On the Use of Convolutional Neural Networks in Pairwise Language Recognition

Alicia Lozano-Diez, Javier Gonzalez-Dominguez, Ruben Zazo,
Daniel Ramos, and Joaquin Gonzalez-Rodriguez

ATVS - Biometric Recognition Group
Universidad Autonoma de Madrid (UAM), Spain
{alicia.lozano,javier.gonzalez,daniel.ramos,joaquin.gonzalez}@uam.es,
ruben.zazo@estudiante.uam.es

Abstract. Convolutional deep neural networks (CDNNs) have been successfully applied to different tasks within the machine learning field, and, in particular, to speech, speaker and language recognition. In this work, we have applied them to pair-wise language recognition tasks. The proposed systems have been evaluated on challenging pairs of languages from NIST LRE'09 dataset. Results have been compared with two spectral systems based on Factor Analysis and Total Variability (i-vector) strategies, respectively. Moreover, a simple fusion of the developed approaches and the reference systems has been performed. Some individual and fusion systems outperform the reference systems, obtaining $\sim 17\%$ of relative improvement in terms of $minC_{DET}$ for one of the challenging pairs.

Keywords: Convolutional networks, CDNNs, pair-wise language recognition.

1 Introduction

Deep Neural Networks (DNNs) are a new paradigm within machine learning. They have shown to be successful in many tasks such as acoustic modelling [16,7,12,8] or speaker recognition [10,4].

Considering this, our work is focused on a related problem: the automatic language recognition (or Spoken Language Recognition, SLR) task. This problem has been addressed for many years by NIST Language Recognition Evaluations (LRE). Many of the state-of-the-art approaches to this problem are based on acoustic systems. For instance, GMM-based systems where a session variability compensation scheme via Factor Analysis (FA) is applied [6], and, also, i-vector approaches that have been proved to be successful to deal with the SLR task [18].

However, new approaches to the problem of SLR based on DNNs have been recently published [15,13]. We propose the use of convolutional deep neural networks (CDNNs), which is a less demanding approach in terms of memory and computational resources than the one proposed in [15]. Moreover, we have applied them to the pair-wise language recognition task, which has been one of

J.L. Navarro Mesa et al. (Eds.): IberSPEECH 2014, LNAI 8854, pp. 79–88, 2014.

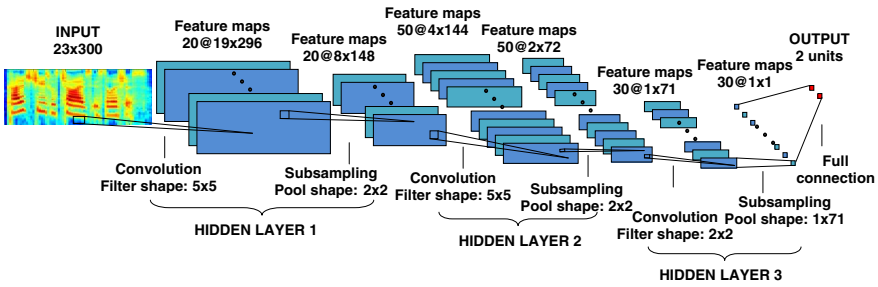

Fig. 1. Representation of architecture used in the experimental part of this work with three hidden layers of 20, 50 and 30 filters respectively (*Model 1*). The other model used (*Model 2*) has the same structure but with 12 filters in each layer.

the tasks proposed by NIST in their Language Recognition Evaluations (LRE). Besides, unlike [13], our proposal is based on the use of CDNNs as a complete system that is directly fed with filter-bank outputs. Figure 1 shows an example of structure used in the experimental part of this work.

The rest of this paper is organized as follows. In Section 2, the proposed system based on CDNNs is described. The reference systems are presented in Section 3 and the database and experimental framework used are exposed in Section 4. Finally, Sections 5 and 6 present the results and conclusions of this work.

2 Convolutional DNNs for Language Recognition

2.1 Convolutional DNNs Architecture

The proposal of this work is to develop a system based on convolutional networks applied to the problem of pairwise language recognition.

