DRAGON SYSTEMS RESOURCE MANAGEMENT BENCHMARK RESULTS—FEBRUARY 1991[1]

James Baker, Janet Baker, Paul Bamberg, Larry Gillick,
Lori Lamel, Robert Roth, Francesco Scattone, Dean Sturtevant,
Ousmane Ba, Richard Benedict

Dragon Systems, Inc.
320 Nevada Street
Newton, Massachusetts 02160
DRAGON@A.ISI.EDU
TEL: (617) 965-5200
FAX: (617) 527-0372

ABSTRACT

In this paper we present preliminary results obtained at Dragon Systems on the Resource Management benchmark task. The basic conceptual units of our system are Phonemes-in-Context (PICs), which are represented as Hidden Markov Models, each of which is expressed as a sequence of Phonetic Elements (PELs). The PELs corresponding to a given phoneme constitute a kind of alphabet for the representation of PICs.

For the speaker-dependent tests, two basic methods of training the acoustic models were investigated. The first method of training the Resource Management models is to re-estimate the models for each test speaker from that speaker's training data, keeping the PEL spellings of the PICs fixed. The second approach is to use the re-estimated models from the first method to derive a segmentation of the training data, then to respell the PICs in a largely speaker-dependent manner in order to improve the representation of speaker differences. A full explanation of these methods is given, as are results using each method.

In addition to reporting on two different training strategies, we discuss N-Best results. The N-Best algorithm is a modification of the algorithm proposed by Soong and Huang at the June 1990 workshop. This algorithm runs as a post-processing step and uses an A*-search (an algorithm also known as a 'stack decoder').

1. INTRODUCTION

In this paper we report on some preliminary work done at Dragon Systems' on the Resource Management benchmark task. First, a brief overview of Dragon Systems speaker-dependent, continuous speech recognition system is given. Next, the modifications necessary to evaluate this system on the RM task are described. Our goal has been to make changes to the standard continuous speech recognition system in ways that are in line with Dragon's long term aims. The primary modifications so far have been in the areas of signal processing and speaker-dependent training. The speaker-dependent training is described in detail in Section 4.

Recognition results are given for the RM1 speaker-dependent development test data and for the Feb91 evaluation test material. In presenting these results, we make a start at evaluating the transfer characteristics of our system when responding to changes in the speaker, the hardware, and the signal processing algorithm. Our experimentation was performed using the speaker-dependent development test data, and these data are used to compare system configurations in this paper. Since we believe that we are still on a steep learning curve, the February 1991 evaluation test material was run through the system only one time, and thus comparative results using the evaluation data are not yet available.

2. OVERVIEW OF THE DRAGON CSR SYSTEM

Dragon Systems' continuous speech recognition system was presented at the June 1990 DARPA meeting [1,2,3]. The system is speaker-dependent and was demonstrated to be capable of near real-time performance on an 844 word task (mammography reports), when running on a 486-based PC. The signal processing is performed by an additional TMS32010-based board. The speech is sampled at 12 kHz and the signal representation is quite simple: there are only eight parameters — 7 spectral components covering the region up to 3 kHz and an overall energy parameter — a complete set of which are computed every 20 ms and used as input to the HMM-based recognizer.

The fundamental conceptual unit used in the system is the "phoneme-in-context" or PIC, where the word "context" in

1. This work was sponsored by the Defense Advanced Research Projects Agency and was monitored by the Space and Naval Warfare Systems Command under Contract N000-39-86-C-0307.

principle refers to as much information about the surrounding phonetic environment as is necessary to determine the acoustic character of the phoneme in question. Several related alternative approaches have appeared in the literature [5,6,7]. Currently, context for our models includes the identity of the preceding and succeeding phonemes as well as whether the phoneme is in a prepausally lengthened segment. PICs are modeled as a sequence of PELs (phonetic elements), each of which represents a "state" in an HMM. PELs may be shared among PIC models representing the same phoneme. A detailed description of models for PICs and how they are trained may be found in [2]. Modifications made to the PIC training procedure are presented in Section 4.

Recognition uses frame-synchronous dynamic programming to extend the sentence hypotheses subject to the beam pruning used to eliminate poor paths. Another important component of the system is the rapid matcher, described in [3], which limits the number of word candidates that can be hypothesized to start at any given frame. Some alternative approaches to the rapid match problem have also been outlined by others [8,9,10].

3. MODIFICATIONS TO THE SYSTEM FOR USE WITH THE RM TASK

In order to be able to run the RM benchmark task on the Dragon speaker-dependent continuous speech recognition system, several modifications were necessary. These modifications primarily concerned the signal acquisition and preprocessing stages. Prior to this evaluation, the system had only been evaluated on data obtained from Dragon's own acquisition hardware.

