Modelling Context Dependency in Acoustic-Phonetic and Lexical Representations[1]

Michael Phillips, James Glass, and Victor Zue

Spoken Language Systems Group
Laboratory for Computer Science
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

ABSTRACT

In 1989, our group first reported on the development of SUMMIT, a segment-based speaker-independent continuous-speech recognition system [13] . The initial version of SUMMIT made use of fairly simple context-independent models for the lexical labels. Recently, we have begun to incorporate more complex models of lexical labels that take into account a variety of contextual factors. These changes, along with an improved corrective training procedure for adapting pronunciation arc weights and a larger set of training data, have resulted in the reduction of error rate by almost a factor of two on the Resource Management task.

INTRODUCTION

Variability in speech arises from many different sources. For example, acoustic variability can be due to noise or channel characteristics, phonetic variability can be due to contextual or speaker-specific effects, and dialect effects can alter speakers' pronunciations of words. Speech recognition systems must have mechanisms to model these various types of variability, and sometimes it may be necessary to deal with different types of variability with different mechanisms. For example, it may be difficult to find a single model that is able to deal effectively with both low-level acoustic variability and dialect differences among speakers.

In the SUMMIT system, we have made a rough distinction between the sort of variability that we can deal with within our phonetic models (including acoustic variability and speaker differences at a phonetic level), and higher level phonological variation (including dialect effects and word-boundary effects). In both cases, our goal is to account for as much of the variability as possible, and it is clear that at least some of the variability is due to contextual effects. Just as there are many types of variability, there are many types of contextual effects, including local phonetic effects (coarticulation), effects of stress, phrase-level effects (such as prepausal lengthening), and higher level effects (such as sentential stress or dialect differences). Therefore, we need to find mechanisms that are able to account for many different types of contextual factors.

In this paper, we will describe a number of experiments intended to address some of the problems mentioned above. So far, we have attempted to account for some of the contextual effects on our phonetic models, although the approach that we have taken should apply to the higher levels of the system also. Briefly, we have found that we can increase recognition performance by creating context-specific models or by using more flexible models. However, we did not see a performance increase when we combined the two in a straightforward manner, presumably due to the fact that more flexible models tend to require more training data. If, instead of using context-specific models, we accounted for context by adjusting the input to the phonetic models (creating a context-normalized input vector), we were able to account for contextual effects and were able to use more flexible phonetic models, resulting in the highest performance for our system.

In the following sections, we will first provide an overview of the system. This will be followed by a more detailed description of the changes we have made to the system, and evaluation results on the Resource Management task.

SYSTEM OVERVIEW

Component Description

A block diagram of the SUMMIT system is shown in Figure 1. The acoustic processing consists of a model of the human peripheral auditory system as a front-end, a hierarchical segmentation algorithm to produce a network of possible acoustic segments, an automatically defined set of segmental measurements for each hypothesized segment, and finally, a statistical classifier for providing a probability of each label given a segment. The result of this analysis branch of the system is a network of possible phonetic interpretations of the speech signal. Each arc in the network has a list of probabilities of the labels used to represent the lexicon [13].

[1]This research was supported by DARPA under Contract N00014-89-J-1332, monitored through the Office of Naval Research.

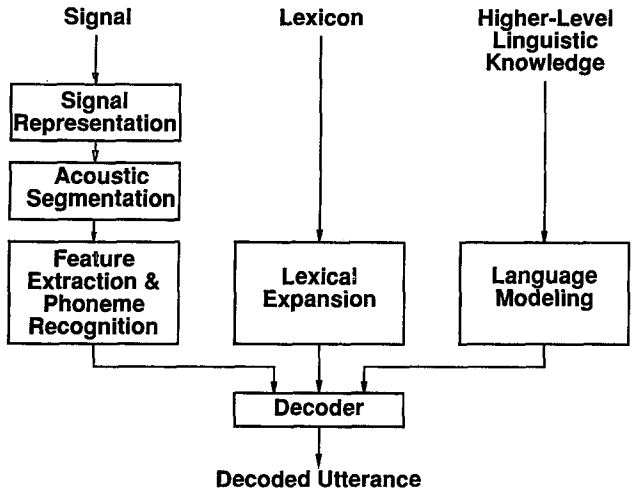

Figure 1: The major components of the SUMMIT system.

