Odyssey 2020 The Speaker and Language Recognition Workshop
1-5 November 2020, Tokyo, Japan

Adaptive Mean Normalization for Unsupervised Adaptation of Speaker Embeddings

Mitchell McLaren, Md Hafizur Rahman, Diego Castan, Mahesh Kumar Nandwana, Aaron Lawson

Speech Technology and Research Laboratory, SRI International, Menlo Park, California, USA
{mitchell.mclaren, mdhafizur.rahman, diego.castan, mahesh.nandwana, aaron.lawson}@sri.com

Abstract

We propose an active learning approach for the unsupervised normalization of vector representations of speech, such as speaker embeddings, currently in widespread use for speaker recognition systems. We demonstrate that the traditionally used mean for normalization of speaker embeddings prior to probabilistic linear discriminant analysis (PLDA) is suboptimal when the evaluation conditions do not match the training conditions. Using an unlabeled sample of target-domain data, we show that the proposed adaptive mean normalization (AMN) technique is extremely effective for improving discrimination and calibration performance, by up to 26% and 65% relative over out-of-the-box system performance. These benchmarks were performed on four distinctly different datasets for a thorough analysis of AMN robustness. Most notably, for a range of data conditions, AMN enabled the use of a calibration model trained on data mismatched to the conditions being evaluated. The approach was found to be effective when using as few as thirty-two unlabeled samples of target-domain data.

Index Terms: speaker recognition, mean normalization, speaker embeddings, calibration, dynamic selection, adaptive normalization.

1. Introduction

Unsupervised adaptation of speaker recognition systems has received considerable attention in the research field [1, 2, 3, 4, 5, 6, 7] due to the degrading impact of domain shift on system performance. Domain shift occurs when a speaker recognition system is trained and tuned on conditions that are subtly or dramatically different from the conditions to which the system is applied. A system that is not robust to domain mismatch suffers reduced discrimination performance and inadequate calibration, which in turn can increase either missed or false system detections. The purpose of system adaptation is counteracting domain-shift impact and enabling the speaker recognition system to perform as expected in the target domain.

System adaptation can be performed in a supervised or unsupervised paradigm. In the supervised scenario, the system leverages target-domain data along with ground-truth speaker labels. Given the difficulty and cost associated with labeling target-domain data for each speaker, unsupervised adaptation approaches are typically sought. Unsupervised adaptation involves leveraging information from target-domain samples to improve system performance without the need for data labeling. The focus of this work is the unsupervised adaptation scenario.

Numerous approaches to unsupervised adaptation have been proposed. Several approaches attempt to reduce domain variability from the i-vector space by directly modeling the domain mismatch into the i-vector space or mapping the source-domain data into a domain-invariant space prior to PLDA training [1, 2, 3]. The adaptation process becomes challenging for PLDA adaptation without speaker-labeled data. Villalba and Lleida [4] proposed a variational Bayesian approach for unsupervised PLDA adaptation, where unlabeled data was modeled as a latent random variable. Although this approach explicitly models speaker uncertainly into the adapted model, it lacks in performance due to low channel variability. Several other approaches aim to estimate speaker labels using techniques including agglomerative hierarchical clustering and subsequently use supervised adaptation techniques, such as retraining the PLDA or calibration model of the system [5, 6]. These techniques are effective if the speaker-label estimates are of high quality, but they often result in degraded system performance due to the confusion of the clusters and low channel diversity of the clustered samples. More recently, different autoencoder-based unsupervised domain-adaption techniques—including a denoising autoencoder [7], a maximum mean discrepancy (MMD) based autoencoder [8], and a correlation alignment (CORAL/CORAL+) loss based autoencoder [9, 10]—have been introduced. Drawbacks of these techniques include the requirement for relatively more computational power and target-domain data to successfully compensate for the domain mismatch with respect to the training data.