The signal processing, as described above, has always been performed by the signal acquisition board. Thus it was thought possible that the performance of the system would be highly tuned to the hardware. In order to run the RM data through the system, software was written to emulate the hardware. One question to be addressed is how well the signal processing software does in fact emulate the hardware. To assess this, a small test was performed using new data from Dragon's reference speaker. The speaker recorded, using the Dragon hardware, three sets of 100 sentences selected from the development test texts (those of BEF, CMR, and DAS). Recognition was performed, using the reference speaker's base models after adapting to the standard training sentences, and an average word error rate of 3.5% was recorded. The fact that the rate is comparable to error rates of some of the better RM1 speakers suggests that we have emulated our standard signal processing reasonably well. An explicit comparison of performance on the reference speaker using our standard hardware and our software emulation will be available soon.

A lexicon for the RM task had to be specified before models could be built. Pronunciations were supplied for each entry in the SNOR lexicon by extracting them from our standard lexicon. Any entries not found in Dragon's current general English lexicon were added by hand. The set of phonemes used for English contains 24 consonants, 17 vowels (each of which may have 3 degrees of stress), and 3 syllabic consonants. Approximately 22% of the entries in the SNOR lexicon have been given multiple pronunciations. These pronunciations may reflect stress differences, such as stressed and unstressed versions of function words, and expected pronunciation alternatives.

Roughly 30,000 PICs are used in modeling the vocabulary for this task. The set of PICs was determined by finding all of the PICs that can occur given the constraint that sentences must conform to the word pair grammar. The training data used to build PIC models for the reference speaker comes primarily from general English isolated words and phrases, supplemented by a few hundred phrases from the RM1 training sentences. The generation and training of PICs is discussed in more detail in the next section.

The language model used in the CSR system returns a log probability indicating the score of the candidate word. This was modified to return a fixed score if the word is allowed by the word-pair grammar or a flag denoting that the sequence is impermissible.

The standard rapid match module was used in all of the experiments reported in this paper, in order to reduce processing time. We have not focused on the issue of processing time in the current phase of our research, and have therefore modified our standard rapid match parameter settings to be sufficiently conservative so as to insure that only a small proportion of the errors are due to rapid match mistakes.

4. TRAINING ALGORITHMS FOR THE SPEAKER-DEPENDENT MODELS

Dragon's strategy for phoneme-based training was described in detail in an earlier report[2]. We have used a fully automatic version of the same strategy to build speaker-dependent models for each of the RM1 speakers, using the reference speaker's models to provide an initial segmentation. The goal was to build models in which the acoustic parameters and duration estimates were based almost entirely on the 600 training utterances for each speaker, using the reference speaker's models only in rare cases for which no relevant training data is available.

The recognition model for a word (or sentence) is obtained by concatenating a sequence of PICs, each of which is, in turn, were selected in the course of the semi-automatic labeling of

a large amount of data acquired from the reference speaker: about 9000 isolated words and 6000 short phrases. In changing to the Resource Management task, an additional set of task-specific training utterances from the reference speaker were added. Although less than 10% of the training data was drawn from the Resource Management task, most of the PICs that are legal according to the word-pair grammar are represented somewhere in the total training set. Legal PICs missing from the training set are typically like the sequence "ah-uh-ee" that would occur in "WICHITA A EAST": for the most part, they do not occur in the training sentences and seem unlikely to occur in evaluation sentences.

The reference speaker's models are speaker-dependent in three distinct ways:

1. The parameters of the PELs depend on the spectral characteristics of the reference speaker's voice.

2. The durations for the PELs in each Markov model for a PIC depend on the reference speaker's speaking rate and other features of his speech.

3. The sequence of PELs used in the Markov model for a PIC depends on what allophone the reference speaker uses in a given context.

We report on two techniques for creating speaker-dependent PICs starting with the reference speaker's models. The first is a straightforward adaptation algorithm, in which a new speaker's training utterances are segmented into PICs and PELs using a set of base models, and the segments are then used to re-estimate the parameters of the PELs and of the duration models. This algorithm is typically run multiple times. This approach is very effective in dealing with (1), since the 600 training sentences include data for almost all of the PELs. This strategy is less effective in dealing with (2), since only about 6000 of the 30000 PICs occur in the training scripts. Adaptation alone, however, can do nothing to change (3) the "spelling" of each PIC in terms of PELs.

The first technique uses the following two steps:

Step 1: The data from all 12 of the speakers were used to adapt the reference speaker's models. Three passes of adaptation were performed with these data. Since Dragon's algorithm does not yet use mixture distributions, this has the effect of averaging together spectra for male and female talkers and generally "washing out" formants in PELs for vowels. The resulting "multiple speaker" models are not good enough to do speaker-independent recognition, but they serve as a better basis for speaker adaptation than do the reference speaker's models.

