We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

4,800
Open access books available

122,000
International authors and editors

135M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS BOOK CITATION INDEX INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

3

Nonlinear Dimensionality Reduction Methods for Use with Automatic Speech Recognition

Stephen A. Zahorian and Hongbing Hu
Binghamton University, NY, USA

1. Introduction

For nearly a century, researchers have investigated and used mathematical techniques for reducing the dimensionality of vector valued data used to characterize categorical data with the goal of preserving "information" or discriminability of the different categories in the reduced dimensionality data. The most established techniques are Principal Components Analysis (PCA) and Linear Discriminant Analysis (LDA) (Jolliffe, 1986; Wang & Paliwal, 2003). Both PCA and LDA are based on linear, i.e. matrix multiplication, transformations. For the case of PCA, the transformation is based on minimizing mean square error between original data vectors and data vectors that can be estimated from the reduced dimensionality data vectors. For the case of LDA, the transformation is based on minimizing a ratio of "between class variance" to "within class variance" with the goal of reducing data variation in the same class and increasing the separation between classes. There are newer versions of these methods such as Heteroscedastic Discriminant Analysis (HDA) (Kumar & Andreou, 1998; Saon et al., 2000). However, in all cases certain assumptions are made about the statistical properties of the original data (such as multivariate Gaussian); even more fundamentally, the transformations are restricted to be linear.

In this chapter, a class of nonlinear transformations is presented both from a theoretical and experimental point of view. Theoretically, the nonlinear methods have the potential to be more "efficient" than linear methods, that is, give better representations with fewer dimensions. In addition, some examples are shown from experiments with Automatic Speech Recognition (ASR) where the nonlinear methods in fact perform better, resulting in higher ASR accuracy than obtained with either the original speech features, or linearly reduced feature sets.

Two nonlinear transformation methods, along with several variations, are presented. In one of these methods, referred to as nonlinear PCA (NLPCA), the goal of the nonlinear transformation is to minimize the mean square error between features estimated from reduced dimensionality features and original features. Thus this method is patterned after PCA. In the second method, referred to as nonlinear LDA (NLDA), the goal of the nonlinear transformation is to maximize discriminability of categories of data. Thus the method is patterned after LDA. In all cases, the dimensionality reduction is accomplished with a Neural Network (NN), which internally encodes data with a reduced number of dimensions. The differences in the methods depend on error criteria used to train the network, the architecture of the network, and the extent to which the reduced dimensions are "hidden" in the neural network.

The two basic methods and their variations are illustrated experimentally using phonetic classification experiments with the NTIMIT database and phonetic recognition experiments with the TIMIT database. The classification experiments are performed with either a neural network or Bayesian maximum likelihood Mahalanobis distance based Gaussian assumption classifier For the phonetic recognition experiments, the reduced dimensionality speech features are the inputs to a Hidden Markov Model (HMM) recognizer that is trained to create phone level models and then used to recognize phones in separate test data. Thus, in one sense, the recognizer is a hybrid neural network/Hidden Markov Model (NN/HMM) recognizer. However, the neural network step is used for the task of nonlinear dimensionality reduction and is independent of the HMM. It is shown that the NLDA approach performs better than the NLPCA approach in terms of recognition accuracy. It is also shown that speech recognition accuracy can be as high as or even higher using reduced dimensionality features versus original features, with "properly" trained systems.

2. Background

Modern automatic speech recognition systems often use a large number of spectral/ temporal "features" (i.e., 50 to 100 terms) computed with typical frame spaces on the order of 10 ms. Partially because of these high dimensionality feature spaces, large vocabulary continuous speech automatic speech recognition often have several million parameters that must be determined from training data (Zhao et al., 1999). In this scenario, the "curse of dimensionality" (Donoho, 2000) becomes a serious practical issue; for high recognition accuracy with test data, the feature dimensionality should be reduced, ideally preserving discriminability between phonetically different sounds, and/or large databases should be used for training. Both approaches are used. Despite growing computer size and computer storage densities, it should be noted that in principle the amount of data needed for adequate training grows exponentially with the number of features; for example, increasing dimensionality from 40 to 50, increases the need for more data by a factor proportional to $k^{(50-40)} = k^{10}$, where k is some number representing the average number of data samples distributed along each dimension, and almost certainly 2 or larger. Thus, for good training of model parameters, increasing dimensionality from 40 to 50, considered a modest increase, could easily increase the need for more data by a factor of 1000 or more. Therefore it seems unlikely that increased database size alone is a good approach to improved ASR accuracies by training with more and more features.

In this chapter, some techniques are presented for reducing feature dimensionality while preserving category (i.e., phonetic for the case of speech) discriminability. Since the techniques presented for reducing dimensionality are statistically based, these methods also are subject to "curse of dimensionality" issues. However, since this dimensionality reduction can be done at the very front end of a speech recognition system, with fewer model parameters tuned than in an overall recognition system, the "curse" can be less of a problem. We first review some traditional linear methods for dimensionality reduction before proceeding to the nonlinear transformation, the main subject of this chapter.

2.1 Principal Components Analysis (PCA)

Principal Components Analysis (PCA), also known as the Karhunen-Loeve Transform (KLT), has been known of and in use for nearly a century (Fodor, 2002; Duda et al., 2001), as a linear method for dimensionality reduction. Operationally, PCA can be described as follows:

Let $\mathbf{X} = [x_1, x_2, ..., x_n]^T$ be an n-dimensional (column) feature vector, and $\mathbf{Y} = [y_1, y_2, ..., y_m]^T$ be an m-dimensional (column) feature vector, obtained as the linear transform of **X**, using the n by m transformation matrix **A**, i.e. $\mathbf{Y} = \mathbf{A}^T\mathbf{X}$.

Let $\hat{\mathbf{X}} = \mathbf{BY}$ be an approximation to $\mathbf{X}$. Note that $\mathbf{X}, \mathbf{Y}$ and $\hat{\mathbf{X}}$ can all be viewed as (column) vector-valued random variables. The goal of PCA is to determine **A** and **B**, such that $E\left\{(\mathbf{X} - \hat{\mathbf{X}})^2\right\}$ is minimized. That is, $\hat{\mathbf{X}}$ should approximate **X** as well as possible, in a mean square error sense. As has been shown in several references (for example, Duda et al., 2001), this seemingly intractable problem has a very straightforward solution, provided **X** is zero mean and multivariate Gaussian. The rows of transformation $\mathbf{A}^T$ have been shown to be the eigenvectors of the covariance matrix of **X**, corresponding to the m largest eigenvalues of this matrix. The columns of **B** are also the same eigenvectors. Thus the "forward" and "reverse" transformations are transposes of each other. The components of **Y** are uncorrelated. Furthermore the expected value of this normalized mean square error between original and re-estimated **X** vectors can be shown to equal the ratio of the sum of "unused" eignevalues to the sum of all eigenvalues. The columns of **A** are called the principal components basis vectors and the components of **Y** are called the principal components.

If the underlying assumption of zero mean multivariate Gaussian random variables is satisfied, then this method of feature reduction generally performs very well. The principal components are also statistically independent for the Gaussian case. The principal components "account for" or explain the maximum amount of variance of the original data. Figure 1 shows an example of scatter plot of 2-D multivariate data and the resulting orientation of the first primary principal components basis vector. As expected this basis vector, represented by a straight line, is oriented along the axis with maximum data variation.

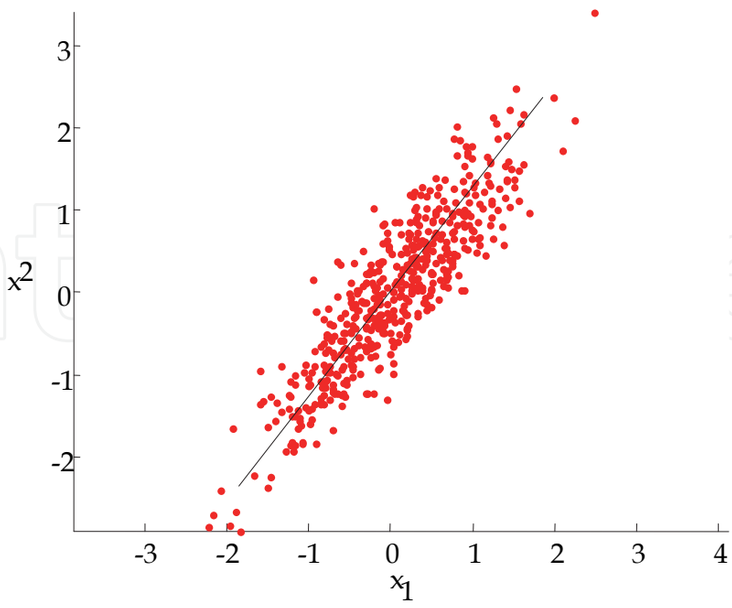

Fig. 1. Scatter plot of 2-D multivariate Gaussian data and first principal components basis vector. Line is a good fit to data.

Figure 2 depicts data which is primarily aligned with a U shaped curve in a 2-D space, and the resulting straight line (PCA) basis vector fit to this data. Since the original data is not multivariate Gaussian, the PCA basis vector is no longer a good way to approximate the data. In fact, since the data primarily follows a curved path in the 2-D space, no linear transform method, resulting in a straight line subspace, will be a good way to approximate the data with one dimension.

