COMPARATIVE EXPERIMENTS ON LARGE VOCABULARY SPEECH RECOGNITION

Richard Schwartz, Tasos Anastasakos, Francis Kubala,John Makhoul, Long Nguyen, George Zavaliagkos

BBN Systems & Technologies
70 Fawcetr Street, Cambridge, MA 02138

ABSTRACT

This paper describes several key experiments in large vocabulary speech recognition. We demonstrate that, counter to our intuitions, given a fixed amount of training speech, the number of training speakers has little effect on the accuracy. We show how much speech is needed for speaker-independent (SI) recognition in order to achieve the same performance as speaker-dependent (SD) recognition. We demonstrate that, though the N-Best Paradigm works quite well up to vocabularies of 5,000 words, it begins to break down with 20,000 words and long sentences. We compare the performance of two feature preprocessing algorithms for microphone independence and we describe a new microphone adaptation algorithm based on selection among several codebook transformations.

1. INTRODUCTION

During the past year, the DARPA program has graduated from medium vocabulary recognition problems like Resource Management and ATIS into the large vocabulary dictation of Wall Street Journal (WSJ) texts. With this move comes some changes in computational requirements and the possibility that the algorithms that worked best on smaller vocabularies would not be the same ones that work best on larger vocabularies. We found that, while the required computation certainly increased, the programs that we had developed on the smaller problems still worked efficiently enough on the larger problems. However, while the BYBLOS system achieved the lowest word error rate obtained by any site for recognition of ATIS speech, the error rates for the WSJ tests were the second lowest of the six sites that tested their systems on this corpus. The reader will find more details on the evaluation results in [1].

In the sections that follow, we will describe the BBN BYBLOS system briefly. Then we enumerate several modifications to the BBN BYBLOS system. Following this we will describe four different experiments that we performed and the results obtained.

2. BYBLOS

All of the experiments that will be described were performed using the BBN BYBLOS speech recognition system. This system introduced an effective strategy for using context-dependent phonetic hidden Markov models (HMM) and demonstrated their feasibility for large vocabulary, continuous speech applications [2]. Over the years, the core algorithms have been refined with improved algorithms for estimating robust speech models and using them effectively to search for the most likely sentence.

The system can be trained using the pooled speech of many speakers or by training separate models for each speaker and then averaging the resulting models.

The system can be constrained by any finite-state language model, which includes probabilistic n-gram models as a special case. Nonfinite-state models can also be used in a post process through the N-best Paradigm.

The BYBLOS speech recognition system uses a multi-pass search strategy designed to use progressively more detailed models on a correspondingly reduced search space. It produces an ordered list of the N top-scoring hypotheses which is then reordered by several detailed knowledge sources.

1. A forward pass with a bigram grammar and discrete HMM models saves the top word-ending scores and times [6].
2. A fast time-synchronous backward pass produces an inital N-best list using the Word-Dependent N-best algorithm[5].
3. Each of the N hypotheses is rescored with cross-word-boundary triphones and semi-continuous density HMMs.
4. The N-best list can be rescored with a trigram grammar (or any other language model).

Each utterance is decoded with each gender-dependent model. For each utterance, the N-best list with the highest top-1 hypothesis score is chosen. The top choice in the final list constitutes the speech recognition results reported below. This N-best strategy [3, 4] permits the use of otherwise computationally prohibitive models by greatly reducing the search space to a few (N=20-100) word sequences. It has enabled us to use cross-word-boundary triphone models and trigram language models with ease.

During most of the development of the system we used the 1000-Word RM corpus [8] for testing. More recently, the system has been used for recognizing spontaneous speech from the ATIS corpus, which contains many spontaneous speech effects, such as partial words, nonspeech sounds, extraneous.noises, false starts, etc. The vocabulary of the ATIS domain was about twice that of the RM corpus. So there were no significant new problems having to do with memory and computation.

2.1. Wall Street Journal Corpus

The Wall Street Journal (WSJ) pilot CSR corpus contains training speech read from processed versions of the Wall Street Journal. The vocabulary is inherently unlimited. The text of 35M words available for language modeling contains about 160,000 different words. The data used for speech recognition training and test was constrained to come from sentences that contained only the 64,000 most frequent words.