CDNNs are models based on the structure of the visual system and are composed of two kinds of layers: convolutional layers and subsampling layers [11]. The first ones act as a feature extractor where each unit is connected to a local subset of units in the layer below. Some units that are related because of their location share their parameters, allowing the network to extract the same features from different locations in the input. This also decreases the amount of parameters to tune. Subsampling layers reduce the size of the representations obtained by convolutional layers by applying a subsampling operation, and making the network, in some way, invariant to small translations and rotations [1]. Moreover, convolutional nets can be trained as a classic *feedforward* network, by using, for instance, supervised learning based on gradient descent algorithms [11].

All these features make them be easier to train and cheaper in terms of resources than other approaches within the *deep learning* paradigm. Then, we have

used them as a complete system to perform pair language recognition. In particular, they have been trained in a supervised way to discriminate between two languages, which are considered challenging pairs due to their similarities. The database used has been a subset of the one provided by NIST in the LRE'09.

2.2 Proposed System: CDNN-Based System

The details of the CDNN-based system are as follows. The input of the network consists of a 2-dimensional *time-frequency* representation of the speech signal. In our case, 23 Mel-scale filter-bank outputs have been used to feed the network for each segment of 3 seconds of speech, normalized to have zero mean and unit variance for each coefficient over the whole training set. Those 3 seconds correspond with 300 frames, since windows of 20 ms of duration have been applied with 10 ms of overlap. Moreover, in order to suppress silences, a voice activity detector based on energy has been used. This last filtering process makes test segments contain less than 3 seconds of actual speech, which was a problem since the network input dimensions were fixed. It was solved by applying a *right padding* by using the first frames of the segment to fit this requirement.

Depending on the configuration of the network, two different models have been considered. Both of them have 3 hidden *convolutional-maxpooling* layers. Each of these layers are composed of two stages: 1) computation of the activation for each hidden unit in each feature map by convolving the input with a linear filter (*weights*), adding a bias term and applying the non-linear transformation *tanh* ($h = tanh(W * x + b)$); and 2) application of a sub-sampling phase based on partitioning the input into non-overlapping regions and choosing the maximum activation of each region. For both models, the shape of the linear filters is 5×5 for the first two hidden layers, and 2×2 for the third one. Regarding the max-pooling regions, they have a shape of 2×2 in the first two hidden layers, and 1×71 in the third one in order to have a single value as output of the last hidden layer. Then, the difference between the two mentioned models relies on the number of filters or *feature maps* considered for each hidden layer, which is related to the idea of how many different features want to be extracted in each layer. The first model (*Model 1*) has 12 filters in each layer and the second one (*Model 2*), has 20, 50 and 30 in each of the three mentioned hidden layers, respectively. All this information is summarized in Table 1.

Table 1. Configuration parameters for the developed models

Conf. Parameter	Model 1	Model 2
# Layers	3	3
# Filters/layer	[12, 12, 12]	[20, 50, 30]
Filter shapes	[(5, 5), (5, 5), (2, 2)]	[(5, 5), (5, 5), (2, 2)]
Pool shapes	[(2, 2), (2, 2), (1, 71)]	[(2, 2), (2, 2), (1, 71)]

As far as the output layer is concerned, it consists of a *fully-connected* layer that computes a $softmax$ function according to the following expression:

$$P(Y = i|x, W, b) = softmax_i(Wx + b) = \frac{e^{W_i x + b_i}}{\sum_j e^{W_j x + b_j}}$$

where i is a certain class, and W and b are the parameters of the model (weights and bias, respectively).

The output value is considered as a score or likelihood measure of belonging to a certain language, between the two languages involved, since the performed experiments are based on *language-pairs*. The final score for a test segment is computed as the difference between the logarithms of each likelihood.

Regarding the training of the network, the algorithm that has been used is the stochastic gradient descent algorithm with a learning rate of 0.1 and based on *minibatches* of 500 samples each one. The cost function that the algorithm tries to optimize (minimize in this case) is the negative log-likelihood, defined as follows:

$$NLL(\theta, D) = -\sum_{i=1}^{|D|} \log P(Y = y^{(i)}|x^{(i)}, \theta)$$

where D is the dataset, θ represents the parameters of the model ($\theta = W, b$, weights and bias respectively), $x^{(i)}$ is an example, $y^{(i)}$ is the label corresponding to example $x^{(i)}$, and P is defined as the output of the $softmax$ function defined above.

