Neural Networks in Automatic Speech Recognition

Françoise Beaufays [a] Hervé Bourlard [b,c]
Horacio Franco [d] Nelson Morgan [e]

IDIAP–RR 01-09

March 30, 2001

PUBLISHED IN
The Handbook of Brain Theory and Neural Networks, second edition,
M. A. Arbib (Ed.), Bradford Books, The MIT Press, 2000.

[a] Nuance Communications, Speech R&D, Menlo Park, CA 94025
[b] Swiss Federal Institute of Technology at Lausanne (EPFL), CH
[c] Dalle Molle Institute for Perceptual Artificial Intelligence (IDIAP), Martigny, CH
[d] SRI International, Speech Technology Laboratory, Menlo Park, CA 94025
[e] Intl. Computer Science Institute (ICSI), Berkeley, CA 94704

IDIAP Research Report 01-09

Neural Networks in Automatic Speech Recognition

Françoise Beaufays Hervé Bourlard Horacio Franco Nelson Morgan

March 30, 2001

published in
The Handbook of Brain Theory and Neural Networks, second edition, M. A. Arbib (Ed.), Bradford Books, The MIT Press, 2000.

Contents

1 Introduction

Automatic speech recognition (ASR), the technology that allows computer systems to transcribe speech waveforms into words, relies essentially on traditional digital signal processing and statistical modeling methods to analyze and model the speech signal. The core ASR technology is typically not based on connectionnist methods, even though neural network processing is commonly seen as a promissing alternative to some of the current algorithms. Because of the maturity of the current technology, neural networks have to compete with high performance algorithms to gain acceptance in the speech recognition field, and it is only recently that significant performance improvements over state-of-the-art systems have been obtained by using neural networks in specific subsystems of the speech recognizer.

In this paper, we review the main neural network approaches to speech recognition. To limit the scope of this review, we will focus on speech recognizers intended to process large-vocabulary continuous speech (as opposed to small-scale systems to be integrated in electronic devices and that recognize only a few command words), and we will discuss exclusively systems based on multi-layer feedforward neural networks, or multi-layer perceptrons (MLP).

Automatic Speech Recognition

Traditional ASR systems follow a hierarchical architecture. A grammar specifies the sentences allowed by the application. (Alternatively, for very large vocabulary systems, a statistical language model may be used to define the probabilities of various word sequences in the domain of application.) Each word allowed by the grammar is listed in a dictionary that specifies its possible pronunciations in terms of sequences of phones (*e.g.* "k-aa-t" for the word "cat"). Phones are further decomposed into smaller units whose acoustic realizations are represented by statistical acoustic models.

When a speech waveform is input to a recognizer, it is first processed by a front-end unit that extracts from the raw signal a sequence of observations or features. This sequence of observations is then decoded into the sequence of speech units whose acoustic models best fit the observations, and that respect the constraints imposed by the dictionary and language model.

Front End Processing

The purpose of the front-end is to extract from the speech signal a set of features that are robust to acoustic variations but representative of the lexical content of the signal. Typically, the waveform is divided in overlapping "frames" of approximately 25 msec over which the signal is assumed to be stationary. The speech frames then undergo a spectral analysis from which a feature vector is derived. Many variants of spectral analysis have been used, including Linear Predictive Coding (LPC), Perceptual Linear Prediction (PLP) (Hermansky, 1990), and log power spectral or cepstral coefficients (which are the Fourier transform of the log spectrum) computed from a spectrum with "mel scale" spacing, which roughly corresponds to auditory "critical bands" (Davis and Mermelstein, 1980).

To model the time dynamics of the signal, the feature vector is augmented with the first and second time derivatives of its components.

Acoustic Modeling

The acoustic models used to represent the acoustic realizations of the speech units are based on the underlying assumption that words consist of sequences of units that have varying length, and that each unit has constant spectral properties. The nonstationary characteristics of speech arise as the sequence of units is traversed in time. While this is not an accurate model for many speech sounds, it is a simplifying assumption that allows powerful statistical techniques to be applied.

The most commonly used tool to model the speech units is the hidden Markov model (HMM). HMMs assume that the sequence of feature vectors is a piecewise stationary process. Accordingly, an utterance is modeled as a succession of stationary states with instantaneous transitions between these states (modeling the temporal structure of speech), and a set of state output processes –one output process associated to each state– that model the actual observed feature vectors (modeling the locally stationary character of the speech signal). This output process is typically represented by a Gaussian mixture model (GMM) over the feature vectors. The "hidden" qualifier in HMMs refers to the fact that the underlying stochastic process (i.e. the sequence of states) is not directly observable, but still affects the observed sequence of acoustic vectors.

