DNN-Based Score Calibration with Multi-Task Learning for Noise Robust Speaker Verification

Zhili TAN, Man-Wai MAK, *Senior Member, IEEE,* Brian Kan-Wing MAK, *Senior Member, IEEE,*

***Abstract*—This paper proposes and investigates several deep neural network (DNN)-based score compensation, transformation and calibration algorithms for enhancing the noise robustness of i-vector speaker verification systems. Unlike conventional calibration methods where the required score shift is a linear function of SNR or log-duration, the DNN approach learns the complex relationship between the score shifts and the combination of i-vector pairs and uncalibrated scores. Furthermore, with the flexibility of DNNs, it is possible to explicitly train a DNN to recover the clean scores without having to estimate the score shifts. To alleviate the overfitting problem, multi-task learning is applied to incorporate auxiliary information such as SNRs and speaker ID of training utterances into the DNN. Experiments on NIST 2012 SRE show that score calibration derived from multi-task DNNs can improve the performance of the conventional score-shift approch significantly, especially under noisy conditions.**

***Index Terms*—Deep learning, speaker verification, score calibration, multi-task learning, noise robustness.**

I. Introduction

Automatic speaker verification aims to verify whether a test utterance is spoken by a target speaker. Since 2011, the i-vector approach [1] together with probabilistic linear discriminant analysis (PLDA) [2] have dominated this area. Under this framework, each utterance is represented by a low-dimensional i-vector that captures speaker- and channel-dependent characteristics, and the PLDA model aims to separate the speaker variability from channel variability in the i-vector space. During verification, given an i-vector pair derived from the utterance of a target (claimed) speaker and the utterance of a claimant, a likelihood ratio score (namely PLDA score)

$$S = \frac{p(\text{i-vector pair}|\text{same speaker})}{p(\text{i-vector pair}|\text{different speaker})}$$

is computed to determine whether the i-vectors in the i-vector pair are from the same speaker or not. When computing the PLDA score, the separation of speaker and channel subspaces in the PLDA model is leveraged to marginalize the channel effect on the i-vector pair. This marginalization process leads to a likelihood ratio score that is mainly dependent on speaker characteristics of the i-vector pair.

One of the main challenges in speaker verification is to enhance the noise robustness of speaker verification systems. In particular, the speaker characteristics in i-vectors may be distorted if the background noise is very severe. In recent years, a lot of effort has been made to mitigate the problem. For example, Hasen and Hansen [3] proposed to enhance and normalize acoustic features by feature-domain factor analysis. Denoising autoencoders (DAE) [4] have been applied to restore speech either in the spectral domain [5], [6] or in the i-vector space [7], [8].

Attempts have also been made to improve noise robustness in PLDA models. For example, Hasan *et al.* [9] and Garcia-Romero *et al.* [10] trained a PLDA model by pooling speech from multiple conditions, and Li and Mak [11], [12] modeled the noise-level variability in utterances by introducing an SNR factor and an SNR subspace into the PLDA model. In [13], [14], Mak *et al.* advocated that utterances of different SNR levels will not only cause the i-vectors to fall on different regions of the i-vector spaces but also change the orientation of the speaker subspace. A mixture PLDA model with mixture alignments determined by the SNR level of utterances was then derived to model the SNR-dependent i-vectors.

Observing that adverse acoustic conditions and duration variability in utterances could have detrimental effect on PLDA scores, researchers explored the potential of other backends to replace the PLDA models, e.g., support vector machines (SVMs) [15] or even end-to-end learning [16]. Besides, a number of score calibration methods [17]–[19] have been proposed to compensate for the detrimental effect on the PLDA scores. While many of these methods can compensate for the duration mismatch only, there are techniques also take the SNR mismatch into account [20]–[23]. All of these methods compensate for the detrimental effect by modeling it as a shift in the PLDA scores. The goal is to estimate the appropriate shift from some meta data (e.g., duration and SNR [20], [21]) or from the i-vectors [22] to counteract the effect.

In [17], [20], [21], the score shift was deterministic and was assumed to be linearly related to the SNR of utterances and/or to the logarithm of utterance duration. In [23], the shift was stochastic and was assumed to follow a Gaussian distribution with mean and variance dependent on the speech quality. Given an observed noisy score, a Bayesian network was used to infer the posterior distribution of the target and non-target hypotheses, from which a calibrated likelihood-ratio is computed. On the other hand, the score shift in [22], [24] was assumed to be simple functions (bilinear transformation and cosine distance) of the two quality vectors derived from the i-vectors involved in the scoring. While promising results have been achieved, the relationship between score shift and SNR and log-duration may not be linear, and the bilinear

Z. L. Tan and M. W. Mak are with The Department of Electronic and Information Engineering, The Hong Kong Polytechnic University, Hong Kong SAR (Email: eddy.zhili@connect.polyu.hk; enmwmak@polyu.edu.hk). Brian K. W. Mak is with The Department of Computer Science, The Hong Kong University of Science and Technology, Hong Kong SAR (Email: mak@cse.ust.hk). This work was supported in part by The RGC of Hong Kong SAR (Grant Nos. PolyU 152518/16E and PolyU 152068/15E).

transformation and cosine distance scores may not accurately reflect the true relationship between the score shift and i-vector quality.

In this paper, we attempt to obviate the above restrictions by directly modeling the complex relationship between score shift and distorted i-vectors. Inspired by the recent findings that deep neural networks (DNNs) have a high capacity in modeling complex relationship, we trained a DNN using i-vector pairs (derived from both clean and noisy speech) as inputs and the ideal score shifts as target outputs. This method, however, requires parallel training data comprising clean and noisy i-vectors so that ideal score shifts can be computed during the training stage. To obviate this requirement, we trained a second DNN that can produce close-to-ideal (clean) PLDA scores by using i-vector pairs augmented with the PLDA score as input. To further leverage the meta data (SNR and speaker labels) that can be easily obtained from training utterances, we used multi-task learning to train a third network whose input is identical to the second DNN but its outputs aim to achieve two tasks: regression and classification. For the former, the network was trained to produce ideal score shift, clean score, and the SNRs of target and test utterances, whereas for the latter, the network aims to classify whether the i-vector pairs come from the same speaker or from different speakers.

The paper is organized as follows. We will introduce the previous score calibration methods in Section II; based on these methods, we propose the DNN-based calibration methods in Section III, where the DNNs are trained to output the calibrated score directly without estimating the score shift. Experiments on NIST 2012 SRE in Section IV show that the auxiliary tasks in multi-task learning help the DNNs to find a better solution, which makes the multi-task DNNs outperform the single-task DNNs under all SNR conditions significantly.

II. QUALITY-BASED SCORE CALIBRATION

To improve the robustness of speaker verification, Mandasari *et al.* [20], [21] and Hasan *et al.* [17] proposed several quality measure functions (QMFs) to compensate for the score shift caused by background noise and short utterance duration. A QMF is a function of some quality measures such as SNR and duration that can be directly obtained from utterances. Denote S as the *uncalibrated* verification score of a target-speaker utterance and a test utterance. Also denote λ_{tgt} and λ_{tst} as the quality measures of the target-speaker and test utterances, respectively. Then, the *calibrated* score S' can be computed as follows:

$$S' = w_0 + w_1 S + Q(\lambda_{tgt}, \lambda_{tst}), \tag{1}$$

where $Q(\lambda_{tgt}, \lambda_{tst})$ is a QMF. In [20], [21], the QMFs were based on the duration and SNR of test speech:

$$\begin{aligned} Q_{\text{SNR}}(\text{SNR}_{tst}) &= w_2 \text{SNR}_{tst} \\ Q_{\text{Dur}}(d_{tst}) &= w_2 \log(d_{tst}) \\ Q_{\text{SNR+Dur}}(\text{SNR}_{tst}, d_{tst}) &= w_2 \text{SNR}_{tst} + w_3 \log(d_{tst}), \end{aligned} \tag{2}$$

where SNR_{tst} and d_{tst} are the SNR and duration of the test utterance, and w_2 and w_3 are their corresponding weight. If the effect of noise in the target utterance is also considered, Q_{SNR} becomes:

$$Q_{\text{SNR2}}(\text{SNR}_{tgt}, \text{SNR}_{tst}) = w_2 \text{SNR}_{tgt} + w_3 \text{SNR}_{tst}, \tag{3}$$

where SNR_{tgt} is the SNR of the target-speaker utterance. In Eqs. 1–3, the weights w_i, $i = 0, \ldots, 3$, can be estimated by logistic regression [25].

By assuming that i-vectors are acoustic-condition dependent, Ferrer *et al.* [24] and Nautsch *et al.* [22] derived a quality vector $\boldsymbol{q}$ based on the posterior probabilities of various acoustic conditions given an i-vector. Thus, each i-vector (either from target speaker or test speaker) is associated with a quality vector, and the score shift of a verification trial is a function of the quality vectors derived from the i-vectors in that trial. In [22], the function is called the function of quality estimate (FQE). Specifically, i-vectors derived from utterances of 55 combinations of different durations and SNRs were used to train 55 Gaussian models $\Lambda_j = \{\boldsymbol{\mu}_j, \boldsymbol{\Sigma}\}_{j=1}^{55}$. Each of these Gaussian models has its own mean $\boldsymbol{\mu}_j$ estimated from the i-vectors of the respective condition, but they share the same global covariance matrix $\boldsymbol{\Sigma}$. The j-th element of $\boldsymbol{q}$ for an i-vector $\boldsymbol{x}$ is defined as the posterior of the j-th condition:

$$q_j = \frac{\mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_j, \boldsymbol{\Sigma})}{\sum_{j'} \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_{j'}, \boldsymbol{\Sigma})}, \; j = 1, \ldots, 55. \tag{4}$$

