INCORPORATING LINGUISTIC FEATURES IN A HYBRID HMM/MLP SPEECH RECOGNIZER

Victor Abrash†, Michael Cohen†, Horacio Franco†, and Isao Arima‡

†SRI International, Speech Research and Technology Program, Menlo Park, CA, 94025
‡NTT Data Communications Systems Corporation, Kanagawa, Japan

ABSTRACT

We have developed a hybrid speech recognition system which uses a multilayer perceptron (MLP) to estimate the observation likelihoods associated with the states of a HMM. In this paper, we propose two schemes for incorporating distinctive speech features (sonorant, fricative, nasal, vocalic, and voiced) into the MLP component of our system. We show a small improvement in recognition performance on a 160-word speaker-independent continuous-speech Japanese conference room reservation database. Further experiments simulating an improved distinctive feature classifier indicate that this approach can potentially lead to substantial performance improvements.

1. INTRODUCTION

The Hidden Markov Model (HMM) is the most common and successful approach for state-of-the-art speech recognition today. Its success is due to the unified framework it provides for modeling many levels of acoustic and linguistic information, providing a globally optimal recognition hypothesis that simultaneously considers all types of acoustic, phonetic, phonological, and syntactic knowledge that has been incorporated in its model structure. However, this approach also has a limited ability to explicitly incorporate some types of linguistic knowledge; in particular, it is computationally expensive to add new features because distributions for each feature must be included in every state of the HMM. Therefore, it is impractical to add features that may be useful only for particular phonetic discriminations. In addition, the practical use of HMMs for speech recognition is dependent on its *feature independence assumption,* which states that the probability of observing any particular speech feature is independent of the probabilities of observing other simultaneous or non-simultaneous speech features.

We have developed a hybrid speech recognition system which uses a multilayer perceptron (MLP) to estimate the observation likelihoods associated with the states of a HMM. This approach has many advantages: it is non-parametric, requiring no assumptions of a specific (usually gaussian or gaussian mixture) set of model distributions; it leads to a reduction in the number of parameters, since the distributed internal representation in a connectionist network enables model parameters to be shared between phonetic classes; we no longer need separate distributions or mixture weights for each HMM state for each feature; finally, this approach eliminates the feature independence assumption to the extent that all input features can be considered jointly rather than independently, with multiple frame input windows used easily. As a result, any number of new features can be included easily in the MLP paradigm simply by providing additional inputs to the network.

In this work, we explicitly incorporate distinctive speech features (e.g., manner or place of articulation) into our hybrid HMM/MLP system. Distinctive features are the minimal characteristics distinguishing two phonemes, and can be used to classify phonemes or cluster them into broad classes. Other researchers have used distinctive features to construct broad phonetic classes for use in large-vocabulary lexical access [1], for letter or language identification [2], and for coarse phonetic segmentation and classification of input speech in segment-based approaches [1, 2, 3, 4]. To our knowledge, they have not been utilized within the unified framework of a stochastic, frame-based HMM speech recognition system.

2. HYBRID HMM/MLP

2.1. The DECIPHER™ System

The baseline system into which we incorporated MLP probability estimation is the DECIPHER™ system, a phone-based, speaker-independent, continuous-speech recognition system, based on semicontinuous (tied Gaussian mixture) HMMs [5]. The system extracts four features from the input speech waveform, including 12th-order mel cepstrum, log energy, and their smoothed derivatives. The front end produces the 26 coefficients for these four features for each 10-ms frame of speech.

Training of the phonetic models is based on maximum-likelihood estimation using the forward-backward algorithm [6]. Recognition uses the Viterbi algorithm [6] to find the HMM state sequence (corresponding to a sentence) that has the highest probability of generating the observed acoustic sequence.

