RAPID MATCH TRAINING FOR LARGE VOCABULARIES

Larry Gillick, Barbara Peskin, and Robert Roth

Dragon Systems, Inc.
320 Nevada Street
Newton, Massachusetts 02160

ABSTRACT

This paper describes a new algorithm for building rapid match models for use in Dragon's continuous speech recognizer. Rather than working from a single representative token for each word, the new procedure works directly from a set of trained hidden Markov models. By simulated traversals of the HMMs, we generate a collection of sample tokens for each word which are then averaged together to build new rapid match models. This method enables us to construct models which better reflect the true variation in word occurrences and which no longer require the extensive adaptation needed in our original method. In this preliminary report, we outline this new procedure for building rapid match models and report results from initial testing on the Wall Street Journal recognition task.

1. INTRODUCTION

In this paper, we report on a new algorithm for building rapid match (prefiltering) models for Dragon's continuous speech recognizer. The rapid match module is intended to supply the recognizer with a relatively short list of word candidates at every point where the recognizer hypothesizes a new word may begin. To accomplish this, the rapid match module performs a quick but very approximate calculation using a short interval of acoustic data – usually no more than 240 milliseconds of speech – and passes on to the recognizer a list of word candidates which can then be analyzed in detail.

When the rapid match module for Dragon's continuous speech recognizer was first presented nearly two years ago [1], we evaluated its performance on a test corpus of mammography reports involving a vocabulary of under 1,000 words. At that time, the performance of the module was more than adequate to meet the demands of this recognition task. But as we move to larger vocabularies, the demands on rapid match have become greater at the same time that its role in recognition has become more crucial: if we hope to approach anything like real-time recognition on a large-vocabulary task using moderately priced personal computers, the recognizer can entertain word hypotheses for only a tiny fraction of its complete vocabulary. Thus, not only must prefiltering provide models for more words, but those models must be capable of making finer distinctions.

Until now, we had been generating rapid match models based on a single artificially constructed token representing the "average" behavior of each word. But working from a single token made it impossible to adequately model potential variability, and extensive adaptation of the models was necessary both to estimate variances and to adjust model parameters to new speakers. In our new training procedure, we instead build word models directly from hidden Markov models for each speaker's vocabulary. As reported below, these new models have allowed us to significantly improve prefiltering performance.

After a brief review of the rapid match module in the next section, we go on to describe in detail our new procedure for building rapid match models. Results from preliminary testing of these models using the Wall Street Journal recognition task are reported in section 4. We close with a discussion of the future directions we hope to explore.

2. REVIEW OF THE RAPID MATCH MODULE

The main job of the rapid match module is to provide the recognizer with a short list of words that may begin at any particular time by looking at speech data beginning at that time and extending only a brief period into the future. To accomplish this, we first construct "smooth frames" of speech by taking a (possibly weighted) average of several frames of acoustic data. For our continuous speech recognition, we have been using three smooth frames of information, each obtained by averaging together four successive 20-millisecond frames of speech. Such smooth frames have the dual benefit of condensing the acoustic information into a much smaller number of parameters and doing so in a way that reduces the sensitivity to potential variation in phoneme duration. The number of speech frames used in calcu-

*This work was sponsored by the Defense Advanced Research Projects Agency and was monitored by the Space and Naval Warfare Systems Command under contract N00039-86-C-0307.

lating a smooth frame, the number of smooth frames, and the offset from one smooth frame to the next are all adjustable parameters in the rapid match module.

As the smooth frames are computed, they are scored against models for word start clusters, which are groups of words whose beginnings are acoustically similar. These word start groups are formed automatically using a specialized clustering algorithm starting from smooth models for the words in the vocabulary. Clearly, this clustering of words into acoustically-similar groupings – a step performed during the rapid match training – results in further efficiencies at recognition time. Each word start cluster is represented by a sequence of probability distributions, one for each smooth frame of the model. We currently assume that each probability density is a product of double exponential distributions, one corresponding to each of the smoothed acoustic parameters. Thus each smooth frame of a word start model is determined by a collection of (mean, deviation)-pairs. We reduce run-time calculations still further by allowing several word start clusters to share the same probability densities for some of their smooth frames. This second level of clustering, like the first, is performed automatically as part of the training process and results in a collection of "position clusters" used for the spelling of all word start groups.