Given the i-vectors $\boldsymbol{x}_{tgt}$ and $\boldsymbol{x}_{tst}$ from a target-speaker and a test speaker, respectively, the corresponding quality vectors $\boldsymbol{q}_{tgt}$ and $\boldsymbol{q}_{tst}$ are obtained from Eq. 4 and the score shift can be obtained from a symmetric bilinear matrix $\boldsymbol{W}$ or cosine-distance score as follows:

$$\begin{aligned} Q_{\text{UAC}}(\boldsymbol{q}_{tgt}, \boldsymbol{q}_{tst}) &= w_2 \boldsymbol{q}_{tgt}^{\mathsf{T}} \boldsymbol{W} \boldsymbol{q}_{tst} \\ Q_{\text{qvec}}(\boldsymbol{q}_{tgt}, \boldsymbol{q}_{tst}) &= w_2 \text{cos}(\boldsymbol{q}_{tgt}, \boldsymbol{q}_{tst}), \end{aligned} \tag{5}$$

where

$$\cos(\boldsymbol{a}, \boldsymbol{b}) = \frac{\boldsymbol{a}^{\mathsf{T}} \boldsymbol{b}}{\|\boldsymbol{a}\| \|\boldsymbol{b}\|}.$$

In Eq. 5, the elements in a quality vector are the posteriors with respect to the corresponding SNR/duration groups, and the simple functions (bilinear transformation and cosine distance) of the two quality vectors could only reflect their similarity in terms of SNR and duration. It is still very close to the QMFs in Eq. 2 and Eq. 3, where the score shift is assumed to be linear with respect to SNR of utterances and/or to the logarithm of utterance duration. As we will discuss in Section IV and Fig. 9, the relationship between score shift and SNR and log-duration is complex, and only the SNR and log-duration information is not enough to estimate the ideal score shift. The i-vectors are essential for estimating the ideal score shift.

III. DNN-BASED SCORE CALIBRATION

This paper proposes an innovative score calibration algorithm to mitigate the limitations of the score calibration algorithms described in Section II. The main idea is to use deep neural networks (DNNs) to estimate the appropriate score shift or to output clean PLDA scores given noisy i-vectors and noisy PLDA scores as input. When the DNN is used for estimating

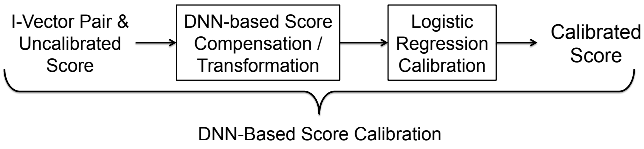

Fig. 1. DNN-based score calibration.

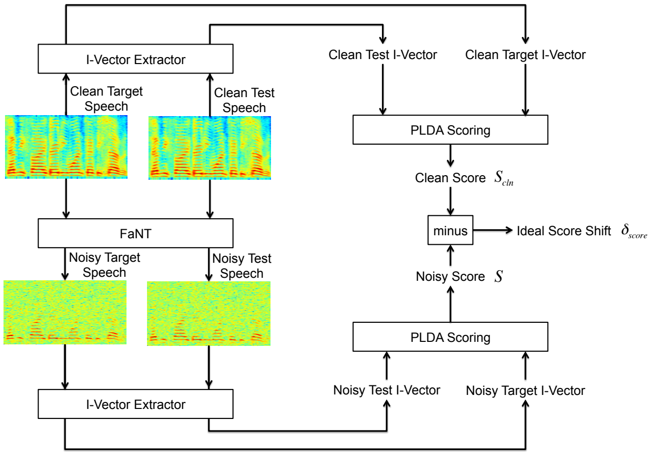

Fig. 2. Procedure of computing clean scores, noisy scores and score shifts during the training stage. The SNRs of the target speech and the test speech can be different, and even one or both of them could be clean.

the score shift, it essentially performs *score compensation* and its role is the same as that of the function Q in Eq. 1. However, when the DNN is used for outputting clean PLDA scores, it essentially performs *score transformation*. For whatever roles, a further *calibration* process is essential because the DNN cannot guarantee that the resulting scores are true log-likelihood ratios. To avoid cluttering with terminologies, we collectively refer to the score compensation, transformation, and calibration processes as DNN-based score calibration. Fig. 1 shows the full process.[1]

A. DNN Score Compensation: Estimating Score Shifts by DNNs

The proposed algorithm uses a DNN to estimate the appropriate score shift given the target and test i-vector pairs ($\boldsymbol{x}_{tgt}$ and $\boldsymbol{x}_{tst}$) and the uncalibrated PLDA score S as shown in Fig. 3. Specifically, given an uncalibrated PLDA score S of a verification trial, the compensated score is given by:

$$S'_1 = S + \text{DNN}_1(\boldsymbol{x}_{tgt}, \boldsymbol{x}_{tst}, S), \tag{6}$$

where DNN_1 denotes the output of a DNN that receives i-vector pairs and uncalibrated scores as input. With these inputs, the DNN outputs the shift of PLDA scores due to the deviation of the acoustic conditions from the clean one:

$$\text{DNN}_1(\boldsymbol{x}_{tgt}, \boldsymbol{x}_{tst}, S) \approx \delta_{score} = S_{cln} - S, \tag{7}$$

where S_{cln} is the PLDA score if both $\boldsymbol{x}_{tgt}$ and $\boldsymbol{x}_{tst}$ were derived from clean utterances. Substituting Eq. 7 to Eq. 6, we have:

$$S'_1 \approx S + (S_{cln} - S) = S_{cln},$$

which means that the clean score can be recovered.

To train DNN_1, we need i-vectors derived from both clean and noise contaminated utterances where the amount of noise contamination should be varied to give a rich set of δ_{score}'s. This can be done by using the FaNT tool [28] with the target SNR set to various levels. The procedure of computing S_{cln}, S and δ_{score} at the training stage is illustrated in Fig. 2. Note that Fig. 2 depicts the situation where both of the target-speaker and test utterances are noisy. However, in real environments, there are situations where either the target-speaker utterance or the test utterance is clean, or both are clean. To accommodate these situations, some of the "noisy i-vectors" in Fig. 2 should be derived from clean speech. Therefore, if both of the i-vectors in the lower branch of Fig. 2 are derived from clean utterances, we have $S = S_{cln}$ and $\delta_{score} = 0$. This is exactly what we want the DNN to produce when the input i-vectors are clean.

The PLDA score of i-vector pair ($\boldsymbol{x}_{tgt}$, $\boldsymbol{x}_{tst}$) can be expressed in terms of the log-likelihood ratio $\text{LLR}(\boldsymbol{x}_{tgt}, \boldsymbol{x}_{tst})$:

$$\begin{aligned} S &= \text{LLR}(\boldsymbol{x}_{tgt}, \boldsymbol{x}_{tst}) \\ &= \frac{1}{2}\boldsymbol{x}_{tgt}^{\mathsf{T}}\boldsymbol{Q}\boldsymbol{x}_{tgt} + \frac{1}{2}\boldsymbol{x}_{tst}^{\mathsf{T}}\boldsymbol{Q}\boldsymbol{x}_{tst} + \boldsymbol{x}_{tgt}^{\mathsf{T}}\boldsymbol{P}\boldsymbol{x}_{tst} + \text{const}, \end{aligned} \tag{8}$$

where $\boldsymbol{Q}$ and $\boldsymbol{P}$ are the matrices derived from the total covariances and across-speaker covariances of i-vectors [29]. Using Eq. 8, the general form of score shift is:

$$\begin{aligned} \delta_{score} &= \text{LLR}(\boldsymbol{x}_{tgt\_cln}, \boldsymbol{x}_{tst\_cln}) - \text{LLR}(\boldsymbol{x}_{tgt}, \boldsymbol{x}_{tst}) \\ &= \frac{1}{2}\boldsymbol{x}_{tgt\_cln}^{\mathsf{T}}\boldsymbol{Q}\boldsymbol{x}_{tgt\_cln} - \frac{1}{2}\boldsymbol{x}_{tgt}^{\mathsf{T}}\boldsymbol{Q}\boldsymbol{x}_{tgt} \\ &\quad + \frac{1}{2}\boldsymbol{x}_{tst\_cln}^{\mathsf{T}}\boldsymbol{Q}\boldsymbol{x}_{tst\_cln} - \frac{1}{2}\boldsymbol{x}_{tst}^{\mathsf{T}}\boldsymbol{Q}\boldsymbol{x}_{tst} \\ &\quad + \boldsymbol{x}_{tgt\_cln}^{\mathsf{T}}\boldsymbol{P}\boldsymbol{x}_{tst\_cln} - \boldsymbol{x}_{tgt}^{\mathsf{T}}\boldsymbol{P}\boldsymbol{x}_{tst}. \end{aligned} \tag{9}$$

Note the difference between Eq. 9 and the bilinear transformation in Eq. 5. The score shift in Eq. 9 involves not only a bilinear transformation between the target-speaker and test i-vectors in its last term, but also the bilinear transformation of clean and noisy test i-vectors. If we were to know the clean test i-vector ($\boldsymbol{x}_{tst\_cln}$) for every noisy test i-vector ($\boldsymbol{x}_{tst}$), then Eq. 9 can be easily computed without a DNN. However, as we do not know $\boldsymbol{x}_{tst\_cln}$, we resort to relying on the DNN to learn the complex relationship between the input i-vector pairs and the score shifts.

B. DNN Score Transformation: Recovering Clean PLDA Scores by DNNs

The score calibration method in Section III-A and previous scoring methods such as QMF and FQE use the concept of score shift to compensate or calibrate the scores. However, if the clean score can be restored, the estimation of score shifts seems to be redundant. To make the restored scores close to the

[1] In some studies [26], [27], the term *calibration* strictly referred to the process of adjusting the scores without affecting the equal error rate (EER). Here, we follow the terminology in [20]–[22] and relax the definition of calibration to include the processes that lead to better EER performance.