The lexicon is also represented as a network, which is derived by applying a set of transformation rules to a set of baseform pronunciations of the words in the lexicon. These transformation rules are defined by hand and are intended to account for some known phonological effects such as flapping and gemination. The pronunciation networks for the individual words are combined into a single network allowing all possible word strings. Inter-word pronunciation rules and local grammatical constraints are taken into account when the words are combined into this network.

Finding the highest scoring word sequence is accomplished by finding the best match between a path in the acoustic network and a path in the lexical network. The initial version of the system used Viterbi search to find the single best match. More recently we have been using the A^*, N-best search described in [15] and [11] to find a list of top scoring sentence hypotheses.

Scoring Strategy

Since the overall score of a path consists of a number of components (acoustic model score, duration model score, segmentation score, and, in some cases, language model score), we must determine a way to combine them. If these were statistically independent probabilities of paths given the acoustics, we could simply combine them by multiplication. Unfortunately, it is unlikely that the component scores are statistically independent. Besides, they are likely to be poor estimates of probabilities both because of lack of training data and because the models used by these components also make mistaken assumptions about their probability distributions and about the statistical independence of the segments making up the path.

In addition, we have the problem in a segment-based system that different paths contain different acoustic segments and therefore have different observation spaces [6]. We cannot simply compare probabilities of word sequences given acoustic observations since the probabilities are computed using different observations. Normalizing the probabilities by the length of the segments helps to some degree (since all paths have the same duration), but then longer duration segments have a greater influence on the path score than short segments.

In the past, we have dealt with these problems by using a weighted linear combination of estimates of the log probability of component scores along with a segment-transition penalty and word-transition penalty as our overall path score. The component weights and transition penalties were obtained by optimizing performance on a portion of the training data.

Recently we have begun to use the N-best search mentioned previously to obtain the top N scoring paths. With the availability of these paths, we can then use the individual component scores as input to a classifier which can be trained to discriminate between correct and incorrect paths. So far, we have been using a linear discriminate function as this classifier, but more complex classifiers can clearly be used. Treating this as a classification problem allows us to not make assumptions about the meaning of the component scores (other than the assumption that we would like them to help discriminate correct from incorrect paths).

This new scoring strategy also permits us to apply, as a post-proccess, constraints that do not fit well into the initial search strategy. For example, we can make use of context dependent models that can consider the global utterance context in addition to the local context.

RECOGNITION EXPERIMENTS

All the experiments described in this paper are performed on the 1,000-word Resource Management (RM) task [7]. In all cases, we have used the perplexity 60 word-pair language model. Except for the baseline system, we have used the now standard 109-speaker training set. To facilitate a meaningful comparison, all the experiments were conducted using the February 1989 speaker-independent test set consisting of 300 utterances, 30 each from 10 different talkers. The experiments that we conducted are summarized in Figure 2, and will be described in this section.

Lexical Models

In the initial version of SUMMIT reported in [13], each label used in the pronunciation of words in the lexicon is represented by a single diagonal Gaussian model. This procedure is illustrated by path (a) in Figure 2. The input to these models is a transformation of a set of segmental acoustic measurements, which were determined automatically using an

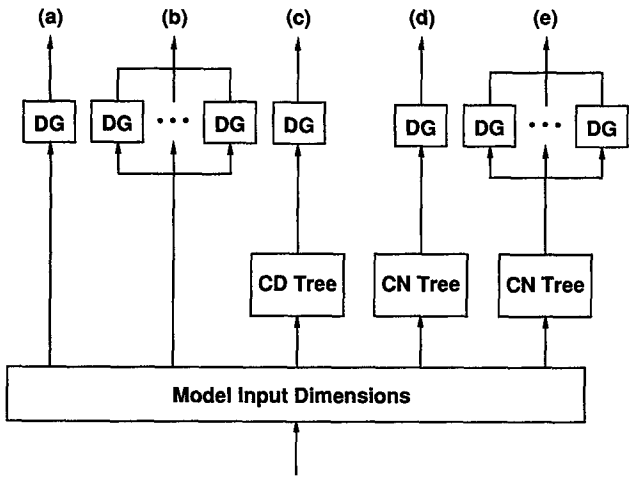

Figure 2: Illustration of the various experimental conditions. DG denotes a diagonal Gaussian classifiers, whereas CD Tree and CN Tree denote context-dependent and context-normalized tree classifiers, respectively, as described in text.