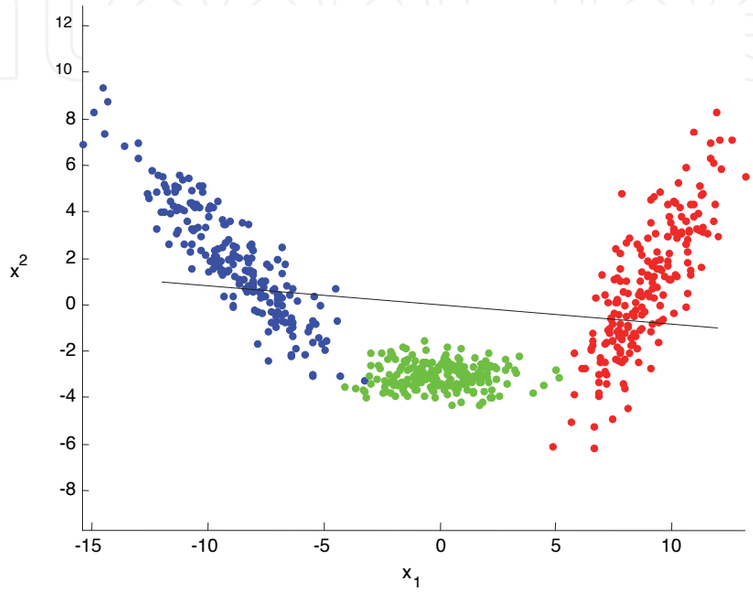

Fig. 2. Scatter plot of 2-D multivariate Gaussian data and first principal components basis vector. No straight line can be a good fit to this data.

2.2 Linear Discriminant Analysis (LDA)

Linear transforms for the purpose of reducing dimensionality while preserving discriminability between pre-defined categories have also long been known about and used (Wang & Paliwal, 2003), and are usually referred to as Linear Discriminant Analysis (LDA). The mathematical usage of this is identical to that for PCA. That is $\mathbf{Y} = \mathbf{A}^T\mathbf{X}$, where $\mathbf{X}$, $\mathbf{Y}$ are again column vectors as for PCA.

The big difference is in how $\mathbf{A}$ is computed. For LDA, it has been shown that the columns of $\mathbf{A}$ correspond to the m largest eigenvalues of $\mathbf{S}_W^{-1}\mathbf{S}_B$, where $\mathbf{S}_W$ is the within class covariance matrix and $\mathbf{S}_B$ is the between class covariance matrix.

Often $\mathbf{S}_B$ is computed as the covariance of the category means; alternatively, it is sometimes computed as the "grand" covariance matrix over all data, ignoring category labels, identical to the covariance matrix used to compute PCA basis vectors. $\mathbf{S}_W$, the within class covariance matrix, is generally computed by first determining the covariance matrix for each category of data, and then averaging over all categories. The explicit assumption for LDA is that the within class covariance of each category is the same, which is rarely true in practice. Nevertheless, for many practical classification problems, features reduced by LDA often are

as effective or even advantageous to original higher dimensional features. Figure 3 depicts 2-D 2-class data, and shows the first PCA basis vector as well as the first LDA basis vector. Clearly, for this example, the two basis vectors are quite different, and clearly the projection of data onto the first LDA basis vector would be more effective for separating the two categories than data projected onto the first PCA basis vector.

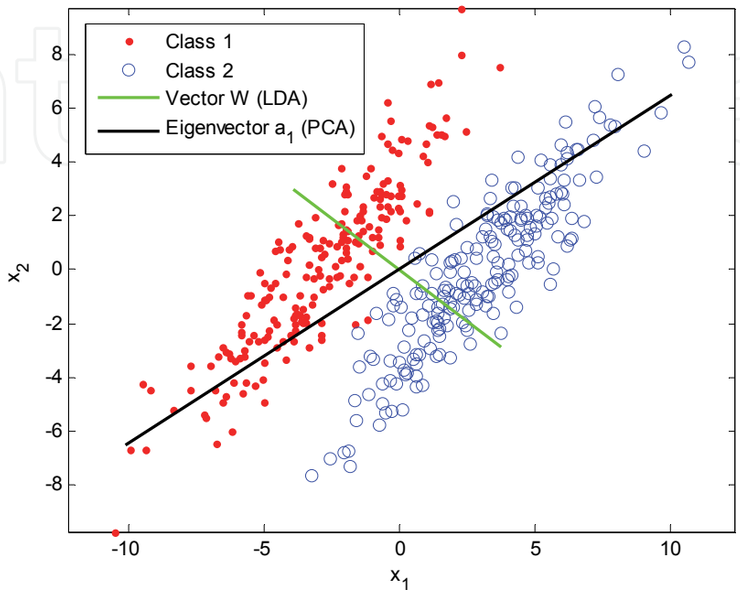

Fig. 3. 2-D 2-class data, along with first LDA basis vector and first PCA basis vector. The classes would be well separated by a projection onto the first LDA basis vector, but poorly separated by a projection onto the first PCA basis vector.

2.3 Heteroscedastic Discriminant Analysis (HDA)

Another linear transformation technique, related to linear discriminant analysis, but which accounts for the (very common) case where within class covariance matrices are not the same for all classes, is called Heterocscedastic Discriminant Analysis (HDA) (Saon et al., 2000). The process for HDA is described in some detail by Saon et al., and illustrated in terms of its ability to better separate (as compared to LDA) data in reduced dimensionality subspaces, when the covariance properties of the individual classes are different.

HDA suffers from two drawbacks. There is no known closed form solution for minimizing the objective function required to solve for the transformation—rather a complex numerically based gradient search is required. More fundamentally, with actual speech data, HDA alone was found to perform far worse than LDA. Nevertheless, if an additional transform, called the Maximum Likelihood Linear Transform (MLLT) (Gopinath, 1998) was used after HDA, then overall performance was found to be the best among the methods tested by Saon et al. However, ASR accuracies obtained with a combination of LDA and MLLT were nearly as good as those obtained with HDA and MLLT. A detailed summary of HDA and MLLT are beyond the scope of this chapter; however, these methods, either by themselves, or in conjunction with the nonlinear methods described in this chapter warrant further investigation.

3. Nonlinear dimensionality reduction

If the data are primarily clustered on curved subspaces embedded in high dimensionality feature spaces, linear transformations for feature dimensionality reduction are not well suited. For example, the data depicted in Figure 2 would be better approximated by its position with respect to a curved U-shape line rather the straight line obtained with linear PCA. (Bishop et al. 1998) discusses several theoretical methods for determining these curved subspaces (manifolds) within higher dimensionality spaces. Another general method, and the one illustrated and explored in more detail in this chapter, is based on a "bottleneck" neural network (Kramer, 1991). This method relies on the general ability of a neural network with nonlinear activation functions at each node, with enough nodes and at least one hidden layer, to be able to determine an arbitrary nonlinear mapping. The general network configuration is shown in Figure 4.

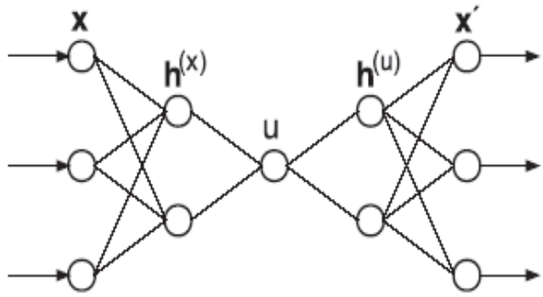

Fig. 4. Architecture of Bottleneck Neural Network.

Presumably, although not necessarily, with enough nodes in each of the hidden layers, and with "proper" training, the network will represent data at the outputs of the bottleneck layer as well as possible in a subspace with a number of dimensions equal to the number of nodes in the bottleneck layer. If data lies along a single curved line in a higher dimensionality space, 1 node in the bottleneck layer should be sufficient. If data lies on a curved surface embedded in a higher dimensionality space, 2 nodes in the bottleneck layer should be sufficient.

3.1 Nonlinear Principal Components Analysis (NLPCA)

If the bottleneck neural network is trained as an identity map, that is with outputs equal to inputs, and using a mean square error objective function, then the neural network can be viewed as performing nonlinear principal components analysis (NLPCA) (Kramer, 1991). Since the final NN outputs are created from the internal NN representations at the bottleneck layer, the bottleneck outputs can be viewed as the reduced dimensionality version of the data. This idea was tested using pseudo-random data generated so as to cluster on curved subspaces.

NLPCA is first illustrated by an example depicted in Figure 5. For this case, 2-D pseudo random data was created to lie along a U shaped curve, similar to the data depicted in Figure 2. A neural network (2-5-1-5-2) was then trained as an identify map. The numbers in parentheses refer to the number of nodes at each layer, proceeding from input to output. All hidden nodes and output nodes had a bipolar sigmoidal activation function. After training with backpropagation, all data were transformed by the neural network. In Figure 5, the original data is shown as blue symbols, and the transformed data is shown by red. Clearly, the data have been projected to a curved U shaped line, as would be expected for the best line fit to the original data.