Also, an *"early stopping"* technique has been used during the training in order to avoid the *overfiting* problem, so the performance of the model is evaluated in a validation set, and if the improvements over that set are not considered relevant, the training stops.

All this development has been done by using Python and, specifically, *Theano* [2], following the ideas of [14].

3 Reference Systems: FA-GMM and i-Vector

In order to have a baseline to compare with, two different systems have been taken as reference and have been evaluated on the same datasets that the proposed method based on CDNNs.

The first one consists of a Factor Analysis GMM Linear Scoring (FA-GMM-LS) [6], which is a GMM system with linear scoring and session variability compensation applied in the statistic domain. The speech signal is represented by a parameterization consisting of seven MFCCs with CMN-Rasta-Warping concatenated to 7-1-3-7 SDC-MFCCs. Two Universal Background Models (UBMs) with 1024 Gaussian components were trained. One of them (UBM_{CTS}) was trained with Conversational Telephone Speech (hereafter, CTS). The other one (UBM_{VOA}) was train with data from VOA (Voice of America radio broadcasts through Internet), provided by NIST. Thereby, two different systems were developed, one for each UBM. Two session variability subspaces matrices were obtained (U_{CTS} and U_{VOA}). The subspaces were initialized with PCA (Principal Component Analysis) based on [9,20], taking into account just top-50 eigenchannels, and trained by using the EM algorithm.

The second reference system, the i-vector system, is based on GMMs where a Total Variability modeling strategy [3] is employed in order to model both language and session variability. Unlike FA, a *total space* represented by a low-rank T matrix jointly includes language and session variability. Moreover, a session variability compensation stage is applied directly to the low dimensional space driven by T by means of Linear Discriminant Analysis (LDA) and Within-Class Covariance Normalization (WCCN) [5].

The speech signal is represented as in the first reference system and the T matrix has been trained with CTS and broadcast data as well.

Both systems output a score for each test segment computed as the difference between the scores given for each of the two language models involved in each pair.

Moreover, as scores from reference and CDNN-based systems are in the same domain (real numbers), a simple sum fusion has been performed.

4 Database and Experimental Protocol

4.1 Database Description

The database used to perform the experiments has been that provided by NIST in LRE'09 [17].

LRE'09 database includes data coming from different audio sources: conversational telephone speech (CTS), used in previous evaluations, and broadcast data that contain telephone and non-telephone speech. That broadcast data consist of two corpora from past Voice of America (VOA) broadcast in multiple languages (VOA2 and VOA3). Some language labels of VOA2 might be erroneous since they have not been audited. More details can be found in [17].

Regarding evaluation data, segments of 3, 10 and 30 second of duration from CTS and broadcast speech data are available to test the developed systems. However, the experiments shown in this paper are only based on segments of 3 seconds (*short duration*).

We have selected five challenging pairs of languages for the experiments of this work: Bosnian-Croatian (BC), Farsi-Dari (FD), Hindi-Urdu (HU), Portuguese-

Table 2. Amount of data used for the experiments per language (in hours)

	Amount of data (# Hours)		
	Training	**Validation**	**Test**
Bosnian	12.27	5.26	0.28
Croatian	9.16	3.92	0.29
Dari	25	10.72	1.07
Farsi	25	10.72	0.28
Hindi	25.9	11.10	0.51
Portuguese	11.79	5.05	0.32
Russian	20.27	8.69	0.66
Spanish	13.85	5.94	0.31
Ukrainian	15.89	6.81	0.31
Urdu	26.63	11.41	0.29

Spanish (PS) and Russian-Ukrainian (RU). These pairs are among the proposed tasks of the language-pair evaluation in the NIST LRE'09, since they are considered of particular interest due to their similarities. Indeed, all of them except Portuguese-Spanish are considered mutually intelligible.

The available datasets have been split into three separate subsets: training, validation and test. The first two datasets includes just broadcast data (VOA2 and VOA3) from the development data provided by NIST LRE'09. However, test segments come from CTS and VOA datasets and are the actual evaluation data of NIST LRE'09. The amount of data (in hours) per language used in the experiments is shown in Table 2.

4.2 Performance Evaluation

The performance of the systems has been evaluated according to the cost measure (C_{DET}) defined in the NIST LRE'09 evaluation plan [17]. This measure takes into account the false alarm and false rejection probabilities and the cost of a bad classification of the segment of speech. As this measure shows the cost with the optimal threshold, it corresponds with the minimum cost operating point, so we will refer to it as $minC_{DET}$ [19].