optimization procedure where the optimization criterion was a measure of phonetic discrimination performance [8]. These measurements are based on an entire segment and therefore can potentially take into account both the static and dynamic properties of the segment and its surroundings. The outputs of these measurements form a vector for each segment. This vector is transformed by a combination of linear discriminant functions and principle components analysis to allow for better modelling by the diagonal Gaussian models. The resulting vector has 52 dimensions. This context-independent system achieved a word error rate of 13.6% on the RM task, as shown in the first row of Table 1. This baseline system was trained on the then standard 72 speaker training set. By increasing the training data to 109 speakers and using an improved corrective training procedure described in [14] for training the pronunciation weights, we reduced the word error rate to 12.9%. This new context-independent result is shown in the second row, marked 109-TRAIN, of Table 1.

The intention of using such simple models of the lexical labels was to serve both as a baseline for experiments with more complex models and to allow us to use a simple distortion measure as a criterion for selecting a set of context-dependent models. We have begun both sets of experiments and have been exploring the trade-offs between adding flexibility to our models (which generally require more training data per model) and making use of more specific context-dependent models (which generally allow us to use less training data per model).

Our initial attempts at using more complex models have focused on the use of mixtures of diagonal Gaussians, since this is a natural extension of our baseline system, and mixtures of Gaussians have been shown to be effective in other continuous-density speech-recognition systems [3]. This is illustrated by path (b) in Figure 2. Our mixtures are seeded with a VQ codebook generated with standard hierarchical procedures. A threshold is used to prune away mixtures with too few members. When we replaced the single Gaussian model for each label in system 109-TRAIN (cf. Table 1) by a mixture Gaussian model with a maximum of 16 mixtures per class, the error rate decreased from 12.9% to 10.3%. The detailed results can be seen in the row marked CI-MIXTURES (context-independent mixtures) of Table 1.

Thus far we have kept the transformation of the original acoustic input dimensions intact when using these more flexible models. There are some indications that this transformation may not be necessary, and in fact its elimination may lead to better performance. In addition, we have been experimenting with the use of distinctive features as an intermediate representation [5]. The use of distinctive features may turn out to be a better representation in which to account for factors such as context, speaker, and dialect effects.

Context Dependent Models

Many researchers have found that the use of context-dependent models can lead to an increase in word recognition performance [10,4]. We have been concerned not only with context-dependent modelling but also with the more general problem of lexical representation. The choice of lexical representation involves not only the choice of an inventory of units (such as context-independent or context-dependent models) but also the structure of the pronunciation networks. Many systems currently make use of a rather complex set of units, but then rely on only a single pronunciation path for each word in the lexicon. Although context-dependent models can account for some of the variability due to context, altering the structure of the pronunciation networks may be a more natural way to account for phonological effects such as flapping and gemination, as well as certain types of inter-speaker variability due to dialect differences. Since we are interested in this more general problem of lexical representation, it has been our goal to find a mechanism to automatically define both an inventory of lexical units and a set of pronunciation networks for a given lexicon. We have been treating this as an optimization problem where the goal is to find a set of transformation rules that, when applied to a set of baseform pronunciations, results in a lexical network that optimizes some measure of recognizer performance.

These transformation rules can alter both the labels on the arcs in the network (resulting in context-dependent units) *and* can also alter the structure of the networks (resulting in networks of alternate pronunciations). The rules are able to take into account a variety of contextual factors including local contexts (e.g., whether the left label is a stressed vowel or whether the right label is a /t/), as well as global contexts (e.g. whether the segment is in the last syllable of the sen-

tence). For the experiments reported in this paper, we have limited the optimization to use only rules that alter the labels on the arcs in order to compare to performance increases achieved by other researchers using only context-dependent modelling.

When applying only label-alteration rules, the optimization procedure that we use is basically a top-down tree growing procedure similar to that used by other researchers [1,2,9]. We start with all samples of a given class in the top node of the tree and then in each iteration, try splitting each leaf node in the tree with each of the available contextual factors (such as whether the left label is a stressed vowel), keeping the split that maximizes the criterion over all leaf nodes of the tree. We only allow splits that result in nodes with at least some minimum number of samples in each node. The resulting leaf nodes define the set of context-dependent models. In our case, we would like to use a splitting criterion that is related to the overall recognition performance (since we are trying to obtain the set of context-dependent labels that maximizes recognizer performance). So far we have only experimented with the total squared distance from the mean for the resulting lexical models.