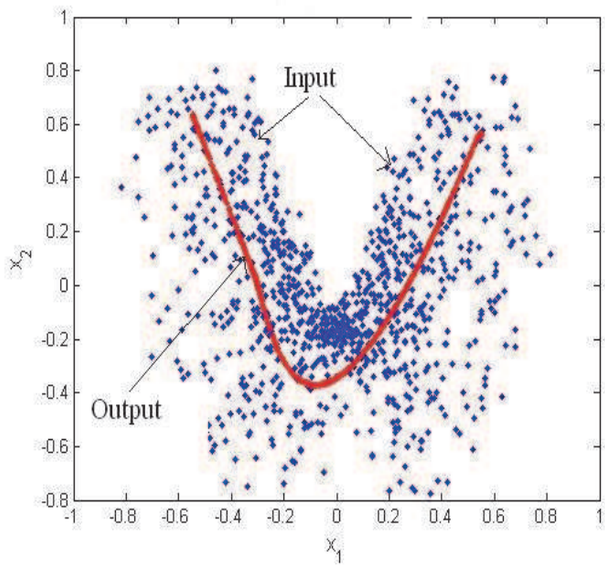

Fig. 5. Plot of input and output data for pseudo-random 2-D data. The output data (red line) is reconstructed data obtained after passing the input data through the trained neural network.

In Figure 6, NLPCA is illustrated by data which falls on a 2-D surface embedded in a 3-D space. For this case, 2-D data pseudo random data are created, but confined to lie on the surface of a 2-D Gaussian shaped surface, as depicted in the left panel of Figure 6. Then a neural network (3-10-2-10-3) was trained as an identity map. After training, the outputs of the neural network are plotted in the right panel of Figure 6. Clearly the neural network "learned" a 2-D internal representation, at the bottleneck layer, from which it could reconstruct the original data.

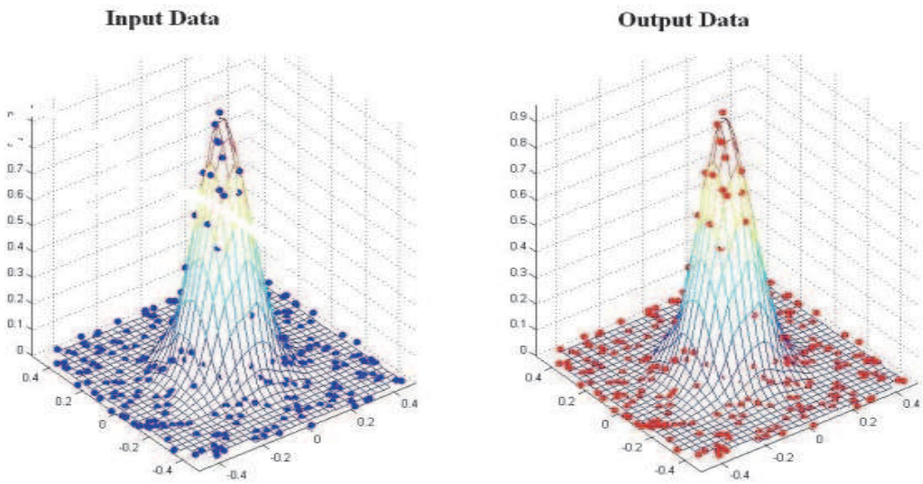

Fig. 6. Input and output plot of the 3-D Gaussian before (left) and after (right) using neural network for NLPCA.

4. Nonlinear Discriminant Analysis (NLDA)

Despite the apparent ability of a neural network to well represent data from reduced dimensions, as illustrated by the examples depicted in Figure 5 and Figure 6, for applications to machine pattern recognition, including automatic speech recognition, a nonlinear feature reduction analogous to linear discriminant analysis might be advantageous to NLPCA. Fortunately, only a minor modification to NLPCA is needed to form NLDA. The same bottleneck network architecture is used, but trained to recognize categories rather than as an identity map. In the remaining part of this chapter, two versions of NLDA based on this strategy are described, followed by a series of experimental evaluations for the phonetic classification and recognition tasks.

4.1 Nonlinear dimensionality reduction architecture

In a previous work (Zahorian et al., 2007), NLPCA was applied to an isolated vowel classification task, and the nonlinear method based on neural networks was experimentally compared with linear methods for reducing the dimensionality of speech features. It was demonstrated that NLPCA which minimizes mean square reconstruction error from a reduced dimensionality space can be very effective for representing data which lies in curved subspaces, but did not appear to offer any advantages over linear dimensionality reduction methods such as PCA and LDA, for a speech classification task. A summary of this work is presented in Section 4.5. In contrast, the nonlinear technique NLDA based on minimizing classification error was quite effective for improving accuracy.

The general form of the NLDA transformer and its relationship to the HMM recognizer are depicted in Figure 7. NLDA is based on a multilayer bottleneck neural network and performs a nonlinear feature transformation of the input data. The outputs of the network are further (optionally) processed by PCA to create transformed features to be the inputs of an HMM recognizer. Note that in this usage, "outputs" may be from the final outputs or from one of the internal hidden layers.

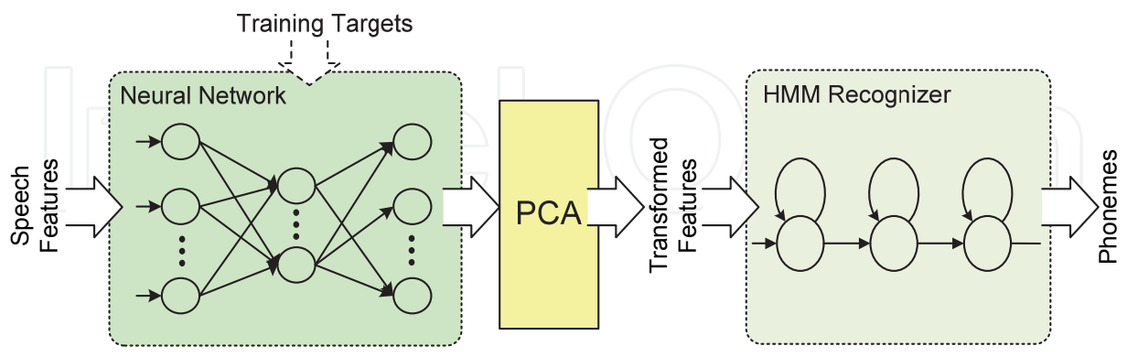

Fig. 7. Overview of the NLDA transformation for speech recognition.

The multilayer bottleneck neural network employed in NLDA contains an input layer, hidden layers including the bottleneck layer, and an output layer. The numbers of nodes in the input and output layers respectively correspond to the dimensions of the input features and the number of categories in the training target data. The targets were chosen as the 48 (collapsed) phones in the training data. The number of hidden layers was

experimentally determined as well as the number of nodes included in those layers. However, most typically three hidden layers were used. Two NLDA approaches were investigated as different layers of networks are used to obtain dimensionality reduced data.

4.2 NLDA1

In the first approach, which is referred to as NLDA1, the transformed features are produced from the final output layer of the network. This approach is similar to the use of tandem neural networks used in some automatic speech recognition studies (Hermansky & Sharma, 2000; Ellis et al., 2001). Figure 8 illustrates the use of network outputs in NLDA1.

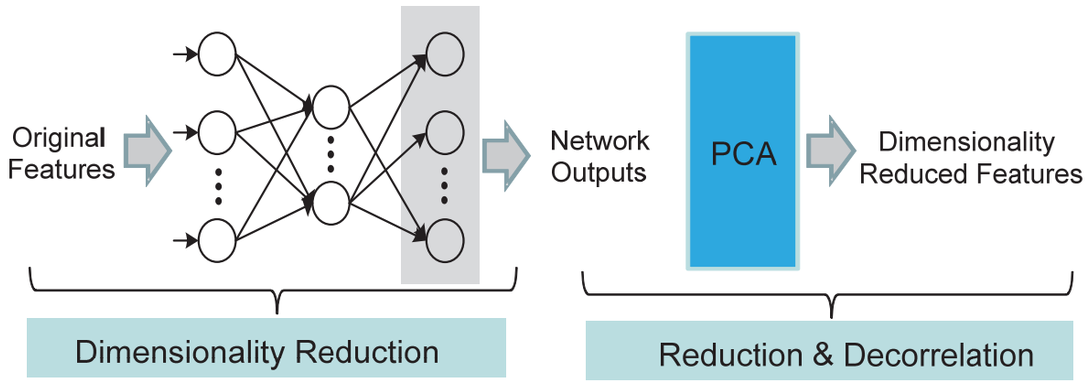

Fig. 8. Use of network outputs in NLDA1.

4.3 NLDA2

Since the activations of the middle layer represent the internal structure of the input features, in the second approach named NLDA2, the outputs of the middle hidden layer, with fewer nodes than the input layer, are used as transformed features to form reduced but more discriminative dimensions. Figure 9 illustrates the use of network outputs in NLDA2. These two versions of NLDA were experimentally tested, with and without PCA following the neural network transformer, with some variations of the nonlinearities in the networks.

