EVALUATION AND INTEGRATION OF NEURAL-NETWORK TRAINING TECHNIQUES FOR CONTINUOUS DIGIT RECOGNITION

John-Paul Hosom, Ronald A. Cole*, and Piero Cosi***

* Center for Spoken Language Understanding (CSLU)
Oregon Graduate Institute of Science and Technology (OGI)
P.O. Box 91000, Portland Oregon 97291-1000 USA
e-mail: {hosom,cole}@cse.ogi.edu www: http://www.cse.ogi.edu/CSLU

** Institute of Phonetics -- C. N. R.
Via G. Anghinoni, 10 - 35121 Padova ITALY
e-mail: cosi@csrf.pd.cnr.it www: http://www.csrf.pd.cnr.it

ABSTRACT

This paper describes a set of experiments on neural-network training and search techniques that, when combined, have resulted in a 54% reduction in error on the continuous digits recognition task. The best system had word-level accuracy of 97.52% on a test set of the OGI 30K Numbers corpus, which contains naturally-produced continuous digit strings recorded over telephone channels. Experiments investigated effects of the feature set, the amount of data used for training, the type of context-dependent categories to be recognized, the values for duration limits, and the type of grammar. The experiments indicate that the grammar and duration limits had a greater effect on recognition accuracy than the output categories, cepstral features, or a 50% increase in the amount of training data.

1. INTRODUCTION

The recognizers in the CSLU Toolkit use a hybrid HMM/ANN framework [1]. In these systems, frame-based recognition is done with context-dependent sub-phonetic states, where the state probability estimation is computed using a neural network.

We have developed a set of procedures within the Toolkit for training special-purpose recognizers for tasks such as continuous digit recognition. This method is simple enough that a bright high-school student can complete the tutorial in a few days. On the continuous digits task, the training procedure yields recognition results that compare favorably to standard HMM systems [1]. This paper shows how competitive performance was achieved by optimizing several of the parameters used in training and incorporating new training techniques.

2. CORPUS

The OGI 30K Numbers corpus [2] was used for training, development, and testing. The data in this corpus were collected from thousands of people within the United States who recited their telephone number, street address, zip code, or other numeric information over the telephone in a natural speaking style. Because the data were collected from a large number of speakers from different backgrounds in different environments, the corpus contains a noticeable amount of breath noise, glottalization, background noise (including music), and other "real-life" complications. Of almost 15,000 utterances, approximately 6600 utterances have been transcribed and time-aligned at the phonetic level by professional labelers. For the experiments reported here, we used only those utterances that consist entirely of digits (zero through nine and "oh"). Before separating the data into training, development, and test sets, about 5% of the corpus was culled for independent testing and set aside. Three speaker-independent partitions were created from the remaining data: 3/5 for training (6087 files, of which 2547 were hand-labeled), 1/5 for development (2110 files), and 1/5 for testing (2169 files). The development partition was further split into five sets, and the development results reported in this paper are for the first of these five sets (423 files).

3. BASELINE SYSTEM

The baseline system was trained using approximately the same method and parameters as the digits recognizer in the March 1998 release of the Toolkit. For training the baseline system, hand-labeled phonetic symbols are mapped, if necessary, to a consistent set of symbols for each word, /oU 9r/ (in "four") is merged into one />r/ phone, and /kh s/ (in "six") is merged into one /ks/ phone. (Phonetic symbols are written in Worldbet).

The system is trained to recognize context-dependent units. For left and right contexts, pauses and stop closures are mapped to the symbol /uc/ (unvoiced closure), and dentals (/th/, /s/, and the right half of /ks/) are mapped to the broad-category symbol /den/; otherwise the contexts are phoneme-specific. Each phoneme can be split into one, two, or three parts. The left part is dependent on the context of the preceding phoneme (or phonetic broad category), the center part (if any) is context independent, and the right part is dependent on the following phoneme (or phonetic broad category). Phonemes that remain as a one-part phoneme can either be context-independent or be dependent on the following phoneme.