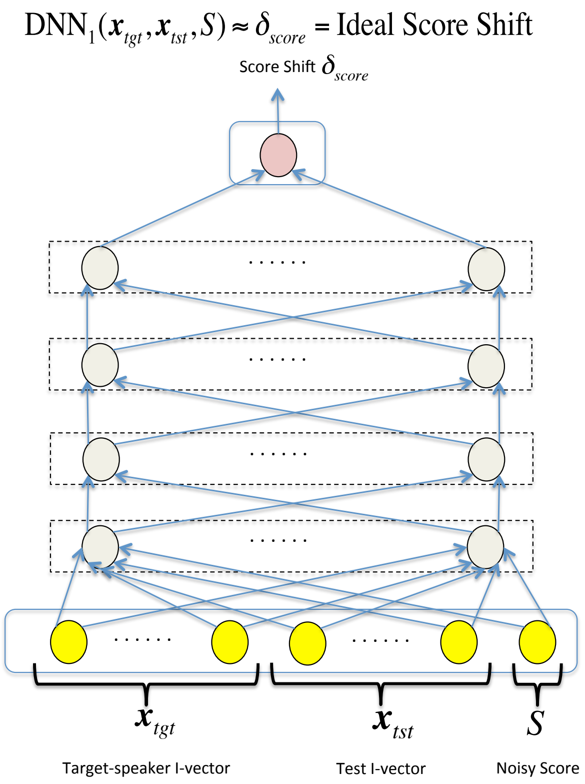
$\text{DNN}_1(\boldsymbol{x}_{tgt}, \boldsymbol{x}_{tst}, S) \approx \delta_{score}$ = Ideal Score Shift
Score Shift δ_{score}
......
$\boldsymbol{x}_{tgt}$
$\boldsymbol{x}_{tst}$
S
Target-speaker I-vector
Test I-vector
Noisy Score

Fig. 3. Single-task DNN with score shift as output.

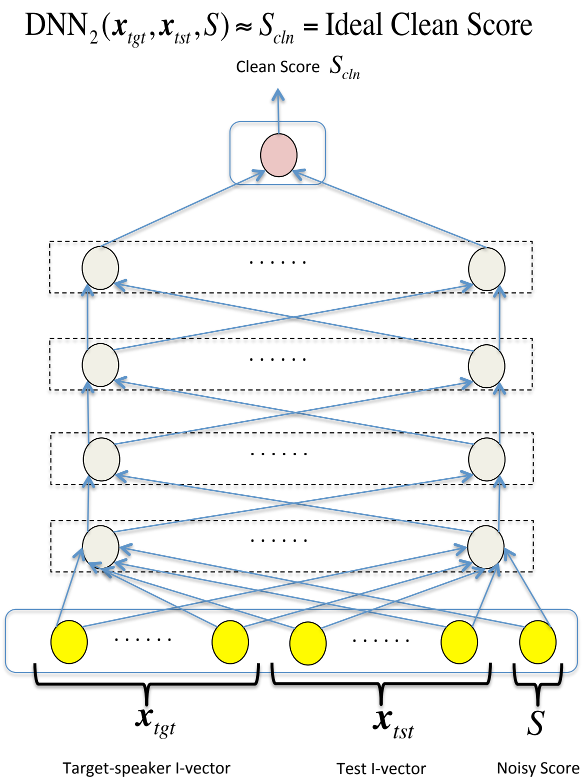

Fig. 4. Single-task DNN that recovers the clean scores.

ideal clean scores, we can use a DNN to model the complex relationship between the i-vector pairs, noisy scores (S), and the clean scores (S_{cln}):

$$S_2' = \text{DNN}_2(\boldsymbol{x}_{tgt}, \boldsymbol{x}_{tst}, S) \approx S_{cln}. \tag{10}$$

In this model, the DNN (see Fig. 4) receives an i-vector pair and its corresponding noisy score as input, and it is trained to output the clean score.

C. Multi-task DNNs for Score Compensation/Transformation

The DNNs in Fig. 3 and Fig. 4 have hundreds of input nodes but only one output node. Their goal is to learn a regression task to produce the desired score shifts or clean scores. During training, the squared errors in the output node will need to be propagated to hundreds of nodes in both the hidden and input layers. Our experience is that having a single source of errors makes the backpropagation (BP) of error gradients very inefficient. One possible solution to assisting the network to learn the regression task is to introduce some auxiliary tasks for the network to learn. In the literature, this is known as multi-task learning [30], [31]. Therefore, a multi-task DNN with auxiliary information in the output layer may help to improve the learning efficiency.

Fig. 5 shows a DNN that uses multi-task learning to learn not only the main task (producing score shift δ_{score} and clean score S_{cln}) but also the auxiliary tasks (producing the SNRs of target-speaker and test utterances and same-speaker and different-speaker posteriors). To incorporate the auxiliary information, we may add auxiliary nodes to either the input layer, the output layers, or both. However, we opt for adding the auxiliary nodes to the output layers rather than to the input layer for four reasons:

1) Adding more input nodes will require the error signals from the error source to be propagated to more input nodes, which is contradictory to our goal of easing the BP training.
2) Given the large number of input nodes corresponding to the i-vector pairs, the squared errors in the output node will mainly depend on the i-vectors rather than the small number of auxiliary inputs. As a result, BP training will tend to find a network that is insensitive to the variability in the auxiliary inputs.
3) Adding auxiliary nodes to the input layer means that it is necessary to estimate this information during the calibration stage. While some information such as SNRs of utterances can be easily estimated, others such as whether the input i-vector pair belongs to the same person or not (see Fig. 5) is not that trivial. In fact, the latter is the goal of the application in the first place.
4) Having more output nodes means that more error signals can be propagated to the hidden layers. The error signals from the auxiliary tasks can guide the network to learn the main task [32]. They also serve as a regularizer to avoid overfitting the main task [30].

Both the clean score S_{cln} and ideal score shift δ_{score} can be the target outputs of the multi-task DNN. Once the multi-task DNN has been trained, the calibrations defined in Eq. 6 and Eq. 10 can be obtained from the output of this DNN.

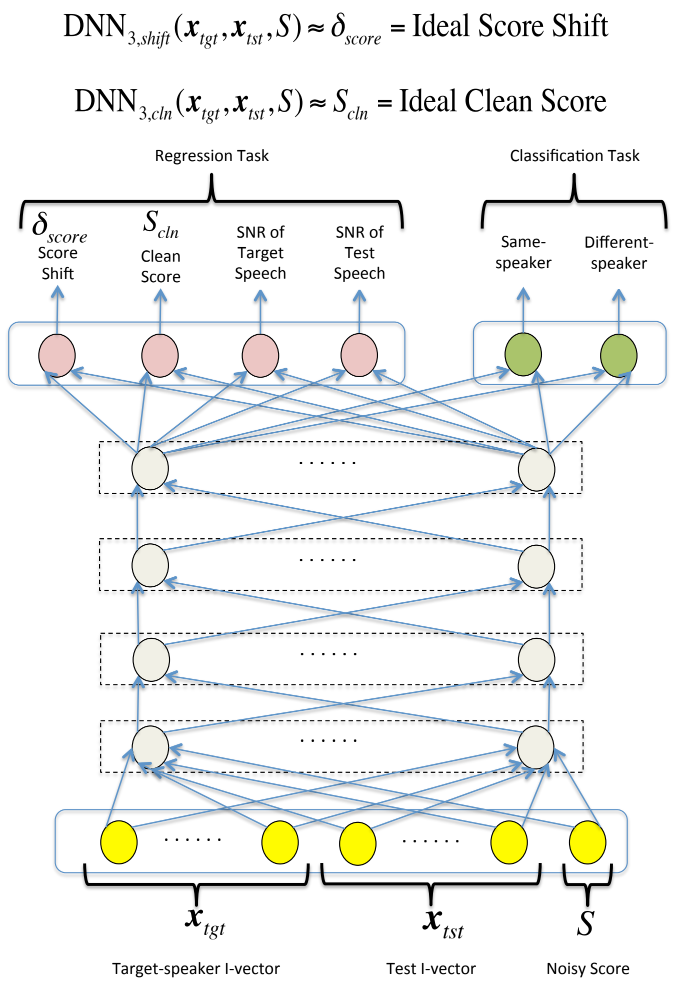
DNN$_{3,shift}(\boldsymbol{x}_{tgt},\boldsymbol{x}_{tst},S)\approx\delta_{score}$ = Ideal Score Shift
DNN$_{3,cln}(\boldsymbol{x}_{tgt},\boldsymbol{x}_{tst},S)\approx S_{cln}$ = Ideal Clean Score
Regression Task
Classification Task
δ_{score} Score Shift
S_{cln} Clean Score
SNR of Target Speech
SNR of Test Speech
Same-speaker
Different-speaker
......
$\boldsymbol{x}_{tgt}$
$\boldsymbol{x}_{tst}$
S
Target-speaker I-vector
Test I-vector
Noisy Score

Fig. 5. Multi-task DNN with classification and regression capacities.

Besides, according to Eq. 3, the SNRs of the target-speaker's utterance and the test utterance, SNR$_{tgt}$ and SNR$_{tst}$, are useful for estimating the score shift. Therefore, we have 4 output nodes in the regression task as shown in Fig. 5. In addition to the regression task, a classification task can be added. Because our goal is to verify speakers, two classification output nodes indicating whether the input i-vector pair is from the same speaker or not is added to the network. In this paper, the regression part of the DNN uses linear output nodes and minimum mean squared error as the optimization criterion, whereas the classification part uses softmax outputs and cross-entropy as the optimziation criterion.

The outputs of a multi-task DNN with 4 regression nodes and 2 classification nodes are the concatenation of two vectors:

$$\begin{aligned}&\mathrm{DNN}_3(\boldsymbol{x}_{tgt},\boldsymbol{x}_{tst},S)\approx\\&\Big[\underbrace{[\delta_{score},S_{cln},\mathrm{SNR}_{tgt},\mathrm{SNR}_{tst}]}_{\text{Regression}},\ \underbrace{[p^+,p^-]}_{\text{Classification}}\Big]^\mathsf{T},\end{aligned}\tag{11}$$

where p^+ and p^- are the posterior probabilities of same-speaker and different-speaker hypotheses, respectively. Similar to the notation in Eq. 7, Eq. 11 means that the DNN uses the i-vector pair $(\boldsymbol{x}_{tgt},\boldsymbol{x}_{tst})$ and the original score S as input. With the multi-task learning strategy, the network outputs the score shift δ_{score}, the clean score S_{cln}, the SNRs of target-speaker speech and test speech SNR$_{tst}$, and the posterior probabilities (p^+ and p^-).