While it is theoretically conceivable to have an HMM for every possible utterance, this is clearly unfeasible for all but extremely constrained tasks. Instead, a hierarchical scheme is typically adopted where a sentence is modeled as a sequence of words, and each word is modeled as a sequence of subword units. The units most commonly used are the phones, i.e., the acoustic realizations of phonemes.

A further refinement is the use of context-dependent phone units, usually referred to as triphones, which are the acoustic realizations of phonemes in the specific context defined by the previous and following phonemes. Typically three HMM states with a left-to-right structure are used to model phone or triphone units.

The simplest and most efficient algorithms to train the parameters of the HMM probability estimators (the Gaussian mixture models as well as the inter-state transition probabilities) are based on maximum likelihood (ML) techniques, that is on finding the model parameters that maximize the likelihood of the training data. The most common of these is the forward-backward algorithm (Baum *et al.*, 1972) a form of the well-known Estimation Maximization (EM) algorithm.

Discriminative Training

Some of the efforts aimed at improving speech recognition have focussed on extending the model training paradigm beyond that of maximum likelihood, and to optimize training criteria more closely related to the actual recognition metric, the recognition error rate.

It has been argued that ML training has the potential for poor discrimination due to the fact that it maximizes the likelihoods of individual models independently of the likelihoods of other (possibly competing) models. Neural network classifiers based on MLPs provide the simplest architecture for discriminative training: feature vectors extracted from the data frames can be input to an MLP and classified into N classes corresponding to the speech units to be modeled, phones or triphones. When appropriately trained, such an MLP classifier can be shown to minimize the classification error-rate over the training data. This form of training explicitly maximizes the discrimination between the correct output class and its competitors. This approach underlies the hybrid MLP/HMM models (Morgan and Bourlard, 1993) that will be described in a later section.

This type of discrimination is "local" in the sense that it makes the models compete over a single frame of data at a time. Speech recognition however is not so much concerned with correctly recognizing individual frames of data as it is with correctly recognizing entire word sequences composed of many data frames. This observation calls for what is referred to as "global" discrimination, where the training algorithm attempts to increase the likelihood of the correct word string while decreasing the likelihoods of all other competing word strings.

A number of approaches to "global" discriminative training have been proposed. The first approach is that of maximum mutual information (MMI), which maximizes the mutual information between the spoken word sequence and its acoustic observations. As a consequence of this formulation, the training procedure maximizes the probability of the correct word sequence while minimizing the probability of the other possible word sequences (Bahl *et al.*, 1986). A common implementation of MMI uses the N-best hypotheses from the recognizer to approximate the likelihood of the competing word sequences.

Another popular training criterion is minimum classification error (MCE), where an approximation to the number of classification errors in the training data is computed and minimized. The approximation is based on computing the difference between a discriminant function for the correct class and a geometric average of the discriminants for the competing classes, and passing this difference through a sigmoid that outputs one when any of the competing classes has a higher discriminant value than the correct class, and zero otherwise (Juang *et al.*, 1992). This method has been used initially to train neural network classifiers and later to train hybrid HMM/MLP hybrid systems.

Yet another approach, REMAP, uses transition-based MLP models with local conditional transition probabilities to estimate the global posterior probabilities of sentences (Konig *et al.*, 1996).

Even though discriminative training had been introduced early in the speech area under the form of MMI, it is the advent of neural networks that spured the interest in discriminative approaches.

2 Acoustic Modeling with MLPs

The idea of combining HMMs and MLPs for the acoustic modeling of speech signals was motivated by the observation that HMMs and MLPs have complementary properties: (1) HMMs are clearly dynamic and very well suited to temporal data, but several assumptions limit their generality; (2) MLPs can approximate any kind of nonlinear discriminant functions, are very flexible, and do not need strong assumptions about the distribution of the input data, but they cannot properly handle time sequences. However, HMMs are based on strict formalisms, making them difficult to interface with other modules in a heterogeneous system. The first such implementations were the so-called hybrid HMM/MLP models.