Proposed in this work is a simple yet effective technique that operates on a single aspect of a traditional speaker recognition system—mean normalization. We

10.21437/Odyssey.2020-13

demonstrate both the need for appropriate mean normalization specific to the conditions being evaluated and, in doing so, the positive impact that this approach has on system discrimination power and calibration performance. We then propose a dynamic selection mechanism for estimating an appropriate mean for each enrollment and test audio file based on the similarity of the conditions between the file in question and candidate examples from the target domain. The technique—termed adaptive mean normalization (AMN)—is benchmarked on four different evaluation sets, on which significant improvements to discrimination and calibration performance are observed. Finally, we highlight the robustness of the approach as well as identify areas of AMN for future work.

The remainder of this paper is organized as follows: In Section 2, we detail the speaker recognition system used in this study. We give a description of the proposed AMN technique in Section 3. In Sections 4 and 5, we describe the evaluation protocol and results, respectively.

2. Speaker Recognition System

In this section, we detail the speaker recognition system used throughout this study. We leverage several of the advances from the SRI-CON-USC team in the NIST SRE'18 [11], including multi-bandwidth DNN-based speaker embeddings based on power normalized cepstral coefficients (PNCC) [12], source-normalized LDA/PLDA, and linear calibration.

Speech activity detection is based on a DNN with two hidden layers containing 500 and 100 nodes. The DNN was trained using 20-dimensional mel-frequency cepstral coefficients (MFCC) features, stacked with 31 frames, and mean and variance normalized over a 201-frame window.

We use a multi-bandwidth speaker embeddings network based on 16 kHz PNCC features of 30 dimensions extracted from a bandwidth of 100–7600 Hz using 40 filters and root compression of 1/15. All audio was sampled at 16 kHz prior to feature extraction. The process for training the embedding network that we developed for the NIST SRE'18 enables high performance on both 8 kHz and 16 kHz audio instead of tailoring the system to a specific bandwidth. This training process involves adding to the pool of embeddings training audio, a downsampled copy of any 16 kHz or higher data to 8 kHz. In doing so, the embeddings network will observe the same spoken content and conditions at both 8 kHz and 16 kHz bandwidths, enabling it to suppress bandwidth variation from the embeddings. The architecture of our embeddings extractor DNN follows the Kaldi recipe [13] with specific details and our general augmentation process can be found in[14].

Training data plays an important role in the design of a robust embeddings extractor [15, 14]. System training data included 234,288 signals from 14,630 speakers. This data was compiled from NIST SRE 2004–2008 [16], NIST SRE 2012 [17], Mixer6 [18], Voxceleb1 [19], and Voxceleb2 (train set) data [20]. Voxceleb1 data had 60 speakers removed that overlapped with Speakers in the Wild (SITW). Augmentation of data was applied using four categories of degradations as in [14], including music and noise, both at a 5 dB signal-to-noise ratio, compression, and low levels of reverb. We used 412 noises compiled from both freesound.org and the MUSAN corpus [21]. Music degradations were sourced from 645 files from MUSAN and 99 instrumental pieces purchased from Amazon music. For reverberation, examples were collected from 47 real impulse responses for low-level reverb available on echothief.com and 400 signals sourced from MUSAN. Compression was applied using 32 different codec-bitrate combinations with open source tools. In addition to these augmentations, we duplicated the 16 kHz or higher data (74,447 files) by downsampling to 8 kHz per the multi-bandwidth goal above.

The focus of this paper is the unsupervised adaptation of the system backend, specifically the mean normalization process. The training data for the backend was a one-third sampling of the embeddings training dataset, excluding the duplicated 16-to-8 kHz data. All embeddings are first transformed by source-normalized LDA [22], in which the sources were defined as each of the data sources of the system training data (each SRE corpus and Voxceleb1/2). Source normalization ensures the domain shift between these data sources is not interpreted as between-class variation as would be the case with traditional LDA. Mean normalization is then applied to the LDA-reduced embeddings, with the mean calculated from the backend training dataset. Length normalization [23] is then applied before PLDA modeling [24]. The PLDA scores are calibrated with scale and shift parameters learned via linear logistic regression over a calibration dataset.