During score compensation and transformation, only the score shift and clean score produced by the multi-task DNN will be used:

$$\mathrm{DNN}_{3,shift}(\boldsymbol{x}_{tgt},\boldsymbol{x}_{tst},S)\approx\delta_{score},\tag{12}$$

and

$$\mathrm{DNN}_{3,cln}(\boldsymbol{x}_{tgt},\boldsymbol{x}_{tst},S)\approx S_{cln}.\tag{13}$$

Therefore, we have

$$\begin{aligned}S'_3&=S+\mathrm{DNN}_{3,shift}(\boldsymbol{x}_{tgt},\boldsymbol{x}_{tst},S)\\&\approx S_{cln},\end{aligned}\tag{14}$$

and

$$\begin{aligned}S'_4&=\mathrm{DNN}_{3,cln}(\boldsymbol{x}_{tgt},\boldsymbol{x}_{tst},S)\\&\approx S_{cln}.\end{aligned}\tag{15}$$

D. Producing Likelihood-Ratio Scores

Our goal is to use DNNs to estimate the ideal clean scores or ideal score shifts in order to improve the performance in terms of equal error rate (EER) and minimum detection cost (minDCF). Therefore, the DNNs are not trained to produce *true* likelihood ratios so that the decision thresholds for which these performance metrics are minimized are application dependent.

For real-world deployments, it is desirable to have application-independent decision thresholds [33] such that not only the EER and minDCF are minimized, but also the actual DCF (actDCF) or C_{primary} at specific thresholds are also small. To this end, all of the *compensated/transformed* scores are subject to further calibration to produce true likelihood-ratio scores using the logistic regression (LR) in the Bosaris toolkit [34]:

$$S''=w_0+w_1S'_i,\tag{16}$$

where S'_i, $i=1,2,3$, and 4, are the compensated/transformed scores in Eqs. 6, 10, 14 and 15, respectively. This calibration step only shifts and scales the DNN calibrated scores, which reduce the actDCF without affecting the EER and minDCF.

IV. Experiments

A. Experimental Setup

Score calibration experiments were conducted on the NIST 2012 SRE under Common Condition 4 (CC4, male). This evaluation condition involves 723 male target speakers, with a total of 7,116 telephone utterances for enrollment and 3,900 telephone utterances for performance evaluation. Totally, there are 2,775 target-speaker trials and 122,624 impostor trials. The duration of these utterances ranges from 10 to 300 seconds (before voice activity detection (VAD)). All utterances were spoken in English. These utterances cover a wide range of SNR, from 0dB to 30dB as shown in Fig. 6.

To investigate the capability of various calibration methods under noisy environments, we used the FaNT tool [28] to add babble noise to the target-speaker utterances and test utterances at an SNR of 15dB, 6dB, and 0dB, respectively. Therefore, we have four groups of training utterances and four groups of test utterances, with the first group being the original utterances and the last three groups having SNRs close to 15dB, 6dB, and 0dB, respectively. Hereafter, we refer to these 4 groups as SNR groups. The SNR distributions of the 4 groups of test utterances in CC4 are shown in Fig. 6. Note that although the target SNRs that we applied to FaNT are 0dB, 6dB, and 15dB, Fig. 6 shows that the peaks of the SNR distributions do not align to these targets. The misalignments are due to the discrepancy in the VAD decisions for adding noise and for measuring SNRs. Specifically, FaNT has its own VAD for estimating the amount of noise to be added to the clean signals, whereas the *measured* SNRs in Fig. 6 were based on the voltmeter function in FaNT and the decisions of our noise-robust VAD [35].

The whole training set comprises $7116 \times 4 = 28,464$ target-speaker utterances from 723 target speakers in CC4 of NIST 2012, leading to $28464^2 \approx 810$ million i-vector pairs. Using the procedure shown in Fig. 2, these i-vector pairs give rise to 810 million PLDA scores and score shifts. A random subset of these scores and score shifts were selected for training the DNNs in Fig. 3 to Fig. 5 (see Section IV-B) and for estimating the calibration weights in Eqs. 1, 3 and 16. To ensure that all DNNs were trained by the same set of i-vector pairs and PLDA scores, only one PLDA model was trained. Specifically, it was trained by using the utterances from the 4 SNR groups mentioned earlier and the i-vectors derived from the microphone utterances (interview speech) of the same set of target speakers in NIST 2006–2010 SREs.

The evaluation protocol of the NIST 2012 SRE defines the target trials and impostor trials (in the .ndx files). For each trial, a target speaker is defined but not his/her enrollment utterances. It is up to the evaluator to select the appropriate enrollment utterances for each trial. Because CC4 in 2012 SRE involves noise contaminated test utterances, we used the original target-speaker utterances and their noise contaminated counterparts from the 6dB and 15dB SNR groups as enrollment utterances. In the sequel, we refer to this test condition as "original". In addition to this "original" test condition, we created three test conditions based on the noise contaminated test utterances. Specifically, for the 15dB test condition, test utterances at the SNR of 15dB were used for scoring, and similarly for the 6dB and 0dB test conditions. The enrollment utterances were different for different test conditions. Specifically, for the 15dB test condition, the enrollment utterances were obtained from the 15dB and 6dB SNR groups; for the 6dB and 0dB test conditions, the enrollment utterances were respectively obtained from the 6dB and 0dB SNR groups.

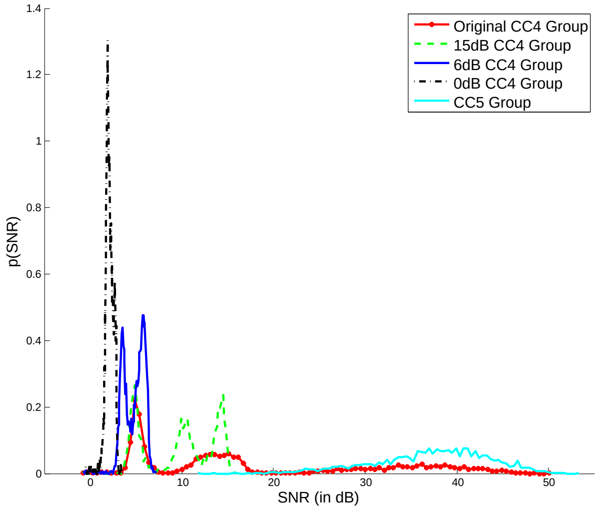

Fig. 6. The SNR distributions of the original and noise contaminated test utterances in NIST 2012 SRE (CC4 and CC5, male). For the noise contaminated utterances, babble noise was added to the original utterances at an SNR of 0dB, 6dB, and 15dB, respectively.

The weights of the QMF in Eq. 3 and the calibration weights in Eq. 1 and Eq. 16 were trained by using 1.5 million same-speaker utterance pairs and 400 million different-speaker utterance pairs derived from the target speakers in the four SNR groups. For Eq. 1 and Eq. 3, using the logistic regression program in the FoCal toolkit, we obtained the weights $w_0 = -21.5197$, $w_1 = 0.1966$, $w_2 = 0.1284$ and $w_3 = 0.1284$, leading to:

$$S' = 0.1966S + 0.1284\text{SNR}_{tst} + 0.1284\text{SNR}_{tgt} - 21.5197.$$

Speech regions in the speech files were extracted by using a two-channel voice activity detector [35]. 19 MFCCs together with energy plus their 1st and 2nd derivatives were extracted from the speech regions, followed by cepstral mean normalization and feature warping with a window size of 3 seconds. A 60-dim acoustic vector was extracted every 10ms, using a Hamming window of 25ms.

B. DNN Training

To highlight the advantages of multi-task learning, a multi-task DNN (Fig. 5) that implements Eq. 12 and Eq. 13 was compared with two single-task DNNs (Fig. 3 and Fig. 4) that implement Eq. 6 and Eq. 10. For the single-task DNNs, we trained restricted Boltzmann machines (RBM) layer-by-layer using the contrastive divergence algorithm [36], [37]; then we fine-tuned the networks using the backpropagation algorithm with sigmoid nonlinearity in the hidden layers. Mini-batch gradient descent with a batch size of 1000 was used. The learning rates for the classification task and regression task are 0.005 and 0.05, respectively. Because there are over 810 million i-vectors pairs that can be used for training, to speed up the training process, 3.2 million pairs were randomly chosen for every 10 iterations of backpropagation training. Totally, we applied 200 iterations of backpropagation to fine-tune the networks.

The RBM at the bottom layer has Gaussian visible nodes and Bernoulli hidden nodes. The remaining RBMs use Bernoulli distributions in both visible and hidden layers. Both the inputs and desired outputs (except for the classification outputs in Fig. 5) of the DNNs were processed by z-normalization. The last layer of the classification part in Fig. 5 were initialized with small random weights. All DNNs have 4 hidden layers, with each layer comprising 256 hidden nodes.

The multi-task DNN (DNN$_3$) was trained in a slightly different manner. Because having a balanced training set is beneficial for the network to learn the classification task, for every iteration we extracted 700,000 same-speaker i-vector pairs and 700,000 different-speaker i-vector pairs for training. We applied 300 iterations of backpropagation to fine-tune DNN$_3$.

C. Denoised Senone I-vectors

We used a senone i-vector/PLDA system [6] to produce the uncalibrated (noisy) scores, which form our baseline results. The system is equipped with a denoising deep classifier that extracts frame-based bottleneck features from the MFCCs of utterances. The deep classifier is formed by stacking two layers of RBMs on top of a denoising autoencoder (DAE) [38]. This structure allows us to extract bottleneck features and estimate the posterior of senones for i-vector extraction [39], which effectively incorporates phonetic information into the senone i-vectors.

We followed the standard procedure to pre-process the i-vectors for Gaussian PLDA modeling. Specifically, the 500-dimensional senone i-vectors were whitened by within-class covariance normalization (WCCN) [40] and length normalization [29], followed by linear discriminant analysis to reduce the dimension to 200 and variance normalization by WCCN [41]. These 200-dimensional i-vectors were input to the PLDA model and the DNNs.