The system is trained using 13 MFCC features (12 cepstral coefficients and 1 energy parameter) plus their delta values, with a 10-msec frame rate. The input to the network consists of the features for the frame to be classified, as well as the features for frames at -60, -30, 30, and 60 msec relative to the frame to be classified (for a total of 130 input values). As many as 2000 samples per category are collected for training. Neural-network training is done with standard back-propagation on a fully-connected feed-forward network. The training is adjusted to use the negative penalty modification proposed by Wei and van

Vuuren [3]. With this method, the non-uniform distribution of context-dependent classes that is dependent on the order of words in the training database is compensated for by flattening the class priors of infrequently occurring classes; this compensation allows better modeling for an utterance in which the order of the words can not be predicted.

During the Viterbi search, transition probabilities are set to be all equally likely, so that no assumptions are made about the likelihood of one category following another category. The search was constrained to minimize insertion errors by having minimum duration values for each category, where the minimum value for a category was computed as the value at two standard deviations from the mean duration. During the search, category durations less than the minimum value are penalized by a value proportional to the difference between the minimum duration and the proposed duration.

The grammar allows any number of digits in any order, with an optional silence between digits. In addition, a "garbage" word is allowed at the beginning and end of each utterance to account for sounds not in the vocabulary. The "garbage" word is defined as a word with a single context-independent category; the value of this category is not an output of the neural network, but is computed as the N^{th}-highest output from the neural network at each frame [4]. In this study, N was set to 5.

Training is done for 30 iterations, and the "best" network iteration is determined by word-level evaluation of each iteration on the development set data. This "best" network is then used to force-align the same training utterances, and training and evaluation are repeated to determine the final digits network.

4. EXPERIMENTS

We evaluated several aspects of training a digit recognition system, including the feature set, the amount of data used for training, the type of context-dependent categories, the values for duration limits, and the type of grammar. Each of these aspects is described in more detail below.

4.1. Features

Ten sets of features were evaluated: 13^{th}-order MFCC with delta values (as used in the baseline system, referred to as *MFCC13D*), 13^{th}-order MFCC with no delta values (*MFCC13*), 9^{th}-order MFCC with and without delta values (*MFCC9D* and *MFCC9*), 13^{th}-order and 9^{th}-order PLP with and without delta values (*PLP13D, PLP13, PLP9D, PLP9*), a combination of 13^{th}-order PLP and 13^{th}-order MFCC (*PM13*), and a combination of 9^{th}-order PLP and 9^{th}-order MFCC (*PM9*). All PLP features were computed using RASTA pre-processing, and all MFCC features were computed using CMS pre-processing.

The evaluation of the combination of PLP and MFCC features was motivated by the hypothesis that training with the two slightly different representations would provide somewhat more robustness to noise, and that the combination of RASTA (which emphasizes regions of transition) and CMS (which does not emphasize transitions) would provide complimentary information. The evaluation of each type of feature with and without delta values was motivated by the belief that the neural networks should, in theory, be able to learn the information provided by the delta values without having these values provided explicitly. Two different cepstral orders (9 and 13) were used to test if the default value of 13 is an over-representation of the signal; with a sampling rate of 8000 Hz, there are on average only 4 formants, and the signal should be adequately represented by 2 cepstral coefficients per formant plus an additional coefficient to approximate the effect of the glottal source.

4.2. Duration Limits

We evaluated each of the 10 recognizers trained with the features described above using four types of duration limits: with minimum duration values taken at two standard deviations from the mean (the default, referred to as *2SD*), from the 2^{nd} percentile of all duration values (*2P*), from the 5^{th} percentile of all duration values (*5P*), and from the 8^{th} percentile of all duration values (*8P*). The reason for selecting a minimum duration value above the absolute minimum duration observed in the data is to remove outliers.

The motivation for comparing the standard-deviation based limits with the percentile-based limits was related to assumptions about the distribution of the data. It was thought that although two standard deviations from the mean might be an appropriate value if the data are normally distributed, a percentile-based method may be a more reasonable method of removing outliers if the data have a different distribution.

