로 동일하다.

3.3 DNN-PCA 평행이동

제안한 방법의 마지막 버전은 DNN 학습 단계에서 PCA를 이용하여 탐색 공간을 축소하는 것이다. 이를 통해 DNN을 학습할 때, 상대적으로 덜 중요한 탐색 공간에서 지역적 최소점에 수렴하는 문제를 완화하고자 하였다. 이때의 변환 함수를 다음과 같이 정의한다.

$$\hat{\boldsymbol{w}}_{proposed} = \Phi_{DPCATrans}(\boldsymbol{o}_{dnn}) = \boldsymbol{w}_{in} + C^{T}\boldsymbol{o}_{dnn}, \quad (14)$$

여기서 C는 DNN 학습세트에서 추정한 t개의 d차원 주성분 벡터로 구성된 $(t \times d)$ 행렬이다. Fig. 3에서 주성분 벡터는 "PC"로 표기하였다. 이 주성분 행렬을 계산하기 위해 DNN 학습세트의 모든 긴 발성 특징과 이로부터 획득한 짧은 발성 특징간의 차 벡터를 계산한다[Fig. 3(c)]. 이 차 벡터들의 공분산 행렬에 대한 고유벡터들을 계산[Fig. 3(d)]하고, 탐색 공간의 차원 축소를 위해 고유값이 큰 순으로 상위 t개의 고유벡터들을 취해 주성분 행렬을 구성한다. 이 버전에서는 선택한 주성분들의 방향을 따라 주어진 짧은 발성 특징 벡터로부터의 평행이동을 수행한다[Fig. 3(e)]. 따라서 목표 출력은 다음과 같다.

$$\boldsymbol{o}_{target} = \Phi^{-1}_{DPCATrans}(\boldsymbol{w}^{l,tr}) = C^{-1^{T}}(\boldsymbol{w}^{l,tr} - \boldsymbol{w}^{s,tr}), \quad (15)$$

이 경우 DNN의 각 출력은 t개 주성분 방향으로의 가중치를 나타내므로, DNN의 입력과 출력의 차원이 $t \le d$로 다를 수 있다.

IV. 실험 및 결과

4.1 음성 데이터베이스

본 연구에서는 NIST 2008 SRE 코퍼스(NIST08

SRE)를 이용하여 제안한 방법의 성능을 평가한다. NIST08 SRE는 다양한 조건의 평가세트를 포함하는데, 본 연구에서는 이 중 하나인 short 2-10 s 조건 가운데에서도 det 6, 7과 8의 남성 화자 발성을 대상으로 한다. Short 2-10 s 조건은 긴 학습 발성(2.5 min가량)과 짧은 테스트 발성(10 s)으로 구성된 평가세트이다. 개발세트로는 NIST 2004 SRE(NIST04 SRE), NIST 2005 SRE(NIST05 SRE), Switchboard2 (SWB2) 코퍼스들의 남성 화자 발성들을 이용하였다. 개발세트 및 평가세트 코퍼스들의 상세한 구분은 Table 1에 표기하였다.

음향특징은 20차 MFCCs(Mel-Frequency Cepstral Coefficients)와 이의 △(속도, 20차) 및 △△(가속도, 20차)를 덧붙여 총 60차로 추출하였다. 음향특징 레벨에서 에너지 기반 VAD와 CMN을 적용하였다. 배경 화자 모델로는 512개의 가우시안 성분으로 구성된 하나의 성별 종속(남성) 모델을 학습하였다. 전체변이 부공간의 차원은 400이며, 모든 i-벡터를 각각 동일한 길이가 되도록 길이 정규화하였다. 추출한 특징을 분류하기 위해 PLDA 모델링 및 스코어링을 적용하였다. 이 과정들은 Kaldi[13] 툴킷을 사용하여 수행하였다. S-정규화[1]를 위해 NIST04 SRE에서 200개의 발성을 랜덤 선택하여 레퍼런스 데이터로 활용하였다.

