INDEPENDENT COMPONENT ANALYSIS AND MLLR TRANSFORMS FOR SPEAKER IDENTIFICATION

Sandro Cumani[1], *Oldřich Plchot*[2], *Martin Karafiát*[2]

(1) Politecnico di Torino, Italy
(2) Brno University of Technology, Speech@FIT, Czech Republic

ABSTRACT

In this paper, we explore the use of Independent Component Analysis (ICA) and Principal Component Analysis (PCA) techniques to reduce the dimensionality of high-level LVCSR features and at the same time to enable modelling them with state-of-the-art techniques like Probabilistic Linear Discriminant Analysis or Pairwise Support Vector Machines (PSVM). The high-level features are the coefficients from Constrained Maximum-Likelihood Linear Regression (CMLLR) and Maximum-Likelihood Linear Regression (MLLR) transforms estimated in an Automatic Speech Recognition (ASR) system. We also compare a classical approach of modeling every speaker by a single SVM classifier with the recent state-of-the-art modelling techniques in Speaker Identification. We report performance of the systems and score-level combination with a current state-of-the-art acoustic i–vector system on the NIST SRE2010 dataset.

Index Terms— Speaker Recognition, MLLR, ICA, PLDA, SVM

1. INTRODUCTION

Maximum–Likelihood Linear Regression (MLLR) transforms [1, 2], while not achieving by themselves state–of–the–art performance for speaker recognition, have been successfully used in combination with cepstral systems to improve recognition accuracy [3]. While cepstral systems have greatly evolved in the last few years, however, MLLR–based systems for speaker recognition are still mostly tied to the Support Vector Machine with Nuisance Attribute Projection framework [3]. Cepstral systems have first moved towards Joint Factor Analysis based techniques [4], and, more recently, a JFA–derived new representation of an entire utterance called *i–vector* [5] has opened the way to successful generative and discriminative models (e.g. Probabilistic Linear Discriminant Analysis [6, 7] and pairwise SVM [8, 9]) which work directly on these low–dimensional features.

Only very recently some effort was done to investigate how some of these techniques can be applied to MLLR–based systems [10]. The results of this work show that MLLR features compressed with Probabilistic Principal Component Analysis (PPCA) and combined with generative cepstral models like PLDA can achieve better performance than NAP–SVM based techniques.

Starting from these results, we investigate an extension of PCA, known as Independent Component Analysis [11, 12, 13], which has proven to be able to give better results than PCA in different contexts (e.g. face recognition [14, 15]) . The goal of our work is to show that an ICA–based system can produce low–dimensional MLLR features that can be effectively modeled using techniques developed for i – vectors. In particular, we show that both pairwise SVM (PSVM) and Probabilistic Linear Discriminant Analysis (PLDA) achieve better results than the SVM–NAP based system and that ICA–based dimensionality reduction outperforms simple PCA. Finally, we analyze how these techniques combine with a cepstral i–vector system.

The paper is organized as follows. Section 2 describes MLLR and Constrained MLLR transforms. In Section 3 we recall the classical SVM–NAP model. Section 4 presents the MLLR dimensionality reduction process for PCA and ICA–based systems. The description of the PLDA and PSVM classifiers are given in Section 5. Experimental results are presented in Section 6 and conclusions are given in Section 7.

2. MLLR

Due to the sparsity of training data, it is usually not possible to train speaker-dependent models for the LVCSR system. MLLR and its variant CMLLR are techniques used to adapt speaker-independent models on the small amount of available speaker-specific data. MLLR is a set of linear transforms, operating in the space of the model parameters, which maximizes the likelihood of the adaptation data by rotating all HMM model parameters. The transforms are estimated using the EM algorithm [16, 1].

CMLLR is similar to MLLR, but mean and variance transforms are constrained to be the same [17]. CMLLR has a significant advantage, it can be applied online by transforming only the input features and therefore, there is no need to transform model parameters [2] . Note, that this attribute of CMLLR causes it to be also called feature-space MLLR (FMLLR).

3. SVM-NAP TECHNIQUES FOR MLLR

The use of Support Vector Machines combined with Nuisance attribute Projection has allowed in the past to build discriminative speaker recognition systems based on MLLR transforms which provide useful complementary information to cepstral–based models [3]. In this framework, SVMs are used in a one–versus–all fashion, that is, for each enrollment speaker an SVM is trained to separate the utterances of that speaker from a set of background impostors utterances. The resulting hyperplane is then used to score test segments.

