MLP Internal Representation as Discriminative Features for Improved Speaker Recognition

Dalei Wu, Andrew C. Morris and Jacques Koreman

Institute of Phonetics, Saarland University, P.O. Box 15 11 50,
D-66041 Saarbrücken, Germany
{daleiwu,amorris,jkoreman}@coli.uni-saarland.de
http://www.coli.uni-saarland.de

Abstract. Feature projection by non-linear discriminant analysis (NLDA) can substantially increase classification performance. In automatic speech recognition (ASR) the projection provided by the pre-squashed outputs from a one hidden layer multi-layer perceptron (MLP) trained to recognise speech sub-units (phonemes) has previously been shown to significantly increase ASR performance. An analogous approach cannot be applied directly to speaker recognition because there is no recognised set of "speaker sub-units" to provide a finite set of MLP target classes, and it for many applications it is not practical to train an MLP with one output for each target speaker. In this paper we show that the output from the second hidden layer of an MLP with three hidden layers, trained to identify a subset of 100 speakers selected at random from the full set of 630 speakers in Timit, can provide a 77% relative error reduction for common Gaussian mixture model (GMM) based speaker identification.

1 Introduction

NLDA based data enhancement by MLP has proved to be very effective for improving performance in automatic *speech* recognition [6, 13]. This has been achieved by training an MLP with one output per phoneme to estimate phoneme posterior probabilities, and then using this MLP to project each data frame onto an internal representation of the data which the MLP has learnt (see Figure 1). This representation may be the input to or output from one of its hidden layers or the input to its output layer, i.e. the "pre-squashed MLP outputs" (see Figure 2).

The success of this simple data-driven approach to data enhancement in ASR has led to analogous procedures being attempted for *speaker* recognition. Despite some positive results with LDA based feature enhancement for speaker recognition [8], application of the more powerful NLDA based enhancement technique has had limited success, except when used in combination with other techniques [9].

There are both practical and theoretical reasons for the lack of success of NLDA based enhancement for speaker recognition. From the practical point of view, if the MLP has one output for each speaker in the closed speaker set then it would require retraining every time a new speaker is added, while from the theoretical point of view, when the number of speakers is large the number of free parameters in the MLP

would become so great that it could not learn to generalise well from the limited training data available. Furthermore, while phoneme data is well clustered and relatively easy to classify, data for each speaker is clustered around every phoneme centre and is therefore harder to separate so that the MLP classification error remains high, in which case the features it generates may reduce, rather than enhance, speaker recognition performance [9]. Reasoning that

> *the internal representation which the MLP learns to enhance separation between a small number of speakers (covering the required range of speaker types) should also be of some use in separating other speakers*,

in this paper we train an MLP to recognise (i.e. estimate posterior probabilities for) a limited number of speakers selected at random from the population. By limiting the number of speakers on which the MLP is trained, both the practical and theoretical problems mentioned above are avoided.

Before training the speaker model for each new speaker to be enrolled into the GMM or HMM based speaker recognition system, and also before processing the data for a speaker to be recognised, each frame of speech data is now projected through the first few layers of this MLP onto its discriminative internal representation (see Fig.1).

