Using Neural Network with Speaker Applications

*Alaa noori mazher ** *Samira faris khlibs**

Date of acceptance 23/12 / 2009

Abstract:

In Automatic Speech Recognition (ASR) the non-linear data projection provided by a one hidden layer Multilayer Perceptron (MLP), trained to recognize phonemes, and has previous experiments to provide feature enhancement substantially increased ASR performance, especially in noise. Previous attempts to apply an analogous approach to speaker identification have not succeeded in improving performance, except by combining MLP processed features with other features. We present test results for the TIMIT database which show that the advantage of MLP preprocessing for open set speaker identification increases with the number of speakers used to train the MLP and that improved identification is obtained as this number increases beyond sixty. We also present a method for selecting the speakers used for MLP training which further improves identification performance.

Key words: Speaker recognition, data enhancement, MLP

Introduction:

It has previously been shown that the projection provided by the pre-squashed outputs from a one hidden layer MLP pre-trained to output a probability for each phoneme, can significantly increase Automatic Speech Recognition (ASR) performance. In attempting to apply the same technique to speaker (rather than speech) recognition, a number of questions arise. What target classes should the MLP is trained to recognize if want the features it generates to provide enhanced discrimination between speakers? If the MLP is trained to recognize some closed subset of speakers, would the mapping learnt also provide? Discriminative features for speakers not seen during training? The number of classes which an MLP can successfully learn to separate with a manageable amount of training data is quite limited. If speech data is available for a large number of speakers, which subset of these speakers would be most effective for MLP training [1].

Perceptron and MLP

A perception is a simple neuron model that has a set of inputs, a weight for each input and an (often nonlinear) activation function that the neuron performs to the weighted sum of inputs (plus possible bias) before sending the value to its output. The perception model is shown in Figure 1, where *y* is an input vector, *w* is a weight vector, *w1* is the bias and the activation function is a step function. See figure (1 a).

A multi-layer perceptron (MLP) consists of at least two layers of perceptions: it has an input layer, one or more hidden layers and output layer. The hidden layers act as a feature extractor and use a nonlinear function such as sigmoid or a radial-basis function to generate (often complex) functions of input. The outputs of all the neurons in the hidden layer serve as input to all of the neurons on the next layer. The output layer acts as a logical net that chooses an index to send to the output on the basis of inputs it receives

*Department Of Computer Science and Information System of the University Of Technology

from the hidden layer, so, that the classification error is minimized, see figure (1 b) .

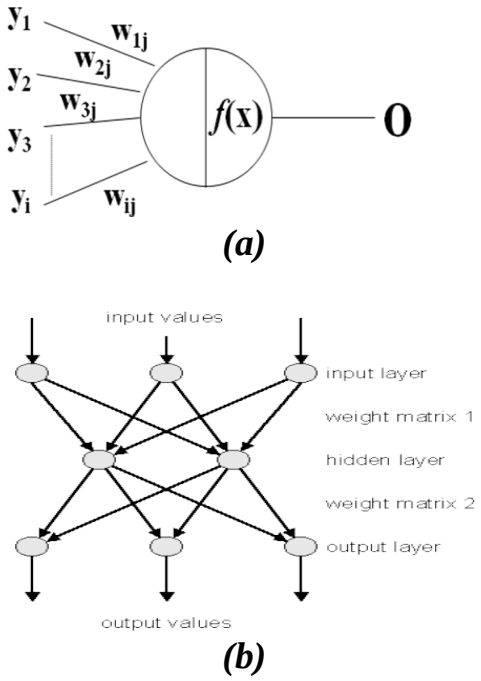

Figure 1: (a) perception, (b) MLP

Speaker Recognition Baseline:

The speaker recognition problem may be one of identification or verification. Given a certain amount of preprocessed speech data X, in the case of identification the problem is to identify the speaker from some given set of speakers, while with verification to decide whether or not the speaker is who they claim to be.[2,3] .

1. Speaker Identification:

A Gaussian Mixture Model (GMM), with some fixed number of Gaussians and diagonal covariance, is trained to model the speech frame Portable Data Frame (PDF) for each speaker. When training data is very limited it can be advantageous to train each GMM by Mean Average Precision (MAP) mean adaptation from a Universal Background Model (UBM). Speaker identification is then performed by selecting the speaker Sj with the largest posterior probability, P(Sj|X) (which corresponds to the largest data likelihood p(X| Sj) if all speaker priors P(Sj) are equal). For identification the ideal speech features must be independent of the true speaker identity, which is not known. Speaker independent feature enhancement is therefore well suited to speaker identification [2].

