EVALUATING THE AURORA CONNECTED DIGIT RECOGNITION TASK – A BELL LABS APPROACH

M. Afify, H. Jiang, F. Korkmazskiy, C.-H. Lee, P. Li, O. Siohan, F. K. Soong, A. Surendran

Dialogue Systems Research Department
Bell Laboratories, Lucent Technologies
Murray Hill, NJ 07974, USA
{afify,hui,yelena,chl,qli,siohan,fks,acs}@research.bell-labs.com

ABSTRACT

Connected digit recognition has always been an ideal task for fundamental research in speech recognition due to its relatively low complexity and potential applications. In Bell Labs we have developed a number of techniques targeting directly or indirectly at connected digit recognition. For the Aurora task, we study a few such algorithms for the entire spectrum of the issues, including feature extraction, context-dependent digit modeling, minimum classification error acoustic modeling, unsupervised noise compensation, and utterance verification. We show how each component contributes to the reduction of digit recognition and verification errors in adverse conditions. Average over all three test sets we obtained 84.6% and 91.3% digit accuracies for clean- and multi-condition training, respectively. This represents an average of 48.6% error rate reduction when compared to the official Aurora baseline results.

1. INTRODUCTION

Connected digit recognition is one of the most important speech recognition tasks for any language due to its low complexity and implied applications. In the past twenty years, numerous advances have been developed that we have witnessed many digit-related services being deployed over traditional telephone networks. Due to the increasing demand for mobile information access, speech recognition in noisy environment is now attracting a new level of interest both in research and for practical applications. The Aurora connected digit recognition database was created to establish a framework for facilitating common evaluation of speech recognition systems in noisy conditions [4].

In Bell Labs, we have developed a number of useful techniques in the areas of feature extraction, model topology and context dependency, acoustic model training, noise robustness and robust decision strategies. In this study, we evaluate some of these algorithms on the Aurora task. Detail of the techniques can be found in the referred publications. Here we only briefly describe how each algorithm contributes to reducing digit recognition and verification errors in noisy conditions.

The rest of the paper is organized as follows. In Section 2, we discuss the important issue of modeling context dependency in connected digit recognition. Both decision tree tied state modeling and the classical head-body-tail modeling are used to replace the whole digit model in the baseline system of the Aurora task. In Section 3, we evaluate Mel frequency cepstral coefficient (MFCC) based systems. It was found that noise compensation is a key factor to improving performance in clean-condition training. In Section 4, we evaluated a newly-developed auditory feature set which is shown to be robust to noise distortion. Noise compensation (NC) is needed again to improve clean-condition training. We also found that minimum classification error (MCE) training effectively reduces digit errors especially in multi-condition training. Utterance verification to reject unreliable recognized digits is evaluated in Section 5. Finally we present a complete set of results in Section 6.

2. MODELING CONTEXT-DEPENDENCY

In the past few years, the use of decision tree state tying (e.g. [9]) has become a standard practice to build context dependent acoustic models like triphone models using conventional maximum likelihood (ML) training.

The decision tree state tying algorithm however has rarely been used to build context dependent (CD) digits models, mainly since the total number of contexts is relatively small. Instead context independent whole digit and cross-digit coarticulation models, e.g. the head-body-tail (HBT) model [3] structure, have been used. The HBT model assumes that context dependent digit models can be built by concatenating a left-context dependent unit (head) with a context independent unit (body) followed by a right-context dependent unit (tail). In other words, each digit consists of 1 body, 12 heads and 12 tails (representing all left/right contexts), for a total of 276 units (11(digits) $\times$ (1(body) + 12(head) + 12(tail)) + 1(silence))[3]. We typically use a 3-state HMM to represent each head and tail unit, and a 4-state HMM for each body. Overall it corresponds to a 10-state digits model for a total number of 837 states (including a 1-state silence model).