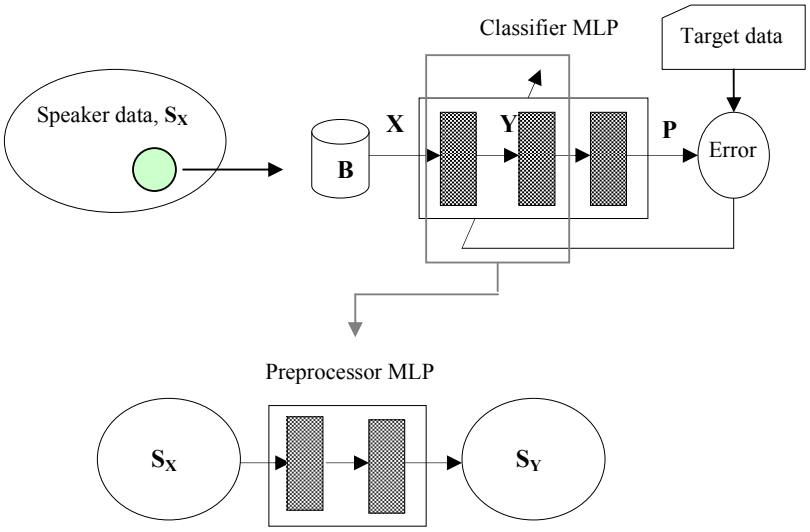

Fig. 1. Data enhancement procedure. A small random set of basis speakers, B, is selected. This is used to train an MLP with several hidden layers to estimate a-posteriori probabilities (P) only for speakers in B. All data S_X from speakers in the full closed set of speakers to be recognised is then passed through the first 2 layers of the trained MLP to produce new data features S_Y, with enhanced speaker discrimination.

Other approaches to harness the discriminative power of MLPs for speaker recognition, such as [7], tend to be rather complex compared to the conceptually simple direct application of MLPs for data enhancement.

In Section 2 we present the baseline GMM based speaker identification model whose performance we are aiming to improve [10]. In Section 3 we give the procedure used for the design and training of the MLP which we use for data enhancement. Section 4 describes the data features and procedures used for system testing, and in Section 5 we present experimental results. These results show that the data enhancement procedure described can give significantly improved speaker recognition performance. This is followed by a discussion and conclusion.

2 Speaker Identification Baseline

In GMM based speaker identification a GMM data pdf $p(x|S)$ (1) is trained for each speaker for some fixed number M of Gaussians. This models the pdf for a single data frame, x, taking no account of the time order of the data frames in the full speech sample, X.

$$p(x|S) = \sum_{i=1}^{M} P(m_i)\, p(x|m_i, S) \tag{1}$$

When the speaker needs to be identified from speech data X, the speaker S_k is chosen as the most probable (2).

$$\begin{aligned} S_k &= \arg\max_{S_k} P(S_k|X) \cong \arg\max_{S_k} P(X|S_k) \\ &= \arg\max_{S_k} \log p(X|S_k) = \arg\max_{S_k} \sum_t \log p(x_t|S_k) \end{aligned} \tag{2}$$

The GMM design, feature data and database used here (32 Gaussians, MFCC features, Timit) are taken from [12]. This simple model gives state-of-the-art speaker recognition performance. With Timit (though not with other databases, such as the CSLU speaker recognition database) no gain is found in training speaker models by adaptation from a global model.

As in [12], GMMs were trained by k-means clustering, followed by EM iteration. This was performed by the Torch machine learning API [3]. We used a variance threshold factor of 0.01 and minimum Gaussian weight of 0.05 (performance falling sharply if either was halved or doubled).

3 MLP Design and Training

The four MLP types tested are shown in Figure 2. Types *a,b,c* have previously been used successfully for data enhancement in ASR [6,13]. These are all feedforward MLPs in which each layer is fully connected to the next. The "neurons" in each layer comprise the usual linear net-input function followed by a non-linear squashing func-

tion, which is the sigmoid function for all layers except the output layer, which uses the softmax function to ensure that all outputs are positive and sum to 1 [1].

Also using Torch [3], each MLP is trained, by gradient descent, to maximise the cross entropy objective (i.e. the mutual information between the actual and target outputs). We trained in batch mode, with a fixed learning rate of 0.01. The data in each utterance was first normalised to have zero mean and unit variance. The estimated probabilities are often close to 0 or 1 and data with such a peaked distribution is not well suited as feature data. The enhanced features taken from the trained MLP of types *a* and *b* are therefore usually taken as the net input values in the output layer, prior to squashing. For type *c* they are normally taken as the squashed output from the last hidden layer (these values having less peaked distributions than the outputs from the output layer), but here we have taken the enhanced features from MLPs *c* and *d* both as the net input to the second hidden layer.