2. Speaker Verification:

For speaker verification, a GMM is trained for each speaker as with identification, but the claimant is accepted if the likelihood ratio of p(X|Sj)/p(X|U) exceeds some fixed threshold, where p(X|U) is the UBM which models the likelihood that X is from any speaker but Sj. For verification the problem is to distinguish a given speaker from all other speakers, so the optimal feature enhancement may be speaker dependent and therefore not so well suited to the approach used here [3].

The Proposed System:

This section for the speaker recognition and an enhansment by Neural network consist of:

1. Speaker Basis Selection:

A random selection of the speaker subset which the MLP is trained to separate (which we call the speaker basis) would be expected to represent the open set speaker population. However, classifier training can be more effective when training data is selected from class boundaries, while many errors in speaker identification are often traceable to a small number of problem speakers. We have tested several strategies for speaker basis selection based on a matrix of the distances between each speaker GMM PDF. We show here that this distance matrix can be estimated using only the speaker posterior probabilities $P_{ji} = P(S_j|X_i)$ for a set of development test data. P_{ji} are obtained by dividing the

development data log likelihood for each speaker by their sum over all speakers for one utterance. As a distance measure between speaker pdfs , we use the symmetric Kullback-Leibler distance KL(S_j, S_k) [1]. This cannot be evaluated in closed form when p(X|S_j) is modeled by a GMM. However, provided P(S_j)=P(S_k),

$$KL(S_j,S_k)=\int\left(p(X\backslash S_j)-p(X\backslash S_k)\right)log\frac{p(X\backslash S_j)}{p(X\backslash S_j)}dX \quad (1)$$

$$\propto\int p(X)\left(p(S_j\backslash X)-p(S_k\backslash X)\right)log\frac{p(S_j\backslash X)}{p(S_k\backslash X)}dX \quad (2)$$

$$=\int p(X)K(S_j,S_k.K)dX=E[K(S_j,S_k,X)] \quad (3)$$

$KL(S_j,S_K)$ Can therefore be estimated by averaging

$K(S_j,S_k,X)$ Over the development test data.

$$KL(S_{ji},p_{ki})\cong\sum_{X1\epsilon Devtestset}K(S_j,S_K,X_i) \quad (4)$$

$$=\sum_i(p_{ji}-p_{ki})(logp_{ji}-logp_{ki}) \quad (5)$$

The resulting speaker-distance matrix KLjk can then be used in various ways to select a subset of speakers for MLP training of the methods we have tested, that which has given the best results to choose speakers in order of decreasing average distance from every other speaker. We refer to this as the Maximum Average Distance (MaxAD) method for speaker basis selection.

2. MLP and GMM Training:

The TIMIT database is used for all the analyses. Since TIMIT is an excellent, phonetic-abundant database, hand-labelled in a precise manner with other speaker-related information such as speaker identity, gender and dialect region included, it is highly suitable for the present analysis, the TIMIT speech database was selected because, although it is only read speech, it is well suited for proof of concept tests and it is well known. As in [4], we first down-sampled TIMIT from 16 kHz to 8 kHz. At 16 kHz our baseline system (as in [4]) obtains 100% correct MLP 1 MLP 2 MLP 3 speaker identification (see figure 1b). However, it is of interest here to work with speech data which is close to telephone quality.

3. Baseline Feature Processing:

We used 20 ms frames and 20 Mel scaled filter bank log power features were extracted every 10 ms, using a Hamming window and a pre-emphasis factor of 0.97. A Discrete Cosine Transform (DCT) was then applied to obtain Mel-Frequency Cepstrum Coefficients (MFCC) features, from which the c0 energy coefficient was dropped. Neither silence removal, dynamic features nor cepstral mean subtraction were used, since none of these improved performance with TIMIT [5].