3. Active Learning Mean Normalization

Mean normalization of vector representations (such as i-vectors or embeddings) is commonplace in speaker recognition systems, as it centralizes the data distribution prior to applying length normalization and PLDA modeling. In this section, we detail the impact of this process followed by our proposed method of dynamically defining the mean for normalization based on the observed conditions of the trial data.

3.1. The Vital Role of Mean Normalization

The aim of mean normalization is to distribute the data evenly over a unit hypersphere that is formed through the length normalization process. This process enables the data to better fulfill the assumptions of the PLDA model in that it then appears more Gaussianly distributed. To perform effectively, the system assumes the mean calculated during system training is appropriate for the operating data conditions. In practice, this assumption is typically unfulfilled, leading to a suboptimal mean vector for the normalization process. To illustrate this issue, we present the results of benchmarking the SRE'18 CMN2 dataset when using the traditional system mean compared to using the mean of the unlabeled adaptation data for SRE'18 CMN2 in Table 2. Using the more appropriate mean significantly improves system performance from an equal error rate (EER) of 9.7% to 7.9% and a cost of likelihood ratio (Cllr) of 0.371 to 0.293. Of significant interest is the improvement in calibration performance (Cllr) when using an appropriate mean in combination with a calibration model trained on data that does *not* represent the evaluation conditions. Surprisingly, using a suitable mean for normalization appears more important than the data used to learn the calibration parameters.

3.2. Adaptive Selection of the Mean

The above benchmark was conducted with a single mean calculated from all available data that was relevant to the evaluation conditions. This static mean is quite suitable when data conditions are homogeneous. However, when evaluation conditions vary, we can assume that dynamically adapting the mean for the conditions would enhance performance. In essence, we can apply an active learning paradigm in which a pool of data is queried using test conditions to find an appropriate mean. We propose a solution to active learning approach below, which also accounts for the case in which no suitable mean data can be found for normalization.

The approach's aim is determining, if possible, an appropriate mean for normalizing each individual test file at test time by selecting from a candidate pool of vector data, based on similarity of the conditions between the candidate and test data (and similarly for the enrollment samples).

The first stage involves ranking the set of candidate data by condition similarity to the test sample in question. For this purpose, we leverage our recent work in condition PLDA (CPLDA) [25]. The CPLDA model is trained using a subset of the system training data but replaces speaker labels with condition labels. The condition labels include compression type, reverb type, noise type, language, and gender information. In combination, these define 11,453 unique conditions. CPLDA takes as input the speaker embeddings after transforming them with LDA that is specific to the CPLDA space, mean normalization, and length normalization.

Once the CPLDA model has been trained, a collection of candidate embeddings for adaptive mean normalization must be defined. Here, this set is formed by using a held-out dataset that is closely matched to the conditions being evaluated; however, in practice, the enrollment samples or test samples could serve as candidates. We analyze the number of candidate mean norm files required for effective AMN in the results section.

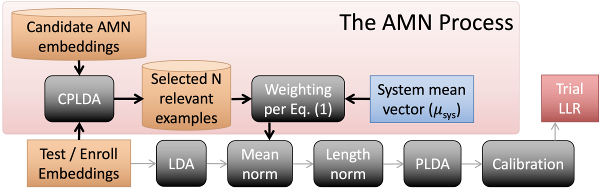

Figure 1: The proposed method of adaptive mean normalization.

As more relevant data is located for the mean estimation, the dynamic mean shifts from the system mean toward this condition-specific mean. A summary of the AMN process is depicted in Figure 1.

AMN is applied to each individual embedding used for speaker enrollment and testing. The aim of AMN is reliably estimating the most appropriate mean for normalizing an embedding given the condition exhibited in the embedding. We must also consider the case that zero or very few suitable embeddings exists for the process. As such, we define the AMN process for an embeddings as follows.

1. Define a similarity threshold α that is used to locate suitable candidate embeddings for the mean normalization of embedding, e.

