INTERSPEECH 2017
August 20–24, 2017, Stockholm, Sweden

Analysis and Description of ABC Submission to NIST SRE 2016

Oldřich Plchot[1], Pavel Matějka[1], Anna Silnova[1], Ondřej Novotný[1], Mireia Diez[1], Johan Rohdin[1], Ondřej Glembek[1], Niko Brümmer[2], Albert Swart[2], Jesús Jorrín-Prieto[2], Paola García[2], Luis Buera[2], Patrick Kenny[3], Jahangir Alam[3] and Gautam Bhattacharya[3]

[1]Brno University of Technology, Speech@FIT and IT4I Center of Excellence, Brno, Czech Republic
[2] Nuance Communications, Inc.
[3]CRIM, Montreal (Quebec), Canada

{iplchot, matejkap, isilnova, inovoton, mireia, rohdin, glembek}@fit.vutbr.cz
{Niko.Brummer, Albert.Swart, Jesus.Jorrin, Paola.Garcia, Luis.Buera}@nuance.com
{patrick.kenny, jahangir.alam, gautam.bhattacharya}@crim.ca

Abstract

We present a condensed description and analysis of the joint submission for NIST SRE 2016, by Agnitio, BUT and CRIM (ABC). We concentrate on challenges that arose during development and we analyze the results obtained on the evaluation data and on our development sets. We show that testing on mismatched, non-English and short duration data introduced in NIST SRE 2016 is a difficult problem for current state-of-the-art systems. Testing on this data brought back the issue of score normalization and it also revealed that the bottleneck features (BN), which are superior when used for telephone English, are lacking in performance against the standard acoustic features like Mel Frequency Cepstral Coefficients (MFCCs). We offer ABC's insights, findings and suggestions for building a robust system suitable for mismatched, non-English and relatively noisy data such as those in NIST SRE 2016.

Index Terms: speaker recognition, i-vector, DNN, fusion

1. Introduction

Four years have passed since the last NIST SRE in 2012 and researchers in the field have been developing and advancing speaker recognition technology, often training and testing on the data released by NIST and LDC. As the test sets of NIST SREs usually serve as a common benchmark in scientific publications, researchers often direct their work to tune their systems to perform well on these tasks.

Since the last NIST SRE evaluations in 2012 and 2010, there was only a moderate progress in the general speaker recognition system design. Most of the state-of-the-art systems rely on i-vectors [1] that are modeled by Probabilistic Linear Discriminant Analysis (PLDA) [2] or its variations [3, 4, 5].

We have however seen an advance in using Deep Neural Networks (DNN) in various fields of speech processing, including ASR [6], language recognition [7, 8, 9] and speaker recognition [10, 11, 12].

In the field of speaker recognition, DNNs are often used for extracting frame-by-frame speech bottleneck features (BNF) taken from a narrow hidden layer compressing the relevant information into low dimensional feature vectors [13, 7, 14]. Using these features, especially in combination with standard MFCCs lead to excellent results [15, 16] on NIST SRE 2010. All of these great results are however achieved on English data (and mostly telephone). We have recently published a study [17] that analyzes the BNF performance on non-English data and the results were mixed. Indeed, the results that most participants achieved in NIST SRE 2016 [18] have confirmed these disturbing conclusions.

SRE'16 brought a completely new non-English dataset and a tough challenge in the domain adaptation. It revealed the weak side of current BNF that are tuned for English, brought back the issue of score normalization [19, 20] and in general significantly increased the difficulty which will undoubtedly inspire a lot of research.

We present our SRE'16 submission, which can be taken as an inspiration to tackle these problems. The raw, but more detailed description of our systems can be found in [21].

2. Datasets and Feature Extraction

At the beginning of the development, all three labs (Agnitio, BUT and CRIM) had agreed to use the data from previous NIST evaluations (NIST SRE 2004 - 2008, Fisher English and Switchboard) for training and to leave out the labeled NIST SRE16 development data for testing, final calibration and fusion. Each site had partitioned the training data in a different way: Agnitio used all of the data for UBM, i-vector extractor, NDA and the PLDA. BUT split the data in order to develop additional *dev* and *test* sets. CRIM split them to define their *Oriental Background data* that included recordings from Chinese, Mandarin and Tagalog, and *Primary Background data* that contained the rest of the data.