The main issue of this approach is that the SVM classes are highly unbalanced: the utterances for the enrolled speaker are very few, and in some cases only one utterance is available. Nevertheless, the use of techniques for nuisance compensation such as Nuisance

This work was partly supported by Technology Agency of the Czech Republic project No. TA01011328, Czech Ministry of Education project No. MSM0021630528 and Grant Agency of the Czech Republic project No. 102/08/0707.

Attribute Projection (NAP) have allowed in the past one–versus–all SVM systems to achieve good performances for both cepstral [18] and MLLR–based systems [3].

4. MLLR PREPROCESSING

4.1. Rank normalization

In SVM–NAP based systems rank normalization can be useful to increase performance of the system [19]. For every dimension, a feature value is replaced by the position the feature would have occupied in the ordered set of values taken from a background training set. In the following we assume MLLR features have been rank–normalized.

4.2. Probabilistic Principal Component Analysis

Principal Component Analysis is a well–known feature reduction technique [20]. PCA can be defined as the linear projection that minimizes the average reconstruction cost for the data, where the cost is given by the mean squared distance between data points and their projections. The solution is given by the eigenvectors of the data covariance matrix corresponding to the largest eigenvalues.

A probabilistic formulation of PCA, known as Probabilistic Principal Component Analysis, describes PCA as the maximum–likelihood solution of a latent variable probabilistic model [20]. The model is given by

$$\mathbf{x} = W\mathbf{s} + \epsilon,$$

where $\mathbf{x}$ is assumed to be a zero–mean d–dimensional observable random variable, $\mathbf{s}$ is a p–dimensional Gaussian distributed latent random variable and ϵ represents Gaussian distributed random noise. PPCA solution can then be estimated by means of EM algorithm [20]. It is possible to show that the subspace spanned by the PPCA is the same subspace identified by standard PCA [20].

4.3. Independent Component Analysis

Independent Component Analysis is a technique that allows to linearly transform a multidimensional random vector into statistically independent components [11, 13]. ICA can be interpreted as an extension to PCA, in the sense that PCA looks for dimensions which decorrelate data, while ICA looks for dimensions which make data independent.

Given a zero–mean random vector of observable variables $\mathbf{x} = (x_1, \ldots, x_d)$, the ICA problem can be formulated [12] as the estimation of a transformation matrix A which maps a (latent) p–dimensional variable $\mathbf{s}$ into $\mathbf{x}$ according to

$$\mathbf{x} = A\mathbf{s}$$

under the assumption that the components of $\mathbf{s}$ are statistically independent. Independent components are assumed to be non–Gaussian.

Another formulation for ICA was given in [11]. In this interpretation, ICA looks for an invertible transformation matrix W such that $\mathbf{s} = W\mathbf{x}$ that minimizes the mutual information of the variables s_i [13].

ICA has been successfully applied in the past for face recognition problems [14, 15], and two different flavors of ICA have been developed, known as Architecture I and Architecture II.

4.3.1. Architecture I

In Architecture I MLLR features are considered as the linear combination of statistically independent basis MLLR transforms. ICA learns the transformation matrix W in order to estimate this statistically independent MLLR basis [14, 15]. PCA is used to reduce dimensionality and to whiten the MLLRs before ICA is computed. In this case, each MLLR is transformed to a zero–mean vector before PCA and ICA are computed.

4.3.2. Architecture II

In Architecture I we look for a set of independent basis vectors. The independence of the basis, however, does not imply the independence of the coefficient of the projected MLLR features. Architecture II looks for a transformation matrix such that the transformed coefficients are statistically independent [15]. Again, PCA is used to reduce dimensionality and to whiten the data. This time each single MLLR dimension is transformed to a zero–mean vector over the different utterances.

5. PLDA AND PAIRWISE SVM

After the introduction of i–vectors cepstral modelling started to focus on low–dimensional generative and discriminative techniques for i–vector based speaker identification. One of the most successful techniques is, in this field, the Probabilistic Linear Discriminant (PLDA) model, a generative model whose different flavors have proven to achieve state–of–the–art results [7, 21]. Starting from PLDA, a new framework for discriminative training of speaker recognition systems was recently introduced in [8, 9], showing that SVM–based systems can also achieve state–of–the–art performance for i–vector modelling.