The dimensionality of the reduced feature space is determined only by the number of nodes in the middle layer. Therefore, an arbitrary number of reduced dimensions can be obtained, independent of the input feature dimensions and the nature of the training targets. A lower dimensional representation of the input features is easily obtained by simply deploying fewer nodes in the middle layer than the input layer. This flexibility allows dimensionality to be adjusted so as to optimize overall system performance (Hu & Zahorian, 2008; Hu & Zahorian, 2009; Hu & Zahorian, 2010).

In contrast with NLDA1 where dimensionality reduction is assigned to PCA, for NLDA2, since the dimensionality reduction can be accomplished with the neural network only, the linear PCA is used specifically for reducing the feature correlation.

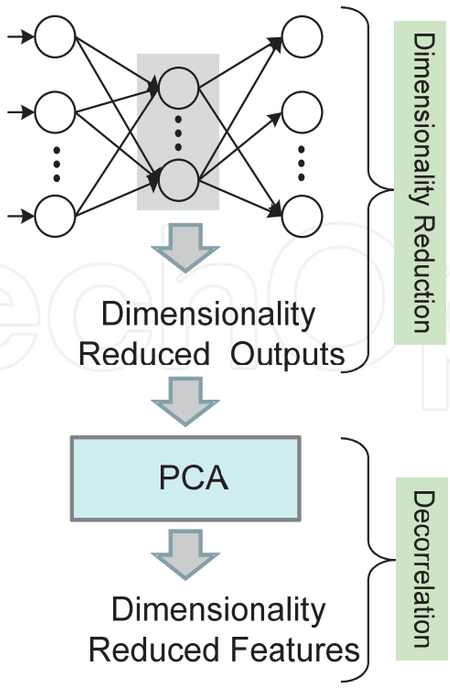

Fig. 9. Middle layer outputs used as dimensionality reduced features in NLDA2.

4.4 Neural networks

In optimizing the design of a neural network, an important consideration is the number of hidden layers and an appropriate number of hidden nodes in each layer. A neural network with no hidden layers can form only simple decision regions, which is not suitable for highly nonlinear and complex speech features. Although it has been shown that a neural network with a single hidden layer is able to represent any function with a sufficient number of hidden nodes (Duda et al., 2001), the use of multiple hidden layers generally provides a flexible configuration such as distributed deployment of hidden nodes, and diverse nonlinear functions for different layers. On the number of hidden nodes, a small number reduces the network's computational complexity. However, the recognition accuracy is often degraded. The more hidden nodes a network has, the more complex a decision surface can be formed, and thus better classification accuracy can be expected (Meng, 2006). Generally, the number of hidden nodes is empirically determined by a combination of accuracy and computational considerations, as applied to a particular application.

Another important consideration is selecting an activation function, or the nonlinearity of a node. Typical choices include a linear activation function, a unipolar sigmoid function and a bipolar sigmoid function as illustrated in Figure 10.

The activation function should match the characteristic of the input or output data. For example, with training targets assigned the values of "0" and "1", a sigmoid function with the outputs in the range of [0, 1] is a good candidate for the output layer. Most typically a mean square error, between the desired output and actual output of the NN, is the objective function that is minimized in NN training. As another powerful approach, the softmax function takes all the nodes in a layer into account and calculates the output of a node as a posterior probability. When the outputs of the network are to be used as

transformed features for the HMM recognition, a linear function or a softmax function is appropriate to generate the data with a more diverse distribution, such as one that would be well-modeled with a GMM. Moreover, equipped with various nonlinearities, the neural network is expected to have a stronger discriminative capability and thus it is enabled to cope with more complex data.

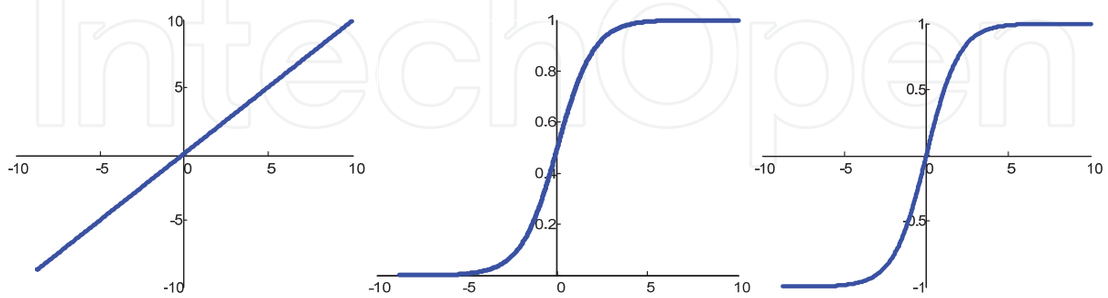

Fig. 10. Illustrations of a linear activation function (left), a unipolar sigmoid function (middle) and a bipolar sigmoid function (right).

The weights of the neural network are estimated using the backpropagation algorithm to minimize the distance between the scaled input features and target data. The update of the weights in each layer depends on the activation function of that layer, thus the network learning can be designed to perform different updates when dissimilar activation functions are used. In addition, a difficulty in neural network training is that the input data has a wide range of means and variances for each feature component. In order to avoid this, the input data of neural networks is often scaled so that all feature components have the same mean (zero) and variance (so that range of values is approximately ± 1).

4.5 Investigation of basic issues

Before performing a series of time consuming phonetic recognition experiments involving the entire TIMIT database, extensive neural network training, and HMM training and evaluation, a more limited set of phonetic classification experiments was conducted using only the vowel sounds extracted from NTIMIT, the telephone version of TIMIT. Note that unlike phonetic recognition experiments, for the case of classification, the timing labels in the database are explicitly used for both training and testing. Thus classification is "easier" than recognition, and accuracies typically higher, since phone boundaries are known in advance and used. In this first series of experiments, classification experiments were conducted using PCA, LDA, NLPCA, and NLDA2 transformations, as well as the original features.

The 10 steady-state vowels /ah/, /ee/, /ue/, /ae/, /ur/, /ih/, /eh/, /aw/, /uh/, and /oo/ were extracted from the NTIMIT database (see section 5.1) and used.

All the training sentences (4620 sentences) were used to extract a total of 31,300 vowel tokens for training. All the test sentences (1680 sentences) were used to extract a total of 11,625 vowel tokens for testing. For each vowel token, 39 DCTC-DCS features were computed using 13 DCTC terms and 3 DCS terms.

For all cases, including original features, and all versions of the transformed features, a neural network classifier with 100 hidden nodes and 10 output nodes, trained with backpropagation, was used as the classifier. In addition, a Bayesian maximum likelihood Mahalanobis distance based Gaussian assumption classifier (MXL) was used for evaluation.

For the neural network transformation cases, the first and third hidden layers had 100 nodes (empirically determined). The number of hidden nodes in the second hidden layer was varied from 1 to 39, according to the dimensionality being evaluated. For the case of NLDA2, the network used for dimensionality reduction was also a classifier. For the sake of consistency, the outputs of the hidden nodes from the bottleneck neural network were used as features for a classifier, using either another neural network or the MXL classifier. [1] In these initial experiments, the bottleneck neural network outputs were not additionally transformed with PCA.

4.5.1 Experiment 1

In the first experiment, all training data were used to train the transformations including LDA, PCA, NLPCA, and NLDA2, and the classifiers. Figure 11 shows the results based on the neural network and MXL classifiers for each transformation method in terms of classification accuracy, as the number of features varies from 1 to 39.

For both the neural network and MXL classifiers, highest accuracy was obtained with NLDA2, especially with a small numbers of features. For the MXL classifier, NLDA2 features result in approximately 10% higher classification accuracies as compared to all other features. For both the neural network and MXL classifiers, accuracy with NLPCA features was very similar to that obtained with linear PCA. For reduced dimensionality features and/or a MXL classifier, the NLDA2 transformation was clearly superior to original features or any of the other feature reduction methods. However, with a neural network classifier and much higher dimensionality features, all feature sets perform similarly in terms of classification accuracy.

As just illustrated, dimensionality reduction is not necessarily advantageous in terms of accuracy for classifiers trained with enough data and the "right" classifier. However, for the case of complex automatic speech recognition systems, there is generally not enough training data.

4.5.2 Experiment 2

To simulate lack of training data, another experiment was conducted. In this experiment, the training data was separated into two groups, with about 50% in each group. One group of data (group 1) was used for "training" transformations while the other data (group 2) was used for training classifiers. In contrast to experiment 1 for which all the training data was used for both the training of transformations and classifiers, for experiment 2, a fixed 50% of the training data was used for "training" transformations and a variable percentage, ranging from 1% to 100% of the other half of the training data, was used for training classifiers.

[1] It was, however, experimentally verified that classification results obtained directly from the bottleneck neural network were nearly identical to those obtained with this other network.