There are two speech training sets. One has 600 sentences from each of 12 speakers (6 male and 6 female). The other has a total of 7,200 sentences from 84 different speakers. The total vocabulary in the training set is about 13,000 words. There are two different standard bigram language models that are typically used – one with 5,000 (5K) words and one with 20,000 (20K) words. The 5K language models were designed to include all of the words in the 5K test set. The 20K language models contain the most likely 20K words in the corpus. As a result, about 2% of the words in the test speech are not in this vocabulary. In addition, there are two variants depending on whether the punctuation is read out loud: verbalized punctuation (VP) and nonverbalized punctuation (NVP).

Most of the test speech is read. In addition to test sets for 5K-word and 20K-word vocabularies, there is also spontaneous speech collected from journalists who were instructed to dictate a newspaper story.

3. IMPROVEMENTS IN ACCURACY

In this section, we describe several modifications that each resulted in an improvement in accuracy on the WSJ corpus. In all cases, we used the same training set (SI-12) and the standard bigram grammars. The initial word error rate when testing on a 5K-word closed-vocabulary VP language model was 12.0%. Each of these methods is described below.

3.1. Silence Detection

Even though the training speech is read from prompts, there are often short pauses either due to natural sentential phrasing, reading disfluency, or running out of breath on long sentences. Naturally, the orthographic transcription that is provided with each utterance does not indicate these pauses. But it would be incorrect to model the speech as if there were no pauses. In particular, phonetic models that take into account acoustic coarticulation between words (cross-word models) do not function properly if they are confounded by unmarked pauses between words.

We developed a two-stage training process to deal with this problem. First we train HMM models assuming there are no pauses between words. Then we mark the missing silence locations automatically by running the recognizer on the training data constrained to the correct word sequence, but allowing optional silence between words. Then we retrain the model using the output of the recognizer as *corrected* transcriptions.

We find that this two-stage process increases the gain due to using cross-word phonetic models. The word error was reduced by 0.6% which is about a 5% reduction in word error.

3.2. Phonetic Dictionary

Two distinct phonetic dictionaries were supplied for training and testing purposes. We found the dictionaries for training and testing were not consistent. That is, there were many words that appeared in both dictionaries, but had different spellings. We also modified the spellings of several words that we judged to be wrong. However, after correcting all of these mistakes, including the inconsistency between the training and testing dictionary, the improvement was only 0.2%, which is statistically insignificant.

One inadequacy of the supplied dictionary was that it did not contain any schwa phonemes to represent reduced vowels. It did, on the other hand, distinguish three levels of stress. But we traditionally remove the stress distinction before using the dictionary. So we translated all of the lowest stress level of the UH and IH phonemes into AX and IX (We will use Random House symbols here). This resulted in another 0.2% reduction in word error.

Another consideration in designing a phonetic dictionary is the tradeoff between the number of parameters and the accuracy of the estimates. Finer phonetic distinctions in the dictionary can result in improved modeling, but they also increase the need for training data. Lori Lamel had previously reported [7] that the error rate on the RM corpus was reduced when the number of phonemes was reduced, ignoring some phonetic distinctions. In particular, she suggested replacing some diphthongs, affricates, and syllabic consonants with two-vowel sequences. She also suggested removing some phonetic distinctions. The list of substitutions is listed in Table 1 below.

Original	New
AY	AH-EE
OY	AWH-EE
OW	AH-OOH
CH	T-SH
IX	AX
UN	AX-N
UM	AX-M
UL	AX-L
AE	EY
OO	UH
ZH	Z
AH	AW

Table 1: These phonemes were removed by mapping them to other phonemes or sequences.

When we made these substitutions, we found that the word error rate decreased by 0.2% again. While this change is not significant, the size of the system was subtantially decreased due to the smaller number of triphone models.

Finally, we reinstated the last three phonemes in the list, since we were uncomfortable with removing too many distinctions. Again, the word error rate was reduced by another 0.2%.