All labs used to some extent also the unlabeled SRE16 data. This data found most use in the score-normalization cohort and other domain adaptation techniques like NAP or WCC. In case of CRIM, adding this data into the *Oriental Background set* results in obtaining their *Oriental set*.

The English part of the Fisher database was used by both Agnitio and BUT to train DNNs for BN feature extraction.

This work was supported by the DARPA RATS Program un- der Contract No. HR0011-15-C-0038. The views expressed are those of the author and do not reflect the official policy or posi- tion of the Department of Defense or the U.S. Government. The work was supported by Czech Ministry of Interior project No. VI20152020025 "DRAPAK", European Union's Horizon 2020 project No. 645523 BISON, by Google research award, Grant Agency of the Czech Republic project No. GJ17-23870Y, and by Czech Ministry of Education, Youth and Sports from the National Programme of Sustainability (NPU II) project "IT4Innovations excellence in science - LQ1602". It also received funding under European Union's Horizon 2020 Marie Sklodowska-Curie grant No. 748097 / M.SC. South Moravian Region grant agreement No. 665860.

http://dx.doi.org/10.21437/Interspeech.2017-1498

2.1. Additional BUT Sets

To provide more analysis and discussion later, we also describe the datasets that BUT created to obtain independent development and test sets which reflect as close as possible the SRE'16 evaluation data. The main motivation for this work was the small size and design of the labeled minor data. First of all, we did not want to base all of our development decisions on the results obtained with data that contains only 20 speakers and different languages that we would be facing in the evaluation set. Next, we believed that the labeled SRE'16 dev data contains dangerously small amount of trials to reliably train both calibration and fusion.

In order to obtain the BUT's split between training (for PLDA, LDA, NDA, SNORM ...) and development (testing, calibration, fusion), we followed the split designed in the PRISM dataset [22]. To obtain the initial trial set for both BUT dev and test sets, we take the *lan-lan* condition of the PRISM set and we cut all of the enrollment and test segments into shorter pieces. The duration of short cuts reflects the evaluation plan for NIST SRE 2016 - more precisely we based our cuts on the actual detected speech in the NIST SRE 2016 development labeled data. We chose the cuts to follow the uniform distribution between $25 - 50$ seconds of speech for enrollment segments and $3 - 40$ seconds for test segments.

When doing the cuts, we also added short versions ($10 - 60$ seconds of speech) of non-English telephone training data into our training set.

2.2. Voice Activity Detection

Each lab used their own pre-processing: Agnitio used a combination of Long-Term Spectral Divergence (LTSD) VAD and energy-based VAD to compute speech/nonspeech labels. The first 50ms of every audio was removed to avoid inconsistent VAD behavior. BUT used a phone recognizer trained on Fisher with with added noise at different levels of SNR. All frames that were marked as silence or noise were dropped. CRIM removed all non-speech frames using an unsupervised GMM-based voice activity detector.

2.3. Feature Extraction

As each lab in the ABC system has their proven VAD and feature extraction recipes, the detailed description of every single feature that was used in our submission would take too much space and we refer the reader to our system description[21] for more details. In general, the core feature extraction for each lab was the acoustic features such as MFCCs. We always extract 20 static coefficients including either C0 or Energy together with their deltas and double deltas. Afterwards we apply normalization. Agnitio applies Cepstral Mean Normalization (CMN), RelAtive SpecTral Amplitude (RASTA) processing and warping (3 seconds window), BUT and CRIM apply short-term Cepstral Mean and Variance Normalization (sCMVN). On top of the standard MFCCs, CRIM and BUT extracted also other types of acoustic features and applied the same normalization. BUT developed systems based on Perceptual Linear Prediction coefficients (PLP) and Perseus[23], while CRIM used Linear Frequency Cepstral Coefficients (LFCC) and Linear Prediction Cepstral Coefficients (LPCC).

Agnitio and BUT used Fisher data to train a DNN for extraction of bottleneck features (BNF)—for BUT the stacked bottleneck features (SBN). For exact configuration of Agnitio's BNF, please refer to [21], BUT's SBN features are well described in our analysis of DNN-based SRE systems [17]. Dimensionality of these features was set to 60 and 80 for Agnitio and BUT, respectively. In line with the recent research [15, 16], BUT has concatenated SBNs with MFCCs in the hope of obtaining the best possible results. Agnitio used the BNF on their own and performed the concatenation with MFCCs at the i-vector level.