Furthermore, DET curves have been used in order to evaluate the performance of the systems in different operating points. In the legend of the DET curves shown in Section 5 the EER (in %) is also shown.

Apart from the performance evaluation of the individual systems considered in this work, the performance of fusion systems has been also included. Those fusion schemes consist of a score level fusion where both mentioned reference systems (FA-GMM and i-vector) and the corresponding CDNN-based model are involved. A simple sum of the scores output by each system involved in the fusion scheme has been used to obtain the final score for a certain segment of speech.

Table 3. Performance of individual (left) and fusion (right) systems ($minC_{DET} \times 100$)

	Individual Systems				Fusion Systems	
	Reference		CDNNs		Ref. Systems +	Ref. Systems +
	FA-GMM	i-vector	Model1	Model 2	Model1	Model 2
BC	34.45	37.24	34.89	37.76	**32.13**	35.48
FD	**33.81**	45.51	49.92	49.88	49.46	49.79
HU	43.30	41.93	36.09	37.72	**35.16**	36.90
PS	11.51	**9.15**	17.08	14.71	10.07	9.53
RU	35.29	**35.06**	45.69	44.53	41.72	42.50

5 Results

The experiments shown in this paper are based on the five challenging language pairs mentioned in Section 4.1. For each of these pairs, two different models (according to the configurations shown in Table 1) and the two reference systems described in Section 3 have been evaluated on the same test samples. Furthermore, the amount of data used for training the CDNN-based system (see Table 2) is approximately the same that the used for training the reference systems, although some languages datasets have been reduced in order not to have big differences between the datasets of the two languages involved in each experiment.

The performance of each individual system can be seen in the left side of Table 3. According to the results, CDNN-based models outperform the best reference systems in the case of Hindi-Urdu, with a relative improvement of $\sim$14% in $minC_{DET}$. As it is shown in the right side of Table 3, by performing a simple sum-fusion of the reference systems and the CDNNs systems, the relative improvement yields up to $\sim$ 17% for the Hindi-Urdu pair. For the Bosnian-Croatian experiment, the fusion system gives $\sim$ 7% of relative improvement, and the performances of all individual systems are pretty similar for this pair.

By way of contrast, the models obtained for the language-pairs Farsi-Dari, Portuguese-Spanish and Russian-Ukrainian, even the fusion ones, give worse results than those yielded by the reference systems. Possible reasons might be that the configuration parameters used are not adequate for the available data or that the development dataset has not been adequately selected (with little variability among utterances).

Regarding the comparison between the two CDNN-models, although *Model 2* has more filters (*feature maps*) and, thereby, its capability to extract a better abstract representation of the input signal is bigger, just in three pairs it gives better results than *Model 1*. This might be caused by a lack of data or variability within them that leads to the problem of *overfitting*. More evidence of occurrence of that problem is that we have observed a big *gap* between validation and test errors.

Finally, Figure 2 shows the DET curves obtained for each language-pair according to the performance of both reference systems, the best CDNN system and the best fusion model. As it was observed with the $minC_{DET}$ performance