Currently, we are using the following contextual functions for the splits in the context trees:

```
LEFT-LABEL-IN-CATEGORY (class)
RIGHT-LABEL-IN-CATEGORY (class)
LEFT-WB ()
RIGHT-WB ()
```

where class refers to one of a number of categories that we have defined by hand. So far, we have defined 64 categories for the left and right labels. These categories include classes based on broad categories, stress, and distinctive features. Examples of categories include front-vowel, nasal, stressed vowel, etc. The `LEFT-WB ()` and `RIGHT-WB ()` functions return `TRUE` or `FALSE` depending on whether the segment in question is at a left or right word boundary.

If we grow a tree using these contextual factors, using a minimum of 50 samples per leaf node as a stopping criterion, we are able to reduce the squared error in the resulting models by approximately 30%. Using single diagonal Gaussian models in each of the leaf nodes of the tree, we compute a context-dependent model score for each of the N-best paths obtained from the context-independent recognition system. This is illustrated by path (c) in Figure 2. Since we are currently only using local constraints in the context-dependent models, we could have incorporated the models into the initial search. Applying the context-dependent models to the N-best paths saves computation for the current experiments, but more importantly allows us to begin to incorporate more global constraints without changing the experimental paradigm. Using these models as another input to the discrimination classifier discussed above to reorder the N-best paths, we obtain a word error rate of 10.1%. The detailed results are shown in Table 1 in the row marked CD-TREE. In this experiment, we are using a total of 1,300 context-dependent models (this number is obtained by counting the number of leaf nodes in all of the contextual trees). The average number of leaf nodes per contextual tree is approximately 17.

Context Normalized Inputs

We have also experimented with accounting for contextual effects separately for each of the model's input dimensions. That is, rather than growing a single contextual tree for each label, we grow a separate tree for each input dimension. This allows for a more detailed accounting for contextual effects, since different input dimensions are likely to be affected differently by the context. In addition, it also alleviates the dimension scaling problem in the distance metric for the distortion criterion. When growing a single contextual tree for a label, our distortion measure must take into account the distortion in all the dimensions at once, so the scaling of the input dimensions will affect the results. This problem disappears if we consider the distortion one dimension at a time. On the other hand, if context somehow affects the relationship among the input dimensions, we could perhaps take that into account in the single contextual tree but not in the separate input dimension trees.

Since diagonal Gaussian models treat each input dimension separately, we can compute statistics for each dimension based on the contextual tree for that dimension. This is illustrated by path (d) in Figure 2. Using these scores as an additional component into the reordering of the N-best paths gives us a word error rate of 8.5%. The detailed results are shown in the row marked CN-TREE (context-normalized tree) of Table 1. Since we have a different contextual tree for each dimension, we can no longer come up with a meaningful count of the number of context dependent models. However, if we count the leaf nodes of each contextual trees, we find we are using an average 6.8 contexts per input dimension for each class.

Since we have found performance increases both by increasing the flexibility of the models (by using mixture Gaussian models) and by using more specific models (by having separate models depending on context), we wonder if even better results can be obtained by combining both of these procedures. Unfortunately, it turns not to be true due to conflicting requirements of the modelling procedures. More flexible models tend to require a larger number of training samples to obtain good performance, and using more specific models causes us to use a smaller portion of the training data for each model. For example, when we replaced single diagonal Gaussian models with mixture Gaussian models in the leaf nodes of the CD-TREE experiment discussed above, we found no increase in performance. Even if we vary the stopping criterion of the tree splitting procedure (thus controlling the number of training samples we allow for the mixture

Gaussian models) we were not able to obtain any significant increase in performance.

Rather than using the contextual trees to define more specific models, we can use this contextual information to adjust the input dimensions for the effects of the context. This procedure permits us to once again train the models using all of the available training data. Specifically, we grow separate contextual trees for each input dimension as discussed above. Then, rather than using the means and variances to train a Gaussian model for each leaf node, we use only the difference between the mean of the leaf node and mean of the overall class as an adjustment to the vector to account for the contextual effects on samples falling into that leaf node. This of course assumes that we can treat the input dimensions separately when accounting for context (because we are using separate contextual trees for each dimension). It also assumes that contextual effects only cause a shift in the observed input dimensions (and no change in the shape of the distribution of the input dimension). Note that using single diagonal Gaussian models on the resulting context-normalized input vectors is equivalent to using single diagonal Gaussian models in the leaf nodes of the separate dimension contextual trees with the variances tied across all of the leaf nodes for a given input dimension for a given label.