Context Independent Hybrid HMM/MLP Models

In hybrid models (Bourlard and Morgan, 1993), an MLP is used to evaluate the likelihoods of the observations given the phones, $p(\mathbf{x}_t|q_j)$, where $\mathbf{x}_t$ denotes the observation vector at time t, and q_j denotes the j^{th} phone-state. The MLP is trained as a classifier that estimates phone-class posterior probabilities, $P(q_j|\mathbf{x}_t)$. The posteriors are then inverted with the Bayes rule, according to $p(\mathbf{x}_t|q_j) = P(q_j|\mathbf{x}_t)p(\mathbf{x}_t)/P(q_j)$. The resulting likelihood is used in lieu of the GMM-generated emission probability computed in a conventional HMM-based ASR system. The rest of the HMM structure (transition probabilities, decoding procedure, etc.) is kept unchanged. Because the MLP learns the boundaries between the phone classes, the hybrid models show better phone discrimination characteristics than traditional ASR systems.

In order to provide the MLP with more contextual information, the phone posterior probabilities can advantageously be conditioned upon a window of several consecutive observations (typically 9 frames: $\mathbf{x}_{t-4}, \mathbf{x}_{t-3}, ...\mathbf{x}_t, ...\mathbf{x}_{t+4}$), rather than upon a single frame of data.

Extension to Context Dependent Hybrid Models

In early implementations of hybrid models, the context of a phone, which is known to influence its acoustic realizations because of co-articulation effects, was ignored. This context, denoted here by c_i, can take the form of one or several phones to the right and/or to the left of the current phone, q_j. With context modeling, the likelihood of an observation $\mathbf{x}_t$ can be expressed as $p(\mathbf{x}_t|q_j, c_i) = P(q_j, c_i|\mathbf{x}_t).p(\mathbf{x}_t)/P(q_j, c_i)$, and, in the context of hybrid models, the posterior probability $P(q_j, c_i|\mathbf{x}_t)$ needs to be estimated by an MLP. Simply extending the context-independent approach to estimate biphone or triphone posteriors would greatly increase the number of outputs of the MLP, and thus the number of parameters to estimate. Because this is undesirable, factorization approaches were proposed in which the posterior probability of a phone and of its context, $P(q_j, c_i|\mathbf{x}_t)$, is decomposed into products of probabilities that can be estimated with smaller MLP classifiers. In (Bourlard *et al.*, 1992), this posterior is expressed as $P(q_j, c_i|\mathbf{x}_t) = P(c_i|q_j, \mathbf{x}_t)P(q_j|\mathbf{x}_t)$, where both terms on the right-hand side of the equation can be estimated by separate MLPs (further context factorization is possible in the case of left and right context modeling). In (Franco *et al.*, 1994), the phone and context posterior is factored as $P(q_j, c_i|\mathbf{x}_t) = P(q_j|c_i, \mathbf{x}_t)P(c_i|\mathbf{x}_t)$, where $p(q_j|c_i, \mathbf{x}_t)$ is estimated by a context-dependent MLP whose training data contains only token whose context is c_i, and that is initialized as the context-independent MLP estimating $P(q_j|\mathbf{x}_t)$. $P(c_i|\mathbf{x}_t)$ is estimated by a MLP classifier similar to that estimating $P(q_j|\mathbf{x}_t)$.

Hierarchical Connectionist Acoustic Models

In very large systems (tens of hours of training data – thousands of HMM states), context classifiers are difficult to train because of the nonuniform distribution of context classes for different phones, and because of the large amount of training data. To overcome this problem, (Fritsch, 1997) proposed to cluster the HMM states in a tree-structured hierarchy, and to estimate the posterior probability of a given state as a product of the posteriors of the parent nodes in the tree. This approach is very modular, and allows as many small MLPs as necessary to be trained for a given task. This was the first connectionist approach to outperform state-of-the-art standard HMM systems on the notoriously difficult Switchboard corpus, a large database of conversational telephone speech.

Recurrent Neural Networks

Recurrent MLPs have been used to overcome the independence assumption made by the HMM framework, that is, the assumption that each frame of data is independent from the surrounding frames and that the probability of a sequence of frames can be computed as the product of the probabilities of the individual frames (Robinson, 1994).