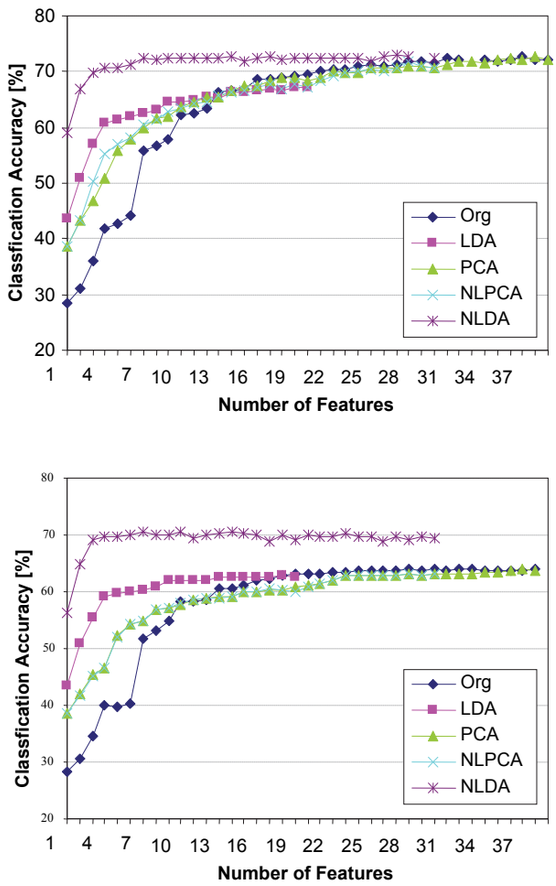

Fig. 11. Classification accuracies of neural network (top panel) and MXL (bottom panel) classifiers with various types of features.

The results obtained with the neural network and MXL classifiers using 10% of the group 2 training data (that is, 5% of the overall training data) are shown in Figure 12. The numbers of features evaluated are 1, 2 4, 8, 16 and 32. For both the neural network and MXL classifiers, NLDA2 clearly performs much better than the other transformations or the original features. However, the advantage of NLDA2 decreases with an increasing number of features, and as the percentage of group 2 data increases (not shown in figure).

4.6 Category labels for discriminatively based transformations

For all discriminatively based transformations, either linear or nonlinear, an implicit assumption is that training data exists which has been labeled according to category. For the case of classification, such as the experiments just described, this labeled data is needed anyway, for both training and test data, to conduct classification experiments; thus the need for category labeled data is not any extra burden. However, for other cases, such as the phonetic recognition experiments described in the remainder of this chapter, there may or may not be easily available and suitable labeled training data. This issue of category labels is described in more detail in the following two subsections.

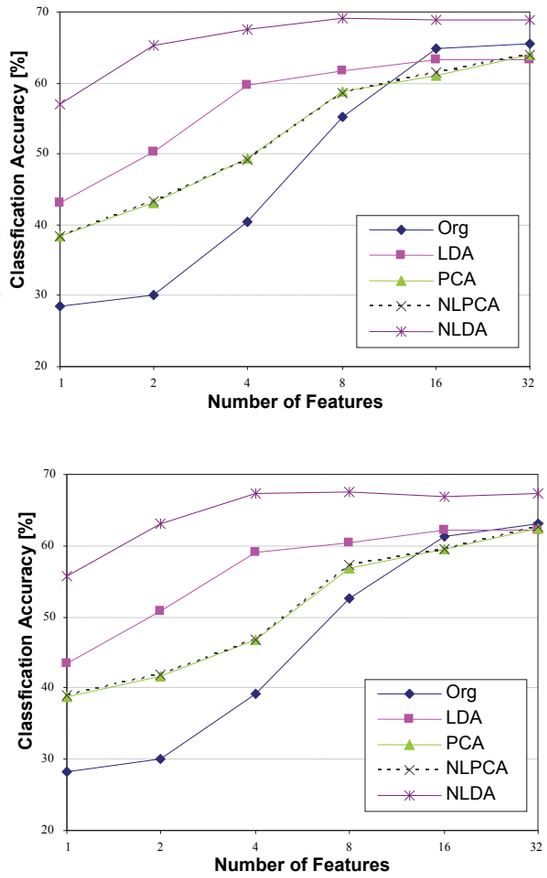

Fig. 12. Classification accuracies of neural network (top panel) and MXL (bottom panel) classifiers using 10% of group 2 training data for training classifier.

4.6.1 Phonetic-level targets

The training of the neural network (NLDA1 and NLDA2) requires category information for creating training targets. For the case of databases such as TIMIT, the data is labeled using 61 phone categories, and the starting point for training discriminative transformations would seem to be these phonetic labels. In the neural network training, these are referred to as targets. Ideally, the targets are uncorrelated, which enables quicker convergence of weight updates. The targets can also be viewed as multidimensional vectors, with a value of "1" for the target category and "0s" for the non-target categories.

Figure 13 illustrates a sequence of phoneme training targets for the TIMIT database using 48 phoneme categories. These vectors have 48 dimensions and each vector consists of only one peak value to indicate the category. Note that, in the TIMIT case, other reasonable choices for targets would be 61 (the number of phone label categories), or 39 (the number of collapses phone categories). However, empirically, the choice of 48 categories, with only some phones combined, seemed to be the best choice for both neural network training targets and for the creation of HMM phone models.

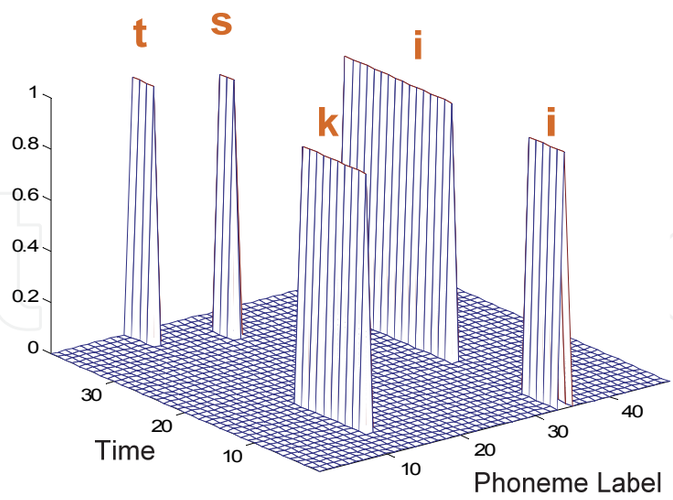

Fig. 13. Training target vectors of the neural network.

4.6.2 State-level targets

Due to the nonstationarity of speech signals, a speech signal varies even in a very short time interval (e.g. a phoneme). For speech recognition tasks, instead of phone level training targets, state (as in hidden states of an HMM) dependent targets could be advantageous in training a versatile network for more highly discriminative speech features. However, the boundaries between states in a phoneme are likely to be indistinct; even more importantly, from a practical perspective is that, unlike phonemes, the (HMM) state boundaries are unknown in advance of training. Thus the estimation of state boundary information is required. This boundary information may be in error due to the nature of unclear state boundaries and the lack of a reliable estimation approach. Therefore, in the discriminative training process, "don't cares" were used to account for this lack of precision in determining state boundaries. In the neural network training process, the errors of output nodes corresponding to "don't cares" are not computed and thus these "don't cares" have no effect on weight updates.

The state training targets with "don't cares" uses "don't care" states for each phoneme model, so that one neural network trained with the targets can generate state dependent outputs. As illustrated in Figure 15, the phone-specific training targets in Figure 14 are expanded to 144 dimensions by duplicating the phoneme specific target by the required number of the states. In the training process, for each point in time, one state target is considered as a "1," and the other two state targets for that point in time are considered as a "don't care," and the state targets for all other categories are considered as "0" value targets. As time progresses during a phone, the "1" moves from state 1 to state 2, to state 3.

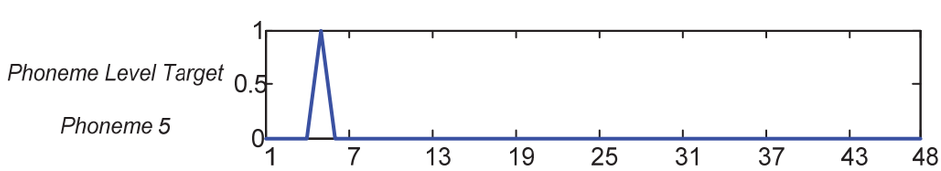

Fig. 14. Illustration of a phoneme level target.

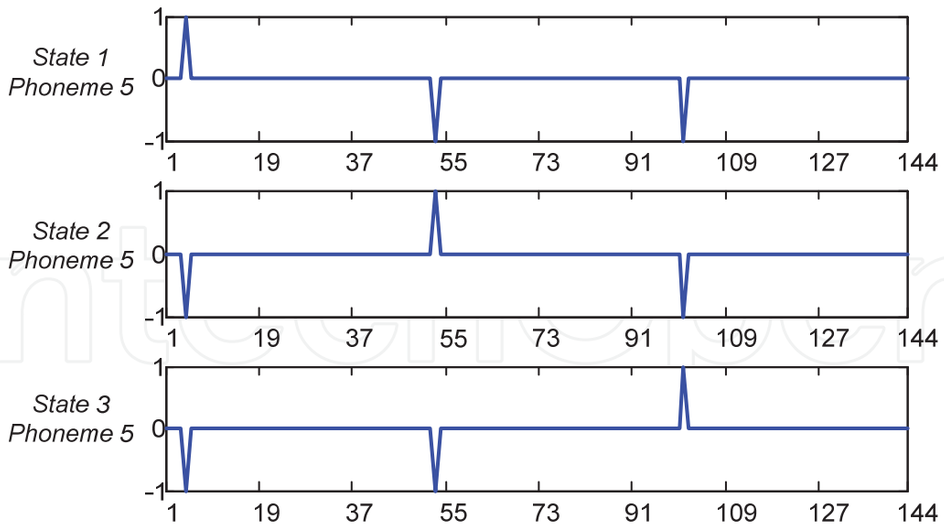

Fig. 15. Illustration of the state level training targets with "don't cares." The -1 values are used to denote "don't cares."