While each of the above improvements was miniscule, the total improvement from changes to the phonetic dictionary was 0.8%, which is about a 7% reduction in word error. At the same time, we now only have a single phonetic dictionary to keep track of, and the system is substantially smaller.

3.3. Weight Optimization

After making several changes to the system, we reoptimized the relative weights for the acoustic and language models, as well as the word and phoneme insertion penalties. These weights were optimized on the development test set automatically using the N-best lists [4]. Optimization of these weights reduced the word error by 0.4%.

3.4. Cepstral Mean Removal

One of the areas of interest is recognition when the microphone for the test speech is unknown. We tried a few different methods to solve this problem, which will be described in a later section. However, during the course of trying different methods, we found that the simplest of all methods, which is to subtract the mean cepstrum from every frame's cepstrum vector actually resulted in a very small improvement in recognition accuracy even when the microphone was the same for training and test. This resulted in a 0.3% reduction in word error rate.

3.5. 3-Way Gender Selection

It has become a standard technique to model the speech of male and female speakers separately, since the speech of males and females is so different. This typically results in a 10% reduction in error relative to using a single speaker-independent model. However, we have found that there are occassional speakers who do not match one model much better than the other. In fact, there are some very rare sentences in which the model of the wrong gender is chosen. Therefore we experimented with using a third "gender" model that is the simple gender-independent model, derived by averaging the male and the female models. During recognition, we find the answer independently using each of these models and then we choose the answer that has the highest overall score. We find that about one out of 10 speakers will typically score better using the gender-independent model than the model for the correct gender. In addition, with this third model, we no longer ever see sentences that are misclassified as belonging to the wrong gender. The reduction error associated with using a third gender model was 0.4%.

3.6. Improvement Summary

The methods we used and the corresponding improvements are summarized in Table 2 below.

Improvement	Method
0.6%	silence-detection
0.8	improvements to phonetic dictionary
0.2	consistent dictionary
0.2	addition of schwa
0.2	reduced phoneme set
0.2	less reduced phoneme set
0.4	Automatic optimization of weights
0.3	Removing mean cepstrum, and
0.4	3-way gender selection
2.5%	Total improvement

Table 2: Absolute reduction in word error due to each improvement.

All the gains shown were additive, resulting in a total of 2.5% reduction in absolute word error, or about a 20% relative change.

4. COMPARATIVE EXPERIMENTS

In this section we describe several controlled experiments comparing the accuracy when using different training and recognition scenarios, and different algorithms.

4.1. Effect of Number of Training Speakers

It has always been assumed that for speaker independent recognition to work well, we must train the system on as many speakers as possible. We reported in [9] that when we trained a speaker-independent system on 600 sentences from each of 12 different speakers (a total of 7,200 sentences), the word error rate was only slightly higher than when the system was trained on a total of 3,990 sentences from 109 speakers. These experiments were performed on the 1000-word Resource Management (RM) Corpus. The results were difficult to interpret because the number of sentences were not exactly the same for both conditions, the data for the 109 speakers covered a larger variety of phonetic contexts than the data for the 12 speakers, and the 12 speakers were carefully selected to cover the various dialectic regions of the country (as well as is possible with only 7 male and 5 female speakers).

For the first time we were able to perform a well-controlled experiment to answer this question on the large vocabulary WSJ corpus. The amount of training data is the same in both cases. In one condition, there are 12 speakers (6 male and 6 female) with 600 sentences each. In the other case, there are 84 speakers with a total of 7,200 sentences. In both cases, all of the sentence scripts are unique. The speakers in both sets were selected randomly, without any effort to cover the general population. In both cases, we used separate models for male and female speakers.

In a second experiment, we repeated another experiment that had previously been run only on the RM corpus. Instead of pooling all of the training data (for one gender) and estimating a single model, we trained on the speech of each speaker separately, and then combined all of the resulting models simply by averaging the densities of the resulting models. We had previously found that this method worked well when each speaker had a substantial amount of training speech (enough to estimate a speaker-dependent model), and all of the speakers had the same sentences in their training. But in this experiment, we also computed separate speaker-dependent models for the speakers with 50-100 utterances, and each speaker had different sentences.

The results of these comparisons are shown in Table 3.