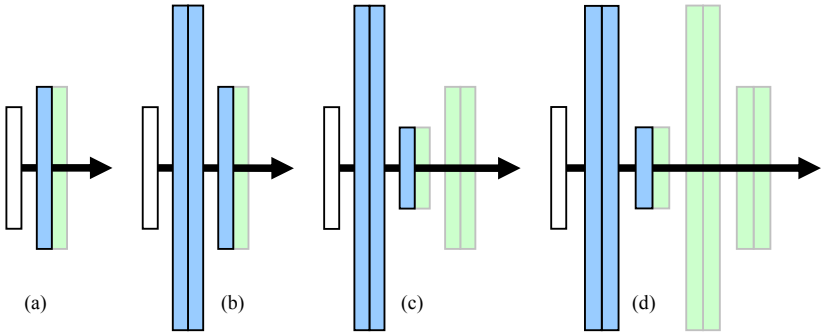

Fig. 2. Four MLP types *(a-d)* tested for data enhancement. Each active layer is shown as a (net-input function / non-linear activation function) sandwich. Only the dark sections of each MLP were used in data projection. The light parts were used only in training.

In ASR the MLP is trained to output a probability for each phoneme. In the model used here we select a random subset of the Timit speakers available for training (which we will refer to as the *speaker basis set*) and train the MLP to output a probability for each of these speakers. Although none of the MLPs *a-d* gave a high basis speaker classification score, the test results in Section 5 show that the speaker discriminative internal data representation which some of them learn can be very beneficial for GMM based speaker modelling

4 Test Procedure

Our baseline system is taken from the state of the art GMM based speaker identification system in [12], using the Timit speech database [5], GMMs with just 32 Gaussians, and 19 MFCC features.

4.1 Baseline Feature Processing

As in [12], all of the Timit signal data was first downsampled to 8 kHz, to simulate telephone line transmission (without down-sampling, GMMs already achieve above 99.7% correct speaker identification). No further low- or high-pass filters were applied. Also as in [12], MFCC features, obtained using HTK [14], were used, with 20ms windows and 10ms shift, a pre-emphasis factor of 0.97, a Hamming window and 20 Mel scaled feature bands. All 20 MFCC coefficients were used except c0. On this database neither silence removal, cepstral mean subtraction, nor time difference features increased performance, so these were not used.

4.2 Test Protocol

Timit does not have a standard division into training, development and test sets which is suitable for work with speaker recognition. For this we first divided the 630 speakers in Timit into disjoint training, development and test speaker sets of 300, 162 and 168 speakers respectively.

Data enhancement MLPs a-d (Figure 2) were trained using a speaker basis set of between 30 and 100 speakers selected at random from the training set. In each case the number of units in hidden layer 1, and also in hidden layer 3 in MLP *d*, was fixed at 500. The number of units in hidden layer 2 in MLPs *c* and *d* was fixed at 20 (a reasonable size for a feature vector for use with GMMs). Performance could have been improved by stopping MLP training when identification error on the development test set (using GMMs trained on data preprocessed by the MLP in its current state) stopped increasing. However, in the tests reported here, each MLP was simply trained for a fixed number (35) of batch iterations, after which mean squared error on the training basis stopped significantly decreasing.

Each MLP type was tested just once with each number of basis speakers. For the best performing MLP (MLP *d*), test-set tests were made with multiple different speaker basis subsets obtained by dividing the training data into as many equal parts as each speaker basis size would permit.

Timit data is divided into 3 sentence types, $SX_{1\text{-}5}$, $SI_{1\text{-}3}$ and $SA_{1\text{-}2}$. The text independent GMM for each speaker to be tested was trained on MLP projected sentences of type ($SX_{1\text{-}2}$, $SA_{1\text{-}2}$, $SI_{1\text{-}2}$) and tested on MLP projected sentences of type (SX_4, SX_5). Baseline GMMs were trained on MFCC features. The speaker identification procedure was as described in Section 2. Both training and testing used Torch [3].

5. Test Results

Test set speaker identification scores, for MLP type *a-d* against speaker basis size, are shown in Table 1 and Figure 3. The baseline test set identification error was 3.87%.