V. Results and Discussions

A. Distributions of PLDA Scores and Score Shifts

To investigate the property of PLDA scores under various background noise levels, we scored clean target-speaker i-vectors against noisy test i-vectors and plotted the distributions of the resulting scores. Fig. 7 shows the distributions of these uncalibrated scores under four background noise levels of test utterances: Clean, 15dB, 6dB, and 0dB. Evidently, the scores tend to be larger and their variances tend to be smaller when the background noise level increases.

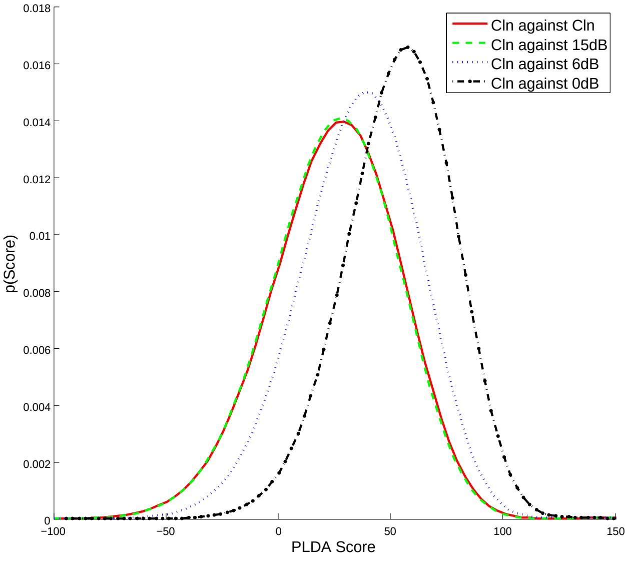

Fig. 7. The distributions of PLDA scores produced by scoring clean target-speaker i-vectors against noisy test i-vectors. Each distribution corresponds to one group of test i-vectors whose utterances have SNRs belonging to one of the four groups: Clean (Cln), 15dB, 6dB, and 0dB.

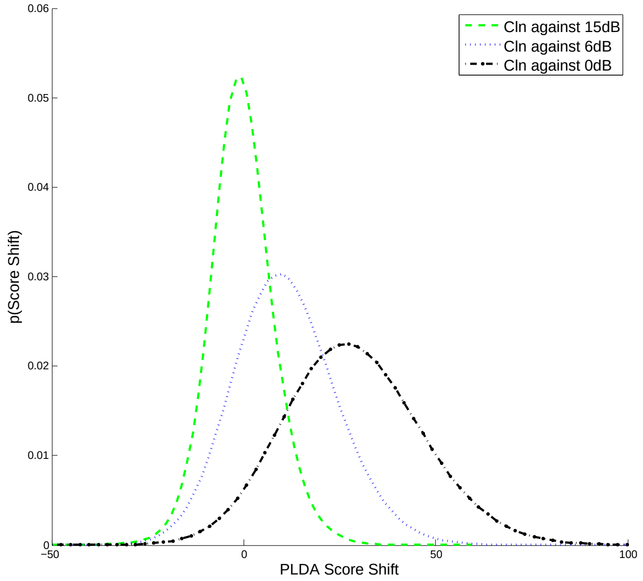

Fig. 8. The distributions of ideal score shifts ($S - S_{cln}$) for 3 SNR conditions of the test utterances. As indicated in the legend, the target-speaker utterances are always clean. The clean scores S_{cln} were obtained by scoring clean target-speaker i-vectors against clean test i-vectors.

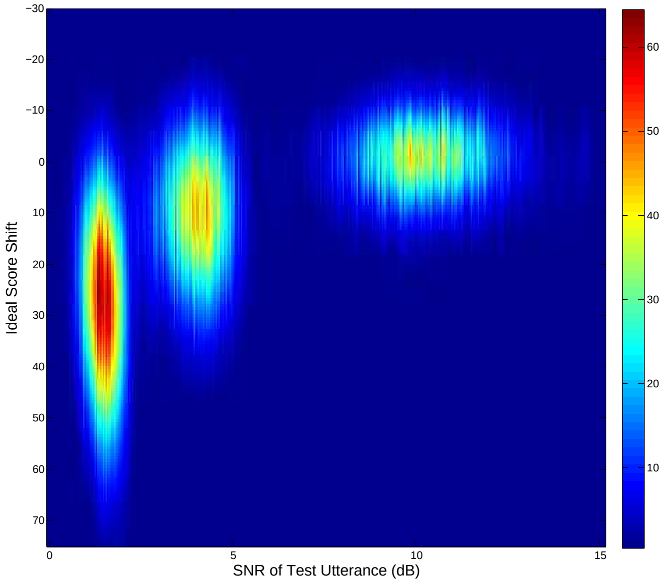

Fig. 9. The distributions of score shifts with respect to the SNR of test utterances when the target-speaker utterances is clean. The SNRs follow the distribution shown in Fig. 6, and the score shifts follow the distribution shown in Fig. 8. The figure shows that the relationship between SNRs and score shifts is non-linear, and that at low SNR, the variability of score shifts is very large.

Because our goal is to use DNNs to compute the ideal score shifts, it is of interest to inspect the relationship between the ideal score shifts and test utterances' SNR. To this end, we plot the distributions of ideal score shifts ($S - S_{cln}$) under three SNR conditions for the test utterances in Fig. 8 and against all SNRs in the test utterances in Fig. 9. Interestingly, the

TABLE I

Performance of various score calibration methods in NIST 2012 SRE (CC4, male speaker, core task) with test utterances contaminated with different levels of babble noise. All networks have 4 hidden layers. LR: Logistic regression.

	Original			15dB		
Score Calibration Method	EER(%)	minDCF	actDCF	EER(%)	minDCF	actDCF
Baseline (LR on noisy PLDA scores)	1.56	0.218	0.855	2.27	0.225	0.778
SNR-dep Score Shift (Eq. 3)	1.68	0.209	0.780	2.24	0.215	0.770
Score Shift by Multi-task DNN (Eq. 12)	1.65	**0.178**	0.694	2.32	**0.203**	0.626
Recovered Clean Score by Multi-task DNN (Eq. 13)	**1.50**	0.189	**0.517**	**2.21**	0.211	**0.455**
	6dB			0dB		
Score Calibration Method	EER(%)	minDCF	actDCF	EER(%)	minDCF	actDCF
Baseline (LR on noisy PLDA scores)	2.29	0.276	0.749	5.37	0.753	0.779
SNR-dep Score Shift (Eq. 3)	2.28	0.269	0.811	5.35	0.754	0.794
Score Shift by Multi-task DNN (Eq. 12)	2.34	**0.231**	0.604	4.00	0.488	0.547
Recovered Clean Score by Multi-task DNN (Eq. 13)	**2.16**	0.243	**0.470**	**3.48**	**0.409**	**0.516**

TABLE II

Performance of various score calibration methods in NIST 2012 SRE (CC5, male speaker, core task). The DNN has 4 hidden layers. LR: Logistic regression.

	Original		
Score Calibration Method	EER(%)	minDCF	actDCF
Baseline (LR on noisy PLDA scores)	**2.48**	0.267	0.861
Score Shift by Multi-task DNN (Eq. 12)	2.54	0.260	**0.640**
Recovered Clean Score by Multi-task DNN (Eq. 13)	2.51	**0.242**	0.716

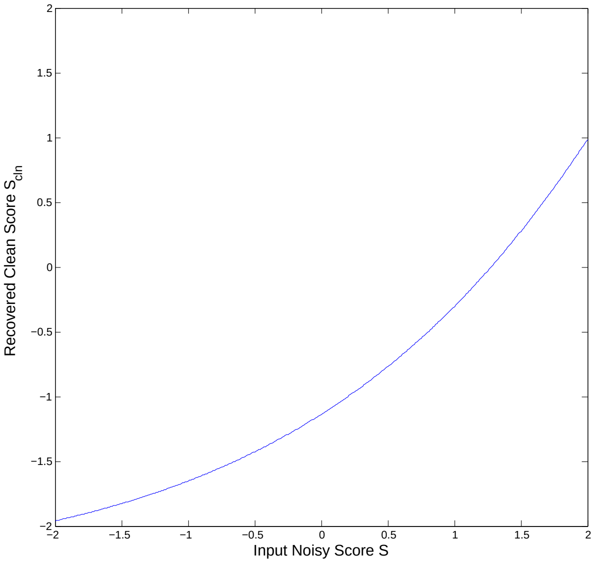

Fig. 10. Graph showing the relationship between the recovered clean scores S'_4 in Eq. 15 produced by a multi-task DNN and the noisy input scores S when the i-vector pair is fixed. The input i-vector pair is a fixed non-target pair. Both S and S'_4 were normalized by the same set of z-norm parameters. The DNN is a multi-task one with 4 hidden layers.

score shifts exhibit a large variability when the SNR of test utterances is very low (0dB). This high variability is definitely not because of the high variability in SNR, as evident in Fig. 6 where the SNR distribution of test utterances is very narrow near 0dB. Quite the opposite, the high SNR variability in the 15dB group shown in Fig. 6 leads to the least variability in score shifts (green dashed curve) in Fig. 8. Therefore, at low SNR, the score shifts will become more difficult to estimate, which demonstrates a major drawback of the methods that entirely rely on SNR of utterances (e.g., QMF in Eq. 3). In theory, the FQE in Eq. 5 is better in the sense that it does not use SNR information directly but instead uses it implicitly through the i-vectors and the Gaussian models. However, whether the bilinear transformation and cosine distance can accurately estimate the score shift at high background noise level is unclear. As demonstrated in Fig. 9, the relationship between score shifts and utterances' SNR are fairly complex and definitely non-linear.

Because i-vectors are noise-level dependent [13], it makes sense to directly predict the score shifts from i-vectors rather than implicitly through the Gaussian models of the i-vectors as in FQE. Therefore, we advocate that through multi-task supervised learning, the DNNs can estimate the score shifts accurately and even recover the clean scores. This is supported by the results to be presented next.