4. Train and Test Set Divisions:

The experiments we make are intended to test the use of MLP data enhancement for identification systems which are both speaker and text independent. The standard TIMIT division into training and test data is not suitable for this purpose so we defined our own gender and dialect region balanced division into speaker-disjoint training, development and evaluation sets, comprising 300, 168 and 162 speakers, respectively, which we denote SpkTr, SpkDv and SpkEv. Each of the 630 speakers in TIMIT has 10 utterances which are labeled as belonging to three sentence types: 6 types “X”, covering a wide range of acoustic contexts; 3 types “I”, being acoustically diverse, and 2 types “A” sentences which were the same for each speaker. We also divided these 11 sentences into disjoint training, development and evaluation sets: SenTr (SA1-2, SI1-2, and SX1-2); SenDv (SX3, SI3) and SenEv (SX4, SX5).

5. GMM Training:

All GMM and MLP training and testing were performed by the Torch machine learning Application Program Interface (API). GMMs used 32 Gaussians, a variance threshold factor of 0.01 and minimum Gaussian weight

of 0.05. TIMIT MAP adaptation did not help and was not used.

6. MLP Training:

Hidden layer 1 in MLP2 and also layer 3 in MLP3 used 100 sigmoid units. The bottle-neck hidden layer, the net input values from which comprise the MLP transformed features, always had the same number of units as the input layer (19). The output layer always had N log-softmax units, where N is the number of speakers which the MLP is being trained to recognize. MLP training was on-line, with an initial learning rate of 0.01 with a learning rate decay factor of 0.1. The data in each utterance was first normalized to have zero mean and unit variance. The training objective was maximum cross entropy. Initial MLP tests looked at making use of the learning curve for an MLP development set (SpkBs^SenDv) to decide when to stop iterative MLP training. However, for all MLPs tested the development set error continued to decrease even after several hundred training epochs, while the identification performance of GMMs trained on the resulting MLP preprocessed features always stopped increasing after about 30 training epochs. In all of the tests here, MLP training was stopped after 30 training epochs.

7. Feature Transformation (MLP and PCA):MFCC data is first normalized in the same way as in MLP training. For MLP 3 the 19 coefficient MFCC data is then passed through the net-input function and sigmoid functions in hidden layer 1 (100 units) and through the net input function to hidden layer 2 (19 units), this MLP data is then orthogonalised by Principal Components Analysis (PCA) projection (onto the unit eigenvectors of the covariance matrix for the MLPC features for the MLP training set SpkBs^SenTr).

8. Train and Test Set Procedures: For the purpose of speaker basis selection, a GMM is trained for each speaker Sj in SpkTr on MFCCs for Sj^SenTr (^ denotes set intersection). Each of these GMMs is then tested on MFCCs for SpkTr^SenDv. Making use of these test likelihoods, a speaker basis for MLP training, comprising a given number of speakers,N, is selected either at random or by MaxAD from SpkTr. Denote this SpkBs. The MLP is trained on MFCCs for SpkBs^SenTr. The trained MLP and PCA matrix is then used to transform the data to be used for GMM training and testing first from MFCCs to MLPCs and then (by PCA) to MLPAs. A GMM is then trained for each speaker in Sj in SpkEv on MLPA data for Sj^SenTr. Each GMM is then tested using MLPA data for every sentence in SpkEv^SenEv.

9. Identification Tests:

In order to confirm the MLP architecture previously proposed in [6, 7], tests were made with MLPs having 1, 2 and 3 hidden layers. In each case the number of speakers whose data was used for MLP training (the basis size) was varied from 2 to 256, using random selection (see Fig.2).

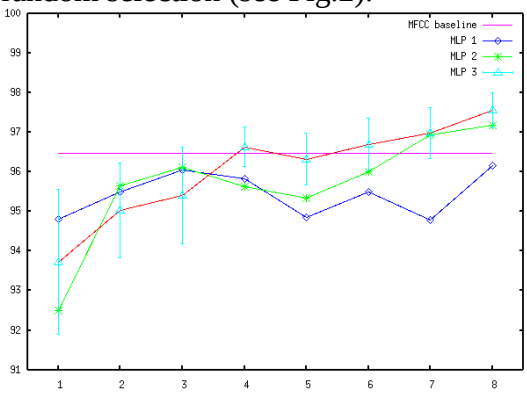

Figure (2) Identification performance for enhancement using MLPs 1, 2 or 3, against log2 speaker basis size, using random basis selection. Error bars shown only for MLP3.