Step 2: For a given speaker, a maximum of six passes of adaptation are carried out, starting from the multiple-speaker models. The resulting models are used to segment the utterances into phonemes. At this point we have a good speaker-dependent set of PEL models, and a set of segmentations with which to proceed further.

The second technique begins with the models produced by the first technique together with the segmentation of the training data into phonemes done using those same models. Using this automatic labeling, speaker-dependent training is performed for each of the RM1 speakers, to produce a new speaker-dependent set of PIC models — with new PEL spellings and duration models. The algorithm is as follows:

Step 1: For each phoneme in turn, all the labeled training data for that phoneme are extracted from the training sentences. For each PIC that involves the phoneme, an appropriate weighted average of these data is taken to create a spectral model (a sequence of expected values for each frame) for the PIC. Details of this averaging process may be found in our earlier report[2], but the key idea is to take a weighted average of phoneme tokens that represent the PIC to be modeled or closely related PICs.

The number of PICs to be constructed for each phoneme is of the same order of magnitude as the number of examples of the phoneme in the 600 training sentences. Since there are examples of only about 6000 PICs in the RM1 training sentences, for most PICs the models must be based entirely on data with either the left or right context incorrect. For about one-fifth of the 30000 PICs, therewere insufficient related data to construct a spectral model (using the usual criteria for "relatedness"). This is frequently the case when a diphone corresponding to a legal word pair fails to occur in the training sentences.

Step 2: Dynamic programming is used to construct the sequence of PELs that best represents the spectral model for each PIC, thereby "respelling" the PIC in terms of PELs. This results in a speaker-dependent PEL spelling for each PIC. In the process, speaker-dependent durations for each PEL in a PIC are also computed.

Step 3: Step 2 results in respelled PICs for those PICs for which sufficient training data are available. For the remaining approximately 6000 PICs, the adapted PIC models of the reference speaker are used (as in technique 1). Merging these PICs results in a model for every legal PIC in the word-pair grammar.

Table 1: Comparison of recognition results for RM1 speakers using the two methods of speaker training: speaker dependent models (SD-PELs) and speaker-dependent respelling of PICs (SD PICs). Word error rates are reported as percentages for the RM1 development test data and the Feb91 evaluation data.

Speaker	SD-PELs Development	SD-PICs Development	SD-PICs Evaluation
BEF	10.5	7.2	6.3
CMR(f)	6.9	6.8	15.0
DAS(f)	4.3	2.9	1.9
DMS(f)	4.1	3.1	3.6
DTB	7.6	3.6	7.2
DTD(f)	5.6	4.4	7.8
ERS	12.4	10.5	12.6
HXS(f)	3.1	2.5	5.6
JWS	6.3	4.7	4.5
PGH	5.3	5.5	9.1
RKM	13.9	9.8	9.9
TAB	3.6	4.3	5.3
Average	7.0	5.4	7.5

Step 4: A final pass of adaptation consists of resegmenting the training data into PELs and then re-estimating the parameters of the speaker-dependent PELs. In the process, duration distributions are also re-estimated.

The above algorithm to create speaker-dependent PIC models provides two sets of models with which we have experimented. The first set is referred to as speaker-dependent RM models. The second set is the output of the final stage, and is referred to as the respelled speaker-dependent RM models. Both sets of speaker-dependent models may contain unchanged PICs from the original reference speaker when no training data was available — mainly unchanged duration models, since most PELs are used in a variety of PICs.

5. RECOGNITION EXPERIMENTS AND DISCUSSION

In this section we present results making use of the two sets of speaker dependent models, as well as results on post processing with the N-best algorithm.

5.1 Comparison of two methods for speaker-dependent training

The error rates using each of the training strategies are shown in Table 1. In this table we display the word error rates on the 100 development test sentences for each of the 12 RM1 speakers, and we also display the performance of the respelled models on the Feb91 evaluation data, which consisted of 25 sentences for each speaker.

Table 2: Cumulative percentage of correct sentences on the choice list using the N-Best algorithm.