Training	Pooled	Averaged
SI-84	11.2	12.3
SI-12	11.6	12.0

Table 3: Word error rate for few (SI-12) vs many (SI-84) speakers, and for a single (Pooled) model vs separately trained (Averaged) models. The experiments were run on the 5K VP closed-vocabulary development test set of the WSJ pilot corpus using the standard bigram grammar.

We found, to our surprise, that there is almost no advantage for having more speakers if the total amount of speech is fixed. We also that the performance when we trained the system separately on each of the speakers and averaged the resulting models, was quite similar to that when we trained jointly on all of the speakers together. This result was particularly surprising for the SI-84 case, in which each speaker had very little training data.

More recently we ran this experiment again on the 5K NVP closed-vocabulary development test set with an improved system, and found that the results for a pooled model from 84 speakers were almost identical to those with an averaged model from 12 speakers (10.9% vs 11.3

Both of these results have important implications for practical speech corpus collection. There are many advantages for having a small number of speakers. We call this paradigm the *SI-few paradigm* as opposed to the *SI-many paradigm*. There are also practical advantages for being able to train the models for the different speakers separately.

1. It is much more efficient to collect the data; there are far fewer people to recruit and train.
2. In SI-few training, we get speaker-dependent models for the training speakers for free.
3. When new speakers are added to the training data, we just develop the models for the new speakers and average their models in with the model for all of the speakers, without having to retrain on all of the speech from scratch.
4. The computation for the average model method is easy to parallelize across several machines.
5. Perhaps the most compelling argument for SI-few training is that having speaker-specific models available for each of the training speakers allows us to experiment with speaker adaptation techniques that would not be possible otherwise.

Our conclusion is that there is little evidence that having a very large number of speakers is significantly better than a relatively small number of speakers – if the total amount of training is kept the same. Actually, if we equalize the cost of collecting data under the SI-few and SI-many conditions, then the SI-few paradigm would likely yield better recognition performance than the SI-many paradigm.

4.2. Speaker-Dependent vs Speaker-Independent

It is well-known that, for the same amount of training speech, a system trained on many speakers and tested on new speakers (i.e. speaker-independent recognition) results in significantly worse performance than when the system is trained on the speaker who will use it. However, it is important to know what the tradeoff is between the amount of speech and whether the system is speaker-independent or not, since for many applications, it would be practical to collect a substantial amount of speech from each user.

Below we compare the recognition error rate between SI and SD recognition. The SI models were trained with 7,200 sentences, while the SD were trained with only 600 sentences, each. Two different sets of test speakers were used for the SI model, while for the SD case, the test and training speakers were the same, but we compare two different test sets from these same speakers. These experiments were performed using the 5K-word NVP test sets, using the standard bigram language models and also rescoring using a trigram language model.

Training	SI-12	SD-1
Test	(7200)	(600)
Dev. Test	10.9	7.9
Nov. 92 Eval	8.7	8.2

Table 4: Speaker-dependent vs Speaker-independent training

As can be seen, the word error rate for the SI model is only somewhat higher than for the SD model, depending on which SI test set is used. We estimate that, on the average, if the amount of training speech for the SI model were 15-20 times that used for the SD model, then the average word error rate would be about the same.

One might mistakenly conclude from the above results that if there is a large amount of speaker-independent training available, there is no longer any reason to consider speaker-dependent recognition. However, it is extremely important to remember that these results only hold for the case where all of the speakers are native speakers of English. We have previously shown [10] that when the test speakers are not native speakers, the error rate goes up by an astonishing factor of eight! In this case, we must clearly use either a speaker-dependent or speaker-adaptive model in order to obtain usable performance. Of course each speaker can use the type of model that is best for him.

4.3. N-Best Paradigm

In 1989 we developed the N-best Paradigm method for combining knowledge sources mainly as a way to integrate speech recognition with natural language processing. Since then, we have found it to be useful for applying other expensive speech knowledge sources as well, such as cross-word models, tied-mixture densities, and trigram language models. The basic idea is that we first find the top N sentence hypotheses using a less expensive model, such as a bigram grammar with discrete densities, and within-word context models. And then we rescore each of the resulting hypotheses with the more complex models, and finally we pick the highest scoring sentence as the answer.