B. *Sensitivity of Score Output to Score Input*

For 200-dimensional pre-processed i-vectors, the number of input nodes corresponding to the i-vector pairs is 400, whereas there is only one score input node. This large ratio may cause the network insensitive to the noisy scores. To find out if it is the case, we plotted the recovered clean scores S'_4 (Eq. 15) produced by a multi-task DNN against the input noisy scores S. To focus on the sensitivity of S'_4 with respect to S, the i-vector pair was fixed to an arbitrary non-target pair. Surprisingly, as shown in Fig. 10, the recovered scores are fairly sensitive to the noisy input scores despite of the large ratio between the two types of input nodes. This result, in fact, agrees with our recent finding that the input noisy scores play an important role in recovering the clean scores [42].

C. *Results on NIST 2012 SRE*

1) Comparing Various Calibration Methods: Table I shows the performance of various score calibration strategies, including SNR-dependent score shifts (Eq. 3) and recovering clean scores by multi-task DNNs (Eq. 13). The baseline refers to using the noisy PLDA scores for computing EER and

TABLE III

Performance of single-task and multi-task DNNs for estimating the score shifts and recovering the clean score in NIST 2012 SRE (CC4, male speaker, core task) with test utterances contaminated with different levels of babble noise. All networks have 4 hidden layers.

	Original		15dB		6dB		0dB	
Score Calibration Method	EER(%)	minDCF	EER(%)	minDCF	EER(%)	minDCF	EER(%)	minDCF
Score Shift by Single-task DNN (Eq. 7)	2.26	0.267	3.25	0.285	3.27	0.322	4.83	0.538
Score Shift by Multi-task DNN (Eq. 12)	**1.65**	**0.178**	**2.32**	**0.203**	**2.34**	**0.231**	**4.00**	**0.483**
Recovered Clean Score by Single-task DNN (Eq. 10)	7.35	0.693	8.11	0.643	8.06	0.733	12.25	0.989
Recovered Clean Score by Multi-task DNN (Eq. 13)	**1.50**	**0.189**	**2.21**	**0.211**	**2.16**	**0.248**	**3.48**	**0.409**

TABLE IV

Performance of single-task and multi-task DNNs with different numbers of hidden layers for estimating the score shifts (Eq. 7 and 12) in NIST 2012 SRE (CC4, male speaker, core task). The test utterances were contaminated with different levels of babble noise.

		Original		15dB		6dB		0dB	
Network Type	No. of Hidden Layers	EER(%)	minDCF	EER(%)	minDCF	EER(%)	minDCF	EER(%)	minDCF
Single-task	1	2.39	0.322	3.49	0.342	3.65	0.413	5.75	0.676
	2	**2.14**	**0.264**	**3.21**	0.298	**3.18**	0.339	4.87	0.575
	3	2.50	0.296	3.70	0.330	3.79	0.398	5.68	0.632
	4	2.26	0.267	3.25	**0.285**	3.27	**0.322**	**4.83**	**0.538**
Multi-task	1	**1.55**	0.202	2.16	0.208	2.21	0.242	4.49	0.624
	2	**1.55**	0.187	2.17	**0.197**	2.14	**0.228**	4.04	0.554
	3	**1.55**	0.193	**2.11**	0.201	**2.08**	0.239	4.20	0.571
	4	1.65	**0.178**	2.32	0.203	2.34	0.231	**4.00**	**0.488**

TABLE V

Performance of single-task and multi-task DNNs with different numbers of hidden layers for recovering the direct clean scores (Eq. 10 and 13) in NIST 2012 SRE (CC4, male speaker, core task). The test utterances were contaminated with different levels of babble noise.

		Original		15dB		6dB		0dB	
Network Type	No. of Hidden Layers	EER(%)	minDCF	EER(%)	minDCF	EER(%)	minDCF	EER(%)	minDCF
Single-task	1	**3.79**	**0.634**	**4.89**	**0.580**	**5.10**	**0.679**	**9.09**	**0.957**
	2	5.02	0.719	5.93	0.675	6.38	0.750	10.81	0.980
	3	7.81	0.779	8.86	0.741	9.52	0.818	14.27	0.997
	4	14.61	0.910	15.41	0.881	17.05	0.942	21.44	1.002
Multi-task	1	1.50	0.234	**2.08**	0.222	**2.11**	0.278	4.48	0.605
	2	**1.48**	0.199	2.37	**0.210**	2.14	**0.241**	3.83	0.502
	3	1.71	**0.187**	2.53	0.213	2.34	0.243	3.58	0.448
	4	1.50	0.189	2.21	0.211	2.16	0.248	**3.48**	**0.409**

minimum detection cost function (minDCF) and using logistic regression for computing the actual DCF (actDCF). The results show that the proposed method achieves the best performance across all of the SNR levels. At 0dB, it also outperforms the baseline significantly.

Fig. 11 shows the normalized Bayes error rates of the minDCF and actDCF of various systems as a function of effective target prior. Among all systems, the one based on score shift computed by the multi-task DNN (green) has a very small margin between the actDCF and minDCF.

The results of the same experiments on CC5 of NIST 2012 SRE are shown in Table II. Unlike the results in CC4, DNN calibration does not show obvious advantage as compared to linear calibration. We suspect that this is because most of the utterances in CC5 have higher SNRs than those in CC4 (see Fig. 6). When the SNR is high, the benefit of DNN score calibration diminishes. Also, since the utterances in CC5 were collected in a noisy environment while the DNN was trained by data with artificially added noise, the DNN may be weak to deal with the natural noises and the Lombard effect.

2) Single-task vs. Multi-task: Table III compares the performance between single-task DNNs and multi-task DNNs. Regardless of which output to be used (the score shift output in Eq. 6 or the direct score output in Eq. 10), the multi-task DNN performs much better than the single-task DNN.

A comparison between the first row of Table III (Score Shift by Single-Task DNN) and the first and second rows of Table I reveals that only under very noisy conditions (0dB), the calibrated scores produced by the single-task DNN (Eq. 7) can improve performance; in all other conditions, this approach performs even poorer than the ones without score calibration or the conventional approach where the score shift is linear with respect to utterances' SNR. This result suggests that estimating the score shifts *entirely* from the i-vector pairs and noisy scores (Eq. 7) is not a good idea. Under the clean condition, the score shifts estimated by the DNN in Eq. 7 have detrimental effect on the uncalibrated scores. However, the good performance in the second row of Table III suggests that once the auxiliary

TABLE VI

Performance of various score calibration methods in a subset of NIST 2012 SRE (CC4, male speaker, core task) with test utterances contaminated with different levels of babble noise. The speech from 500 speakers was used to train the multi-task DNN as in Fig. 5, and the 38,820 trials in CC4 of the other 223 speakers were used in the test trials. The DNN has 4 hidden layers. LR: Logistic regression.

	Original			15dB		
Score Calibration Method	EER(%)	minDCF	actDCF	EER(%)	minDCF	actDCF
Baseline (LR on noisy PLDA scores)	1.21	0.194	0.867	1.96	0.214	0.794
Score Shift by Multi-task DNN (Eq. 12)	0.94	**0.168**	**0.751**	1.96	**0.206**	**0.698**
Recovered Clean Score by Multi-task DNN (Eq. 13)	**0.82**	0.188	0.777	**1.57**	0.214	0.726
	6dB			0dB		
Score Calibration Method	EER(%)	minDCF	actDCF	EER(%)	minDCF	actDCF
Baseline (LR on noisy PLDA scores)	1.89	**0.266**	0.755	5.09	0.709	0.722
Score Shift by Multi-task DNN (Eq. 12)	1.65	0.274	**0.673**	**3.67**	**0.507**	**0.634**
Recovered Clean Score by Multi-task DNN (Eq. 13)	**1.55**	0.309	0.728	3.70	0.548	0.704

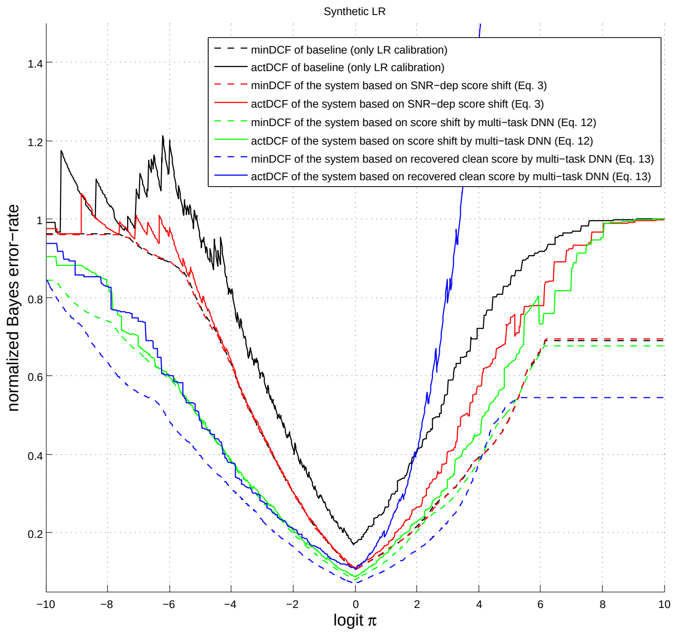

Fig. 11. Normalized bayes error rate plots showing that the minDCF and actDCF of different systems as a function of effective target prior in NIST 2012 SRE (CC4, male speaker, core task) contaminated with babble noise at an SNR of 0dB.