제안한 DNN은 딥러닝 라이브러리인 Keras[14]를 이용하여 실험하였다. 이 때, Keras의 백엔드 엔진으로는 Theano를 사용하였다. 앞서 소개한 세 가지 버전의 DNN은 모두 동일한 방법으로 학습하였다. 각 DNN을 2048개의 은닉 노드들로 구성된 다섯 개의 은닉 층으로 구성하였다. 은닉층의 활성 함수로 ReLU를, 출력층의 활성 함수로 선형 함수를 사용하였다. 평균 제곱 오차를 손실 함수로 이용하였다. 학습률은 10.0에서 시작하여 매 반복마다 감쇠하였다(감쇠 인자 = 0.01). 미니배치 크기는 1000으로 하였고, 모멘텀을 적용(모멘텀 계수 = 0.9)하였다. 각 은닉층 및 출력층에 배치 정규화[15]를 적용하였다. 각 은닉층에 0.5의 비율로 드롭 아웃[16]을 적용하였다. 각 학습은 수렴시까지 반복하였다. DNN 검증세트에 대해 가장 낮은 오류를 갖는 시점의 가중치와 바이어스를 최종적인 DNN 파라미터 $\boldsymbol{\theta}$로 선택하였다.

Table 1. Corpora used for each experimental phase.

Corpus		NIST04 SRE	NIST05 SRE	SWB2	NIST08 SRE
Development set	UBM train	O			
	TVM train	O	O	O	
	PLDA train	O	O	O	
	S-normalization	O			
	DNN train	O	O		
	DNN validation			O	
Evaluation set					O

4.2 실험결과

베이스라인 및 제안한 방법들의 성능 비교 결과를 Table 2에 기재하였다. 성능 평가 척도는 2008년 기준 최소 검출 비용 함수(minDCF)이다. 베이스라인 시스템은 화자 특징에 따라 세 가지로 구분하였는데, 길이 정규화된 i-벡터를 사용하는 경우(baseline1), WCCN(LDA) 변환된 특징을 사용하는 경우(baseline2), SUVN(LDA) 변환된 특징을 사용하는 경우(baseline3)이다.

Table 2. Comparison of baseline and proposed systems on NIST08 SRE short 2-10 s (male).

Method	Feat dim	DNN output dim	Evaluation condition		
			det6	det7	det8
LN (baseline1)	400	-	0.0420	0.0313	0.0323
LN-DProj		400	0.0994	0.0985	0.0983
LN-DTrans			0.0410	0.0305	0.0318
LN-DPCATrans		150	0.0404	0.0311	0.0331
		100	0.0418	0.0312	0.0320
		50	0.0405	0.0294	0.0305
WCCN(LDA) (baseline2)	250	-	0.0393	0.0297	0.0304
WCCN(LDA)-DProj		250	0.1000	0.1000	0.1000
WCCN(LDA)-DTrans			0.0389	0.0294	0.0287
WCCN(LDA)-DPCATrans		150	0.0398	0.0306	0.0299
		100	**0.0387**	0.0296	0.0299
		50	0.0399	0.0293	0.0284
SUVN(LDA) (baseline3)		-	0.0400	0.0294	0.0301
SUVN(LDA)-DProj		250	0.0728	0.0659	0.0645
SUVN(LDA)-DTrans			0.0402	0.0299	0.0297
SUVN(LDA)-DPCATrans		150	0.0406	0.0284	0.0278
		100	0.0395	**0.0276**	**0.0268**
		50	0.0399	0.0281	0.0276

각 det 별로 가장 좋은 실험 결과는 굵은글씨로 표시하였다. 본 연구에서는 WCCN(LDA)와 SUVN(LDA)를 적용한 후 추가적인 길이 정규화를 수행하였다. 전체 변이 부공간의 본래 차원은 400차이고, WCCN (LDA) 혹은 SUVN(LDA)를 적용하는 경우에는 250차로 축소하였다. 실험 결과, WCCN(LDA) 혹은 SUVN (LDA)를 적용하였을 때, 길이 정규화만 수행한 경우보다 낮은 최소 검출 비용을 보였다.