4.3. Grammar

We evaluated two types of grammars: the first allowed optional silence between digits (the default, referred to as *SIL*), and the second allowed an optional "garbage" word as well as optional silence between digits (*GAR*).

The motivation for evaluating these two grammars was to test whether the optional pauses between words are modeled sufficiently well by the silence category, or whether a more complex model is needed. The risk of using the GAR grammar was that the number of deletions would increase, by having valid words recognized as garbage. On the other hand, it was thought that the GAR grammar might provide better modeling of the non-speech sounds that may occur between words.

4.4. Categories

We evaluated all ten sets of features with two types of categories: phonetic categories that are dependent on the context of specific neighboring phonemes (the default, *PHON*), and phonetic categories that are dependent on the context of broad classes of phonemes (*BC*). The *PHON* recognizer has 218 outputs, and the *BC* recognizer has 163 outputs.

The motivation for using the PHON set of categories was that the phoneme-specific differences in a particular context may provide additional information about the word. The motivation for using the BC set of categories was the belief that the phoneme-specific differences within one broad class are

minimal, and that trying to determine minor phonetic differences in multi-speaker data might be futile.

4.5. Amount of Data

We trained all of the systems described above using as many as 2000 samples per category. For five of the ten most promising feature sets, we trained with all available hand-labeled data. The motivation for this comparison was to estimate the effect on recognition performance by increasing the amount of training data by 50%.

4.6. Evaluation Methodology

Due to the large number of possible combinations of tests, we conducted the evaluation using the following methodology:

1. Creating the baseline system using the method outlined in Section 3. (We confirmed that the results of this recognizer are comparable to the results of the CSLU Toolkit digits recognizer.)
2. Training and evaluating the 10 sets of features with 2000 samples per category, using the SIL grammar and 2SD limits, the GAR grammar and 2SD limits, the SIL grammar and 5P limits, and the GAR grammar and 5P limits. Training was done using the PHON set of categories.
3. Selecting the better grammar based on the results from step 2, and evaluating the better grammar with the remaining 2P and 8P limits.
4. Repeating steps 2 and 3 for each set of features using the BC set of output categories.
5. Selecting the five most promising sets of features with the best grammar, limits, and categories, and training networks with these features using all available hand-labeled data.

To create a final recognizer, we force-aligned all available data with the current best recognizer, trained another system using these force-aligned data, and then trained again using the forward-backward method [5]. We selected the best recognizer based on the word-level development-set results.

For evaluating the selected recognition system and the baseline system on the test set, we computed the significance level using McNemar's test (at the 5% level) and confidence intervals for both systems (at 95%). For computing the confidence intervals, we divided the test set into ten subsets (with approximately 217 digit strings per subset) and determined the recognition accuracy on each of these subsets.

5. RESULTS

The baseline system that we trained had word-level accuracy of 94.54% and sentence-level accuracy of 80.61%, which is comparable to the performance of the digits recognizer in the March 1998 release of the CSLU Toolkit, with 94.63% word accuracy and 82.27% sentence accuracy. The sentence-level results are not significantly different at the 5% level (P=0.44).

It can be seen in Figure 1 that the GAR grammar did better than the SIL grammar with the SD2 limits as well as with the 5P limits, so GAR was chosen as the best grammar. The duration limits for the 2P, 5P, 8P, and 2SD conditions using the GAR grammar are shown in Figure 2; it can be seen that the difference between the 2P and 5P results depends on the feature set, but that the 2P and 5P results are usually better than the 8P results and almost always noticeably better than the SD results. As a result, the 2P limits were chosen.