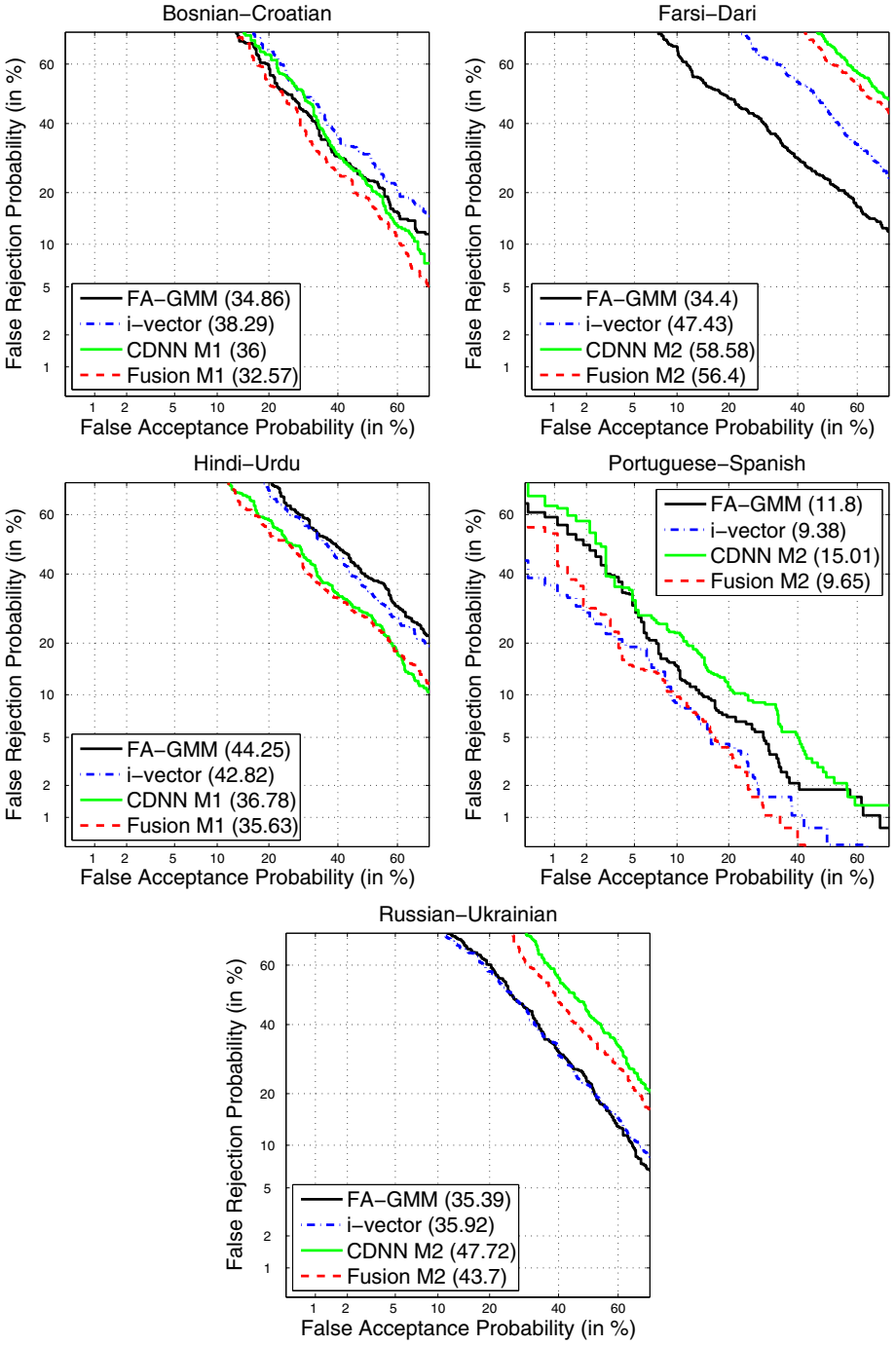

Fig. 2. DET curves corresponding to reference systems, the best CDNN system and the best fusion according to the EER for each language pair. The EER (in %) is shown in brackets.

measure, our individual approach outperforms the reference systems in the experiments with Hindi-Urdu, and the fusion one, in the Bosnian-Croatian pair. Relative improvements and general behaviour or the systems are similar to the one observed with $minC_{DET}$ measure.

6 Conclusions

Considering recent work, CDNNs can be considered a powerful tool to be applied to SLR tasks with a tractable amount of data. It can be considered one of the less costly approaches within the deep learning paradigm.

In this work, we have applied them to the problem of language-pair recognition. The proposed systems have been trained to discriminate between two languages, which are considered challenging due to their similarities. Results have been compared with the ones obtained from two spectral systems.

The proposed models manage to outperform the reference systems in two out of the five pairs considered. It should be pointed out that the test utterances have a duration of just 3 seconds of speech. Moreover, the CDNN systems are fed with the Mel-Filter bank outputs in blocks of 3 seconds. However, this can be considered an exploratory work and more configurations and different treatment of data should be studied.

Acknowledgments. This work has been developed within the project *CMCV2: Caracterización, Modelado y Compensación de Variabilidad en la Señal de Voz* (TEC2012-37585-C02-01), funded by *Ministerio de Economía y Competitividad*, Spain.