Choice #	Cumulative %
1	72
2	83
3	87
4	88
5	90
6	91
7	92
8	92
9	93
10	93
11	93
12	93
13	93
14	93
15	94

Analysis of Errors for Speaker-Dependent Respelled PICs

In the course of our research it has been enlightening to investigate the errors. We will now focus our discussion on the performance of the respelled models when recognizing the development data. The word error rates are seen to range from a low of 2.5% for speaker HXS to 10.5% for ERS, with an overall average error rate of 5.4%. When the very same system is run without the rapid match module, the amount of computation is vastly increased, but there is only a small reduction of the observed overall error rate from 5.4% to 5.1%. Roughly 62% of the errors involve function words only, and the remaining 38% involve a content word (and may also include a function word error). Function words have an error rate of 7.6% compared to 2.5% for content words. The most common content word error is "SPS-40" which is often misrecognized as "SPS-48". Other content word errors often involve homophones (such as "ships+s" —> "ships"). Function word deletions are more common than insertions, and substitutions may be symmetric ("and" —> "in" are as frequent as "in" —> "and") or asymmetric ("their" —> "the" but the reverse confusion does not occur). Other common errors involve contractions: "what is" -> "what+s" and "when will" —> "when+ll".

Use of alternate pronunciations

Approximately 22% of the lexical entries have alternate pronunciations. These variants are used to express expected pronunciation alternations and/or stress differences.

5.2 N-Best Algorithm Test.

A recognition pass using an N-Best algorithm was performed on the development test data. The N-Best algorithm which we have implemented is similar to the one proposed by Soong and Huang[4]. It runs as a post-processing step and is essentially a stack decoder which processes the speech in reverse time. Computational results saved during the forward pass are used to provide very close approximations to the best score of a full transcription which extends a reverse partial transcription. Although a more complete description of the algorithm is beyond the scope of the paper, we note that a key difference between the algorithm we use and that of Soong and Huang is that we do a full acoustic match in the reverse pass (i.e., we process the speech data). Also, the reason our extension scores are only approximate is that in our current implementation, the forward and reverse acoustic match scores are different.

The test was run on the 1200 utterances from the RM1 development sentences, 100 each from the 12 RM1 speakers. The parameters controlling the N-Best were set conservatively. With high confidence, the 100 best alternative sentence transcriptions were delivered (slowing down the recognition by about a factor of six). These transcriptions included ones differing only in placement of internal pauses and/or alternative pronunciations. If such transcriptions are considered identical, 17 choices were delivered on average. The results given below do consider such transcriptions as being identical.

The forward algorithm determined the correct transcription 70% of the time, and the N-Best algorithm delivered it as a choice 94% of the time (almost always as one of the top 15). That is, for around 80% of the misrecognitions, the correction was on the choice list. A cumulative count (based on the 1200 test utterances) is given in Table 2. For instance, the correct transcription was one of the top 5 choices 90% of the time.

7. REFERENCES

1. P. Bamberg, Y.L. Chow, L. Gillick, R. Roth, and D. Sturtevant, "The Dragon Continuous Speech Recognition System: A Real-Time Implementation," *Proceedings of DARPA Speech and Natural Language Workshop*, June 1990, Hidden Valley, Pennsylvania, pp. 78-81.

2. P. Bamberg and L. Gillick, "Phoneme-in-Context Modeling for Dragon's Continuous Speech Recognizer," *Proceedings of DARPA Speech and Natural Language Workshop*, June 1990, Hidden Valley, Pennsylvania, pp. 163-169.

3. L. Gillick and R. Roth, "A Rapid Match Algorithm for Continuous Speech Recognition," *Proceedings of DARPA Speech and Natural Language Workshop*, June 1990 Hidden Valley, Pennsylvania, pp. 170-172.

4. F.K. Soong and E.-F. Huang, "A Tree-Trellis Based Fast Search for Finding the N-Best Sentence Hypotheses in Continuous Speech Recognition," *Proceedings of the DARPA Speech and Natural Language Workshop*, June 1990, Hidden Valley, Pennsylvania, pp. 12-19.

5. R. Schwartz et al., "Context-Dependent Modeling for Acoustic-Phonetic Recognition of Continuous Speech", *IEEE International Conference on Acoustics, Speech, and Signal Processing*, April 1985.

6. Bahl et al., "Large Vocabulary Natural Language Continuous Speech Recognition", *IEEE International Conference on Acoustics, Speech, and Signal Processing*, May 1989.

7. K. F. Lee et al., "The Sphinx Speech Recognition System", IEEE *International Conference on Acoustics, Speech, and Signal Processing*, May 1989.

8. Lalit Bahl, Raimo Bakis, Peter V. de Souza and Robert L. Mercer, "Obtaining Candidate Words by Polling in a Large Vocabulary Speech Recognition System", *ICASSP 88*, New York City, April 1988.

9. Xavier L. Aubert, "Fast Look-Ahead Pruning Strategies in Continuous Speech Recognition", *ICASSP 89*, Glasgow, May 1989.

10. Lalit Bahl, P. S. Gopalakrishnan, D. Kanevsky, D. Nahamoo, "Matrix Fast Match: A Fast Method for Identifying a Short List of Candidate Words for Decoding", *ICASSP 89*, Glasgow, May 1989.