2. Define the maximum number of embeddings M used to estimate the mean.

3. Define the default system mean μ_{sys} as the mean of the PLDA training data.

4. Use CPLDA scoring to compare the embedding e to the candidate embeddings and produce condition similarity scores for each.

5. Retain only the N embeddings with the highest condition similarity scores above the threshold α, up to a maximum of M embeddings ($N \leq M$).

6. Estimate the condition-specific mean μ_e from the retained embeddings.

7. Determine the final mean for normalization μ_{AMN} as the weighting of the default system mean μ_{sys} and the condition-specific mean μ_e based on how many of the M desired embeddings were retained, such that,

$$\mu_{\text{AMN}} = \frac{\mu_{\text{sys}}}{2}\left(1 - \frac{N}{M}\right) + \frac{N\mu_e}{2M} \qquad (1)$$

Regarding computation, we have heuristically found that AMN requires very limited overhead. The major factor in determining the overhead is the number of available

Table 1: *Details of evaluation datasets used in this work including norm sets and the number of target/impostor trials.*

Condition	# Spk	# Tgt / Imp	NormSet	# Seg
sre16	201	37k / 1.9mil	sre16-unlb	2400
sre18	188	60.7k / 2.0mil	sre18-unlb	2400
sitw-eval	180	3.7k / 717k	sitw-dev	1200
rats-src	168	3.2k / 539k	rats-src	7152

candidate embeddings. As we demonstrate in the result section, the number of samples for effective AMN can be quite few. It is also worth noting the benefit in treating each embedding individually compared to our prior work on the related and more intense process of trial-based calibration (TBC) [26, 25]. In TBC, the selection of candidate calibration trials is specific to each enroll-test pairing, which is considerably more costly per test file as the number of enrolled speakers increases.

4. Evaluation Protocol

We demonstrate the effectiveness of the proposed approach using several datasets that represent a large scope of conditions: NIST SRE 2016 (tgl and yue) [27], NIST SRE 2018 (CMN2) [11], Speakers in the Wild (SITW) [28], and source signals of the DARPA RATS SID data [29]. For each dataset, we define the candidate AMN data, which is disjoint in terms of actual speakers from the evaluation set. Details of each dataset are provided in Table 1. To optimize space in this article, we direct the readers interested in more details on each of these datasets to the provided references. The maximum number of segments, M, selected for AMN was tuned on the RATS dataset and defined as the half the number of candidate segments. This implies that a mean constructed of 50% of the available candidates would be sufficient, while allowing for a degree of heterogeneity in the candidates.

The system was calibrated using a subset of the RATS source data and held static across the evaluations. The motivation for holding this constant was demonstrating the benefit of AMN in terms of calibration performance when using a calibration model matched or mismatched to the evaluation data.

We report results in terms of equal error rate (EER) and cost of the likelihood ratio (Cllr). EER provides a measure of discrimination power irrespective of how well the system is or is not calibrated, while Cllr measures how well the system is calibrated across a broad spectrum of operating points, which is an important aspect when deploying a system that must make decisions via a score threshold.

5. Results and Analysis

This section evaluates the proposed AMN technique and highlights the approach's associated benefits as well as its sensitivities. Due to the number of datasets used in benchmarking and a thorough analysis of AMN, a single table of results is presented and each relevant comparison is detailed in the text.

5.1. Calibration of Traditional Systems

To grasp the potential impact of mean normalization, we first benchmarked each evaluation set using the speaker recognition system *out-of-the-box*. The first column in Table 2 details the EER and Cllr metrics across the datasets. The EER significantly varies across the sets due to each set being composed of different conditions, while nonetheless competitive for each of these datasets. Of significant note is the poor calibration performance of the sets except for rats-src, as a held-out portion of rats-src was used in the learning of the calibration models. To provide a reference of how well these datasets could potentially be calibrated in an matched scenario, we applied calibration for each when trained on the evaluation data itself (second column of metrics). In summary, these trends indicate that with a traditional speaker recognition system, the application of an appropriate calibration model is vital.