세 가지 베이스라인 특징을 각기 제안한 방법으로 보상하여 성능을 평가하였다. 길이 정규화된 i-벡터에 제안한 방법을 적용한 경우 사용한 방법의 이름이 "LN"으로 시작하도록 표기하였다. WCCN(LDA) 및 SUVN(LDA) 변환 후 제안한 방법을 적용한 경우는 각각 "WCCN (LDA)"와 S"UVN (LDA)"으로 시작하도록 표기하였다. 제안한 방법은 버전에 따라 각각 "DProj" (DNN 사상), "DTrans" (DNN 평행이동), "DPCATrans" (DNN-PCA 평행이동)로 약어 표기하였다. 각 det 별로 가장 좋은 실험 결과는 굵은 글씨로 표시하였다. 또한 제안한 방법을 적용하기 전에 비해 성능이 개선된 경우를 밑줄로 표시하였다.

DNN 사상과 DNN 평행이동은 DNN의 입출력 차원이 동일하다. 이 때, DNN의 출력은 전체 변이 부공간 상의 새로운 특징 벡터로 직접 사상되거나 입력 특징 벡터에 가산하여 평행이동된 특징 벡터를 획득하는데 쓰인다. DNN-PCA 평행이동에서 DNN 출력은 PCA로 추정한 차원 축소된 부공간 상에서 원 특징 벡터를 이동시키기 위한 차 벡터로 쓰인다. 따라서 DNN-PCA 평행이동의 DNN 출력 차원은 입력 특징 벡터의 차원보다 작다. 본 연구에서는 DNN 출력 차원을 50차, 100차, 150차로 실험하였다.

제안한 방법 중 DNN 사상은 다른 방법들과 달리, 모든 경우에 대해 매우 저조한 성능을 보였다. DNN 평행이동은 적용 전에 비해 대체로 소폭 개선된 성능을 보였다. 길이 정규화된 i-벡터에 적용하였을 때에는, 적용하기 전에 비해 det6, 7, 8에서 각각 2.4 %, 2.6 %, 1.5 % 개선된 최소 검출 비용을 보였다. WCCN (LDA) 수행 후 적용하였을 때에는 최소 검출 비용이 더 감소하였으며, 이는 적용 전에 비해 각각 1.0 %, 1.0 %, 5.6 % 개선된 성능이다. 하지만 SUVN(LDA) 수행 후 적용하였을 때에는 적용 전에 비해 각각 -0.5 %, -1.7 %, 1.3 %로 성능 개선을 확인하지 못하였다.

DNN-PCA 평행이동은 적절한 축소 차원을 선택한 경우에 한해 좋은 성능을 보였다. 특히 SUVN (LDA) 수행 후 100차로 축소하여 적용하였을 때, 모든 실험 중에서 가장 낮은 평균 최소 검출 비용을 보였다. 이 때의 최소 검출 비용은 적용 전에 비해 각각 1.3 %, 6.1 %, 11.0 % 개선되었다.

DNN 사상의 성능이 유독 저조한 원인을 분석하기 위해 SUVN(LDA) 수행 후 베이스라인 및 제안한 방법을 적용했을 때의 긴 발성 특징과 짧은 발성 특징 간의 평균 절대치 오차를 계산하였다(Table 3). 긴 발성 특징은 베이스라인이나 제안한 방법에 관계 없이 항상 동일하나, 제안한 방법에서의 짧은 발성 특징은 DNN 학습 수렴 후 보상한 결과를 취하였다. DNN 학습세트 및 검증세트 각각에 대해, 제안한 방법으로 보상하였을 때 짧은 발성 특징이 긴 발성 특징에 더 가까워진 경우를 밑줄로 표시하였다. 평균 절대치 오차 계산 결과, DNN 사상은 DNN 학습세트상의 짧은 발성 특징을 긴 발성 특징에 가깝게 사상하였지만, DNN 검증세트의 데이터에 대해서는 오히려 더 멀어지도록 잘못 사상하는 현상을 보였다. 즉, 일반화 성능이 저조한 것을 확인할 수 있다. 반면, DNN 평행이동은 학습세트 및 검증세트 모두에 대해 적용 전과 유사한 수준의 거리를 유지하였으며, DNN-PCA 평행이동의 경우 학습세트는 물론이고 검증세트에서도 짧은 발성 특징을 긴 발성 특징에 가깝게 보상하는 것을 확인할 수 있었다.

본 연구에서는 평가세트인 NIST 2008 SRE의 short2-

Table 3. Mean average errors between long- and short-utterance features.