Each test was repeated 10 times because of the randomness introduced either by the random basis selection ,

when MaxAD basis selection was used, by the random MLP weights initialization, or by the random GMM weights initialization for the GMMs used to set up the inter speaker distance matrix. The baseline GMM score was also subject to this random factor, so this test was also repeated 10 times (and had a % correct variance of 0.48). Fig.3 shows percent correct identification for enhancement using MLPs 1, 2 or 3, against log2 speaker basis size. Basis selection is random. Further tests were made to compare the performance of random and MaxAD speaker basis selection, again varying the basis size from 2 to 256 and repeating each test 10 times ,where X is data matrix, with X as rows and Y is target output matrix with 0/1 target vectors as rows ,(see Fig.3).

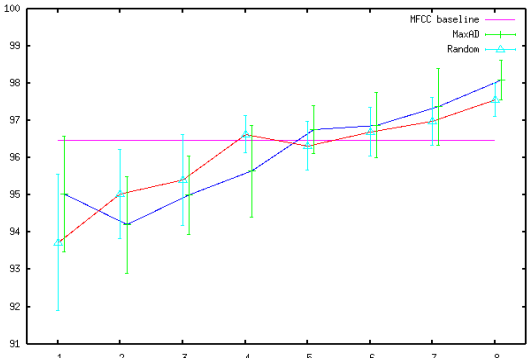

Figure (3) Identification performance for enhancement by MLP 3 using basis selected at random or by MaxAD.

Discussion:

Speaker identification using MLP as from all three MLPs improves with the number of speakers used in MLP training, though MLPs with more hidden layers improve more consistently. No significant improvement over the MFCC baseline occurs until the basis size is at least 26. It looks as if performance would continue to increase with the basis size going well beyond 210. MaxAD basis selection significantly outperforms random selection when the basis size is above 25. That MaxAD gives better results than random selection even when at 256 selected out of 300 speakers most of the speakers selected must be the same, suggests that it is good at avoiding problem speakers rather than selecting useful speakers. The test results with TIMIT show that MLP based feature enhancement can be used to advantage in speaker identification providing that the data used to train the MLP comes from a large enough number of speakers.

References:

1.Bishop, C.M. 1995. "Neural networks for pattern recognition", waltom, street Oxford University Press,2 nd ,pp 162.

2.Daugman, J. 2003. "Demodulation by complex-valued wavelets for stochastic pattern recognition", I J O W,p.p 338.

3.Dalei, Wu. 2006. "Discriminative Preprocessing of Speech: Towards Improving Biometric Authentication", Saarland University Press, pp 315.

4.Reynolds, D.A. 1995. 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, 17,(1):91-108 .

5.Reynolds, D.A. 2000. "Speaker identification and verification using adapted Gaussian mixture models", DSP,10(13), 225.

6.Konig, Y., Heck, L., Weintraub, M. & Sonmez, K. 1998. "Nonlinear discriminate feature extraction for robust text-independent speaker recognition", Proc. RLA2C, ESCA, spring link p.72-75.

7.Wu, D, Morris, A.C. & Koreman, J.2005, "MLP internal representation as discriminate features for improved speaker recognition", proc.NOLISP, 12:25-33.

استخـدام الشبكات العصبيــة مع تطبيقــات المتكـلم

علاء نوري مزهر *** ***سميرة فارس خليبص *

*قسم علوم الحاسبات ونظم المعلومات/ الجامعة التكنولوجية

الخلاصة :

في عمليات تمييز الكلام(ASR) ، توجه البيانات اللاخطية الناتجة من طبقة مخفية من طبقات (MLP) ، لتميز إحدى وحدات الكلام الصغرى (Phonemes) ، لتحسين الخصائص التي تزيد من أداء إل (ASR) خصوصا بوجود الضوضاء. وفي المحاولات السابقة التي طبقت لتمييز الكلام لم تنجح في تحسين الأداء باستثناء الدمج ما بين خصائص المعالجة بوساطة (MLP) مع خصائص أخرى. لقد عرضنا نتائج اختبارات لقاعدة بيانات (TIMIT) والتي بينت فوائد المعالجة الأولية ل (MLP) باتجاه تعريف مجموعة من المتكلمين وذلك بزيادة عدد المتكلمين المستخدمين لتدريب إل (MLP) عن إل (60)التي تمكنت من تحسين الأداء. كذلك بينا طريقة لاختيار المتكلمين المستخدمين لتدريب إل (MLP) والتي أعطت بعدا أكثر في أيجاد التماثل.