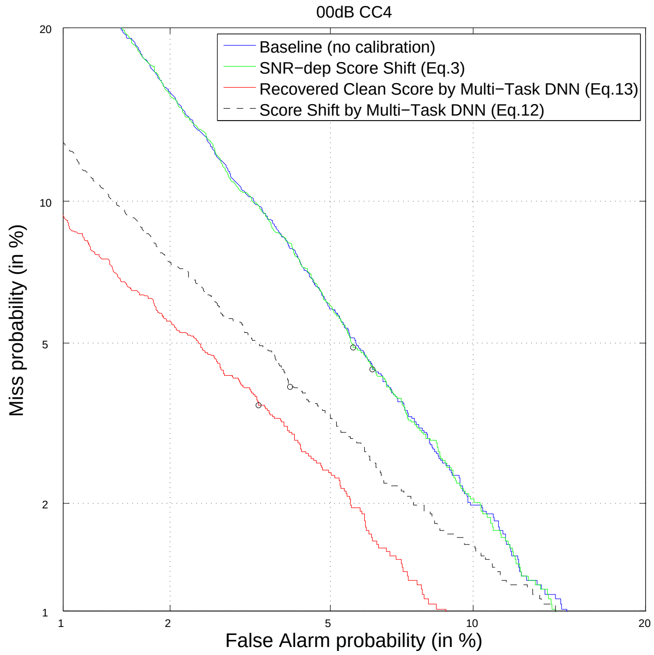

Fig. 12. The DET curves of the four systems in NIST 2012 SRE (CC4, male speaker, core task). Test utterances were contaminated with babble noise at an SNR of 0dB.

information is added to the network, the score shifts estimated by the multi-task DNN become very close to the ideal ones.

Table III also allows us to compare the performance of using the DNN to estimate the score shift (δ_{score}, Fig. 3) and using the DNN to recover the clean scores (S_{cln}, Fig. 4). Specifically, for the single-task case, Row 1 and Row 3 of Table III show that the scores recovered by the DNNs are so wrong that the error is 3 to 4 times that of the baseline (c.f. Table I and the third row of Table III), while the score shifts by the DNNs are comparable to the baseline (c.f. Table I and the first row of Table III). For the multi-task case, Row 2 and Row 4 of Table III suggest that it is better to recover the clean scores directly than to estimate the score shifts, provided that multi-task learning is used to train the DNNs.

Fig. 12 compares the DET performance of the baseline against the multi-task DNNs. The results clearly suggest that recovering the clean scores by DNNs leads to superior performance across a wide range of decision thresholds.

3) Generalization Capability: Using the i-vectors from target speakers to train the PLDA model and the calibration DNNs may lead to overfitting on target speakers. In NIST 2012 SRE, the enrollment utterances of target speakers come from previous NIST SREs. As these utterances were also used for training the PLDA model and the DNN, there may be chance that they can only work for these target speakers. To demonstrate the generalization capability of the multi-task DNNs, we selected 500 target speakers from CC4 and used their i-vectors and PLDA scores (scores between same targets and between different targets) to train a multi-task DNN as in Fig. 5. The target and non-target trials (totally 38,820) derived from the remaining 223 target-speakers were then used for performance evaluation. To the PLDA model and the DNN, these 223 speakers are unseen speakers. As shown in Table VI, the multi-task DNNs perform significantly better than the baseline. More importantly, the performance in Table VI is comparable to and in many cases better than that in Table I. This suggests that both the PLDA model and DNN can generalize to unseen speakers and that using the enrollment utterances for training does not lead to overfitting.

4) Different Numbers of Hidden Layers: All of the DNNs in Tables I and III comprise four hidden layers. It is of interest to see how they perform if the number of hidden layers varies. Table IV and Table V show the performance of single-task and multi-task DNNs with different number of hidden layers for estimating the score shifts and recovering the clean scores, respectively. Theoretically, a network with a deeper structure should posses a larger capacity to perform the mapping task. However, the results do not show such trend, especially for the single-task DNNs in Table IV where they were used for estimating the score shifts. In particular, the single-task DNN with 3 hidden layers performs worse than those with 2 and 4 hidden layers. In Table V, the performance of single-task DNNs (for recovering clean scores) becomes worse when the number of hidden layers increases. But this phenomenon does not occur in the multi-task DNNs. Further investigations on the error profiles during the BP training process reveal that the training errors of single tasks DNNs increase when the number of hidden layers increases. This contradicts to the common belief that networks with a higher capacity should produce a lower training error. One possible cause of this contradiction is that the networks are stuck at bad local minima. The multi-task DNNs do not suffer from this problem, primarily because the auxiliary output nodes can introduce *additional* errors to help the network to learn the main task [30].

As discussed in [43], when a DNN becomes deeper, its accuracy may get saturated; further increase in the depth will lead to performance degradation. This phenomenon can also be observed in Table V. In our case, this phenomenon is not caused by overfitting, because adding more layers leads to higher training error; instead, vanishing gradients and slow convergence are more likely to be the cause. One possible way to alleviate this difficulty is to use residual networks [43]. A special property of residual networks is that they are trained to produce the difference between the desired mapping function $f(\mathbf{x})$ and the input $\mathbf{x}$ instead of producing $f(\mathbf{x})$. With this special arrangement, networks can enjoy performance gain from increased depth. Eq. 7 is similar to the residual function in [43], where the score shift δ_{score} is indeed the residual between the desired clean score S_{cln} and the input noisy score S. This explains why the DNNs that estimate score shifts (Fig. 3) are easier to train than the ones that recover clean scores (Fig. 4).

VI. Conclusions and Future Work

In this paper, we proposed several DNN-based score calibration algorithms, where the calibrated scores and score shifts are estimated from the i-vector pairs of verification trials. The conventional calibration methods such as quality measure functions (QMF) assume that score shift caused by background noise is a linear function of the utterance's SNR. Our results, however, suggest that the score shift is nonlinearly related to SNR. Also, QMFs are deterministic functions of SNR in that when the SNR is fixed, the score shift will also be fixed. Our results, however, show that the lower the SNR, the larger the variability in the ideal score shift, which suggests that SNR alone is not adequate for estimating the score shift. These observations motivate us to use more flexible models such as DNNs to model the complex relationship between the i-vector pairs, uncalibrated scores, and score shifts. This paper has shown that DNNs are flexible enough for recovering the clean PLDA scores directly, allowing us to skip the score-shift estimation entirely. By introducing auxiliary tasks to the DNNs through multi-task learning, we demonstrated that the resulting DNNs can learn the main task better, which is supported by their superior performance across a wide-range of SNRs.

Despite the promising results, the proposed methods have some weaknesses. First, the methods are more computationally demanding than the conventional ones because of the DNNs. However, the extra computation time is insignificant when compared with the time for i-vector extraction. This is especially the case for long test utterances as the complexity of i-vector extraction is proportional to the number of speech frames in the utterances. Second, the DNNs are not trained to produce true likelihood ratios; therefore further calibration is essential. Third, the DNNs require clean and noisy utterance pairs for training, whereas conventional methods such as QMFs and FQEs do not have this requirement.

The multi-task DNNs have five auxiliary output nodes, three for the SNR information and two for the speaker information. The current study did not investigate which of the auxiliary information is more important or helpful. Also, there may be other auxiliary information, such as the duration of utterances, that is also useful. These are the interesting directions that are worth investigation in the future.

In this work, both of the training and testing datasets cover a wide range of SNR, meaning that the DNNs were trained to handle test utterances with different SNRs. Because the SNR ranges are the same for both training and testing, the behaviour of the DNNs under unseen SNR is unclear. In future work, it is of interest to use one SNR group for training a DNN and test it on anther SNR group to investigate the generalization capability of the DNN under SNR mismatch conditions. Currently, utterances were contaminated with babble noise only. The robustness of DNN-based calibration for other types of noise and for reverberated speech also are worth investigation.

We adopted the DNN training procedure in [44]. Over the years, a number of advanced training procedures have been developed. For example, the RBM pre-training can be replaced by discriminative pre-training [45]. The stochastic gradient descent can also be enhanced by using adaptive moment estimation (Adam) [46]. It is also possible to replace the sigmoid activation by other nonlinear functions, such as hyperbolic tangent and softsign, to avoid saturation at the top hidden layer [47] or to achieve better performance [48]. Furthermore, it has been found that rectifier nonlinearity, such as ReLU, is beneficial for acoustic modeling [49]. Training can be sped up by using batch normalization [50]. Deeper networks can be trained by using the strategy in residual networks [43]. These modern DNN training methods can potentially improve the performance of the DNN-score calibration techniques proposed in this paper.

References

[1] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 19, no. 4, pp. 788–798, 2011.

[2] S. Prince and J.H. Elder, “Probabilistic linear discriminant analysis for inferences about identity,” in *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, 2007, pp. 1–8.

[3] T. Hasan and J. Hansen, “Acoustic factor analysis for robust speaker verification,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 4, pp. 842–853, 2013.

[4] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P. Manzagol, “Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion,” *The Journal of Machine Learning Research*, vol. 11, pp. 3371–3408, 2010.

[5] J. H. Liu, W. Q. Zheng, and Y. X. Zou, “A robust acoustic feature extraction approach based on stacked denoising autoencoder,” in *Multimedia Big Data (BigMM), IEEE International Conference on*, 2015, pp. 124–127.

[6] Z. L. Tan, Y. K. Zhu, M. W. Mak, and B. Mak, “Senone i-vectors for robust speaker verification,” in *Int. Sym. on Chinese Spoken Language Processing (ISCSLP’16)*, Oct. 2016.

[7] T. Pekhovsky, S. Novoselov, A. Sholohov, and O. Kudashev, “On autoencoders in the i-vector space for speaker recognition,” in *Odyssey*, 2016.

[8] S. Novoselov, T. Pekhovsky, O. Kudashev, V. Mendelev, and A. Prudnikov, “Non-linear PLDA for i-vector speaker verification,” in *Proc. Interspeech*, September 2015, pp. 214–218.

[9] T. Hasan, S. O. Sadjadi, G. Liu, N. Shokouhi, H. Bořil, and J. Hansen, “CRSS systems for 2012 NIST speaker recognition evaluation,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2013, pp. 6783–6787.

[10] D. Garcia-Romero, X. Zhou, and C. Y Espy-Wilson, “Multicondition training of Gaussian PLDA models in i-vector space for noise and reverberation robust speaker recognition,” in *IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2012, pp. 4257–4260.

[11] N. Li and M. W. Mak, “SNR-invariant PLDA modeling in nonparametric subspace for robust speaker verification,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 10, pp. 1648–1659, 2015.