Since the amount of training data is limited in Aurora, controlling the size of the context dependent digit models is required. The decision tree state tying algorithm becomes a good candidate to build context dependent digit models. When building triphone models, the set of questions used by the decision tree includes questions about the identity of the left and right phone as well as questions about their broad phonetic classes membership. On the other hand the set of questions used to build context dependent digit models is much simpler since it only includes questions about the identity of the left and right context digit. As a direct consequence, when building a 10-state CD digit models, we only use questions related to the left (right) context when

growing the decision tree for the first (last) 3 states of the model, while the question set includes questions about the left and right context for the 4 central states. In addition, when growing and pruning the decision tree, different thresholds are applied to the central states compared to the initial and final states. Hence, we force the central states to be almost context independent by using a high threshold, while a much lower threshold is used at the model extremities to cover most context.

It then appears that the HBT model structure is a special case of tying that can be obtained by setting the decision tree threshold to 0 for the initial and final states of the model (therefore allowing all left/right context) while setting the threshold to infinity for the central states (leading to context independent states). One advantage of the proposed technique is that by controlling one or two parameters (the thresholds) it is possible to generate models lying anywhere between context independent models to fully context dependent models. Such a training paradigm is therefore very flexible in allowing context dependency modeling under tight memory requirements and the limited training data constraint, while leading to improved accuracy over context independent models.

3. MFCC FEATURE EVALUATION

In the first set of experiments, we used a 39-dimension MFCC feature vector including 12 cepstral coefficients and energy, plus their first and second order time derivatives. Context-dependent digit models with ML training described in Section 2 are used for all results reported here. It is noted that the recognizer used here was developed for large vocabulary continuous speech recognition.

3.1. Baseline Results - MFCC Features

When building the CD digit models, we have adjusted the threshold so that almost all contexts are allowed for the initial and final states, while only a small number of contexts is used for the central states. Overall, the total number of tied states is about 900, comparable with the HBT model complexity. We have built CD models for eleven 10-state digits and a 3-state silence unit, plus one single-state context independent silence model. On average, the total number of Gaussian mixtures is about 4600, or about 5 Gaussians per state. We should point out that experimental evaluations have shown that the total number of Gaussians can be halved by increasing the thresholds used by the decision tree state tying and reducing the number of Gaussians per state, with little degradation in performance. The recognition results are reported in Table 1 in terms of word accuracy, as obtained from the NIST scoring tool. Compared to the official baseline, our system reduces the error rate by about 43% and 33% for clean- and multi-condition training, respectively. It is noted that we have not tuned our recognizer for the Aurora task. We have observed a huge amount of deletion errors in low SNR cases, such as at 0dB SNR. Therefore the reported results throughout are heavily biased towards better results for real-time performance at relatively high SNR levels. By modifying the word insertion penalty, it is possible to get a significant improvement in word accuracy at low SNR levels and we believe it will lead to further enhancement of the overall performance.

Training Mode	Set A	Set B	Set C	Overall
Multicondition	90.84	91.12	91.06	90.99
Clean Only	75.80	79.64	77.00	77.57
Average	83.32	85.38	84.03	84.28

Table 1: Average word accuracy (%) - MFCC CD models

3.2. Noise Compensation – MFCC Features

A recent review on adaptation and compensation techniques for speech recognition, can be found in [7]. In this study the compensation algorithm is designed to compensate, in the log spectral domain, for noise added in the linear spectral domain. Working in the log spectral domain is attractive for speech recognition systems using MFCC, where there is a direct linear relationship between the two domains. However, operating on additive noise in the log spectral domain leads to a non-linear mismatch function which is difficult to deal with directly. The use of a vector Taylor series (VTS) expansion facilitates converting this non-linear function into a first or higher order polynomial. This results in mathematically tractable solutions for:

- Estimating the noise mean of the current utterance;
- Estimating the clean speech features from the noisy features in an minimum mean squared error sense.

Moreover, using a sequential estimation technique with an optimal forgetting factor [2] facilitates tracking of time varying noise and is attractive for real time applications

Here we applied the noise compensation algorithm on the test data, in batch mode. The clean training data is used to estimate the Gaussian mixture model used for noise compensation. We only report recognition results using clean-condition training since applying noise compensation under multi-condition training is not likely to give additional enhancement. Results are given in Table 2. It indicates that the proposed technique reduces the error rate by about 31% over that without imposing noise compensation.