In recurrent MLP implementations of hybrid HMM/MLP recognizers, posterior phone probabilities are estimated based not only on the current observations but also on a set of state variables that are themselves functions of the observations and state variables at the previous time step. The recurrence built in the state variables allows the network to capture the dynamic evolution of the speech signal in a more principled and efficient way than by increasing the number of data frames in the input window as in non-recurrent hybrid systems. The weights of the recurrent MLP are typically trained with the backpropagation through time algorithm, a method for “unfolding” the recurrence and treating the time-expanded structure as a multi-layer time-independent feedforward MLP.

This approach proved to be competitive with state-of-the-art HMM-based recognizers on large-vocabulary continuous speech recognition tasks (Hochberg *et al.*, 1994).

Multiband Recognition

Hybrid MLP/HMM systems were recently used in multiband recognition, a technique by which the speech frequency band is divided in several sub-bands that are recognized in parallel. The sub-band segments can be recombined at the frame, phone, or word level, either with a MLP combiner or with a simpler (non-adaptive) mechanism. This approach has shown great potential for speech environments with band-limited noise (by weighting the clean frequency bands more than the noisy bands) (Bourlard *et al.*, 1996) (Tibrewala *et al.*, 1997), and in combination with a full-band hybrid recognizer (Mirghafori *et al.*, 1998).

3 MLP Front-End Processing

The front-end unit, which is responsible for extracting from the waveform vectors of features to be modeled and recognized, is probably the most critical component of an ASR system in that, ultimately, the performance and the simplicity of the acoustic models are determined by the information contained in the speech features and by the relevance of their representation.

Although most of the early attempts at incorporating MLPs in ASR systems focussed on the acoustic models rather than the front end, MLP-based feature extractors have progressively gained more interest in the speech community. There are several reasons for this.

First, traditional front-end units are designed based on classical DSP principles (guided to some extent by our knowledge of the human hearing system). They are mostly non-parametric, rigid, and offer little room for optimization from observed data. The introduction of trainable elements in the front-end allows for a more data-driven design. Second, MLP-based front ends can be jointly optimized with the acoustic models, thereby allowing a closer integration between the two modules. This is especially relevant in the context of discriminative training. Finally, MLPs are powerful tools for introducing new knowledge sources into the front-end unit, without having to make specific modeling assumptions. The following sections will give a few examples of these points.

Speaker Normalization and Adaptation

Speaker recognition systems are typically trained from utterances spoken by different speakers.

These are speaker-independent (SI) recognizers. Speaker-dependent (SD) recognizers, trained from a single speaker's data, typically outperform SI systems, but are understandably harder to obtain, because of the difficulty of collecting massive amounts of speech from each user of the system. Part of the performance gap between SD and SI recognizers can be regained with techniques such as speaker adaptation, *i.e.* adjusting the SI models to fit a specific voice, and speaker normalization (also refered to as feature adaptation), *i.e.* factoring out inter-speaker differences.

There is a vast body of literature on this topic, most of which does not make use of MLPs. Model adaptation algorithms differ in terms of which parameters are adapted (*e.g.* Gaussian means only or means and variances), what criterion is used (maximum likelihood or maximum a posteriori), how the transformations are constrained, etc. Similarly, feature transformations come in different flavors. A few examples of adaptation/normalization implementations using MLPs are given below.

In (Neto *et al.*, 1995), a single layer MLP was inserted between the PLP cepstral front end and a SI connectionist recognizer. It was trained from a small amount of data for each test speaker, to normalize out speaker-specific characteristics from the feature stream. This layer was trained by freezing the parameters of the SI system after it had been trained, and backpropagating errors through the recognizer, down to the MLP layer. This technique brought a 35% WER improvement over the SI recognizer (the SD WER however was 54% lower than the SI WER). Instead, retraining the parameters of the SI recognizer with speaker-specific data (speaker adaptation) didn't work as well (and is trickier because of the larger number of parameters to adjust). (Abrash *et al.*, 1995) reached similar conclusions for native speakers, but found speaker adaptation to work better than speaker normalization for nonnative test speakers, presumably because of the greater difference between the test speakers and the (native) training population.

In order to improve the acoustic resolution of the feature transformation, (Abrash, 1997) followed a "mixture of experts" approach ((Jordan, 1994), (Zhao *et al.*, 1995)). Instead of transforming the speech features with a global transformation according to $\mathbf{x}' = T(\mathbf{x})$, he considered a stack of transformation networks, $T_i(.)$, each specializing at a specific region of the acoustic space, r_i. The overall feature transformation is then expressed as a weighted sum of local transformations, $\mathbf{x}' = \sum_i P(r_i|\mathbf{x})T_i(\mathbf{x})$, where the posterior probabilities $P(r_i|\mathbf{x})$ are estimated by a so-called gating network. In particular, the acoustic regions can be defined to correspond to the different phonemes, or to groups of phonemes.