3. Classifier Schemes

All of our systems are based on i-vectors [1] that are used as features for various classifiers. Each site trained a Universal Background Model (UBM) as a GMM with 2048 components and subsequently also the i-vector extractor on their own training data. In the following paragraphs, we will describe the approaches taken to build the final classifiers. We include the full description of classifiers as the described architecture was tuned for SRE'16 and we believe that the nuances that differentiate them from the usual PLDA recipes are important and might serve for additional research.

3.1. Agnitio

MFCC-PLDA: This system is based on a full covariance UBM, using MFCCs in the whole process. 400 dimensional i-vectors are extracted consequently. Nearest-neighbor Analysis (NDA) performs a dimensionality reduction of those ivectors from 400 to 250. This process is followed by mean normalization, which is adapted to the use case employing unlabeled development data, and length normalization. Scoring between i-vectors is achieved by using gender dependent PLDA (speaker space dimension is fixed to 120).

AGN-MFCC-BNF-PLDA: Two feature extractors are used: MFCC and BottleNeck Features (BNF). Two variants of this system are created by allocating the bottleneck layer as second or fourth hidden layer.

Two separate full covariance UBMs are trained on MFCC and BNF features. For each audio, two 400-dimensional i-vectors are extracted, respectively. They are then stacked to obtain a single 800-dimensional i-vector per audio. Once again, NDA is employed, but this time it performs a dimensionality reduction of those ivectors from 800 to 500. The process is followed by mean normalization, and length normalization. Scoring between i-vectors is achieved using gender dependent PLDA (speaker space dimension fixed to 200).

Finally, gender-dependent s-norm is applied, assisted by automatic gender recognition. The cohorts are obtained from the SRE16 unlabeled development data.

3.2. BUT

All of BUT's systems are based on UBM with diagonal covariance components and i-vector extractors with 600 dimensions, both trained in gender independent fasion only on telephone data from MIXER collections, Fisher English and Switchboard 2. The following paragraphs describe three classifier architectures that were used on top of i-vectors obtained by means of different front-end features.

PLDA: For PLDA model training, telephone data and non-English short cuts were used. All i-vectors were mean (mean was calculated using all training data) and length normalized. Then the Linear Discriminant Analysis (LDA) with Within Class Covariance Correction (WCC) was applied, decreasing dimensions of i-vectors from 600 to 200. The WCC is based on weighted addition of the within-class covariances of different

languages and datasets into the within-class covariance of LDA. We also removed the shift between the training data and the minor and major datasets. Resulting scores were normalized using speaker dependent s-norm with a cohort created from our training data and unlabeled SRE16 data. Speaker dependent means for the s-norm were computed on the 500 closest i-vectors for each speaker.

Discriminative PLDA: For training the DPLDA model [3], telephone data from Mixer+Fisher+Switchboard was used along with unlabeled data from NIST SRE'16. Unlabeled data were used to form non-target trials with labeled telephone data only (e.g. no trials between two unlabeled utterances were used for training). First, NAP was performed on top of all ivectors. As classes for NAP, 20 languages from training list were selected along with one class corresponding to both major and minor unlabeled data. After NAP all ivectors were mean (mean was calculated using all training data available) and length normalized. After the mean normalization, we performed LDA, decreasing the dimensionality of vectors to 250. As an initialization of DPLDA training, we used a corresponding PLDA model. During the DPLDA training, we set the prior probability of target trials to reflect the SRE'16 evaluation operating point (exactly in the middle between the two operating points of SRE'16 DCF).

Support Vector Machines: One SVM per speaker was trained using the enrollment ivector(s) as positive samples and unlabeled major and unlabeled minor data as negative samples. Length normalization, WCCN and NAP were applied to ivectors and zt-norm was applied to the scores [21].

The following BUT subsystems were used in the final fusion: 2 DPLDA systems trained on PLP and MFCC, 4 PLDA systems trained on MFCC, Perseus, PLP, MFCC-SBN and a single SVM system trained on PLP.

3.3. CRIM

All CRIM's systems are also based on 600 dimensional i-vectors obtained by means of a diagonal covariance UBM that was trained on the *primary background data* and iteratively adapted to the *oriental data* using relevance MAP. The i-vector extractor was first trained using sufficient statistics from all of the primary background data and later adapted by performing several iterations of minimum divergence training on the Oriental data.