Training Mode	Set A	Set B	Set C	Overall
Clean Only	84.50	86.36	81.30	84.60

Table 2: Word accuracy (%) after noise compensation

4. AUDITORY FEATURE EVALUATION

In the following, a recently developed auditory feature [8] was applied to the Aurora task with a head-body-tail (HBT) model structure [3]. Furthermore MCE training using a generalized probabilistic descent (GPD) algorithm [3] is used. Noise compensation technique is also incorporated to further improve the performance.

4.1. Baseline Results - Auditory Features

The auditory feature extraction procedure is comprised of the following steps: an outer-middle-ear transfer function, FFT, frequency conversion from linear to the Bark scale, auditory filtering, nonlinearity, and discrete cosine transform (DCT). Here the auditory feature is a 39-dimension vector including 12 cepstral coefficients, energy ($c(0)$ of DCT) [8], plus their first and second order time derivatives.

In the baseline system, ML training was applied, The performances under the two training conditions are listed in Table 3. Compared to the official Aurora baseline, this step reduces the average error rates by 47.76% and 25.84% for clean- and multi-condition training, respectively.

Training Mode	Set A	Set B	Set C	Overall
Multicondition	90.11	89.16	91.00	89.91
Clean Only	77.89	79.74	80.43	79.14
Average	84.00	84.45	85.72	84.53

Table 3: Average word accuracy (%) using auditory features and HBT models with ML training

4.2. GPD and NC - Auditory Features

For multi-condition training, we then applied the MCE-GPD algorithm to discriminatively train the model parameters [3]. This step reduced the average error rate by 35.76% over the official Aurora results. For clean-condition training, noise compensation is needed. It is interesting to note that by interpreting the auditory features as a modified MFCC representation we were able to apply the same principle discussed in Section 3.2 to these features. Compared to the official Aurora baseline, this step reduced the average error rates by 59.28% with the ML-trained HBT models. The mixed set of results are shown in Table 4.

Training Mode	Set A	Set B	Set C	Overall
Multicondition, GPD	91.56	90.69	91.80	91.26
Clean, Noise Comp.	83.57	84.75	82.06	83.74
Average	87.57	87.72	86.93	87.50

Table 4: Average word accuracy (%) using auditory features and HBT models: GPD for multi-condition and noise compensation for clean-condition training

5. DIGIT VERIFICATION WITH BOOSTING

Utterance verification is a technique to assign a confidence measure for acceptance or rejection to a detected word hypothesis. In the past, log likelihood ratio between likelihoods of the recognized digit and its corresponding anti-model is used to perform digit verification (e.g. [10]). Here we apply boosting to digit verification. Boosting is a learning algorithm which combines a set of "weak" classifiers into a "strong" one with a weighted majority vote.

In the Aurora evaluation, we perform boosting based verification on each digit of the recognized string [6]. We have no time to apply more sophisticated techniques (e.g. [10]). Instead we used a simple baseline system to illustrate the importance of digit verification and the utility of boosting based techniques. Therefore the verification results reported here are only for relative comparison. We give results on a subset of Test Set A in multi-condition training using MFCC features and CD digit models. Three sets of receiver operating characteristic (ROC) curves are plotted at three different SNR's, 20dB, 10 dB and 0dB, respectively, in Figure 1. False rejection, $E_r = N_{cr}/N_c$, and false acceptance, $E_a = N_{ia}/N_i$, errors are defined with N_c and N_i, being the numbers of correctly and incorrectly recognized digits, respectively; and N_{cr} and N_{ia}, being the numbers of words falsely rejected and falsely accepted in verification, respectively. Since Test Set A is highly skewed, i.e. much more correctly than incorrectly recognized digits, it implies a need to operate at different point on the ROC curves other than that at equal error rate for the Aurora test. Plots of digit errors against false rejection errors before and after boosting at the 10dB SNR level are given in Figure 2. It is clear that boosting improves verification accuracy and reduces recognition errors if rejection is allowed. From the results in Figures 1 and 2 it is also clear that digit verification in adverse conditions appears to be a very difficult problem. More research is needed.

6. SUMMARY AND DETAILED RESULTS

A number of techniques have been applied to the Aurora connected digit recognition task. Some of algorithms give significant improvement for clean-condition training while the others reduce digit errors in multi-condition training. It is clear that understanding the problem and the available techniques is a key to a successful combination of algorithms. Within limited time, we were able to apply some methods developed at Bell Labs in the last few years to evaluate the Aurora task. A detailed summary of the full set of results is reported in the following.