As mentioned previously, speaker adaptation can also be performed in the model domain, *i.e.* by transforming the parameters of the SI acoustic models. In (Abrash *et al.*, 1996), a nonlinear model transformation was proposed to adapt the means of a standard GMM-based HMM system. The MLP transformations were trained with a generalization of the EM algorithm (GEM), in which the maximization step (M step) is performed with a series of steepest ascend iterations, a technique inspired from backpropagation training.

Joint Feature Extraction and Model Training

Steepest descent/ascent training and the concept of backpropagating error gradients throughout a given structure as commonly done in the field of MLPs finds applications in various aspects of speech recognition. For example, in (Bengio, 1990), a MLP feature transformation was trained along with the HMM-based acoustic models to maximize the MMI criterion for a plosive consonant classification task. Here the feature transformation is SI, and is meant to increase acoustic discrimination rather than focusing on speaker variability. Jointly optimizing both sets of parameters gave higher recognition rates than optimizing the two modules separately.

Similarly, in (Rahim *et al.*, 1998), a feature transformation and a set of acoustic models were jointly trained to minimize the classification errors between subword units (MCE training). The transformation and the acoustic models were iteratively trained by freezing one of the two modules, training the other one, and vice versa. The authors showed that multiple parallel transformations (one for each word class) gave better recognition results than a single transformation, and that a nonlinear MLP transformation performed better than an affine transformation.

It should be noted however that such joint optimization approaches do not require the use of MLP-based/transformed front-ends. For example, in (Biem, 1997), the MCE criterion is used to jointly train the parameters of the front-end filterbank and of a prototype-based classifier.

Noise Robustness

ASR systems typically show lower performances when operated in noisy environments (*e.g.* speech transmitted through a handsfree phone in a moving car). Noises corrupt the speech waveform, and are often difficult to model due to the changing nature of the environment that produces them (speed of the car, quality of the road, weather conditions, etc).

To help solve this problem, MLPs were trained to extract clean speech from noisy input signals. In (Tamura and Waibel, 1988), a signal mapping was performed at the waveform level, by presenting 60 samples of noisy speech to the MLP and using 60 samples from the original clean signal as desired outputs. Good performance was reported in human listening tests. In (Sorensen, 1991), a similar approach was applied to feature vectors rather than to the raw waveform. Experiments with the LPC-cepstrum and with an "auditory feature" defined at an intermediate level between waveform and cepstrum showed from 10 to 65% WER improvement (depending on the SNR) on an isolated word recognition task. In these tests, the auditory feature mapping outperformed the cepstrum mapping.

MLPs have also been used to combine knowledge sources such as instantaneous and sentence-level measures of the SNR, sentence-level estimates of the noise level, and noise-corrupted features to compute an estimate of the feature noise, and subtract this estimate from the noisy feature vector (Weintraub and Beaufays, 1999). This approach reduced the WER on a large-vocabulary conversational speech database corrupted by additive car noise by up to 40%.

Even though these figures are impressive, noise reduction is still an active research topic awaiting more developments, especially in applications where the noise is non-stationary and where it is correlated with the speech signal (*e.g.* speech reverberation in cars).

MLP Front Ends for Speaker Verification

Speaker verification, the task of recognizing a person's identity from his/her voice, is similar in essence to speech recognition: a front-end unit extracts features from the test waveform, and a subsequent classifier uses statistical models of the users's voices to identify the speaker's identity. In (Heck *et al.*, 1999), a MLP was trained to perform frame-level classification of speakers. The inputs to the MLP are 9 adjacent frames of cepstral features, the outputs are speaker posterior probabilities. The MLP has 4 hidden layers, with a bottleneck in the middle. The first 2 layers can be seen as performing discriminative feature compression, whereas the last 2 layers perform speaker classification. After training of the MLP, the last 2 layers are chopped off, and the features computed by remaining layers are fed into a traditional GMM-based speaker classifier. This approach resulted in 15% speaker classification improvement over the baseline GMM classifier. The approach was successfully extended to input in the MLP lower-level features such as filterbank log-energies.