[12] N. Li and M. W. Mak, “SNR-invariant PLDA with multiple speaker subspaces,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 5565–5569.

[13] M. W. Mak, X. M. Pang, and J. T. Chien, “Mixture of PLDA for noise robust i-vector speaker verification,” *Audio, Speech, and Language Processing, IEEE/ACM Transactions on*, vol. 24, no. 1, pp. 130–142, 2016.

[14] N. Li, M. W. Mak, and J. T. Chien, “DNN-driven mixture of PLDA for robust speaker verification,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. PP, no. 99, pp. 1371–1383, 2017.

[15] S. Cumani and P. Laface, “Large-scale training of pairwise support vector machines for speaker recognition,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 11, pp. 1590–1600, 2014.

[16] D. Snyder, P. Ghahremani, D. Povey, D. Garcia-Romero, Y. Carmiel, and S. Khudanpur, “Deep neural network-based speaker embeddings for end-to-end speaker verification,” in *Spoken Language Technology Workshop (SLT), 2016 IEEE*. IEEE, 2016, pp. 165–170.

[17] T. Hasan, R. Saeidi, J. H. L. Hansen, and D. A. van Leeuwen, “Duration mismatch compensation for i-vector based speaker recognition systems,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2013, pp. 7663–7667.

[18] Q. Hong, L. Li, M. Li, L. Huang, L. Wan, and J. Zhang, “Modified-prior PLDA and score calibration for duration mismatch compensation in speaker recognition system,” in *Proc. Interspeech*, 2015.

[19] A. Shulipa, S. Novoselov, and Y. Matveev, “Scores calibration in speaker recognition systems,” in *Speech and Computer: 18th International Conference*, Budapest, Hungary, August 2016, pp. 596–603.

[20] M. I. Mandasari, R. Saeidi, M. McLaren, and D. A. van Leeuwen, “Quality measure functions for calibration of speaker recognition systems in various duration conditions,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 11, pp. 2425–2438, Nov 2013.

[21] M. I. Mandasari, R. Saeidi, and D. A. van Leeuwen, “Quality measures based calibration with duration and noise dependency for speaker recognition,” *Speech Communication*, vol. 72, pp. 126 – 137, 2015.

[22] A. Nautsch, R. Saeidi, C. Rathgeb, and C. Busch, “Robustness of quality-based score calibration of speaker recognition systems with respect to low-SNR and short-duration conditions,” in *Odyssey*, 2016, pp. 358–365.

[23] A. Ortega J. Villalba, A. Miguel and E. Lleida, “Bayesian networks to model the variability of speaker verification scores in adverse environments,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 12, pp. 2327–2340, 2016.

[24] L. Ferrer, L. Burget, O. Plchot, and N. Scheffer, “A unified approach for audio characterization and its application to speaker recognition,” in *Odyssey*, 2012, pp. 317–323.

[25] N. Brümmer, A. Swart, and D. van Leeuwen, “A comparison of linear and non-linear calibrations for speaker recognition,” in *Odyssey*, 2014, pp. 14–18.

[26] N. Brümmer and G. Doddington, “Likelihood-ratio calibration using prior-weighted proper scoring rules,” in *Proc. Interspeech*, 2013, pp. 1976–1980.

[27] N. Brümmer and D. Garcia-Romero, “Generative modelling for unsupervised score calibration,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2014, pp. 1680–1684.

[28] H. Hirsch, “Fant-filtering and noise adding tool,” 2005.

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

[30] R. Caruana, “Multitask learning: A knowledge-based source of inductive bias,” *Machine Learning*, vol. 28, pp. 41–75, 1997.

[31] D. Chen and B. Mak, “Multitask learning of deep neural networks for low-resource speech recognition,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 7, pp. 1172–1183, 2015.

[32] G. Hinton, “Learning distributed representations of concepts,” in *Proceedings of the eighth annual conference of the cognitive science society*. Amherst, MA, 1986, vol. 1, pp. 1–12.

[33] D. Leeuwen and N. Brümmer, “The distribution of calibrated likelihood-ratios in speaker recognition,” in *Proc. Interspeech*, 2013, pp. 1619–1623.

[34] N. Brümmer and E de Villiers, “The BOSARIS toolkit: Theory, algorithms and code for surviving the new DCF,” *arXiv preprint arXiv:1304.2865*, 2013.

[35] M. W. Mak and H. B. Yu, “A study of voice activity detection techniques for NIST speaker recognition evaluations,” *Computer Speech and Language*, vol. 28, no. 1, pp. 295–313, 2014.

[36] G. Hinton, “Training products of experts by minimizing contrastive divergence,” *Neural Computation*, vol. 14, no. 8, pp. 1771–1800, 2002.

[37] Y. Bengio, P. Lamblin, D. Popovici, H. Larochelle, et al., “Greedy layer-wise training of deep networks,” *Advances in neural information processing systems (NIPS)*, vol. 19, pp. 153, 2007.

[38] Z. L. Tan and M. W. Mak, “Bottleneck features from SNR-adaptive denoising deep classifier for speaker identification-adaptive denoising deep classifier for speaker identification,” in *Asia-Pacific Signal and Information Processing Association, Annual Summit and Conference (APSIPA ASC)*, 2015, pp. 1035–1040.

[39] Y. Lei, N. Scheffer, L. Ferrer, and M. McLaren, “A novel scheme for speaker recognition using a phonetically-aware deep neural network,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2014, pp. 1695–1699.

[40] M. Mandasari M. McLaren and D. Leeuwen, “Source normalization for language-independent speaker recognition using i-vectors,” in *Proc. of Odyssey: The Speaker and Language Recognition Workshop*, 2012, pp. 55–61.

[41] A. Hatch, S. Kajarekar, and A. Stolcke, “Within-class covariance normalization for SVM-based speaker recognition,” in *Proc. of the 9th International Conference on Spoken Language Processing*, Pittsburgh, PA, USA, Sep. 2006, pp. 1471–1474.

[42] Z. L. Tan and M. W. Mak, “I-vector DNN scoring and calibration for noise robust speaker verification,” in *Proc. Interspeech*, 2017, accepted.

[43] K. M. He, X. Y. Zhang, S. Q. 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.

[44] G. Hinton and R.R. Salakhutdinov, “Reducing the dimensionality of data with neural networks,” *Science*, vol. 313, no. 5786, pp. 504–507, 2006.

[45] F. Seide, G. Li, X. Chen, and D. Yu, “Feature engineering in context-dependent deep neural networks for conversational speech transcription,” in *Automatic Speech Recognition and Understanding (ASRU), 2011 IEEE Workshop on*. IEEE, 2011, pp. 24–29.

[46] D. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in *International Conference on Learning Representations (ICLR)*, 2015, pp. 1–13.

[47] X. Glorot and Y. Bengio, “Understanding the difficulty of training deep feedforward neural networks.,” in *Aistats*, 2010, vol. 9, pp. 249–256.

[48] B. Karlik and V. Olgac, "Performance analysis of various activation functions in generalized MLP architectures of neural networks," *International Journal of Artificial Intelligence and Expert Systems*, vol. 1, no. 4, pp. 111–122, 2011.

[49] A. Maas, A. Hannun, and A. Ng, "Rectifier nonlinearities improve neural network acoustic models," in *ICML Workshop on Deep Learning for Audio, Speech, and Language Processing (WDLASL)*, 2013.

[50] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proceedings of the 32nd International Conference on Machine Learning (ICML)*, 2015, pp. 448–456.

Zhili TAN received the B.Eng. and M.Sc. degrees in electronic engineering from the Chinese University of Hong Kong in 2013 and 2014, respectively. Since August 2014, he has been pursuing a Ph.D. degree in electronic and information engineering in The Hong Kong Polytechnic University. His research interest include speaker recognition and machine learning.

Man-Wai MAK (M'93–SM'15) received a Ph.D. in electronic engineering from the University of Northumbria in 1993. He joined the Department of Electronic and Information Engineering at The Hong Kong Polytechnic University in 1993 and is currently an Associate Professor in the same department. He has authored more than 180 technical articles in speaker recognition, machine learning, and bioinformatics. Dr. Mak also coauthored a postgraduate textbook *Biometric Authentication: A Machine Learning Approach*, Prentice-Hall, 2005 and a research monograph *Machine Learning for Protein Subcellular Localization Prediction*, De Gruyter, 2015. He served as a member of the IEEE Machine Learning for Signal Processing Technical Committee in 2005–2007. He has served as an associate editor of IEEE/ACM Transactions on Audio, Speech and Language Processing. He is currently an associate editor of *Journal of Signal Processing Systems* and *IEEE Biometric Compendium*. He also served as Technical Committee Members of a number of international conferences, including ICASSP and Interspeech, and gave a tutorial on machine learning for speaker recognition in Interspeech'2016. Dr. Mak's research interests include speaker recognition, machine learning, and bioinformatics.

Brian Kan-Wing MAK received the B. Sc. degree in Electrical Engineering from the University of Hong Kong, the M. S. degree in Computer Science from the University of California, Santa Barbara, USA, and the Ph.D. degree in Computer Science from the Oregon Graduate Institute of Science and Technology, Portland, Oregon, USA. He had worked as a research programmer at the Speech Technology Laboratory of Panasonic Technologies Inc. in Santa Barbara, and as a research consultant at the AT&T Labs – Research, Florham Park, New Jersey, USA. He had been a visiting researcher of Bell Laboratories and Advanced Telecommunication Research Institute — International as well. Since April 1998, he has been with the Department of Computer Science in the Hong Kong University of Science and Technology, and is now an Associate Professor.

He had served or is serving on the editorial board of the IEEE Transactions on Audio, Speech and Language Processing, the Signal Processing Letters, and Speech Communication. He also had served on the Speech and Language Technical Committee of the IEEE Signal Processing Society. His interests include acoustic modeling, speech recognition, spoken language understanding, computer-assisted language learning, and machine learning. He received the Best Paper Award in the area of Speech Processing from the IEEE Signal Processing Society in 2004.