For multi-condition training, we use auditory features, head-body-tail context-dependent digit modeling, noise compensation and minimum classification error discriminative training. The average digit accuracies for Test Sets A, B and C are given in Tables 5, 6 and 7, respectively. For clean-condition training, we use MFCC features, decision tree state tying context-dependent digit modeling, and noise compensation. The average digit accuracies for Test Sets A, B and C are given in Tables 8, 9 and 10, respectively. Additional compensation (e.g. [1, 5]) improves recognition only slightly. It seem the proposed noise compensation algorithm capture most of the mismatches. On the average, we obtained a digit accuracy of 87.93% for both clean- and multi-condition training. It represents a 48.60% digit error reduction compared to the official baseline of the Aurora task. The overall results are summarized in Table 11.

REFERENCES

[1] M. Afify, and O. Siohan, "Constrained Maximum Likelihood Linear Regression for Speaker Adaptation," *Proc. ICSLP-2000*, Beijing, 2000.

[2] M. Afify and O. Siohan, "Sequential Noise Estimation with Optimal Forgetting for Robust Speech Recognition," *Proc. ICASSP-2001*, Salt Lake City, May 2001.

[3] W. Chou, C.-H. Lee and B.-H. Juang, "Minimum Error Rate Training of Inter-word Context Dependent Acoustic Model Units in Speech Recognition," *Proc. ICSLP-94,* pp.439-442, Yokohama, Sept. 1994.

[4] H. G. Hirsch and D. Pearce, "The Aurora Experimental Framework for the Performance Evaluation of Speech Recognition Systems under Noisy Conditions," *Proc. ISCA ITRW ASR2000*, pp. 181-188, Paris, May 2000.

[5] H. Jiang, F. K. Soong and C.-H. Lee, "Hierarchical Stochastic Feature Matching for Robust Speech Recognition," *Proc. ICASSP-2001*, Salt Lake City, 2001.

[6] F. Korkmazskiy, F. Soong, O. Siohan, "Boosting Techniques for Utterance Verification," *Bell Labs Technical Memorandum*, 2001.

[7] C.-H. Lee and Q. Huo, "On Adaptive Decision Rules and Decision Parameter Adaptation for Automatic Speech Recognition," *Proc. IEEE*, Vol. 88, pp. 1241-1269, 2000.

[8] Q. Li, F. K. Soong, and O. Siohan, "A High-Performance Auditory Feature for Robust Speech Recognition," *Proc. ICSLP-2000*, pp. 51-54, Beijing, 2000.

[9] W. Reichl and W. Chou. "Robust Decision Tree State Tying for Continuous Speech Recognition," *IEEE Trans. on Speech and Audio Proc.*, Vol. 8, No. 5, 2000.

[10] R. A. Sukkar and C.-H. Lee, "Vocabulary Independent Discriminative Utterance Verification for Non-Keyword Rejection in Subword Based Speech Recognition," *IEEE Trans. on Speech and Audio Proc.*, Vol. 4, No. 6, pp. 420-429, 1996.

SNR/dB	Subway	Babble	Car	Exhib.	Average
Clean	99.3	99.4	99.4	99.4	99.38
20	99.0	99.1	99.1	98.9	99.03
15	98.2	98.6	98.7	97.8	98.33
10	96.3	97.0	97.2	95.6	96.53
5	92.2	90.7	91.5	89.5	90.98
0	76.7	70.2	72.1	72.7	72.93
-5	44.6	36.9	31.7	39.2	38.10
Average	92.48	91.12	91.72	90.90	91.56

Table 5: Average word accuracy (%) for Test Set A with auditory features in multi-condition MCE-GPD training

SNR/dB	Resta.	Street	Airport	Station	Average
Clean	99.3	99.4	99.4	99.4	99.38
20	99.3	98.8	99.2	99.1	99.10
15	98.5	98.2	98.6	98.3	98.40
10	95.9	96.2	96.7	95.9	96.18
5	88.0	90.4	90.7	88.8	89.48
0	65.7	73.2	74.4	67.8	70.28
-5	33.3	38.0	40.7	31.7	35.93
Average	89.48	91.36	91.92	89.98	90.69