5.1. PLDA

Probabilistic Linear Discriminant Analysis describes the process of generation of an observed feature vector ϕ using a latent variable model

$$\phi = m + U_1 y + U_2 x + \epsilon,$$

where y and x are latent variables related to speaker identity and channel respectively, usually referred to as speaker factor and channel factor, ϵ represent residual noise and U_1 and U_2 are low–rank matrices used to constrain the dimensionality of speaker and channel subspaces. More details about the PLDA model can be found in [7] and [21].

5.2. Pairwise Discriminative Training

The pairwise SVM framework presented in [8, 9] proves to be a competitive discriminative technique for cepstral–based speaker recognition. Instead of training one SVM for each speaker as in the SVM–NAP system, a single model is built which is able to directly score a pair of utterances as belonging to the same speaker or not. This framework not only allows to achieve state–of–the–art performance in i–vector modelling, but allows for fast testing of utterance pairs (trials), since there's no need to train a SVM for each speaker.

6. EXPERIMENTS

In this section we analyze the performance of our featured generative and discriminative systems compared to the standard NAP–SVM ap-

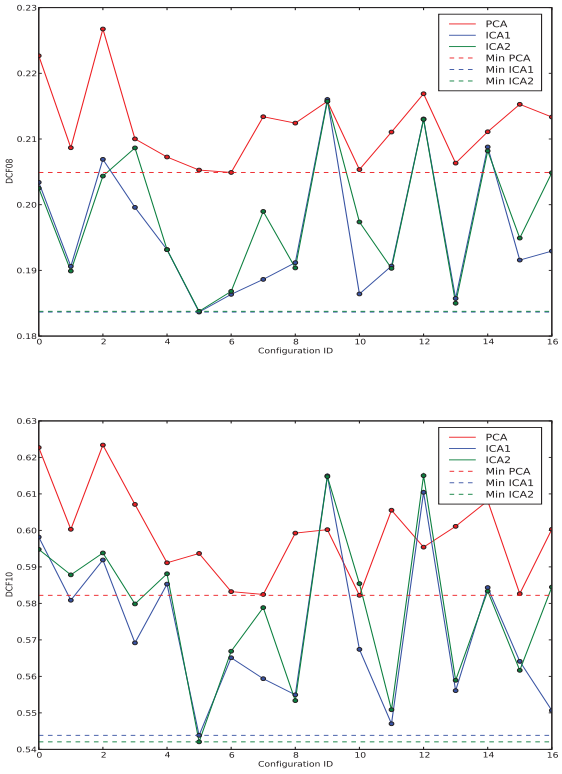

Fig. 1. Female set – DCF08 and DCF10 for different subspace configurations. The dotted lines represents the DCF of the best configuration for each subsystem

proach. Results are given for the 2010 Speaker Recognition Evaluation [22] extended tel-tel condition for both female and male set and are expressed in terms of Equal Error Rate, Minimum Detection Cost Function as proposed by NIST for 2008 SRE (DCF08) and as proposed for 2010 SRE (DCF10).

Our baseline is the NAP–SVM system used by the ABC site for the 2010 SRE submission [19]

6.1. MLLR extraction

An LVCSR system was trained on a set 2000 hours of Fisher data and 300 hours of Switchboard and Callhome data. Used features were PLP12_0_D_A_T (in HTK notation) with VTLN applied and HLDA used for dimensionality reduction. Speaker adaptive training was done using fMPE + MPE models with crossword triphones. 2–class CMLLR was used to model speech and silence, and 3–class MLLR was used to model 2 data clusters and silence. Only speech–related matrices are stacked together for our system resulting in feature vectors of dimensionality 4680. As ASR transcription the NIST transcript were used. While phoneme alignment was estimated using VTLN features, the MLLR and CMLLR transformation matrices are estimated using non-VTLN features.

6.2. NAP–SVM system

MLLR and CMLLR matrices are stacked and rank–normalized as described in section 4 using NIST 2004 and 2005 telephone data. Nuisance Attribute Projection is used to compensate for channel and noise effects. Two NAP matrices are computed: the first is trained on NIST SRE 2004 and 2005 telephone data, while the second is trained on NIST SRE2005 and 2006 microphone data. 20 dimensions from the first and 10 dimensions from the second were used for NAP. The background cohort for SVM training consists of NIST 2004 and 2005 data. For a more detailed description of this baseline system refer to [19].