2.2. Incorporating MLPs into DECIPHER™

The hybrid HMM/MLP DECIPHER system substitutes scaled probability estimates computed with MLPs for the tied-mixture HMM state-dependent observation probability densities. No changes are made in the topology of the HMM system. The experiments described here use an MLP to compute context-independent phonetic probabilities for the 35 phone classes in the Japanese version of the DECIPHER™ system. During the Viterbi recognition search, the probability of acoustic vector Y_t given the phone class q_j, $P(Y_t|q_j)$, is required for each HMM state. Since MLPs can compute Bayesian posterior probabilities, we compute the required HMM probabilities using the formula,

$$P(Y_t|q_j) = \frac{P(q_j|Y_t)\,P(Y_t)}{P(q_j)}$$

The factor $P(q_j|Y_t)$ is the posterior probability of phone class q_j given the input vector Y at time *t*. This is computed by a back-propagation-trained [7] three-layer feed-forward MLP. $P(q_j)$ is the prior probability of phone class q_j and is estimated by counting class occurrences in the examples used to train the MLP. $P(Y_t)$ is common to all states for any given time frame, and can therefore be discarded in the Viterbi computation, since it will not change the optimal state sequence used to get the recognized string.

The MLP has an input layer of 234 units, spanning 9 frames (with 26 coefficients for each) of cepstra, delta-cepstra, log-energy, and delta-log-energy that are normalized to zero mean and unit variance. The hidden layer has 1000 units, and the output layer has 35 units, one for each context-independent phonetic class in the Japanese DECIPHER™ system. Both the hidden and output layers consist of sigmoidal units.

The MLP is trained to estimate $P(q_j|Y_t)$, where q_j is the class associated with the middle frame of the input window. Stochastic gradient descent is used. The training signal is provided by the HMM DECIPHER™ system previously trained by the forward-backward algorithm. Forced Viterbi alignments (alignments to the known word string) for every training sentence provide phone labels, among 35 classes, for every frame of speech. The target distribution is defined as 1 for the index corresponding to the phone class label and 0 for the other classes. A minimum relative entropy between posterior target distribution and posterior output distribution is used as a training criterion. With this training criterion and target distribution, assuming enough parameters in the MLP, enough training data, and that the training does not get stuck in a local minimum, the MLP outputs will approximate the posterior class probabilities $P(q_j|Y_t)$ [8]. Frame classification on an independent cross-validation set is used to control the learning rate and to decide when to stop training as in [9]. The initial learning rate is kept constant until cross-validation performance increases less than 0.5%, after which it is reduced as $1/2^n$ until performance increases no further.

3. APPROACH

Integration of distinctive speech features in our hybrid HMM/MLP speech recognition system was tested using the 160-word vocabulary Japanese Conference Room Reservation (JCRR) database [10]. The JCRR database includes continuous speech from 110 talkers. Each talker reads 84 sentences regarding information about conference rooms. The tests described here are speaker-independent.

In earlier work, a context-dependent tied gaussian mixture HMM system had a word error rate of 11.6% on this database while the hybrid HMM/MLP system (which mixes MLP and HMM probabilities) had a word error rate of 9.5%. For the current experiments, since distinctive features were provided only to the MLP component of our system, a fully context-independent (CI) system in which observation probabilities were estimated solely by the MLP was used as a baseline in order to allow careful examination of the effects of this approach on MLP probability estimation. The baseline word error rate was 16.1%. Five distinctive features (sonorant, fricative, nasal, vocalic, and voiced) were chosen initially, based on their coverage and division of the training data into useful categories. The baseline MLP architecture is as described in the previous section.

We explored two different methods to include the five new features in our standard MLP [11]. The main idea of both approaches was to improve performance by devoting extra network resources to the detailed representation of important linguistic structure (as defined by the broad classes formed by our set of distinctive features) by partially supervising the organization of the hidden layer representation. If the network can make use of the new hidden layer features for phonetic classification, overall accuracy can be improved. One set of experiments involved supervising the training of part of the hidden layer to develop a detailed *local* representation of the distinctive speech features for use in phonetic classification. The other approach aimed to develop a *distributed* feature representation in the entire hidden layer of the network.