One might expect that there would be a severe problem with this approach if the latter knowledge sources were much more powerful than those used in the initial N-best pass. However, we have found that this is not the case, as long as the initial error rate is not too high and the sentences are not too long.

In tests on the ATIS corpus (class A+D sentences only), we obtained a 40% reduction in word error rate by rescoring the N-best sentence hypotheses with a trigram language model. In this test, we used a value of 100 for N. This shows that the trigram language model is much more powerful than the bigram language model used in finding the N-best sentences. But there were many utterances for which the correct answer was not found within the N-best hypotheses. It was important to determine whether the system was being hampered by restricting its consideration to the N-best sentences before using the trigram language model. Therefore, we artificially added the correct sentence to the N-best list before rescoring with the trigram model. We found that the word error only decreased by another 7%. We must remember that in this experiment, the performance with the correct sentence added was an optimistic estimate, since we did not add all of the other sentence hypotheses that scored worse than the 100th hypothesis, but better than the correct answer.

The question is whether this result would hold up when the vocabulary is much larger, thereby increasing the word error rate, and the sentences are much longer, thereby increasing the number of possible permutations of word sequences exponentially. In experiments with the 5K-word WSJ sentences with word error rates around 14% during the initial pass, and with average sentence lengths around 18 words we still found little loss.

However, on the 20K-word development test set, we observed a significant loss for trigram rescoring, but not for other less powerful knowledge sources. The experiment was limited to those sentences that contained only words that were inside the recognition vocabulary. (It is impossible to correct errors due to words that are outside of the recognition vocabulary.) This included about 80% of the development test set. The results are shown below in Table 5 for the actual N-best list and with the correct utterance artificially inserted into the list.

Knowledge Used	Actual N-best	With Correct Answer Added
Initial N-best	19.5	19.5
Cross-word rescoring	16.1	15.6
Trigram rescoring	13.9	10.2

Table 5: Effect of N-best Paradigm on 20K-word recognition with trigram Language model rescoring

While this result is a lower bound on the error rate, it indicates that much of the potential gain for using the trigram language model is being lost due to the correct answer not being included in the N-best list. As a result we are modifying the N-best rescoring to alleviate this problem.

5. MICROPHONE INDEPENDENCE

DARPA has placed a high priority on microphone independence. That is, if a new user plugs in any microphone (e.g., a lapel microphone or a telephone) without informing the system of the change, the recognition system is expected to work as well as it does with the microphone that was used for training.

We considered two different types of methods to alleviate this problem. The first attempts to use features that are independent of the microphone, while the second attempts to adapt the system or the input to observed differences in the incoming signal in order to make the speech models match better.

5.1. Cepstrum Preprocessing

The RASTA algorithm [11] smoothes the cepstral vector with a five-frame averaging window, and also removes the effect of a slowly varying multiplicative filter, by subtracting an estimate of the average cepstrum. This average is estimated with an exponential filter with a constant of 0.97, which results in a time constant of about one third of a second. The blind deconvolution algorithm estimates the simple mean of each cepstral value over the utterance, and then subtracts this mean from the value in each frame. In both cases, speech frames are not distinguished from noise frames. The processing is applied to all frames equally. In addition, there was no dependence on estimates of SNR.

Every test utterance was recorded simultaneously on the same microphone used in the training (a high-quality noise-cancelling Sennheiser microphone) and on some other microphone which was not known, but which ranged from an omni-directional boom-mounted microphone or table-mounted microphone, a lapel microphone, or a speaker-phone. We present the error rates for the baseline and for the two preprocessing methods in Table 6 below.

Preprocessing	Sennheiser	Alternate-Mic
Mel cepstra vectors	12.0	37.7
RASTA preprocessing	12.5	27.8
Cepstral Mean Removal	11.8	27.2

Table 6: Comparison of simple preprocessing algorithms. The results were obtained on the 5K-word VP development test set, using the bigram language model.