5.2. The Impact of Relevant Mean Normalization

The first comparison regarding mean normalization that we make involves the performance difference when using the system mean (based on PLDA training data) and a mean from the predefined, held-out data that is relevant to each condition (Cond-MN). This is implemented as two differences, noting that the adaptive selection from Section 3.2 is not used in this section. Firstly, the mean used to normalize a dataset is calculated from the held out portion of the data (for example, sitw-dev used to calculate a single mean vector for the normalization of sitw.eval). Second, the calibration model is trained in the same manner with the mean of rats-src being used to normalize embeddings prior to generation of the scores for calibration training. Table 2 details results for this comparison in the first and third column of metrics with Mean Norm being baseline or Cond-MN, respectively. It can be observed that EER and Cllr improves relative to

Table 2: Performance metrics (EER/Cllr) across evaluation datasets for each type of mean normalization. The EvalSet for calibration is a positively biased oracle scenario. The two baseline results offer the same EER values due to the ranked order of scores being unchanged. This is in contrast to the alternate mean normalization approaches which apply trial-specific normalization.

	Mean Norm	Baseline	Baseline	Cond-MN	AMN	AMN-mismatch
	Calibration	RATS	EvalSet	RATS	RATS	RATS
Dataset	SRE'16	10.0% / 0.497	10.0% / 0.364	9.4% / 0.406	**8.3%** / **0.321**	13.3% / 0.892
	SRE'18	9.7% / 0.371	9.7% / 0.357	7.9% / 0.293	**7.2%** / **0.283**	9.6% / 0.500
	SITW	2.6% / 0.381	2.6% / **0.100**	**2.5%** / 0.153	2.6% / 0.134	2.6% / 0.126
	RATS	3.6% / 0.149	3.6% / 0.147	**3.5%** / **0.143**	3.6% / 0.152	**3.5%** / 0.182

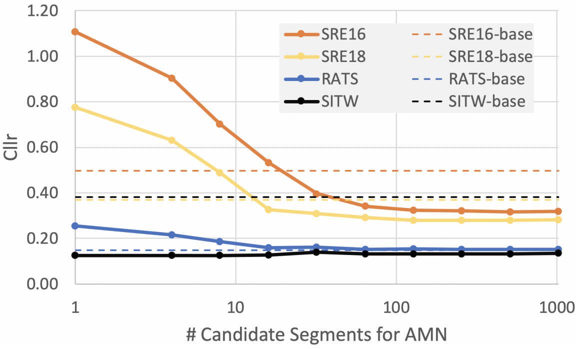

Figure 2: Performance vs. the number of candidate segments available for AMN on the RATS dataset.

the baseline (Baseline/RATS) by up to 19% and 60%, respectively, when normalizing the enrollment and test embeddings using a static mean calculated from an appropriate dataset for each evaluation set (Cond-MN/RATS). While this improvement in discrimination as indicated by EER is impressive when only the mean differs between systems, what is surprising is that the calibration model, which remained static and trained on rats-src data, is suddenly appropriate for the calibration of mismatched evaluation conditions. In fact, the average difference between the Baseline/EvalSet and the Cond-MN/RATS is quite small.

5.3. Adaptive Mean Normalization

While dataset-relevant mean normalization (Cond-MN) significantly improved performance, we take the approach one step further by introducing a dynamic selection mechanism based on condition similarity. AMN was applied to each individual enrollment and test sample, with a maximum number of samples of half the number of segments available to AMN. Results in the AMN column of Table 2 indicate this dynamic process further enhanced performance in most cases over CMN, notably those datasets with less homogeneous conditions (such as rats-src and SITW). Of interest is the average measurement of similarity observed from the perspective of CPLDA. For each embedding, the ratio of segments used in the dynamic mean estimate compared to the desired number $\frac{N}{M}$ can be averaged over the evaluation set to provide an indication of how well the candidate set is fit to the evaluation set. These average fit measures had an average of 0.9 with standard deviation 0.1, suggesting that the candidate dataset is well suited for the AMN processing of each evaluation set. Comparing the first column (Baseline/RATS) to the improvements in the fourth column (AMN/RATS) of Table 2, gains of up to 26% (average of 11%) in EER and 65% (average of 30%) in Cllr can be observed.