Each word of the vocabulary may belong to several different word start clusters, depending on the context in which the word finds itself. We currently generate four models for each word, based on whether the word emerges from silence or speech and whether it is followed by silence or speech. The number of smooth frames representing a word start group is determined by the lengths of its members. In our current implementation, most words have models filling all three smooth frames, but some very short words (most commonly function words like "the", "to", and "of" when embedded in continuous speech) receive models with fewer frames.

During recognition, as smooth frames are generated from incoming acoustic data, they are scored against the various word start clusters using the negative log likelihood for the probability models for each group. The score for a word start group is computed as an average over the scores from each of the smooth frames in its model. For every word start group scoring within a certain threshold, the words belonging to the group are looked up, possible duplicates are removed, and a language model score for each word is added to its word start score. The list of all words whose combined score falls within a second threshold is then passed on to the recognizer for a more complete analysis.

For more details on the rapid match module, consult [1].

3. BUILDING BETTER MODELS

The process of creating word start groups begins from sample tokens for the words in the recognizer's vocabulary. The speech frames are averaged together into smooth frames, just as in the rapid match recognition process, and these smoothed versions are then clustered into word start groups.

Until now, this process began from a single token representing the "average" behavior of each word. Dragon's word models are built up from basic building blocks called phonemes-in-context, or PICs. The representative tokens used by the rapid matcher were constructed by concatenating PIC tokens built by means of a linear alignment routine. Through linear stretching and shrinking operations, examples of the desired phoneme were normalized to a common length and then the acoustic parameters averaged together on a frame-by-frame basis. (See [2] for a more detailed description of PIC models and the construction of aligned tokens.) Unfortunately, in the course of alignment, any usable information about the variability of frame parameters is lost.

Although the models formed in this way were sufficient for a task like the mammography study, the strategy suffers from three main deficiencies:

- Because each word model is based on a single token, there is no way to measure the variability of model parameters. Such estimates must be incorporated during adaptation of the models.

- Because the token is constructed from a linear alignment of phonemic units, the model rigidly expects a particular phoneme in a particular frame and so is relatively intolerant of variation in phoneme duration. While the alignment process involves blending different behaviors within the phonemic unit, the representation does not allow for mixing frames involving different PICs. Averaging together several successive acoustic frames to create the "smooth frames" used in rapid match softens this effect, but cannot eliminate it.

- Finally, because the token is based on the reference speaker's models, extensive adaptation is necessary to adjust the model parameters to other speakers. And while adaptation can successfully modify values for the (mean, deviation)-pairs representing word start clusters, it cannot alter the spelling of word start clusters by position clusters nor the assignment of words to word start groups. Both of these steps

are performed once and for all based on the reference speaker's models.

Our new method for building rapid match models overcomes these difficulties by working directly from HMMs representing the words for each speaker's vocabulary. In the new rapid match training, we begin from the phonemic spelling of each word and, using the speaker's own models, unpack the sequence of nodes representing each PIC. We then generate a collection of sample tokens by simulated traversals of this node sequence. At each node, we determine the duration of the stay by a random draw from a double exponential duration distribution and then, for each of the resulting number of frames, generate parameter values by independent draws from the output distribution for the node. The resulting collection of sample tokens exhibits all the variability one would expect to see in actual occurrences of the word. These tokens are then converted to their smoothed forms, the smoothed versions averaged together smooth frame by smooth frame to obtain both means and deviations for the new word model, and the usual clustering algorithm can then be followed.

Of course, the sample tokens generated by independent draws from the output distributions are probably not themselves accurate representations of actual word occurrences; we would expect a high degree of correlation between successive frames in actual speech. But because these samples are processed through two rounds of averaging – the first combining successive acoustic frames into a single smooth frame and the second averaging smooth frames from the many sample tokens – we expect the resulting means to be fairly well estimated. On the other hand, our assumption of independence of frames probably leads to an underestimate of the true frame deviations. For example, in the extreme (and purely hypothetical) case that the four successive acoustic frames were in fact identical in actual speech, our random draws would underestimate the deviations by a factor of two. In general, we expect to be off by a considerably smaller factor, but we have found that performance of our new models is improved if we scale up all our estimated deviations by a factor in the range 1.3-1.5.