Method	Feat dim	DNN output dim	Mean average error	
			DNN train set	DNN valid set
SUVN(LDA) (baseline3)	250	-	0.784271	0.805483
SUVN(LDA)-DProj		250	0.625202	0.881388
SUVN(LDA)-DTrans			0.785852	0.805839
SUVN(LDA)-DPCATrans		150	0.780929	0.805905
		100	0.720984	0.793687
		50	**0.713959**	**0.793247**

Table 4. Comparison of baseline and proposed systems on truncated test utterances with various durations.

Duration of test utterances	Method	Evaluation condition		
		det6	det7	det8
10 s	SUVN(LDA) (baseline3)	0.0400	0.0294	0.0301
	SUVN(LDA)-DPCATrans(100d)	0.0395	0.0276	0.0268
5 s	SUVN(LDA) (baseline3)	0.0604	0.0486	0.0450
	SUVN(LDA)-DPCATrans(100d)	0.0590	0.0484	0.0475
2 s	SUVN(LDA) (baseline3)	0.0779	0.0729	0.0697
	SUVN(LDA)-DPCATrans(100d)	0.0768	0.0719	0.0680

10 s 조건의 테스트 발성 길이(10 s)에 맞추어 개발세트의 각 발성을 10 s 길이로 분할하여 SUVN(LDA) 및 제안한 DNN을 학습하였다. 만약 테스트 발성의 길이가 10 s로 고정되지 않고 그보다 짧은 길이로 입력될 경우에도 성능 개선을 얻을 수 있는지 확인하고자 하였다. 이를 위해, short2-10 s 조건의 각 테스트 발성에서 5 s 및 2 s 길이의 구간을 임의로 취하여 보다 짧은 길이를 갖도록 만들었다. 베이스라인은 SUVN(LDA)이며, 제안한 방법 중에서는 DNN 출력 차원이 100차인 DNN-PCA 평행이동에 대해서 실험하였다. Table 4에 실험 결과를 기재하였으며, 베이스라인에 비해 성능이 개선된 경우는 밑줄로 표기하였다. 실험 결과, 한 경우를 제외(5 s 절단, det 8)하고는 대부분의 상황에서 베이스라인에 비해 소폭 개선된 성능을 보이는 것을 확인할 수 있었다.

V. 결 론

본 연구에서는 딥러닝을 이용하여 화자 확인시 짧은 발성 특징을 보상하는 방법을 제시하였다. 긴 학습 발성과 짧은 테스트 발성을 이용하는 상황에서, 제안한 방법은 테스트 발성으로부터 추출한 특징 벡터를 임의의 긴 발성에서 추출한 특징 벡터와 유사하게 변환한다. 이 때 사용되는 DNN의 구조에 따라 총 세 가지 버전의 변환 방법을 제안하였다. 이 방법들은 학습 단계에서 정답 출력 레이블을 달리하여 학습한다. 실험 결과 WCCN(LDA) 변환된 특징에 DNN 평행이동을 적용하였을 때 WCCN(LDA)와 SUVN(LDA)만을 적용한 베이스라인 시스템들에 비해 더 낮은 최소 검출 비용을 확인할 수 있었다. 또한 적절한 축소 차원을 선택한 경우 DNN-PCA 평행이동이 이보다 더 좋은 성능을 보일 수 있음을 보였다.

본 연구에서는 DNN 학습세트에서 사용한 짧은 발성의 길이가 테스트 발성의 길이와 완전히 일치하지 않아도 성능 개선을 얻을 수 있음을 확인 하였으며, 향후 보다 다양한 길이의 발성을 보상하는 일반화된 구조로 확장하고자 한다. 또한 DNN-PCA 평행이동 시 최적의 축소 차원을 결정하는 방안을 찾고, 다양한 심층 신경망 구조나 손실 함수를 적용하여 제안한 방법의 성능을 개선하고자 한다. 그 밖에 DNN-PCA 평행이동에서 PCA 대신 비선형 차원 축소가 가능한 딥 오토 인코더[17]를 적용하는 방안을 생각해 볼 수 있다.

감사의 글

본 연구는 산업통상자원부 및 한국산업기술평가관리원의 IT R&D사업의 연구결과로 수행되었음(10041610, 인식센서융합 기반 실환경하에서 임의의 사용자 30명에 대해 인식률 99 %에 근접하는 사용자의 신원과 행위 및 위치 정보 인식 기술 개발).