4 Confidence in Recognition Results

Although speech recognition has made tremendous progress in the last decade, the technology is not faultless, and error correction mechanism must be available for a deployed application to be successful. In particular, it is important to assess how confident the recognizer is in its understanding of the user's request before processing it. One way of implementing such a feature is to estimate the posterior probability that each word in the utterance was correctly recognized, and to decide, based on the word posteriors, whether to execute the request or reprompt the user.

Neural networks have been used to implement such a mechanism (Weintraub *et al.*, 1997). During recognition, one gathers a set of cues relative to the current word and to its decoding by the recognizer, *e.g.* how well the acoustic models match the data, whether there are other likely decodings of the spoken segment, how the durations of the sub-word segments fit their expected distributions, etc. These cues act as knowledge sources that are input to an MLP classifier. The MLP is trained with a target output equal to 1 (if the word was correctly recognized) or 0 (misrecognition). At run-time, the MLP classifier will output the posterior probability that the word is correctly recognized, given the measured knowledge sources.

5 Conclusion

Speech recognition was already a fairly mature field by the time multilayer MLPs and the backpropagation algorithm gained general acceptance from the research community. As a consequence, MLPs were experimented with to replace specific components of a well-defined architecture, rather than as a way to solve a brand new problem. A posteriori, the question is thus where and how have MLPs helped to improve speech recognition systems.

Virtually every possible functionality of multilayer MLPs has been exploited: MLP classifiers were used for phone and triphone classification, MLP estimators proved to be useful in feature transformation and noise estimation, MLP combiners helped in merging different knowledge sources to estimate confidence measures, even MLP predictors were experimented with.

In all these applications, the main advantages of MLPs over other statistical modeling methods are: (1) MLP implementations typically require fewer assumptions and can be optimized in a data-driven fashion (the appropriate use of massive amounts of training data seems to be the key to successful speech recognition), (2) backpropagation training can be generalized to any optimization criterion, including maximum likelihood and all forms of discriminative training, and (3) MLP modules can easily be integrated in non-adaptive architectures.

However, most industrial speech recognizers to date count very few MLP components. This can be attributed in part to historical reasons, but more fundamentally MLPs have some limitations that still hamper their integration in ASR systems. In particular, their training time is typically greater than that of non-connectionist models for which closed-form or fast iterative solutions can be found. This is problematic with very large data sets where weeks of training may be necessary to achieve the desired performance. It is even more critical when one considers that, somewhat disconcertingly, increasing the amount of training data and the number of parameters remains the best way to improve recognition performance! Nonetheless, MLPs have made great progress in the speech recognition field since the first edition of this book: scalable implementations have started to appear in the literature, and MLP-based models have outperformed state-of-the-art traditional ASR systems on some of the most challenging recognition tasks.

References

[1] V. Abrash, H. Franco, A. Shankar, and M. Cohen, 1995, Acoustic Adaptation using Nonlinear Transformations of HMM Parameters, *Proc. ICASSP*.

[2] V. Abrash, A. Shankar, H. Franco, and M. Cohen, 1996, Acoustic Adaptation Using Nonlinear Transformations of HMM Parameters, *Proc. ICASSP*.

[3] V. Abrash, 1997, Mixture Input Transformations for Adaptation of Hybrid Connectionist Speech Recognizers, *Proc. Eurospeech*.

[4] L. Bahl, P. Brown, P. de Souza, and R. Mercer, 1986, Maximum Mutual Information Estimation of Hidden Markov Models Parameters for Speech Recognition, *Proc. of ICASSP*.

[5] L. Baum, 1972, An inequality and associated maximization technique in statistical estimation for probabilistic functions of Markov processes, Inequalities, Vol. 3, pp. 1-8.

[6] Y. Bengio, R. De Mori, G. Flammia, and R. Kompe, March 1992, Global Optimization of a Neural Network-Hidden Markov Model Hybrid, *IEEE Trans. on Neural Networks*, Vol. 3, No. 2, pp. 252-259.

[7] A. Biem and S. Katagiri, 1997, Cepstrum-Based Filter-Bank Using Discriminative Feature Extraction Training at Various Levels, *Proc. ICASSP*, pp. 1503-1506.

[8] H. Bourlard and N. Morgan, 1993, *Connectionist Speech Recognition – A Hybrid Approach*, Kluwer Academic Publishers.