4. INITIAL RESULTS ON THE WALL STREET JOURNAL TASK

Our goal is to ensure that the correct word candidate is returned by the rapid matcher in the list of the top 100-200 words. We do not require that it be the highest ranked – the recognizer will do the hard work of analyzing the top candidates in detail – but it is essential that the correct candidate not be excluded from this analysis. Therefore, our evaluation of the new rapid match training program concentrates on performance in this range.

In order to assess how close we've come to meeting our goal, we have been using an evaluation package which ranks the word candidates nominated by the rapid matcher in any given speech frame. By running the recognizer in a mode where it knows the correct transcription for a text, we can obtain a segmentation of each utterance, marking the frame in which each word is most likely to begin. We then use the evaluation package to look at what rank the correct word has in the list of candidates passed on to the recognizer in that frame.

To provide an initial reading on the new rapid match training and to help set clustering thresholds, we first looked at its performance on the mammography task. While we did not expect the new routine to improve noticeably on our earlier performance – it was, after all, a relatively easy task involving a limited vocabulary and recorded by our reference speaker – it was reassuring to find that the new routine, like the old, returned the correct word in the list of the top 100 candidates over 99% of the time for a test set roughly 4300 words long, and by the top 200 words, the correct candidate failed to appear on the list only about 1 time in 1000.

We then moved on to the more challenging Wall Street Journal task. Here we built new rapid match models for the 5K verbalized punctuation vocabulary for 5 of our 12 speakers, ranging from our worst performer to our best, and compared them to the original models which had already been adapted to each speaker. (For a description of our overall performance on the Wall Street Journal task, see the companion article [3].) The results are summarized in Table 1, which reports what percent of the time the correct word was included in the word candidate list returned by rapid match, as a function of the length of the list. The test sets involved about 40 sentences totaling somewhat over 700 words per speaker. They were drawn from the 5K verbalized punctuation speaker-dependent Wall Street Journal corpus. In all cases the new models improved significantly over the old, usually cutting the error rate by 25-50%.

Although the new training method obviates the need for adaptation of models, we were curious about whether adaptation would further improve the performance of the rapid match system. We therefore have begun experimenting with adapting our new rapid match models. Preliminary results indicate that we can expect to gain about another percentage point improvement even after a single round of adaptation. A sample is given in Table 2, for speaker 00A.

We have also begun building new rapid match models for

speaker I.D.	list length	old models	new models
00A	100	75.3	82.1
	150	82.4	86.9
	200	84.5	90.5
	250	87.2	91.9
00C	100	79.3	83.2
	150	85.6	88.4
	200	88.5	90.4
	250	89.8	91.9
001	100	88.2	92.8
	150	90.8	94.2
	200	93.0	95.2
	250	94.5	95.9
203	100	85.4	91.8
	150	89.3	93.9
	200	91.0	95.2
	250	92.1	95.9
432	100	87.6	92.4
	150	91.5	94.8
	200	93.5	96.7
	250	94.6	97.5

Table 1: Percentage of time correct word returned by rapid match, by list length, for Wall Street Journal 5K task.

the 20K vocabulary. Results for a sampling of speakers on the 20K task are given in Table 3. Clearly the difficulty of the rapid match task grows significantly with vocabulary size. However, it should be noted that while the job of creating sufficiently good models grows enormously as the vocabulary grows, the burden at recognition time does not: the number of word start clusters grows much more slowly than the vocabulary size both because we allow the clustering thresholds to increase gradually with vocabulary size and because large vocabularies permit more sharing of cluster models. For example, the number of word start clusters for the mammography task (with a vocabulary of 860 words) was about 1500, for the 5K Wall Street Journal task about 5000 clusters, and for the 20K vocabulary about 6000 clusters. (Recall that each word is given four context-determined models, so the actual number of word models is four times the vocabulary size.)