The results show that the word error rates increase by a factor of three when the microphone is changed radically. The RASTA algorithm reduced the degradation to a factor of 2.3, while degrading the performance on the Sennheiser microphone just slightly. The blind deconvolution also reduced the degradation, but did not degrade the performance on the training microphone. (In fact, it seemed to improve it very slightly, but not significantly.) This shows that the five-frame averaging used in the RASTA algorithm is not necessary for this problem, and that the short-term exponential averaging used to estimate the long-term cepstrum might vary too quickly.

5.2. Known Microphone Adaptation

We decided to attack the problem of accomodating an unknown microphone by considering another problem that seemed simpler and more generally useful. It would be very useful to be able to adapt a system trained on one microphone so that it works well on another particular microphone. The microphone would not have been known at the time the HMM training data was collected, but it *is* known before it is to be used. In this case, we can collect a small sample of stereo data with the microphone used for training and the new microphone simultaneously. Then using the stereo data we can adapt the system to work well on the new microphone.

For microphone adaptation, we assume we have the VQ index of the cepstrum of the Sennheiser signal, and the cepstrum of the alternate microphone. Given this stereo data, we accumulate the mean and variance of the cepstra of the alternate microphone of the frames whose Sennheiser data falls into each of the bins of the VQ codebook. Now, we can use this to define a new set of Gaussians for data that comes from the new microphone. The new Gaussians have means that are shifted relative to the original means, where the shift can be different for each bin. In addition, the variances are typically wider for the new microphone, due to some nondeterministic differences between the microphones. Thus the distributions typically overlap more, but only to the degree that they should. The new set of means and variances represents a codebook transformation that accomodates the new microphone.

5.3. Microphone Selection

In the problem we were trying to solve the test microphone is not known, and is not even included in any data that we might have seen before. In this case, how can we estimate a codebook transformation like the one described above? One technique is to estimate a transformation for many different types of microphones and then use one of those transformations.

We had available stereo training data from several microphones that were not used in the test. We grouped the alternate microphones in the training into six broad categories, such as lapel, telephone, omni-directional, directional microphones, and two specific desk-mounted microphones. Then, we estimated a transformed codebook for each of the microphones using stereo data from that microphone and the Sennheiser, being sure that the adaptation data included both male and female speakers.

To select which microphone transformation to use, we tried simply using each of the transformed codebooks in turn, recognizing the utterance with each, and then choosing the answer with the highest score. Unfortunately, we found that this method did not work well, because data that really came from the Sennheiser microphone was often misclassified as belonging to another microphone. We believe this was due to the radically different nature of the Gaussians for the Sennheiser and the alternate microphones. The alternate microphone Gaussians overlapped much more.

Instead we developed a much simpler, less costly method to select among the microphones. For each of the seven microphone types (Sennheiser plus six alternate types) we estimated a mixture density consisting of eight Gaussians. Then, given a sentence from an unknown microphone, we computed the probability of the data being produced by each of the seven mixture densities. The one with the highest likelihood was chosen, and we then used the transformed codebook corresponding to the chosen microphone type. We found that on development data this microphone selection algorithm was correct about 98% of the time, and had the desirable property that it *never* misclassified the Sennheiser data.

After developing this algorithm, we found that a similar algorithm had been developed at CMU [12]. There were four differences between the MFCDCN method and our method. First, we grouped the several different microphones into six microphone types rather than modeling them each separately. Second, we modified the covariances as well as the means of each Gaussian, in order to reflect the increased uncertainty in the codebook transformation. Third, we used an independent microphone classifier, rather than depend on the transformed codebook itself to perform microphone selection. And fourth, the CMU algorithm used an SNR-dependent transformation, whereas we used only a single transformation. The first difference is probably not important. We believe that the second and third differences favor our algorithm, and the fourth difference clearly favors the MFCDCN algorithm. Further experimentation will be needed to determine the best combination of algorithm features.

We then compared the performance of the baseline system with blind deconvolution and the microphone adaptation algorithm described above. Since these experiments were performed after improvements described in Section 1, and the test sets and language models were different the results in Table 7 are not directly comparable to those in Table 6 above.