Using context-normalized input dimensions (rather than context-specific models) allows us to use all of the training data for the models for each class. When we replace the single diagonal Gaussian model with the mixture Gaussian models, illustrated by path (e) in Figure 2, we obtained a word error rate of 7%. This represents the best that we have been able to achieve thus far, reducing the error rate of the baseline system by nearly one-half.. The detailed scores for this experiment can be seen in the row labeled CN-MIXTURES (context-normalized mixtures) in Table 1.

System	Correct	Sub	Del	Ins	Error	Sent. Error
Baseline	87.6	10.3	2.1	1.2	13.6	54.7
109-TRAIN	88.4	9.6	2.0	1.3	12.9	54.7
CI-MIXTURES	91.2	7.4	1.4	1.4	10.3	47.7
CD-TREE	90.9	7.7	1.4	1.1	10.1	48.0
CN-TREE	92.6	6.4	1.0	1.1	8.5	43.7
CN-MIXTURES	93.7	5.3	0.7	0.7	7.0	36.0

Table 1: This table shows the results obtained for each of the experiments described in the paper. The columns indicate the percentage of words correct, the percentage of substitutions, deletion, and insertions, the percentage word error (Sub + Del + Ins), and the percentage of sentence error. The systems include: the baseline system, the baseline system trained on the 109 speaker training set, the context-independent mixture Gaussian system, the system using context-dependent trees, the system using context-normalization trees for each input dimension, and finally the system using context-normalization trees along with mixture Gaussian models.

BENCHMARK RESULTS

In connection with the Fourth DARPA Speech and Natural Language workshop, we participated in the benchmark evaluation of the SUMMIT system on the Resource Managenemet task, using the February-91 test set released by NIST. The system used context-normalized input dimensions with mixture Gaussian models, and was trained on the standard 109-speaker training set. The results are shown in Table 2. Comparing the last row of Table 1 with Table 2, we see that our system's performance is quite similar on the two different test sets. We are encouraged by the results of our first attempt at context-dependent modelling. We expect that additional performance gain can be realized when more complex models are introduced.

System	Correct	Sub	Del	Ins	Error	Sent. Error
CN-MIXTURES	93.3	6.0	0.7	1.2	8.0	33.7

Table 2: SUMMIT benchmark performance on the Resource Management task with a perplexity 60 language model, using the February-91 test set released by NIST. The system used context-normalization trees for each input dimension, with mixture Gaussian models.

DISCUSSION & FUTURE PLANS

While the experiments presented here only address local contextual effects, it is important to note that the mechanism that we have developed can account for both local contextua effects and more global contextual effects. Furthermore, the general approach we have taken not only allows us to account for contextual effects on the phonetic models, but also to alter the structure of the pronunciation networks to account for contextual effects. Admittedly, we have only experimented with context-dependent models in these recognition experiments. Even within the limited scope of the current experiments, however, we have achieved substantial performance improvements over our baseline system. In related work, we have experimented with altering the structure of pronunciation networks, resulting in substantial performance increases on the task of recognizing a small set of isolated words over telephone network. We hope that when we extend the present experiments by altering the structure of the pronunciation networks and by considering more contextual effects, we will find further performance increases on the Resource Management task as well.

In the present work we have kept the form of the input representation fixed. Since this particular transformation of the original acoustic dimensions was intended to allow us to model context-independent labels with rather simple diagonal Gaussian models, it may not be an appropriate input representation for the more flexible models discussed here. In particular, since we have so far found that we can achieve the best performance by using the context-normalized input dimensions (which assumes that the normalization can

be carried out for each input dimension independently), we would now like to have input dimensions where context affects the dimensions independently. It is unlikely that the set of dimensions resulting from our current principle components analysis is the best input for this type of normalization. We are now beginning to experiment with applying the normalization to the original input dimensions, which should be more directly affected by contextual effects.