[9] H. Bourlard, N. Morgan, C. Wooters, and S. Renals, 1992, CDNN: A Context Dependent Neural Network for Continuous Speech Recognition, *Proc. ICASSP*.

[10] H. Bourlard and S. Dupont, 1996, A new Approach Based on Independent Processing and Recombination of Partial Frequency Bands, *Proc. ICSLP*.

[11] H. Franco, M. Cohen, N. Morgan, D. Rumelhart, V. Abrash, 1994, Context-Dependent Connectionist Probability Estimation in a Hybrid Hidden Markov Model-Neural Net Speech Recognition System, *Computer Speech and Language*, vol. 8, pp. 211-222.

[12] J. Fritsch, 1997, ACID/HNN: A Framework for Hierchical Connectionist Acoutic Modeling, *1997 Workshop on Automatic Speech Recognition and Understanding Proc.*, ed. S. Furui, B.-H. Juang, W. Chou.

[13] H. Hermansky, 1990, Perceptual linear predictive (PLP) analysis of speech, *Journal of the Acoust. Soc. Am.*, 87(4):1738-1752.

[14] M. Hochberg, S. Renals, A. Robinson, D. Kershaw, 1994, Large Vocabulary Continuous Speech Recognition Using a Hybrid Connectionist-HMM system, *Proc. ICSLP* , vol. 3, pp. 1499-1502.

[15] L. Heck, Y. Konig, M. K. Sonmez, M. Weintraub, 2000, Robustness to Telephone Handset Distortion in Speaker Recognition by Discriminative Feature Design, to appear in *Speech Communication* .

[16] N. Mirghafori and N. Morgan, 1998, Combining Connectionist Multi-Band and Full-Band Probability Streams for Speech Recognition of Natural Numbers, *Proc. ICSLP*.

[17] J. Neto, L. Alameida, M. Hochberg, C. Martins, L. Nunes, S. Renals, and T. Robinson, 1995, Speaker-Adaptation for Hybrid HMM-ANN Continuous Speech Recognition Sustem, *Proc. Eurospeech*, pp. 2171-2174.

[18] M. Rahim, Y. Bengio, and Y. LeCun, 1997, Discriminative Feature and Model Design for Automatic Speech Recognition, *Proc. Eurospeech*, Vol. 1, pp. 75-78.

[19] T. Robinson, March 1994, An Application of Recurrent Nets to Phone Probability Estimation, *IEEE Trans. on Neural Networks*, Vol. 5, No. 2, pp. 298-305.

[20] H. Sorensen, 1991, A Cepstral Noise Reduction Multi-Layer Neural Network, *Proc. ICASSP*, pp. 933-936.

[21] S. Tamura and A. Waibel, 1988, Noise Reduction Using Connectionist Models, *Proc. ICASSP*, pp. 553-556.

[22] S. Tibrewala and H. Hermansky, 1997, Multi-band and Adapttaion Approaches to Robust Speech Recognition, *Proc. Eurospeech*, pp. 2619-2622.

[23] M. Weintraub, F. Beaufays, Z. Rivlin, Y. Konig, and A. Stolcke, 1997, Neural-Network Based Measures of Confidence for Word Recognition, *Proc. ICASSP*, Munich, Germany, vol.2, pp. 887-890.

[24] M. Weintraub and F. Beaufays, 1999, Increased Robustness of Noisy Speech Features Using Neural Networks, *Proc. Workshop on Robust Methods for Speech Recognition in Adverse Conditions*, Tampere, Finland, pp. 207-210.

[25] Y. Zhao, R. Schwartz, J. Sroka, and J. Makhoul, 1995, Hierchical Mixtures of Experts Methodology Applied to Continuous Speech Recognition, *Proc. ICASSP*, pp. 3443-3446.

[26] B. Juang and S. Katagiri, 1992, Discriminative Learning for Minimum Error Classification, *IEEE Trans. on Signal Processing*, Vol. 40, No. 12, 1992.

[27] F. Beaufays, M. Weintraub, and Y. Konig, 1998, DYNAMO: An Algorithm for Dynamic Acoustic Modeling, Proc. 1998 DARPA Broadcast News Transcription and Understanding Workshop

[28] L. Rabiner and B.-H. Juang, 1993, *Fundamentals of Speech Recognition*, Prentice-Hall.

[29] Y. Konig, H. Bourlard, and N. Morgan, REMAP – Experiments with Speech Recognition, *Proc. ICASSP*, 1996.