Table 1. Results for the PLDA system. PCA/ICA dimensions are indicated in parentheses. LDA and PLDA dimensions are both set to 250. Features are length–normalized

System	Female Set			Male set		
	EER	DCF08	DCF10	EER	DCF08	DCF10
MLLR Baseline						
SVM–NAP	6.02%	0.255	**0.559**	6.43%	0.218	0.524
MLLR PSVM SYSTEM						
PCA	5.42%	0.248	0.642	4.99%	0.204	0.525
ICA1	**4.48%**	**0.214**	0.613	4.59%	**0.189**	0.482
ICA2	4.48%	0.215	0.612	**4.56%**	**0.189**	**0.480**
MLLR PLDA SYSTEM						
PCA (400)	4.22%	0.205	0.582	**4.15%**	0.180	0.520
ICA1 (400)	3.84%	0.186	0.567	**4.15%**	**0.172**	**0.470**
ICA2 (400)	4.16%	0.197	0.585	4.16%	0.173	0.473
PCA (600)	4.19%	0.205	0.594	4.76%	0.209	0.527
ICA1 (600)	**3.80%**	**0.184**	**0.544**	4.56%	0.192	0.494
ICA2 (600)	3.83%	0.184	0.542	4.53%	0.189	0.506

6.3. ICA and pairwise SVM

Both ICA and PCA are trained on the same lists used for the SVM–NAP system, comprising SRE 2004 and 2005 data. The same data is also used to train the PSVM and PLDA systems. For the PLDA system we add an LDA–WCCN step followed by length normalization after ICA and PCA [21]. WCCN is applied to the ICA–projected MLLR before SVM training [8].

6.4. Results

A first set of experiments was conducted to evaluate the performance of the models with different subspaces dimensions. We tested 200, 400 and 600 PCA/ICA subspaces combined with different LDA dimensions (100, 150, 250) and different sizes for the PLDA speaker factor subspace (100, 150, 250). The results are summarized in Figure 1, which shows DCF08 and DCF10 for the female set for different configurations for the three subsystems based on PCA, ICA Architecture I (ICA1) and ICA Architecture 2 (ICA2). Due to lack of space the details for all the configurations are not described. However, we can observe that both ICA architectures achieve better performance than PCA in almost all the considered configurations, and the best ICA configurations outperform the best PCA configuration.

Comparing the results for the different techniques we observed that the best results for PCA are obtained with 400 dimensional PCA followed by 250 dimensional LDA and 250 dimensional speaker subspace for PLDA for both the male and female set. The same configuration achieves the best performance also for ICA–based systems on the male set, although 600 dimensional ICA gives a small improvement on the female set. The results for these conditions are shown in the last block of Table 1. Since the same configuration (400 dimension for PCA/ICA, 250 for LDA and 250 for PLDA speaker factors) achieves almost always the best performance for both PCA and ICA, we now restrict our analysis for the PLDA system to these settings.

Table 2. Fusions of a cepstral system with PLDA and PSVM MLLR systems based on 400–dimensional ICA Architecture I

System	Female Set			Male set		
	EER	DCF08	DCF10	EER	DCF08	DCF10
I–vectors PLDA	2.05%	0.104	0.348	1.27%	0.067	0.318
I–vectors PLDA + MLLR PSVM ICA1	1.94%	**0.096**	0.322	1.32%	0.066	**0.272**
I–vectors PLDA + MLLR PLDA ICA1	**1.89%**	**0.096**	**0.319**	**1.21%**	**0.064**	0.285

For pairwise SVM we decided to train a single configuration based on 400–dimensional PCA/ICA. The choice of the systems was dictated by the results of the PLDA systems. The results are shown in Table 1. PSVM achieves on average better performances than SVM–NAP, with ICA giving better results than PCA. However, the results are worse than those obtained by the PLDA system. We believe that this might be due to the low number of available training patterns, which negatively affects the discriminative systems more than the generative PLDA systems.

Finally, Table 2 shows the results obtained by score–level fusion of the ICA–based MLLR systems and a cepstral i–vector system. 400–dimensional i–vectors are used to train a PLDA system with 150 dimensional speaker subspace. WCCN and length normalization is applied to i–vectors before PLDA training. The cepstral system is trained on NIST 2004, 2005 and 2006 data. As expected, the fusion of MLLR systems with a cepstral system allows to improve the recognition performance.