Instead of using the i-vectors in the well-known and standard i-vector/PLDA pipeline, they are used as inputs to train a speaker classifier neural network (SCN). At the time of NIST SRE 2016, the architecture presented below was novel and interested readers can consult [24] for more details and results on NIST SRE 2010.

The inner representation of SCN is then used to obtain inputs into three classifiers - cosine distance (CD), PLDA and Latent Dirichlet Allocation (LDA). CRIM's final system combines 4 sub-systems based on the inner representation of SCN: MFCC-DNN1, MFCC-DNN2, MFCC-DNN3 and MFCC-DNN-LDA. Additionally 4 sub-systems were also created using just i-vectors: LFCC-CD, LPCC-CD LFCC-PLDA and MFCC-CD.

MFCC-DNN Systems: In order to train a speaker classifier network (SCN), a feed-forward neural network was used to learn a mapping between i-vectors and speaker labels. This approach can be viewed as a projection of the i-vectors into high-dimensional label-space which allows for easier discrimination between speakers [25]. The SCN is two layers deep and uses a sigmoid nonlinearity in the hidden layers. Each hidden layer consists of 2000 hidden units. The softmax function is used at the output layer, which represents a probability distribution over the speakers in the training set (Primary Background Data + Oriental Background Data). The speakers were filtered in such a way that each speaker had at least 5 recordings/i-vectors.

The input i-vectors are length normalized before being processed by the SCN. After the model is trained, it is used as a feature extractor for the background, enrollment and test data. Specifically, the activations of the last hidden layer were treated as feature vectors (d-vectors) for speaker verification.

In the case of SRE'16 data (development and evaluation) we only force the d-vectors to be of unit norm and do not perform any mean-centering. Speaker verification is performed using a cosine distance classifier with the SCN-projected features (i.e., d-vectors). We developed three system variations: **MFCC-DNN1:** For speaker models with 3 enrollment d-vectors (2000-dimensional) we average the individual scores during cosine scoring. In all other systems, for speaker models with 3 enrolment i-vectors/d-vectors a single score is produced by averaging the i-vectors/d-vectors. **MFCC-DNN2:** NAP projection is applied to all the d-vectors produced by the SCN. **MFCC-DNN3:** In this case we reduce the dimension of the NAP projected d-vectors using a principal component analysis (PCA) technique.

MFCC-DNN-LDA System: In this system we model the hidden activations of the DNN speaker classifier using Latent Dirichlet Allocation (LDA). The system MFCC-DNN-LDA differs from MFCC-DNN1 in replacing the cosine distance backend with a probabilistic backend which was trained blindly on the unlabeled training data. (We did not attempt to assign speaker, language or gender labels to the training data.)

As in MFCC-DNN1, the feature vector used to represent an utterance consisted of the sigmoid activations of the last hidden layer of the DNN. We viewed these features as noisy binary vectors and modeled them by a hidden vector of Bernoulli probabilities. If speaker labels were available, we would associate one Bernoulli probability vector with each speaker. Since we did not have speaker labels for the training set, we treated the recordings as if they all came from different speakers. We treated the components of the feature vector as being statistically independent and we placed a Beta prior on each of the Bernoulli probabilities. We "estimated" the priors by appealing to the maximum likelihood II principle, using the methods in [25].

4. Fusion

The final submission strategy was a three-way fusion of one system per lab, trained on the labeled minor data. Each lab provided a pre-fused system that went into the final fusion.

For **Agnitio**, normalized scores for the three subsystems are linearly fused by a simple weighted addition. Weights are 0.5, 0.25 and 0.25 for MFCC-PLDA, MFCC-BNF-4-PLDA and MFCC-BNF-2-PLDA, respectively. The scores are assumed to be of comparable scales because of score normalization.

To train the parameters of **CRIM's** fusion, we used the labeled minor SRE'16 development data. After training, the fusion was then applied: (i) to this same data (test-on-train) to pass as training scores for the final ABC fusion; and (ii) to the SRE'16 evaluation data, also as input to the final ABC fusion.