Preprocessing	Sennheiser	Alternate-Mic
Mel cepstra vectors	11.6	
Cepstral Mean Removal	11.3	32.4
Microphone Adaptation	11.3	21.3

Table 7: Microphone Adaptation vs Mean Removal. These experiments were performed on the 5K-word NVP development test set using a bigram language model.

6. SUMMARY

We have reported on several methods that result in some reduction in word error rate on the 5K-word WSJ test. In addition, we have described several experiments that answer questions related to training scenarios, recognition search strategies, and microphone independence. In particular, we verified that there is no reason to collect speech from a large number of speakers for estimating a speaker-independent model. Rather, the same results can be obtained with less effort by collecting the same amount of speech from a smaller number of speakers. We determined that the N-best rescoring paradigm can degrade somewhat when the error rate is very high and the sentences are very long. We showed that a simple blind deconvolution preprocessing of the cepstral features results in a better microphone independence method than the more complicated RASTA method. And finally, we introduced a new microphone adaptation algorithm that achieves improved accuracy by adapting to one of several codebook transformations derived from several known microphones.

Acknowledgement

This work was supported by the Defense Advanced Research Projects Agency and monitored by the Office of Naval Research under Contract Nos. N00014-91-C-0115, and N00014-92-C-0035.

REFERENCES

[1] Pallett, D., Fiscus, J., Fisher, W., and J. Garofolo, "Benchmark Tests for the Spoken Language Program", *DARPA Human Language Technology Workshop*, Princeton, NJ, March, 1993.

[2] Chow, Y., M. Dunham, O Kimball, M. Krasner, G.F. Kubala, J. Makhoul, P. Price, S. Roucos, and R. Schwartz (1987) "BYBLOS: The BBN Continuous Speech Recognition System," *IEEE ICASSP-87*, pp. 89-92

[3] Chow, Y-L. and R.M. Schwartz, "The N-Best Algorithm: An Efficient Procedure for Finding Top N Sentence Hypotheses", ICASSP90, Albuquerque, NM S2.12, pp. 81-84.

[4] Schwartz, R., S. Austin, Kubala, F., and J. Makhoul, "New Uses for the N-Best Sentence Hypotheses Within the BYBLOS Speech Recognition System", ICASSP92, San Francisco, CA, pp. I.1-I.4.

[5] Schwartz, R. and S. Austin, "A Comparison Of Several Approximate Algorithms for Finding Multiple (N-Best) Sentence Hypotheses", ICASSP91, Toronto, Canada, pp. 701-704.

[6] Austin, S., Schwartz, R., and P. Placeway, "The Forward-Backward Search Algorithm", ICASSP91, Toronto, Canada, pp. 697-700.

[7] Lamel, L., Gauvain, J., "Continuous Speech Recognition at LIMSI", *DARPA Neural Net Speech Recognition Workshop*, September, 1992.

[8] Price, P., Fisher, W.M., Bernstein, J., and D.S. Pallett (1988) "The DARPA 1000-Word Resource Management Database for Continuous Speech Recognition," *IEEE Int. Conf. Acoust., Speech, Signal Processing*, New York, NY, April 1988, pp. 651-654.

[9] Kubala, F., R. Schwartz, C. Barry, "Speaker Adaptation from a Speaker-Independent Training Corpus", *IEEE ICASSP-90*, Apr. 1990, paper S3.3.

[10] Kubala, F., R. Schwartz, Makhoul, J., "Dialect Normalization through Speaker Adaptation", *IEEE Workshop on Speech Recognition* Arden House, Harriman, NY, Dec. 1991.

[11] Hermansky, H., Morgan, N., Bayya, A., Kohn, P., "Compensation for the Effect of the Communication Channel in Auditory-Like Analysis of Speech (RASTA-PLP), *Proc. of the Second European Conf. on Speech Comm. and Tech.* September, 1991.

[12] Liu, F-H., Stern, R., Huang, X., Acero, A., "Efficient Cepstral Normalization for Robust Speech Recognition", *DARPA Human Language Technology Workshop*, Princeton, NJ, March, 1993.

[13] Placeway, P., Schwartz, R., Fung, P., and L. Nguyen, "The Estimation of Powerful Language Models from Small and Large Corpora", To be presented at ICASSP93, Minneapolis, MN.