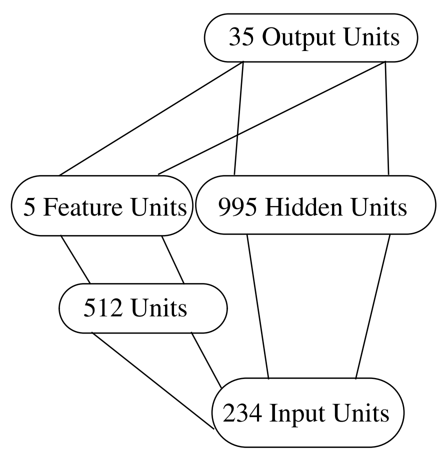

Figure 1: MLP architecture for local feature representation.

Local Representation: In this approach, we pre-train a secondary MLP to estimate the probability of each of the five distinctive features on a frame-by-frame basis. This network uses the same 234 cepstral inputs as before, 512 hidden units, and five outputs. The training target vector is obtained by a straightforward mapping of the training signal used in the phonetic-classifier MLP. Overall accuracy of the distinctive feature extractor MLP on an independent test set was 94.6% correct. The two networks are combined by replacing the first five hidden units of the phonetic MLP with the outputs of the distinctive feature network (Figure 1). The phonetic MLP is then trained as before, with weights inside the secondary network fixed. This architecture guarantees that network resources have been devoted to the features or discriminations judged to be important, and that the hidden layer includes a local representation of those features.

.

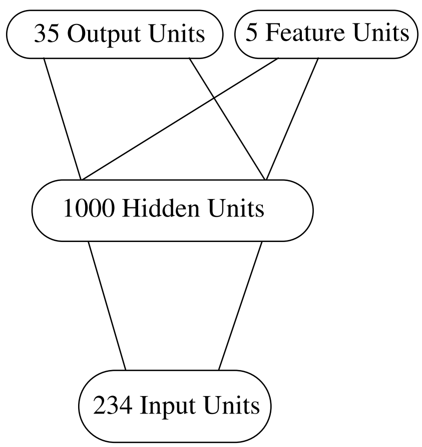

Figure 2: MLP architecture for distributed feature representation.

Distributed Representation: The goal of this approach is to explicitly require the representation of distinctive speech features in the internal hidden layer features developed during training of the MLP, while still allowing the hidden layer to self-organize in a distributed fashion. The architecture is the same as in the baseline system, but with an additional output layer, which has an output unit for each distinctive feature to be represented (Figure 2). In this architecture, there is a complete interconnection between the hidden layer and the new output layer, in addition to the complete interconnection between the hidden layer and the original output layer (which has one output unit for each phonetic class). For each training pattern, the training signal consists of both the correct phonetic class (for the first set of output units) and the correct vector of distinctive speech features (for the second set of output units). The second training vector is obtained as in the first experiment. As a result, during training, signals are back-propagated to simultaneously increase both phonetic discrimination and correct feature classification.

MLP System	Word Error (%)
Baseline	16.1
Local representation	15.4
Distributed representation	15.9
Local representation with perfect oracle	7.0
Local representation with 20% better oracle	12.9

Table 1. Comparison of MLPs using local and distributed methods for integrating distinctive speech features into recognition.

4. RESULTS AND ERROR ANALYSIS

Both approaches improved performance slightly, although the local representation performed better. We examined the issue of whether the lack of substantial improvement was due to inadequate performance in estimating distinctive feature vectors, in which case further error reduction is possible, or in the underlying method. We chose to investigate this issue using the local representation architecture, which is simpler to understand and interpret.

In order to determine whether our architecture could take advantage of better feature detectors in the hidden layer, we ran a set of experiments using an "oracle", simulating a better feature detector than was currently available. In DECIPHER™, we tested a perfect oracle (i.e., one that always knows the feature) and an oracle that had 20% better feature classification than the current feature MLP. In both cases, the rest of the network was computed normally. The 20% improvement was achieved by interpolating between the perfect oracle and the actual classifications by the pre-trained feature MLP. In the case of the perfect oracle, examination of the phonetic classification confusion matrix shows that all network errors are within the broad classes formed by our distinctive features, i.e., between /m/ and /n/ or among the (un)voiced stops. Thus, our approach can potentially eliminate many kinds of phonetic misclassifications. The 16% reduction in error from the interpolation experiment validates that our approach is valid with a realistically better feature classifier.