References

1. P. Kenny, “Bayesian speaker verification with heavy tailed priors,” in Proc. Odyssey, 61-70 (2010).
2. A. Kanagasundaram, R. Vogt, D. Dean, S. Sridharan, and M. Mason, “i-vector based speaker recognition on short utterances,” Interspeech, 2341-2344 (2011).
3. A. K. Sarkar, D. Matrouf, P. M. Bousquet, and J. F. Bonastre, “Study of the effect of i-vector modeling on short and mismatch utterance duration for speaker verification,” in Proc. Interspeech, 2662-2665 (2012).
4. P. Kenny, T. Stafylakis, P. Ouellet, M. J. Alam, and P. Dumouchel, “Plda for speaker verification with utterances of arbitrary duration,” ICASSP, 7649-7653 (2013).
5. M. I. Mandasari, R. Saeidi, M. McLaren, and D. van Leeuwen, “Quality measure functions for calibration

of speaker recognition systems in various duration conditions," IEEE Trans. on Audio, Speech, and Lang. Process. **21**, 2425-2438 (2013).
6. A. Kanagasundaram, D. Dean, S. Sridharan, J. Gonzalez-Dominguez, J. Gonzalez-Rodriguez, and D. Ramos, "Improving short utterance i-vector speaker verification using utterance variance modelling and compensation techniques," Speech Communication **59**, 69-82 (2014).
7. E. Variani, X. Lei, E. McDermott, I. Lopez-Moreno, and J. Gonzalez-Dominguez, "Deep neural networks for small footprint text-dependent speaker verification," ICASSP, 4052-4056 (2014).
8. Y. Lei, N. Scheffer, L. Ferrer, and M. McLaren, "A novel scheme for speaker recognition using a phonetically aware deep neural network," ICASSP, 1695-1699 (2014).
9. *The NIST Year 2008 Speaker Recognition Evaluation Plan*, http://www.itl.nist.gov/iad/mig/tests/sre/2008/sre08_evalplan_release4.pdf, 2008.
10. D. Garcia-Romero, and C. Y. Espy-Wilson. "Analysis of i-vector length normalization in speaker recognition systems," Interspeech, 249-252 (2011).
11. N. Dehak, R. Dehak, J. R. Glass, D. A. Reynolds, and P. Kenny, "Cosine similarity scoring without score normalization techniques," in Proc. Odyssey, 71-75 (2010).
12. V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," ICML, 807-814 (2010).
13. *The Kaldi Speech Recognition Toolkit*, http://kaldi-asr.org/doc/about.html, 2011.
14. *Blocks and Fuel: Frameworks for Deep Learning*, https://arxiv.org/abs/1506.00619, 2015.
15. *Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift*, https://arxiv.org/abs/1502.03167, 2015.
16. *Improving Neural Networks by Preventing Co-Adaptation of Feature Detectors*, https://arxiv.org/abs/1207.0580, 2012.
17. G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," Science **313**, 504-507 (2006).

저자 약력

▸ 양 일 호 (IL-Ho Yang)

2008년 2월: 서울시립대학교 컴퓨터과학부 학사
2010년 2월: 서울시립대학교 컴퓨터과학부 석사
2010년 ~ 현재: 서울시립대학교 컴퓨터과학부 박사과정

▸ 허 희 수 (Hee-Soo Heo)

2013년 2월: 서울시립대학교 컴퓨터과학부 학사
2013년 ~ 현재: 서울시립대학교 컴퓨터과학부 석박사통합과정

▸ 윤 성 현 (Sung-Hyun Yoon)

2015년 8월: 서울시립대학교 컴퓨터과학부 학사
2015년 ~ 현재: 서울시립대학교 컴퓨터과학부 석사과정

▸ 유 하 진 (Ha-Jin Yu)

1990년 2월: KAIST 전산학과 학사
1992년 2월: KAIST 전산학과 석사
1997년 2월: KAIST 전산학과 박사
1997년 ~ 2000년: LG 전자 전자기술원 선임연구원
2000년 ~ 2002년: SL2(주) 연구소장
2002년 ~ 현재: 서울시립대학교 컴퓨터과학부 교수