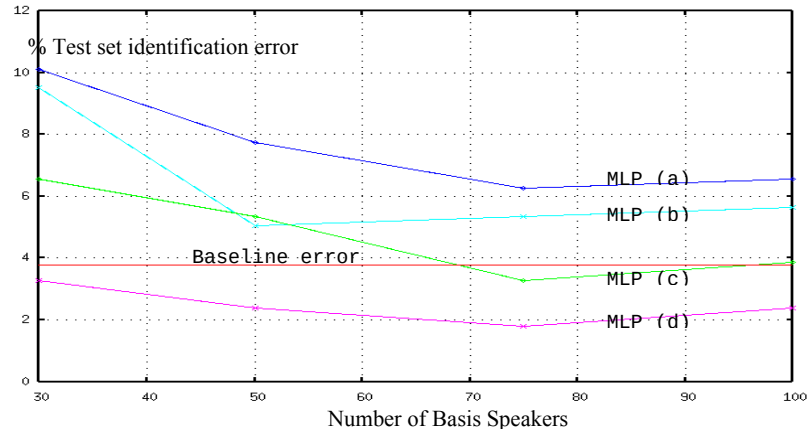

Fig. 3. Speaker identification error rate for the 168 speakers in the test set, for data enhancement using MLPs *a,b,c,d*, with varying numbers of basis speakers

Table 1. Test set speaker identification error for MLPs *a-d* in Fig.2 against speaker basis size

Speaker basis size	30	50	75	100	Best % rel. error reduction
MLP *a*	10.10	7.74	6.25	6.55	-61.5
MLP *b*	9.52	5.06	5.36	5.65	-30.7
MLP *c*	6.55	5.36	3.27	3.87	15.5
MLP *d*	3.27	2.38	1.79	2.38	53.8

The best scoring MLP (MLP *d*) was then tested many times, for each number of basis speakers, also on the test set (Table 2). While results for different repetitions for each speaker basis size varied considerably, in 28 out of 30 tests the speaker identification error was lower than the baseline error. The optimal size of the speaker basis set used for training was 100, giving a relative error reduction of up to 77.0 %.

Table 2. MLP *d* speaker identification test-set % error against speaker basis size. For each number of basis speakers, test-set tests were repeated, using disjoint speaker basis sets, as many times as were permitted by the number of available speakers (Baseline error 3.87%)

Repetition \ Basis size	30	50	60	75	100	150
1	3.57	2.68	2.68	2.37	1.49	2.08
2	2.98	2.68	1.79	2.08	0.89	1.79
3	3.87	2.08	2.68	3.57	1.49	
4	2.08	2.08	1.79	2.08		
5	3.27	1.79	1.79			
6	4.76	1.49				
7	2.68					
8	3.27					
9	1.49					
10	3.57					
Mean % error	3.15	2.13	2.15	2.53	1.29	1.93
Max % rel. err. reduction	61.5	61.5	53.7	46.3	77.0	53.7

6. Discussion

Results reported show up to 2.98% absolute (77.0% relative) performance improvement over the state of the art baseline on the Timit database. This was achieved with minimal fine-tuning and confirms our working hypothesis that the transformation learnt by the MLP to separate a random subset of speakers also substantially enhances separability between any speakers from the same population. An increase in identification accuracy has been found before with LDA when one output was trained for each speaker to be recognised [8]. By contrast, our MLP (a), which performs a linear separation equivalent to LDA [4], performs on average very badly. However, this could be because in our case none of the test speakers are used in training, so that the MLP is required to generalise to new speakers.

It appears that the ability of the features provided by the MLP to enhance speaker discrimination increases with the number of hidden layers. However, from the application viewpoint it would be advantageous to keep the MLP size and data transformation complexity to a minimum. It would be interesting to know whether the quality of data enhancement can be increased by dividing a given number of neurons into a greater number of layers, allowing for a more highly non-linear transformation.

Because of the large search space of possible MLP configurations, our search is still far from being optimised. Our decision to alternate large with small hidden layers

is based on the intuition that the benefits of non-linear vector space expansion and data compression should possibly be balanced. Our choice of MLP types *a-c* for testing was also guided by what has been used successfully before in ASR [6, 13], while MLP *d* was used in [9] for speaker recognition data enhancement, although the features it produced were of use only in combination with other features. In future we could try varying layer sizes, and also test the discriminatory power of features from every compressive hidden layer, not just the second. So far we have seen performance always increasing with the number of hidden layers used in MLP training (while always using just three layers for data enhancement). We have yet to find the point where this benefit stops increasing.