Two approaches are used to determine state boundaries. The first approach uses a fixed state length ratio for all phonemes, with typically about the first 1/6 of each phoneme considered as state 1, the central 2/3 as state 2, and the final 1/6 as state 3 (assuming a 3-state model for each phoneme). The second approach determines state boundaries using the HMM-based Viterbi alignment based on already trained HMMs. As illustrated in one of the experiments presented later, the state dependent targets were shown to perform better than phone level targets. The simpler approach of using fixed ratios for state boundaries was as good as using the Viterbi alignment approach.

5. Evaluation of feature reduction methods with phonetic recognition experiments

Given the high dimensionality of speech feature spaces used for automatic speech recognition, typically 39 or more, it is not feasible to visualize the distribution of data in feature space. It is possible that a reduced dimensionality subspace obtained by linear methods, such as PCA or LDA, forms an effective, or at least adequate subspace for implementing automatic speech recognition systems with a reduced dimensionality feature space. Note that if PCA or LDA do perform well, these methods would be preferred to the nonlinear methods, due to the much simpler implementation methods and the corresponding need for less data. However, it is also possible that one of the nonlinear methods for feature reduction is more effective, that is enable higher ASR accuracy, than any of the linear methods. The comparisons of these various methods can only be done experimentally.

5.1 TIMIT database

The database used for all experiments reported in the remainder of this chapter is TIMIT. The TIMIT database was developed in the early 1980's for expediting acoustic-phonetic ASR research (Garofolo et al., 1993; Zue et al, 1990). It consists of recordings of 10 sentences from each of 630 speakers, or 6300 sentences total. Of the text material in the database, two

dialect sentences (SA sentences) were designed to expose the specific variants of the speakers and were read by all 630 speakers. There are 450 phonetically-compact sentences (SX sentences) which provide a good coverage of pairs of phones. Each speaker read 5 of these sentences and each text was spoken by 7 different speakers. A total of 1890 phonetically-diverse sentences (SI sentences) were selected from existing text sources to add diversity in sentence types and phonetic contexts. Each speaker read 3 of these sentences, with each text being read only by a single speaker. All sentences are phonetically labeled with start and stop times for each phoneme.

The database is further divided into a suggested training set (4620 sentences, 462 speakers) and suggested test set (1680 sentences, 168 speakers). The training and test sets are balanced in terms of representing dialect regions and male/female speakers. Although a total of 61 phonetic labels were used in creating TIMIT, due to the great similarity of many of these phonemes (both from a perception point of view and acoustically), most ASR researchers, ever since the work reported by Lee and Hon (Lee & Hon, 1989) have combined these very similar sounding phones, and collapsed the phone set to 39 total. Similarly, most researchers have not used the SA sentences for ASR experiments, since the identical phonetic contexts (every speaker read the same sentences for the SA sentences), were thought be non representative of everyday speech.[2] Most, but not all researchers, have used the recommended training and test sets. For all ASR experiments reported in this chapter, the SA sentences were removed, the recommended training and test sets were used, and the phone set was collapsed to the same 39 phones used in most ASR experiments with TIMIT. The NTIMIT database, used for the classification experiments described in section 4.5, is the same one as TIMIT, except the data was transmitted over phone lines and re-recorded. Thus NTIMIT is more bandlimited (approximately 300Hz to 3400 Hz), more noisy, but has the identical "raw" speech.

5.2 DCTC/DCSC speech features

For both training and testing data, the modified Discrete Cosine Transformation Coefficients (DCTC) and Discrete Cosine Series Coefficients (DCSC) (Zahorian et al. 1991; Zahorian et al., 1997; Zahorian et al., 2002; Karnjanadecha & Zahorian, 1999) were extracted as original features. The modified DCTC is used for representing speech spectra, and the modified DCSC is used to represent spectral trajectories. Each DCTC is represented by a DCSC expansion over time; thus the total number of features equals the number of DCTC terms times the number of DCSC terms. The number of DCTCs used was 13, and number of DCS terms was varied from 4 to 7, for a total number of features ranging from 52 to 91. These numbers are given for each experiment. Additionally, as a control, one experiment was conducted with Mel-frequency Cepstral Coefficients (MFCCs) (Davis & Mermelstein, 1980), since these MFCC features are most typically used in ASR experiments. A total of 39 features including 13 MFCC features, delta terms, and delta-delta terms were extracted from both the training and test data.

5.3 Hidden Markov Models (HMMs)

Left-to-right Markov models with no skip were used and a total of 48 monophone HMMs were created from the training data using the HTK toolbox (Verion 3.4) (Young et al., 2006).

[2]With all else identical, the use of SA sentences typically improves ASR accuracy by about 2%.

The bigram phone information extracted from the training data was used as the language model. Various numbers of states and mixtures were evaluated as described in the following experiments. In all cases diagonal covariance matrices were used. For final evaluations of accuracy, some of these 48 monophones were combined to create the "standard" set of 39 phone categories.

5.4 Experiment with various reduced dimensions

The first experiment was conducted to evaluate the two NLDA versions with various dimensions in the reduced feature space with and without the use of PCA. As input features, 13 DCTCs, computed with 8 ms frames and 2 ms spacing, were represented with 6 DCSCs over a 500 ms block, for a total of 78 features (13 DCTCs x 6 DCSCs). The 48-dimensional outputs of the neural network were further reduced by PCA in NLDA1, while the dimensionality reduction was controlled only by the number of nodes in the middle layer in NLDA2. The features which were dimensionality reduced by PCA and LDA alone were also evaluated for the purpose of comparison. Figure 16 shows recognition accuracies of dimensionality reduced features using 1-state and 3-state HMMs with 3 mixtures per state. Note that the NLDA1 features without the PCA process are always 48 dimensions. Compared to the PCA and LDA reduced features, the NLDA1 and NLDA2 features performed considerably better for both the 1-state and 3-state HMMs.

For the case of 3-state HMMs, the transformed features reduced to 24 dimensions resulted in the highest accuracy of 69.3% for NLDA1. A very similar accuracy of 69.2% was obtained with NLDA2 using 36-dimensional features. The recognition accuracies were further improved by about 3% with PCA reduced dimensionality features versus the NLDA features for most cases, showing the effectiveness of PCA in de-correlating the network outputs. The accuracies obtained with the original 78 features, and 3 mixture HMMs, are approximately 58% (1 state models) and 63% (3 state models).

5.5 NLDA1 and NLDA2 experiment with various HMM configurations

The aim of the second experiment is a more thorough evaluation of NLDA1 and NLDA2 using a varying number of states and mixtures in HMMs. The 78 DCTC/DCSCs (computed as mentioned in previous section) were reduced to 36 dimensions based on the results of the previous experiment. The 48 phoneme level targets were used in the training of the network. The features which are the direct outputs of the network without PCA processing were also evaluated.

Figure 17 shows accuracies using 1-state and 3-state HMMs with a varying number of mixtures per state. NLDA2 performed better than NLDA1 for all conditions--approximately 2% higher accuracy. The NLDA2 transformed features resulted in the highest accuracy of 73.4% with 64 mixtures, which is about 1.5% higher than the original features for the same condition. The use of PCA improves accuracy on the order of 2% to 10%, depending on the conditions. Although not shown explicitly by the results depicted in Figure 17, it was also experimentally determined that for NLDA1, highest accuracies were obtained using a nonlinearity in the output nodes of the NN for training, but replacing this with a linear node for transforming the features for use with the HMM. In contrast, for NLDA2, best performance was obtained with the nonlinearities used for both training and final transformations. The superiority of the NLDA transformed features is more significant when a small number of mixtures are used. For example, the NLDA2 features modeled by 3-

state HMMs with 3 mixtures resulted in an accuracy of 69.4% versus 63.2% for the original features.