7. CONCLUSIONS

We presented a new technique for extracting low–dimensional features from MLLR tansforms based on Independent Component Analysis. ICA–based feature reduction proves to perform better than PCA–based dimensionality reduction when combined with PLDA and Pairwise SVM modelling, and both ICA and PCA–based PLDA and PSVM outperform the SVM–NAP approach. PLDA gives, on average, better results than the PSVM approach. However, fusion with a state–of–the–art cepstral system shows that both PLDA and PSVM can be effectively used to improve cepstral models.

8. REFERENCES

[1] C.J. Leggetter and P.C. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density hmms," in *Computer, Speech and Language*, 1995, vol. 9, pp. 171–186.

[2] M.J.F. Gales, "Maximum likelihood linear transformations for hmm-based speech recognition," *Computer Speech and Language*, vol. 12, pp. 75–98, 1998.

[3] A. Stolcke, S. Kajarekar, L., and E. Shriberg, "Speaker recognition with session variability normalization based on mllr adaptation transforms," *IEEE Trans. Audio*, pp. 1987–1998, 2007.

[4] Patrick Kenny et al., "A study of interspeaker variability in speaker verification.," *IEEE Transactions on Audio, Speech and Language Processing*, pp. 980–988, 2008.

[5] N. Dehak et al., "Support Vector Machines versus fast scoring in the low-dimensional total variability space for speaker verification," in *Proc. of Interspeech 2009*, 2009, pp. 1559–1562.

[6] S. J. D. Prince and J. H. Elder, "Probabilistic Linear Discriminant Analysis for inferences about identity," in *11th International Conference on Computer Vision*, 2007, pp. 1–8.

[7] P. Kenny, "Bayesian speaker verification with heavy–tailed priors," in *Keynote presentation, Odyssey 2010*, 2010.

[8] S. Cumani et al., "Fast discriminative speaker verification in the i-vector space," in *Proc. of ICASSP 2011*, 2011, pp. 4852–4855.

[9] L. Burget et al., "Discriminatively trained Probabilistic Linear Discriminant Analysis for speaker verification," in *Proc. of ICASSP 2011*, 2011, pp. 4833–4836.

[10] N. Scheffer, Y. Lei, and L. Ferrer, "Factor analysis back ends for MLLR transforms in speaker recognition," in *Proc. of Interspeech 2011*, 2011, pp. 257–260.

[11] P. Comon, "Independent component analysis, a new concept?," *Signal Process.*, vol. 36, no. 3, pp. 287–314, Apr. 1994.

[12] S. Amari, A. Cichocki, and H. H. Yang, "A new learning algorithm for blind signal separation," in *Advances in Neural Information Processing Systems*. 1996, pp. 757–763, MIT Press.

[13] A. Hyvärinen, "Fast and robust fixed–point algorithms for independent component analysis," *IEEE Transactions on Neural Networks*, vol. 10, no. 3, pp. 626–634, 1999.

[14] M. S. Bartlett, J. R. Movellan, and T. J. Sejnowski, "Face recognition by independent component analysis," *IEEE Transactions on Neural Networks*, vol. 13, no. 6, pp. 1450–1464, Nov. 2002.

[15] Bruce A. Draper et al., "Recognizing faces with pca and ica," in *Computer Vsion and Image Understanding, Special Issue on Face Recognition*, 2003, pp. 115–137.

[16] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," .

[17] V. Digalakis and L. Neumeyer, "Speaker adaptation using combined transformation and bayesian methods," in *Proc. ICASSP '95*, Detroit, MI, 1995, pp. 680–683.

[18] W. M. Campbell et al., "SVM based speaker verification using a GMM supervector kernel and NAP variability compensation," in *Proc. of ICASSP 2006*, 2006.

[19] N. Brümmer et al., "Abc system description for nist sre 2010," in *Proc. NIST 2010 Speaker Recognition Evaluation*, 2010, pp. 1–20.

[20] Christopher M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*, Springer, 1st ed. 2006. corr. 2nd printing edition, Oct. 2007.

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

[22] NIST, "The NIST year 2008 and 2010 Speaker Recognition Evaluation plans," http://www.itl.nist.gov/iad/mig/tests/sre.