References

1. Bengio, Y.: Learning deep architectures for AI. Foundations and Trends in Machine Learning 2(1), 1–127 (2009), also published as a book. Now Publishers (2009)
2. Bergstra, J., Breuleux, O., Bastien, F., Lamblin, P., Pascanu, R., Desjardins, G., Turian, J., Warde-Farley, D., Bengio, Y.: Theano: a CPU and GPU math expression compiler. In: Proceedings of the Python for Scientific Computing Conference (SciPy) (June 2010), oral Presentation
3. Dehak, N., Kenny, P., Dehak, R., Glembek, O., Dumouchel, P., Burget, L., Hubeika, V., Castaldo, F.: Support vector machines and joint factor analysis for speaker verification. In: ICASSP, pp. 4237–4240 (2009)
4. Ghahabi, O., Hernando, J.: i-vector modeling with deep belief networks for multi-session speaker recognition. In: Proc. ODYSSEY (2014)
5. Gonzalez-Dominguez, J., Lopez-Moreno, I., Franco-Pedroso, J., Ramos, D., Toledano, D.T., Gonzalez-Rodriguez, J.: Atvs-uam nist sre 2010 system. In: Proceedings of FALA 2010 (November 2010)
6. Gonzalez-Dominguez, J., Lopez-Moreno, I., Franco-Pedroso, J., Ramos, D., Toledano, D.T., Gonzalez-Rodriguez, J.: Multilevel and session variability compensated language recognition: Atvs-uam systems at nist lre 2009. IEEE Journal on Selected Topics in Signal Processing (2010) (article in press)

7. Hinton, G., Deng, L., Yu, D., Dahl, G., Rahman Mohamed, A., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T., Kingsbury, B.: Deep neural networks for acoustic modeling in speech recognition. Signal Processing Magazine (2012)
8. Jaitly, N., Nguyen, P., Senior, A., Vanhoucke, V.: Application of pretrained deep neural networks to large vocabulary speech recognition. In: Proceedings of Interspeech 2012 (2012)
9. Kenny, P., Boulianne, G., Dumouchel, P.: Eigenvoice modeling with sparse training data. IEEE Transactions on Speech and Audio Processing 13(3), 345–354 (2005)
10. Kenny, P., Gupta, V., Stafylakis, T., Ouellet, P., Alam, J.: Deep neural networks for extracting baum-welch statistics for speaker recognition. In: Proc. ODYSSEY (2014)
11. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. In: Intelligent Signal Processing, pp. 306–351. IEEE Press (2001)
12. Lee, H., Largman, Y., Pham, P., Ng, A.Y.: Unsupervised feature learning for audio classification using convolutional deep belief networks. In: Advances in Neural Information Processing Systems 22, pp. 1096–1104 (2009)
13. Lei, Y., Ferrer, L., Lawson, A., McLaren, M., Scheffer, N.: Application of convolutional neural networks to language identification in noisy conditions. In: Proc. ODYSSEY (2014)
14. LISA: Deep Learning Tutorial. University of Montreal, `http://deeplearning.net/tutorial/`
15. Lopez-Moreno, I., Gonzalez-Dominguez, J., Plchot, O.: Automatic language identification using deep neural networks. In: Proc. ICASSP (2014)
16. Mohamed, A.R., Dahl, G.E., Hinton, G.: Acoustic modeling using deep belief networks. IEEE Trans. on Audio, Speech and Language Processing, `http://www.cs.toronto.edu/~hinton/absps/speechDBN_jrnl.pdf`
17. NIST: The 2009 nist language recognition evaluation plan (2009), `http://www.itl.nist.gov/iad/mig/tests/lre/2009/LRE09_EvalPlan_v6.pdf`
18. Penagarikano, M., Varona, A., Diez, M., Rodriguez-Fuentes, L.J., Bordel, G.: Study of different backends in a state-of-the-art language recognition system. In: INTERSPEECH (2012)
19. Van Leeuwen, D.A., Brummer, N.: Channel-dependent gmm and multi-class logistic regression models for language recognition. In: IEEE Odyssey 2006: The Speaker and Language Recognition Workshop, pp. 1–8. IEEE (2006)
20. Vogt, R., Sridharan, S.: Explicit modelling of session variability for speaker verification. Computer Speech & Language 22(1), 17–38 (2008)