We would also like to explore the use of distinctive features as the input representation since there is some evidence that this might be a better representation for accounting for contextual effects [12]. For example, in the environment of a nasal, we could expect the nasality feature of a vowel to be affected in a particular way whereas other features of the vowel would be affected by other contextual effects.

Finally, if we account for context by making specific models for particular contexts (e.g., triphones or the context-dependent tree discussed above), we are constrained to some degree by the amount of training data we would have available to train each of these more specific models. This has led us in the past to use fairly simple and easily trained parametric distributions for these models.

Accounting for context by normalizing the input dimensions reduces the need to split up the training data, and therefore should lead to more flexible and robust models for the labels in the lexicon. We have thus far presented results using mixture Gaussian models, but are now experimenting with other types of models and discriminators including multilayer perceptrons and radial basis functions.

REFERENCES

[1] Breiman, J., R. Friedman, R. Olshen, and C. Stone, *Classification and Regression Trees*, Wadsword International Group, Belmont, CA, 1984.

[2] Chen, F., and J. Shrager,"Automatic Discovery of Contextual Factors Describing Phonological Variation", *Proc. First DARPA Speech and Natural Language Workshop*, pp. 284-289, Philadelphia, PA, February 1989.

[3] Lee, C., L. Rabiner, R. Pieraccini, and J. Wilpon, "Acoustic Modelling for Large Vocabulary Speech Recognition", *Computer Speech and Language*, Vol. 4, pp 127-165, April 1990.

[4] Lee, K., *Large Vocabulary Speaker Independent Continuous Speech Recognition: the Sphinx System*, Doctoral Thesis, Carnegie Mellon University, Pittsburgh, PA, 1988.

[5] Meng, H, V. Zue, and H. Leung,"Investigation of Signal Representation, Attribute Extraction, and the Use of Distinctive Features for Phonetic Classification," *These proceedings*, 1991.

[6] Ostendorf, M., and S. Roukos,"A Stochastic Segment Model for Phoneme-Based Continuous Speech Recognition," *IEEE Acoustics, Speech, and Signal Processing*, Vol. 37, pp 1857-1869, December 1989.

[7] Pallett, D., "Benchmark Tests for DARPA Resource Management Database Performance Evaluations,"*Proc. ICASSP-89*, pp.536-539, Glasgow, Scotland, May 1989.

[8] Phillips, M., "Automatic Discovery of Acoustic Measurements for Phonetic Classification," *J. Acoustic. Soc. Am.*, Vol. 84, S216, 1988.

[9] Sagayama, S., and S. Honma,"Estimation of Unknown Context Using a Phoneme Environment Clustering Algorithm", *Proc. International Conference on Spoken Language Processing*, pp 361-364, Kobe, Japan, 1990.

[10] Schwartz, R., Y. Chow, O. Kimball, S. Roucos, M. Krasner, and J. Makhoul,"Context-dependent Modelling for Acoustic-Phonetic Recognition of Continuous Speech," *Proc. ICASSP-85*, pp 1205-1208, 1985.

[11] Soong, F., and E. Huang, "A Tree-Trellis Based Fast Search for Finding the N-best Sentence Hypotheses in Continuous Speech Recognition," *Proc. Third DARPA Speech and Natural Language Workshop*, pp. 12-19, Hidden Valley, PA June 1990.

[12] Stevens, K., "Phonetic Features and Lexical Access", *Proc. The Second Symposium on Advanced Man-Machine Interface Through Spoken Language*, pp. 10.1 - 10.23, Makaha, HI, 1988.

[13] Zue, V., J. Glass, M. Phillips, and S. Seneff, "The MIT SUMMIT Speech Recognition System: A Progress Report," *Proc. First DARPA Speech and Natural Language Workshop*, pp. 179-189, Philadelphia, PA, February 1989.

[14] Zue, V., J. Glass, D. Goodine, M. Phillips, and S. Seneff, "The SUMMIT Speech Recognition System: Phonological Modelling and Lexical Access," *Proc. ICASSP-90*, 49-52, Albuquerque, NM, 1990.

[15] Zue, V., J. Glass, D. Goodine, H. Leung, M. McCandless, M. Phillips, J. Polifroni, and S. Seneff, "Recent Progress on the VOYAGER System," *Proc. Third DARPA Speech and Natural Language Workshop*, pp. 206-211, Hidden Valley, PA, June 1990.