To reduce the amount of experimentation required the number of MLP batch training iterations was fixed at 35, although it is well known that MLPs tend to over fit to training data after the learning curve begins to flatten out. In future we should use cross validation testing to permit training to stop when MLP preprocessing maximises speaker identification performance on the development set.

Results are only reported here for multiple *random* selections of each given number of basis speakers. While the number of speakers selected was always large enough to guarantee a fairly representative sample from the full speaker population, the somewhat erratic variation in identification performance resulting from different random speaker bases of the same size suggests that it would be instructive to see whether more principled methods could be used for basis speaker set selection.

7. Conclusion

Test results reported show that the negative results which others have found when trying to use MLP based data enhancement directly for speaker identification [9] can be overcome. The number of target speakers which the MLP is trained to recognise must be small enough to avoid the classification problem becoming too difficult to train, but large enough to provide a feature basis sufficient to separate all speakers within a large population. The internal representation learnt by this MLP in separating the small set of basis speakers provides an enhanced feature vector which can improve GMM based speaker recognition performance. This form of data enhancement can be applied to speaker verification as well as to speaker identification. It can be used with growing speaker sets, of unlimited size, with no need for further training as new speakers are added.

Timit data is clean read speech. It needs to be checked whether these kind of benefits are also achievable on other more challenging databases, such as BANCA [2] and NIST [11]. It is expected that this will be the case because in ASR this method of data enhancement was particularly effective when both the training and test data exhibit a wide and varied range of noise conditions [13].

Aknowledgments

This work was supported by the EC SecurePhone project IST-2002-506883.

References

1. Bishop, C.M., Neural networks for pattern recognition, Oxford University Press, (1995).
2. Bengio, S., Bimbot, F., Mariethoz, j., Popovici, V., Poree, F., Bailly-Bailliere, E., Matas, G. & Ruiz, B., "Expermental protocol on the BANCA database", IDIAP-RR 02-05 (2002).
3. Collobert, R., Bengio, S. & Mariéthoz, J., "Torch: a modular machine learning software library", Technical Report IDIAP-RR 02-46 (2002).
4. Duda, O., Hart, P.E. & Stork, D.G., Pattern classification, Wiley (2001).
5. Fisher, W.M., Doddingtion, G.R. & Goudie-Marshall, K.M., "The DARPA speech recognition research database: Specifications and status", Proc. DARPA Workshop on Speech Recognition, pp. 93-99 (1986).
6. Fontaine, V., Ris, C. & Boite, J.-M., "Nonlinear Discriminant Analysis for improved speech recognition", Proc. Eurospeech'97, pp.2071-2074 (1997).
7. Genoud, D., Ellis, D. & Morgan, N., "Combined speech and speaker recognition with speaker-adapted connectionist models", Proc. ASRU (1999).
8. Jin, Q. & Waibel, A., "Application of LDA to speaker recognition", Proc. ICSLP'00 (2000).
9. Konig, Y., Heck, L., Weintraub, M. & Sonmez, K., "Nonlinear discriminant feature extraction for robust text-independent speaker recognition", Proc. RLA2C, ESCA workshop on Speaker Recognition and its Commercial and Forensic Applications, pp.72-75 (1998).
10. Reynolds, D.A., "Speaker identification and verification using Gaussian mixture speaker models", Speech Commun., 17 (1995), pp.91-108 (1995).
11. Reynolds, D.A., Doddington, D.R., Przybocki, M.A. & Martin, .F., "The NIST speaker recognition evaluation – overview, methodology, systems, results, perspective", Speech Communication, v.31, n.2-3, pp.225-254 (2000).
12. Reynolds, D.A., Zissman, M.A., Quatieri, T.F., O'Leary, G.C. & Carlson, B.A. "The effect of telephone transmission degradations on speaker recognition performance", Proc. ICASSP'95, pp.329-332 (1995).
13. Sharma, S., Ellis, D., Kajarekar, S., Jain, P. & Hermansky, H., "Feature extraction using non-linear transformation for robust speech recognition on the Aurora database", ICASSP 2000 (2000).
14. Young, S. et al. HTKbook (V3.2), Cambridge University Engineering Dept. (2002).