Poor performance of the feature network can be attributed to many factors: the training targets might be inaccurate, due to poor Japanese pronunciation models in the recognizer; cepstral coefficients may be inadequate for discriminating between our features; our architecture may not be sufficient — only the feature values for the central frame of our 9 frame input window were provided to the phonetic MLP; finally, the large imbalance between the number of positive and negative training examples may have led to a poor allocation of network resources during training.

5. CONCLUSIONS

We have presented preliminary results for two methods of adding five distinctive speech features (sonorant, fricative, nasal, vocalic, and voiced) to our system. We showed that locally integrating knowledge about distinctive speech features into an MLP by training specific hidden units to recognize specific features can improve performance a small amount, but that allocating network resources for the same purpose in a distributed fashion using the architecture described above does not appreciably improve performance. These results show a small performance improvement; however, "oracle" experiments with improved feature detectors have shown that further improvements in feature classifiers that can be incorporated into specific hidden units are likely to lead to substantial performance improvements.

ACKNOWLEDGMENTS

Part of this work was supported by DARPA Contract MDA904-90-C-5253 and part by a contract with NTT Data Communications Systems Corporation.

REFERENCES

[1] L. Fissore, P. Laface, G. Micca, and R. Pieraccini, "Lexical Access to Large Vocabularies for Speech Recognition," *IEEE Trans. on Acoust. Speech and Signal Proc.*, Vol. 37, n. 8, August 1989.

[2] Y. Muthusamy, R. Cole, and M. Gopalakrishnan, "A Segment-Based Approach to Automatic Language Identification," *ICASSP*, pp. 353-356, 1991.

[3] E. Eide J. Rohlicek, H. Gish, and S. Mitter, "A Linguistic Feature Representation of the Speech Waveform," *ICASSP*, pp. 483-486, Vol. 1, 1993.

[4] H. Meng and V. Zue, "Signal Representation Comparison for Phonetic Classification," *ICASSP*, pp. 285-288, 1991.

[5] H. Murveit, M. Cohen, P. Price, G. Baldwin, M. Weintraub, and J. Bernstein, "SRIs DECIPHER System," *DARPA Speech and Natural Language Workshop*, February 1989.

[6] S. E. Levinson, L. R. Rabiner, and M. M. Sondhi, "An Introduction to the Application of the Theory of Probabilistic Functions of a Markov Process to Automatic Speech Recognition," *Bell Syst. Tech. Journal*, Vol. 62, pp. 1035-1074, 1983.

[7] D. Rumelhart, G. Hinton, and R. Williams, "Learning Internal Representations by Error Propagation," in *Parallel Distributetd Processing: Explorations of the Microstructure of Cognition, Vol. I: Foundations*, Ed. D. Rumelhart & J. McClelland. MIT Press, 1986.

[8] N. Morgan and H. Bourlard, "Continuous Speech Recognition Using Multilayer Perceptrons with Hidden Markov Models," *ICASSP*, pp. 413-416, 1990.

[9] S. Renals, N. Morgan, M. Cohen, H. Franco, "Connectionist Probability Estimation in the DECIPHER Speech Recognition System," *ICASSP*, Vol. 1, pp. 601-604, 1992.

[10] O. Shirotsuka, G. Kawai, M. Cohen, and J. Bernstein, "Performance of a Speaker-Independent Japanese Recognizer as a Function of Training Set Size and Diversity," *ICSLP*, pp. 297-300, 1992.

[11] M. Cohen, H. Franco, N. Morgan, D. Rumelhart, and V. Abrash, "Hybrid Neural Network/Hidden Markov Model Continuous Speech Recognition," *ICSLP*, 1992.