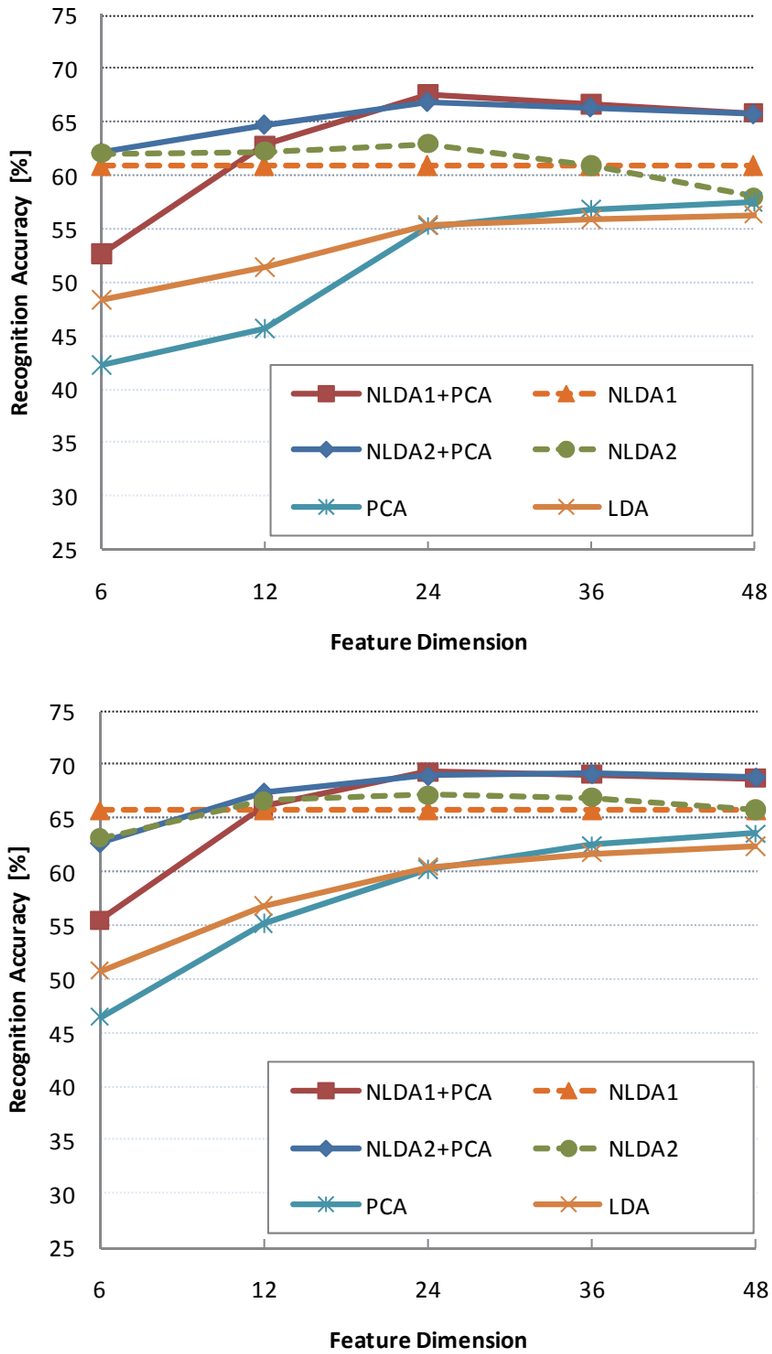

Fig. 16. Accuracies of NLDA1 and NLDA2 with various dimensionality reduced features based on 1-state (top panel) and 3-state HMMs (bottom panel). The NLDA1 features without PCA are always 48 dimensions.

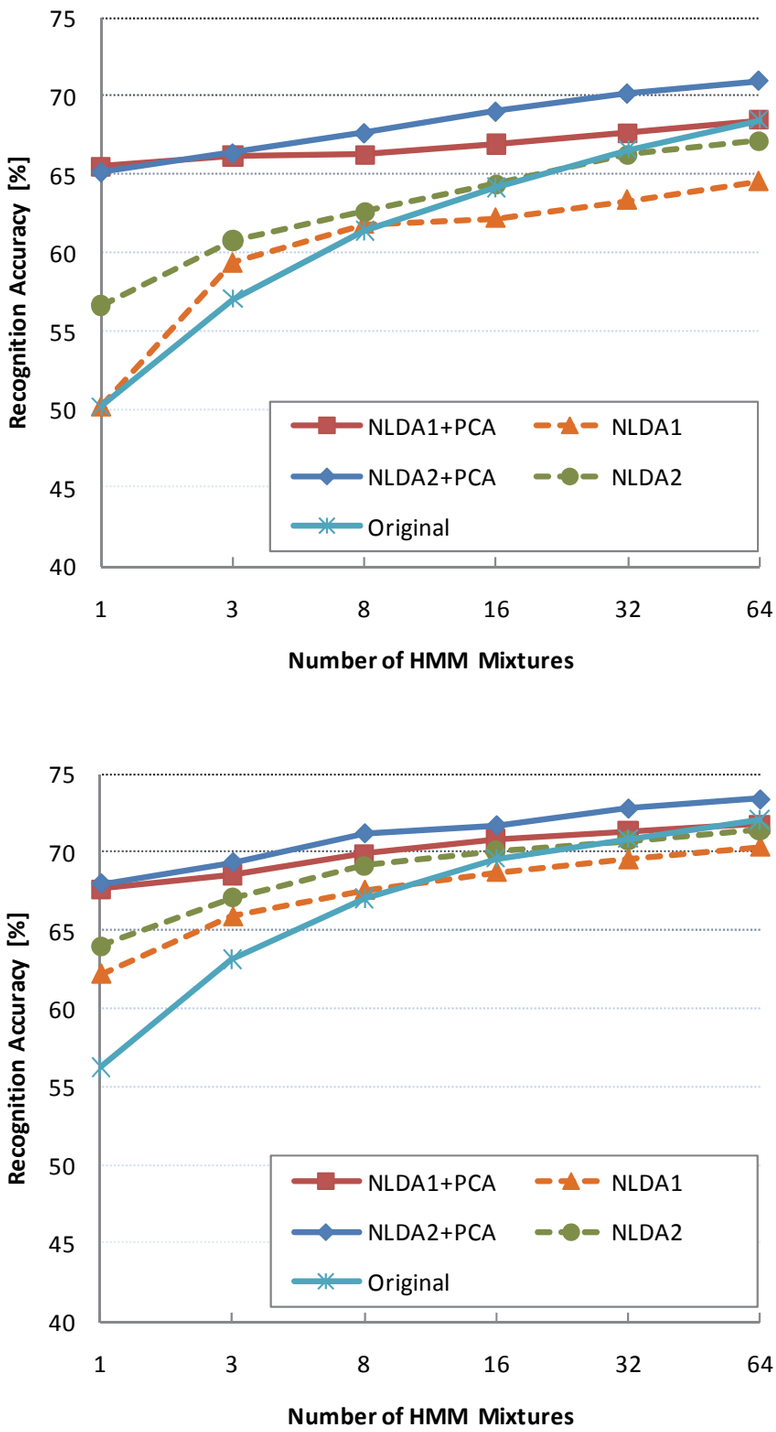

Fig. 17. Accuracies of the NLDA1 and NLDA2 features using 1-state (top panel) and 3-state HMMs (bottom panel) with various numbers of mixtures.

These results imply that the middle layer outputs of a neural network are able to better represent original features in a dimensionality-reduced space than are the outputs of the final output layer. The configuration of HMMs can be largely simplified by incorporating NLDA.

5.6 Experiments with large network training

The results of the previous experiment showed large performance advantages for NLDA2 over NLDA1 and the original features, when using either a small number of features, or a "small" HMM. However, if all original features were used, and a 3- state HMM with a large number of mixtures were used, there was very little advantage of NLDA2, in terms of phonetic recognition accuracy. Therefore, an additional experiment was performed, using the state level targets with "don't cares," as mentioned previously, and a very large neural network for transforming features. The state targets were formed using either a constant length ratio (ratio for 3 states: 1:4:1) or a Viterbi forced alignment approach, as described in Section 4.5. The expanded neural networks had 144 output nodes and were iteratively trained. For both NLDA1 and NLDA2, the networks were configured with 78-500-36-500-144 nodes, going from input to output.

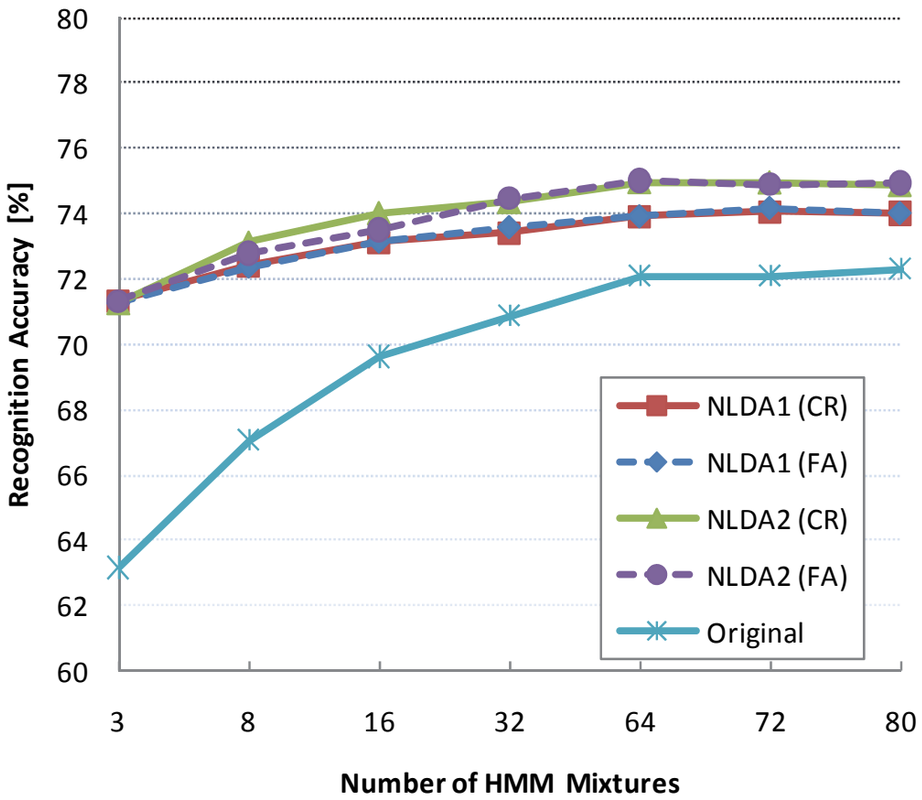

Fig. 18. Recognition accuracies of the NLDA dimensionality reduced features using the state level targets. "(CR)" and "(FA)" indicate the training targets obtained with the constant length ratio and forced alignment respectively.