Because of data scarcity and to combat over-training, we used generative fusion and calibration strategies, with as few as

Table 1: *BUT subsystems and ABC fusions on NIST SRE 2016, C_{min}^{Prm} and C_{act}^{Prm} are resp. minimum and actual DCF'16.*

	SRE16			BUT test		
System Name	EER	C_{min}^{Prm}	C_{act}^{Prm}	EER	C_{min}^{Prm}	C_{act}^{Prm}
DPLDA_PLP	13.46	0.73	0.76	4.31	0.47	0.47
PLDA_PLP	12.6	0.74	0.74	6.2	0.42	0.44
PLDA_MFCC	12.55	0.72	0.73	6.05	0.42	0.43
PLDA_MFCCSBN	15.00	0.82	0.82	9.30	0.48	0.49
SVM_PLP	12.91	0.76	2.78	5.63	0.48	0.71
AGNITIO_NIGCAL	11.62	0.70	0.72	-	-	-
AGNITIO_QCAL	11.62	0.70	0.89	-	-	-
BUT_FIX_NIGCAL	9.99	0.66	0.69	3.48	0.28	0.29
BUT_FIX_QCAL	9.99	0.66	0.68	3.47	0.27	0.28
CRIM_NIGCAL	9.84	0.68	0.75	-	-	-
CRIM_QCAL	15.71	0.68	0.69	-	-	-
ABC_NIGCAL	8.68	0.62	0.76	-	-	-
ABC_QCAL	8.68	0.62	0.63	-	-	-

possible parameters. The fusion strategy was linear-Gaussian pre-calibration of each sub-system, followed by equal-weighted summation. Separate gender-independent calibrations were done for 1-call and 3-call enrollment. The linear-Gaussian calibration is done by computing the log-LR obtained from a generative model with two univariate Gaussians for targets and non-targets, with different means and shared covariance. The parameters were estimated with maximum-likelihood. Pre-calibration was applied before summation, so that (i) missing scores could be replaced by log-LR = 0 and (ii) sub-system scores were roughly at the same scale, with better system contributing a bit more than weaker systems.

Fusion and calibration of the **BUT** subsystems were trained with logistic regression, optimizing the cross-entropy, on the BUT development set. Our objective was to improve the error-rates on the independent BUT test set, but we were also monitoring error-rate trends on the labeled minor SRE'16 development set.

4.1. Primary ABC fusion for fixed condition

The input to the final ABC fusion consisted of 3 sets of scores, each produced by the labs Agnitio, BUT and CRIM. The training scores consisted of SRE'16 minor labeled development data. In the case of CRIM, this constituted a second use of this data. For the Agnitio and BUT systems, this data was unexposed.

Because of the scarcity of data, we did not judge fusion strategies by EER or DCF'16. Instead we looked at regularity of score histograms, DET-curves and normalized DCF curves. We performed Linear-Gaussian generative fusion, followed by non-linear post-calibration.

For post-calibration after fusion, we tried linear, quadratic and NIG [26]. In all cases NIG gave better calibration as judged on the SRE'16 minor labeled development data. The linear-Gaussian calibration is the same as described for CRIM's systems in section 4 above. The quadratic fusion is also generative Gaussian, but with independent (rather than shared) variances for targets and non-targets. The NIG calibration used independent normal-inverse Gaussian (NIG) distributions for targets and non-targets.

For NIG maximum-likelihood parameter estimation in [26], we had used a trust-region Newton algorithm for direct optimization of the likelihood. This time, we used a modified version of the EM algorithm in [27]. The modification is similar to the minimum-divergence trick—the model is over-parametrized during the M-step and then simplified again using a reparametrization of the hidden variable. The EM algorithm was initialized with moment matching. After a few hundred EM iterations, training was completed using direct L-BFGS optimization, which gives faster convergence during the end-game.

5. Experimental Results and Discussion

In table 1, we present results in terms of Equal Error Rate (EER) and SRE'16 primary metric for part of the BUT subsystems and individual fusions. The results on SRE'16 eval are computed with all available trials without any filtering as it is performed in the NIST scoring tool. The complete list of all subsystems evaluated on SRE'16 labeled development set can be found in [21]. On the subsystem results we can observe several trends that hold for us in SRE'16. First, let us mention the competitive performance of discriminatively trained systems (comparing DPLDA_PLP and PLDA_PLP). Next we compare the PLDA_MFCC and PLDA_MFCCSBN and notice the better performance of MFCC-based system over the system with concatenated MFCCs and SBNs. Both observations hold over all operating points and two different datasets.