A word should be said about the relationship between results on these evaluation tests and actual recognition performance. We have found that even if a word has a poor rank in the frame in which the recognizer ideally expects the word to begin, a good score in a neighboring frame will often allow the recognizer to get the word

list length	before adaptation	after adaptation
100	82.1	83.9
150	86.9	88.3
200	90.5	90.8
250	91.9	92.7

Table 2: Effect of one round of adaptation on rapid match models for 00A.

right. On the other hand, if a word fails to be passed on to the recognizer within a small window around the optimal word start, performance will suffer. Being deprived of the correct word, the recognizer is forced to follow a false path through the web of sentence hypotheses, usually resulting in two or three word errors. Thus, even small improvements to the rapid match module can have a significant impact at recognition time. As an example of the relationship between the rapid match evaluation results and actual recognition performance, Table 4 gives rapid match results for both old and new training models for our in-house speaker SAL on the Wall Street Journal 5K test set, along with word error rates in the related recognition tests.

list length	speaker 00A	203	432
200	77.5	90.2	90.1
400	84.0	93.7	93.0
600	87.2	95.0	94.5
800	89.5	96.0	95.5
1000	91.4	96.8	96.6

Table 3: Percentage of time correct word returned by rapid match, for Wall Street Journal 20K task.

5. FUTURE PLANS

The work described above is only the beginning of a long-term project to improve the performance of Dragon's rapid match algorithm for continuous speech recognition. Most immediately, we plan to work at tuning the many parameters involved in rapid match training. In the results cited above, we deliberately chose parameter settings as close as possible to those used in our original training routine, using, for example, three smooth frames of length four with each of the four acoustic frames given equal weight. But there is no reason to believe that these values optimize performance. In our isolated word recognizer, in contrast, rapid match uses five smooth frames

Rapid Match Performance

list length	old models	new models
100	83.1	91.1
200	90.0	94.9
300	92.0	96.6
400	93.3	97.4
500	94.1	98.1

Recognition Results

OLD MODELS		NEW MODELS	
avg #words returned	word error rate	avg #words returned	word error rate
122	17.1	106	10.5
161	13.2	140	9.4
236	11.4	207	7.8
348	10.6	305	7.2
458	9.4	400	7.0

Table 4: Comparison of rapid match performance and recognition results.

computed from overlapping windows of length six with weighting coefficients 1, 4, 6, 6, 4, 1. We plan to experiment with different window sizes and weights, with special attention to the benefits of reading more deeply into a word.

In the past year, Dragon has moved from its original set of 8 signal-processing parameters to a set of 32, adding 12 cepstral and 12 difference cepstral parameters. The rapid match models described above used only the original 8 parameters, but we should be able to improve performance by using information from all 32. To keep the recognition-time computation low, we plan to explore ways of distilling the 32 parameters down to a small but effective collection of smooth parameters, possibly by means of principal component techniques. We have also begun using tied mixture models for our continuous speech recognition (see [3]) and the token generation which forms the heart of our new training strategy must be modified to work for these distributions. We also hope to move from the naive hypothesis of the independence of adjacent speech frames to a token generation system capable of incorporating trends across frames.

We are encouraged by the significant gains produced by the first stages of our new rapid match training program and look forward to further improvements as these additional features are incorporated.

REFERENCES

1. L. Gillick and R. Roth, "A Rapid Match Algorithm for Continuous Speech Recognition," *Proceedings of DARPA Speech and Natural Language Workshop*, Hidden Valley, Pennsylvania, June 1990, pp. 170-172.
2. P. Bamberg amd L. Gillick, "Phoneme-in-Context Modeling for Dragon's Continuous Speech Recognizer," *Proceedings of DARPA Speech and Natural Language Workshop*, Hidden Valley, Pennsylvania, June 1990, pp. 163-169.
3. J. Baker et al., "Large Vocabulary Recognition of Wall Street Journal Sentences at Dragon Systems," *this Proceedings.*
4. L. Bahl, R. Bakis, P.V. deSouza, and R.L. Mercer, "Obtaining Candidate Words by Polling in a Large Vocabulary Speech Recognition System," *ICASSP 88*, New York City, April 1988.
5. X.L. Aubert, "Fast Look-Ahead Pruning Strategies in Continuous Speech Recognition," *ICASSP 89*, Glasgow, May 1989.
6. L. 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.