As shown in Figure 18, both NLDA1 and NLDA2 using the expanded targets lead to a significant increase in accuracy. The NLDA2 accuracies are typically about 2% higher than NLDA1 accuracies. The use of forced alignment for state boundaries resulted in the highest accuracy of 75.0% with 64 mixtures. However, the best result using the much simpler constant ratio method is only marginally lower at 74.9%. Similar experiments, with all identical conditions except using either phone level targets, or state level targets without "don't cares" resulted in about 2% lower accuracies. These results imply that the use of "don't cares" is able to reduce errors introduced by inaccurate determination of state boundaries.

Comparing these results with those from Figure 17, the NLDA2 features in a reduced 36-dimensional space achieved a substantial improvement versus the original features, especially when a small number of mixtures were used. These results show the NLDA methods based on the state level training targets are able to form highly discriminative features in a dimensionality reduced space.

5.7 MFCC experiments

For comparison, 39-dimensional MFCC features (12 coefficients plus energy with the delta and acceleration terms) were reduced to 36 dimensions with the same configurations and evaluated. The results followed the same trend, but the accuracies were about 4% lower than those of the DCTC-DCSC features for all cases, for example, 70.7% with NLDA2 using forced alignment and 32 mixtures.

6. Conclusions

Nonlinear dimensionality reduction methods, based on the general nonlinear mapping abilities of neural networks, can be useful for capturing most of the information from high dimensional spectral/temporal features, using a much smaller number of features. A neural network internal representation in a "bottleneck" layer is more effective than the representation at the output of a neural network. The neural network features also should be linearly transformed with a principal components transform in order to be effective for use by a Hidden Markov Model. For use with a multi-hidden-state Hidden Markov Model, the nonlinear transform should be trained with state-specific targets, but using "don't cares," to account for imprecise information about state boundaries. In future work, linear transforms other than principal components analysis, such as heteroscedastic linear transforms followed by maximum likelihood linear transforms, should be explored for post processing of the nonlinear transforms. Alternatively, the neural network architecture and/or training constraints could be modified so that the nonlinearly transformed features are more suitable as input features for a Hidden Markov Model.

7. References

Bishop, C. M.; Svensén, M. & Williams, C. K. I. (1998). GTM: The generative topographic mapping. *Neural Computation*, 10 (1): 215-234, 1998.

Davis, S. B. & Mermelstein, P. (1980). Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences, *IEEE Trans. on Acoustics, Speech and Signal Processing , 28*, 357-366.

Donoho, D. L. (2000). *Aide-Memoire. High-Dimensional Data Analysis: The Curses and Blessings of Dimensionality.* Department of Statistics, Stanford University.

Duda, R. O.; Hart, P. E. & Stork, D. G. (2001). *Pattern Classification.* (R. O. Duda, Ed.) New York: A Wiley-Interscience Publication.

Ellis, D.; Singh, R. & Sivadas, S. (2001). Tandem Acoustic Modeling In Large-Vocabulary Recognition. *Proc. ICASSP '01*, pp. 517-520, Salt Lake City, USA, May 7-11, 2001.

Fodor, I. (2002). *A Survey of Dimension Reduction Techniques.* Technical Report, Center for Applied Scientific Computing, Lawrence Livermore National Laboratory.

Garofolo, J. S.; Lamel, L. F.; Fisher, W. M.; Fiscus, J. G.; Pallett, D. S. & Dahlgren, N. L. (1993). TIMIT Acoustic-Phonetic Continuous Speech Corpus, http://www.ldc.upenn.edu/Catalog/LDC93S1.html. *TIMIT Acoustic-Phonetic Continuous Speech Corpus*.

Hermansky, H.; Ellis D. P. W. & Sharma, S. (2000). Tandem connectionist feature extraction for conventional HMM systems. *Proc. ICASSP '00*, *3*, pp. 1635-1638. Istanbul, Turkey, June 5-9, 2000.

Hu, H. & Zahorian, S. A. (2008). A Neural Network Based Nonlinear Feature Transformation for Speech Recognition. *Proc. INTERSPEECH '08*, pp. 533-1536, Brisbane, Australia, Sept. 22-26, 2008.

Hu, H. & Zahorian, S. A. (2009). Neural Network Based Nonlinear Discriminant Analysis for Speech Recognition. *Proc. ANNIE 2009.*

Hu, H. & Zahorian, S. A. (2010). Dimensionality Reduction Methods for HMM Phonetic Recognition. *Proc. ICASSP 2010*, pp. 4854 – 4857, Dallas, Texas, March 14-19, 2010.

Jolliffe, I. (1986). *Principal Component Analysis.* New York: Springer-Verlag.

Karnjanadecha, M. & Zahorian, S. A. (1999). Signal modeling for isolated word recognition. *Proc. ICASSP '99*. pp. 293-296, Phoenix, Arizona, March 15-19, 1999.

Kramer, M. A. (1991). Nonlinear principal component analysis using autoassociative neural networks. *AIChE Journal , 37*, 233-243.

Kumar, N. & Andreou, A. G. (1998). Heteroscedastic discriminant analysis and reduced rank HMMs for improved speech recognition. *Speech Communication , 26*, 283-297.

Lee, K.-F. & Hon, H. W. (1989). Speaker-independent phone recognition using hidden Markov models. *IEEE Transactions on Acoustics, Speech, and Signal Processing , 37*, 1641-1648.

Meng, F. (2006). *Whole Word Phonetic Displays for Speech Articulation Training.* Ph.D. Dissertation, Old Dominion University, Norfolk, VA.

Saon, G.; Padmanabhan, M.; Gopinath, R. & Chen, S. (2000). Maximum likelihood discriminant feature spaces. *Proc. ICASSP '00*, pp. II1129--II1132, Istanbul, Turkey, June 5-9, 2000.

Wang, X. & Paliwal, K. K. (2003). Feature extraction and dimensionality reduction algorithms and their applications in vowel recognition. *Pattern Recognition , 36*, 2429-2439.

Young, S.; Evermann, G.; Gales, M.; Hain, T.; Kershaw, D.; Liu, X.; Moore, G.; Odell, J.; Ollason, D.; Povey, D.; Valtchev, V. & Woodland, P. (2006). *The HTK Book (for HTK Version 3.4),* Cambridge University Engineering Department.

Zahorian, S. A.; Qian, D. & Jagharghi, A. J. (1991). Acoustic-phonetic transformations for improved speaker-independent isolated word recognition. *Proc. ICASSP'91*, pp. 561-564. Toronto, Ontario, Canada, May 14-17, 1991.

Zahorian, S.; Silsbee, P. & Wang, X. (1997). Phone Classification with Segmental Features and a Binary-Pair Partitioned Neural Network Classifier. *Proc. ICASSP '97*, pp. 1011-1014, Munich, Germany, April 21-24, 1997.

Zahorian, S. A.; Zimmer, A. M. & Meng, F. (2002). Vowel Classification for Computer-based Visual Feedback for Speech Training for the Hearing Impaired, *Proc. ICSLP 2002,* pp. 973-976, Denver, CO, Sept. 16-20, 2002.

Zahorian, S. A.; Singh, T. & Hu, H. (2007). Dimensionality Reduction of Speech Features using Nonlinear Principal Components Analysis. *Proc. INTERSPEECH '07*, pp. 1134-1137, Antwerb, Belgium, Aug. 27-31, 2007.

Zhao, J.; Zhang, X.; Ganapathiraju, A.; Deshmukh, N. & Picone, J. (1999). Decision Tree-Based State Tying For Acoustic Modeling. *A Tutorial*, Institute for Signal and Information Processing, Department of Electrical and Computer Engineering, Mississippi State University.

Zue, V.; Seneff, S. & Glass, J. (1990). Speech database development at MIT: Timit and beyond. *Speech Communication , 9*, 351-356.

Speech Technologies

Edited by Prof. Ivo Ipsic

ISBN 978-953-307-996-7
Hard cover, 432 pages
Publisher InTech
Published online 23, June, 2011
Published in print edition June, 2011

This book addresses different aspects of the research field and a wide range of topics in speech signal processing, speech recognition and language processing. The chapters are divided in three different sections: Speech Signal Modeling, Speech Recognition and Applications. The chapters in the first section cover some essential topics in speech signal processing used for building speech recognition as well as for speech synthesis systems: speech feature enhancement, speech feature vector dimensionality reduction, segmentation of speech frames into phonetic segments. The chapters of the second part cover speech recognition methods and techniques used to read speech from various speech databases and broadcast news recognition for English and non-English languages. The third section of the book presents various speech technology applications used for body conducted speech recognition, hearing impairment, multimodal interfaces and facial expression recognition.

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

IntechOpen