5.4. Data Requirements

To illustrate the power of AMN, we vary the number of segments in the candidate calibration dataset from 1 to 1024 in powers of two and benchmark the Cllr on each dataset. Figure 2 indicates that with as few as 32 segments, AMN already achieves most of the potential gain observed when using 1024 segments. Although not shown, a similar trend was observed in terms of EER, with particularly significant gains in SRE'16 and SRE'18 datasets. The significance of such limited data improving system performance exemplifies once again the importance of appropriate mean normalization as well as the stability of such a simple approach.

Figure 2 also indicates a difference between 1 candidate segment and the *base* results, with the most obvious difference being within the SRE'16 dataset. This occurs due to our application of a calibration model trained without AMN scoring for the base results, and one with AMN scoring for the AMN benchmarking. In the case that AMN locates few or no relevant examples for mean normalization, μ_{AMN} approximates μ_{sys} and the use of a AMN-influenced calibration model becomes unsuitable. Given this sensitivity of the approach, future work should consider how to dynamically assign calibration model parameters based on the N examples used to estimate μ_{AMN}.

Finally, we attempt to highlight the sensitivities of AMN by using a candidate pool of data that is not matched to the evaluation conditions. This is done using a leave-one-set-out approach. For example, we evaluated the rats-src dataset by defining the candidate AMN set as all other datasets and excluding rats-src. Ideally, the CPLDA similarity scoring process would exclude anything dissimilar to the condition being evaluated, and the results would taper toward the baseline system results as if the system μ_{sys} was used consistently. We observe, however, that performance was not enhanced with this approach and was, in fact, degraded in all cases except SITW. While not presented here, we also evaluated different thresholds, α on the similarity measure without success. This suggests that improvements could be made to the condition similarity measure provided by CPLDA to help filter out less appropriate data in the dynamic mean process.

6. Conclusions

In this work, we proposed the method of adaptive mean normalization (AMN) for the unsupervised adaptation of speaker recognition systems by using data samples relevant to evaluation conditions. The AMN approach was demonstrated to be extremely effective at improving discrimination and calibration performance by up to 26% and 66% relative to out-of-the-box system performance. These benchmarks were performed on four distinctly different datasets for a thorough analysis of AMN robustness. Most notably, results showed that with AMN, a calibration model trained on mismatched condition data provided impressive calibration performance. Further, AMN was found to be highly effective when using as few as eight unlabeled target-domain data samples.

Future work will involve improving AMN's sensitivity to an absence of appropriate condition data, which may degrade performance when using the technique. Additionally, the propagation of errors from the condition predictor should be analyzed to determine the impact of such errors on the final calibration performance.

7. References

[1] H. Aronowitz, "Inter dataset variability compensation for speaker recognition," in *Proc. ICASSP*, 2014, pp. 4002–4006.

[2] M. H. Rahman, A. Kanagasundaram, D. Dean, and S. Sridharan, "Dataset-invariant covariance normalization for out-domain PLDA speaker verification," in *Proc. Interspeech*, 2015.

[3] O. Glembek, J. Ma, P. Matejka, B. Zhang, O. Plchot, L. Burget, and S. Matsoukas, "Domain adaptation via within-class covariance correction in i-vector based speaker recognition systems," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, 2014, pp. 4032–4036.

[4] J. Villalba and E. Lleida, "Unsupervised adaptation of PLDA by using variational Bayes methods," in *Proc. ICASSP*, 2014.

[5] D. Garcia-Romero, A. McCree, S. Shum, N. Brummer, and C. Vaquero, "Unsupervised domain adaptation for i-vector speaker recognition," in *Proc. of Speaker Odyssey*, 2014.

[6] S. H. Shum, D. A. Reynolds, D. Garcia-Romero, and A. McCree, "Unsupervised clustering approaches for domain adaptation in speaker recognition systems," in *Proc. Speaker Odyssey*, 2014.