Table 6: Average word accuracy (%) for Test Set B with auditory features in multi-condition MCE-GPD training

SNR/dB	Subway	Street	Average
Clean	99.3	99.4	99.35
20	99.0	98.6	98.80
15	97.6	98.1	97.85
10	96.2	96.0	96.10
5	92.0	90.6	91.30
0	76.7	73.2	74.95
-5	44.5	38.2	41.35
Average	92.30	91.30	91.80

Table 7: Average word accuracy (%) for Test Set C with auditory features in multi-condition MCE-GPD training

SNR/dB	Subway	Babble	Car	Exhib.	Average
Clean	99.8	99.7	99.7	99.7	99.73
20	97.9	99.1	99.2	99.0	98.80
15	95.7	98.0	98.2	96.6	97.13
10	87.7	94.7	95.2	91.8	92.35
5	72.8	82.0	85.6	80.0	80.10
0	46.7	52.9	59.3	57.5	54.10
-5	23.6	22.6	23.7	28.2	24.53
Average	80.16	85.34	87.50	84.98	84.50

Table 8: Word accuracy (%) for Test Set A in clean-condition training with MFCC and CD digit models

SNR/dB	Resta.	Street	Airport	Station	Average
Clean	99.8	99.7	99.7	99.7	99.73
20	99.3	98.2	99.3	99.1	98.98
15	97.6	97.2	98.4	98.3	97.88
10	93.8	92.2	96.1	94.8	94.23
5	79.7	78.8	86.4	84.7	82.40
0	55.9	53.2	64.3	59.9	58.33
-5	26.1	24.1	30.8	26.2	26.80
Average	85.26	83.92	88.90	87.36	86.36

Table 9: Word accuracy (%) for Test Set B in clean-condition training with MFCC and CD digit models

SNR/dB	Subway	Street	Average
Clean	99.8	99.6	99.70
20	97.6	98.2	97.90
15	95.0	97.0	96.00
10	88.5	91.5	90.00
5	72.0	77.3	74.65
0	46.1	49.8	47.95
-5	21.9	22.6	22.25
Average	79.84	82.76	81.30

Table 10: Word accuracy (%) for Test Set C in clean-condition training with MFCC and CD digit models

Absolute Performance (%)				
Training Mode	Set A	Set B	Set C	Overall
Multicondition	91.56	90.69	91.80	91.26
Clean Only	84.50	86.36	81.30	84.60
Average	88.03	88.53	86.55	87.93
Performance Relative to Official Baseline (%)				
Training Mode	Set A	Set B	Set C	Overall
Multicondition	30.69	32.15	49.45	35.76
Clean Only	59.89	69.18	44.77	61.44
Average	45.29	50.67	47.11	48.60

Table 11: Summary – Average Word Accuracy (%)

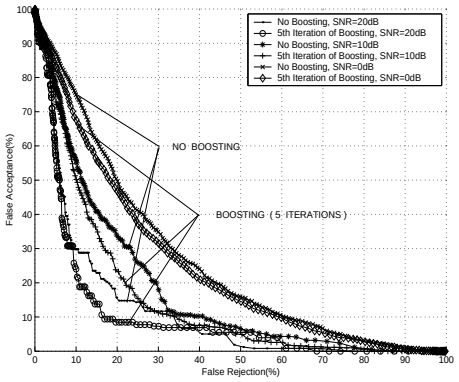

Figure 1: Digit Verification Performance for Test Set A with MFCC Features and Multi-condition Training.

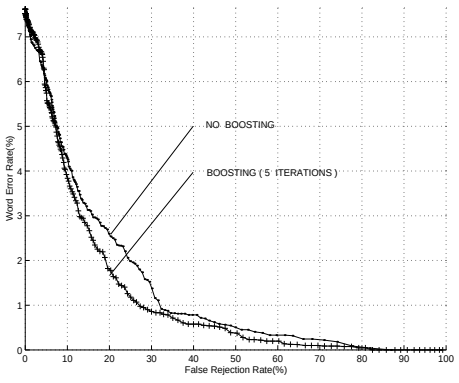

Figure 2: Digit Error after Rejection for Test Set A with MFCC Features and Multi-condition Training.