It is also worth to look at the performance of our SVM system that has seen only enrollment segments and unlabeled SRE'16 data for each trial. Other data were exposed only for the purposes of the score normalization. This rather dated approach provided competitive results on calibration-independent metrics. We were able to reasonably calibrate this system on abundant number of trials and easier data of BUT dev set, but the miscalibration on evaluation set was still too large. This was the reason to exclude it later from the BUT fusion together with one DPLDA system that had also issues with calibration and produce the FIXED version of the BUT fusion. Excluding these systems improved only the actual DCF.

Another important aspect of the whole SRE'16 is score normalization. In fact all of the BUT subsystems presented here contain score normalization. We have dedicated the whole paper [19] to study the effects of score normalization. We also encourage the interested reader to study the theory behind score normalization that was partly inspired by SRE'16 in [20]. Performing score normalization, more precisely the adaptive variant of s-norm was crucial to obtain good results both on SRE'16 labeled dev set and evaluation set.

To analyze the calibration, let us first look at the individual systems and observe that with exception of the non-standard SVM system, there were no problems with calibration as the C_{min}^{Prm} and C_{act}^{Prm} are very close. All of the parameters were trained with logistic regression on BUT dev set and the good calibration has transferred also to the evaluation data.

Finally, we can analyze the individual fusions. For every fusion, we list two post-calibration strategies: NIG and QCAL. During our development, we had chosen NIG as our go-to strategy for evaluation. Comparing NIG and quadratic post-calibrations, we can observe that with the exception of Agnitio system, the simple QCAL was more effective on evaluation data. Comparing fusions of the individual sites we see similar performance of individual sites and a moderate gain from the cross-site fusion.

6. Conclusions

We have presented various sytems of the ABC team that are designed to cope with dataset mismatch and non-English data. We have presented and compared several fusion and calibration strategies and we have uncovered and discussed problems brought by SRE'16.

7. 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. PP, no. 99, 2010.

[2] S. J. D. Prince, "Probabilistic linear discriminant analysis for inferences about identity," in *Proc. International Conference on Computer Vision (ICCV)*, Rio de Janeiro, Brazil, 2007.

[3] L. Burget, O. Plchot, S. Cumani, O. Glembek, P. Matějka, and N. Brümmer, "Discriminatively trained probabilistic linear discriminant analysis for speaker verification," in *Proc. of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Prague, CZ, May 2011.

[4] S. Cumani, N. Brümmer, L. Burget, P. Laface, O. Plchot, and V. Vasilakis, "Pairwise discriminative speaker verification in the i–vector space," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 21, no. 6, pp. 1217–1227, june 2013.

[5] S. Cumani, O. Plchot, and P. Laface, "On the use of i-vector posterior distributions in probabilistic linear discriminant analysis," *IEEE/ACM Transactions on Audio, Speech and Language Processing*, vol. 22, no. 4, pp. 846–857, 2014.

[6] G. Hinton, L. Deng, D. Yu, A. rahman Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. S. G. Dahl, and B. Kingsbury, "Deep neural networks for acoustic modeling in speech recognition," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, November 2012. [Online]. Available: http://research.microsoft.com/apps/pubs/default.aspx?id=171498

[7] Y. Song, B. Jiang, Y. Bao, S. Wei, and L. Dai, "i-vector representation based on bottle neck feature for language identification," in *IEEE Electronics Letters*, 2013.

[8] R. Fér, P. Matějka, F. Grézl, O. Plchot, and J. Černocký, "Multilingual Bottleneck Features for Language Recognition," in *Proceedings of Interspeech 2015*, vol. 2015, no. 09, 2015, pp. 389–393.

[9] I. Lopez-Moreno, J. Gonzalez-Dominguez, and O. Plchot, "Automatic Language Identification Using Deep Neural Networks," in *ICASSP 2014*, Florence, Italy, 2014.

[10] Y. Lei, N. Scheffer, L. Ferrer, and M. McLaren, "A novel scheme for speaker recognition using a phonetically-aware deep neural network," in *ICASSP*, 2014.

[11] Y. Lei, L. Ferrer, M. McLaren, and N. Scheffer, "Comparative study on the use of senone-based deep neural networks for speaker recognition," *Submitted to IEEE Trans. ASLP*, 2014.