[7] S. Shon, S. Mun, W. Kim, and H. Ko, "Autoencoder based domain adaptation for speaker recognition under insufficient channel information," in *Proc. Interspeech*, 2017, pp. 1014–1018.

[8] W. W. Lin, M.-W. Mak, L. Li, and J.-T. Chien, "Reducing domain mismatch by maximum mean discrepancy based autoencoders," in *Proc. Odyssey*, 2018, pp. 162–167.

[9] B. Sun, J. Feng, and K. Saenko, *Correlation alignment for unsupervised domain adaptation*. Springer, 2017, pp. 153–171.

[10] K. A. Lee, Q. Wang, and T. Koshinaka, "The coral+ algorithm for unsupervised domain adaptation of plda," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 5821–5825.

[11] *NIST 2018 Speaker Recognition Evaluation Plan*, 2018, https://www.nist.gov/document/sre18evalplan2018-05-31v6pdf.

[12] C. Kim and R. Stern, "Power-normalized cepstral coefficients (PNCC) for robust speech recognition," in *Proc. ICASSP*, 2012, pp. 4101–4104.

[13] *NIST SRE 2016 Xvector Recipe*, 2017, https://david-ryan-snyder.github.io/2017/10/04/model_sre16_v2.html.

[14] M. McLaren, D. Castan, M. Nandwana, L. Ferrer, and E. Yilmaz, "How to train your speaker embeddings extractor," in *Proc. Speaker Odyssey*, 2018.

[15] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust DNN embeddings for speaker recognition," in *Submitted to ICASSP*, 2018.

[16] *The NIST Year 2008 Speaker Recognition Evaluation Plan*, 2008, http://www.itl.nist.gov/iad/mig/tests/sre/2008/sre08_evalplan_release4.pdf.

[17] *The NIST Year 2012 Speaker Recognition Evaluation Plan*, 2012, http://www.nist.gov/itl/iad/mig/upload/NIST_SRE12_evalplan-v17-r1.pdf.

[18] L. Brandschain, D. Graff, and K. Walker, *Mixer 6 Speech LDC2013S03*, 2013, https://catalog.ldc.upenn.edu/LDC2013S03.

[19] A. Nagrani, J. S. Chung, W. Xie, and A. Zisserman, "Voxceleb: Large-scale speaker verification in the wild," *Computer Science and Language*, 2019.

[20] J. S. Chung, A. Nagrani, and A. Zisserman, "Voxceleb2: Deep speaker recognition," in *Proc. Interspeech*, 2018.

[21] D. Snyder, G. Chen, and D. Povey, "Musan: A music, speech, and noise corpus," *arXiv preprint arXiv:1510.08484*, 2015.

[22] M. McLaren and D. Van Leeuwen, "Source-normalized LDA for robust speaker recognition using i-vectors from multiple speech sources," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 3, pp. 755–766, 2012.

[23] D. Garcia-Romero and C. Y. Espy-Wilson, "Analysis of i-vector length normalization in speaker recognition systems," in *Twelfth Annual Conference of the International Speech Communication Association*, 2011.

[24] S. Prince and J. Elder, "Probabilistic linear discriminant analysis for inferences about identity," in *Proc. IEEE International Conference on Computer Vision*. IEEE, 2007, pp. 1–8.

[25] L. Ferrer, M. K. Nandwana, M. McLaren, D. Castan, and A. Lawson, "Toward fail-safe speaker recognition: Trial-based calibration with a reject option," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2019.

[26] M. McLaren, N. Scheffer, L. Ferrer, and Y. Lei, "Effective use of DCTs for contextualizing features for speaker recognition," in *Proc. ICASSP*, 2014.

[27] *NIST 2016 Speaker Recognition Evaluation Plan*, 2016, https://www.nist.gov/sites/default/files/documents/itl/iad/mig/SRE16_Eval_Plan_V1-0.pdf.

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

[29] K. Walker and S. Strassel, "The RATS radio traffic collection system," in *Proc. Odyssey*, 2012.