[12] D. Garcia-Romero, X. Zhang, A. McCree, and D. Povey, "Improving speaker recognition performance in the domain adaptation challenge using deep neural networks," in *SLT*, 2014.

[13] P. Matějka *et al.*, "Neural Network Bottleneck features for Language Identification," in *IEEE Odyssey: The Speaker and Language Recognition Workshop*, Joensu, Finland, 2014.

[14] N. D. Fred Richardson, Douglas A. Reynolds, "A unified deep neural network for speaker and language recognition," in *Interspeech*, 2015.

[15] P. Matějka, O. Glembek, O. Novotný, O. Plchot, F. Grézl, L. Burget, and J. Černocký, "Analysis of DNN Approaches To Speaker Identification," in *Proceedings of the ICASSP*, Shanghai, CN, 2016.

[16] S. O. Sadjadi, S. Ganapathy, and J. W. Pelecanos, "The IBM 2016 Speaker Recognition System," in *Proceedings of Odysset 2016*, Bilbao, Spain, 2016.

[17] O. Novotný, P. Matějka, O. Glembek, O. Plchot, F. Grézl, L. Burget, and J. Černocký, "Analysis of the DNN-Based SRE Systems in Multi-language Conditions," in *Proceedings of SLT 2016*. IEEE Signal Processing Society, 2016.

[18] "The NIST year 2016 Speaker Recognition Evaluation Plan," https://www.nist.gov/sites/default/files/documents/2016/10/07/sre16_eval_plan_v1.3.pdf, 2012.

[19] P. Matějka, O. Novotný, O. Plchot, L. Burget, and J. H. Černocký, "Analysis of Score normalization in Multilingual Speaker Recognition," in *Proceedings of Interspeech 2017*, Stockholm, Sweden, 2017.

[20] A. Swart and N. Brummer, "A Generative Model for Score Normalization in Speaker Recognition," in *Proceedings of Interspeech 2017*, Stockholm, Sweeden, 2017.

[21] N. Brummer, A. Swart, J. J. Prieto, P. García, P. Matějka, O. Plchot, M. Diez, A. Silnova, X. Jiang, O. Novotný, J. Rohdin, O. Glembek, F. Grézl, L. Burget, L. Ondel, J. Pešán, J. Černocký, P. Kenny, J. Alam, G. Bhattacharya, and H. Zeinali, "ABC NIST SRE 2016 system description," Tech. Rep., 2016. [Online]. Available: http://www.fit.vutbr.cz/research/groups/speech/publi/2016/brummer_NIST_SRE_2016_ABC_1478007109_abc-nist-sre-systemdescription_v2.pdf

[22] L. Ferrer, H. Bratt, L. Burget, H. Cernocky, O. Glembek, M. Graciarena, A. Lawson, Y. Lei, P. Matejka, O. Plchot, and N. Scheffer, "Promoting robustness for speaker modeling in the community: the PRISM evaluation set," in *Proceedings of SRE11 analysis workshop*, Atlanta, Dec. 2011.

[23] O. Glembek, P. Matějka, O. Plchot, J. Pešán, L. Burget, and P. Schwarz, "Migrating i-vectors Between Speaker Recognition Systems Using Regression Neural Networks," in *Proceedings of Interspeech 2015*, 2015, pp. 2327–2331.

[24] G. Bhattacharya, M. J. Alam, P. Kenny, and V. Gupta, "Modelling speaker and channel variability using deep neural networks for robust speaker verification," in *2016 IEEE Spoken Language Technology Workshop, SLT 2016, San Diego, CA, USA, December 13-16*, 2016.

[25] T. Minka, "Estimating a Dirichlet Distribution," 2012. [Online]. Available: http://www.msr-waypoint.com/en-us/um/people/minka/papers/dirichlet/minka-dirichlet.pdf

[26] N. Brummer, A. Swart, and D. van Leeuwen, "A comparison of linear and non-linear calibrations for speaker recognition," in *Odyssey 2014: The Speaker and language Recognition Workshop*, 2014.

[27] D. Karlis, "An EM Type Algorithm for Maximum Likelihood Estimation for the Normal Inverse Gaussian Distribution," *Statistics & Probability Letters*, March 2002.