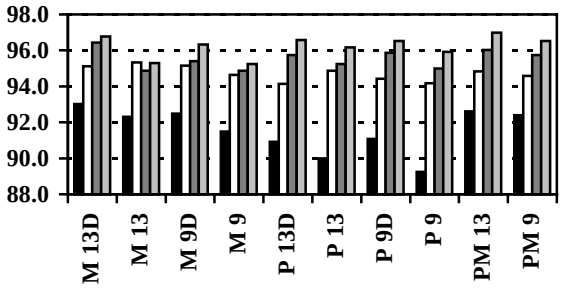

Figure 1: Word-level accuracy results for each of the 10 features using the four initial grammar and duration-limit combinations. The horizontal axis codes are explained in Section 4.1 The black bar is for the SIL grammar and 2SD limits, the white bar is for the GAR grammar and 2SD limits, the dark-gray bar is for the SIL grammar and the 5P limits, and the light-gray bar is for the GAR grammar and the 5P limits.

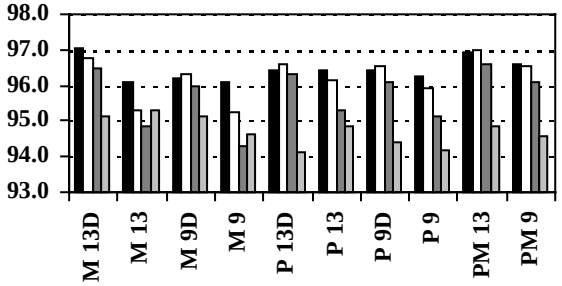

Figure 2: Word-level accuracy results for the same 10 feature sets as in Figure 1, using the four types of duration limits with the GAR grammar and the PHON output categories. The black bar is for the 2P limits, the white bar is for the 5P limits, the dark-gray bar is for the 8P limits, and the light-gray bar is for the 2SD limits.

The recognizers trained with the BC categories (162 outputs) had results similar to the recognizers trained using the PHON categories (218 outputs), but the PHON results had, on average, a 4% reduction in error. Figure 3 shows a comparison of the BC and PHON results for the GAR grammar and 2P limits.

The recognizers trained using all available data instead of 2000 samples per category had, on average, a 3.2% reduction in error. The increase in the amount of training data was 52.2%.

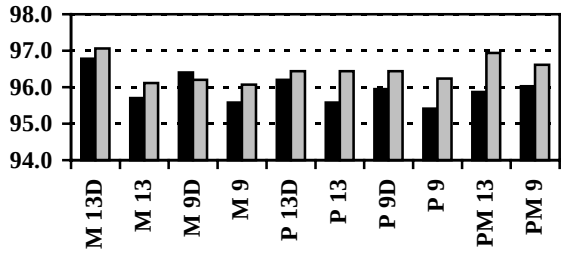

Figure 3: Word-level accuracy results for the same 10 feature sets as in Figure 1 (using the GAR grammar and 2P limits), comparing the BC categories (dark bar) and PHON categories (light bar).

Given these results, the best set of parameters was determined to be the GAR grammar that allows optional garbage between words, the 2P duration limits (which are computed from the 2nd percentile of duration values), the use of all available data, the PHON set of phoneme-specific categories, and 13th order MFCC coefficients with their delta values. The system trained with these features on all available hand-labeled data had 97.15% word accuracy and 89.13% sentence accuracy on the development set.

The development-set results from forced-alignment training were 97.68% (word) and 90.07% (sentence). Finally, the results from forward-backward training were 98.22% (word) and 91.96% (sentence).

The results of test-set evaluation are summarized in Table 1. The 90.36% sentence-level result on 2169 files (12437 words) is significantly better than the 80.08% baseline results, and the confidence interval is ±0.45% for the new recognizer and ±0.73% for the baseline recognizer.

System	Word Accuracy	Sentence Accuracy	Confidence Interval	Reduction in Error
Baseline	94.65%	80.08%	94.65±0.73%	n/a
New	97.52%	90.36%	97.52±0.45%	54%

Table 1: Test-set results for the baseline system and the new system, where the new system was trained with the set of best parameters as determined from the experiments in this paper. Evaluation was done on 2169 utterances (12437 words).

The results indicate that changing the duration limits and grammar had the greatest effect on recognizer performance, and forced alignment of all data and forward-backward training had the second-greatest effect. The use of all available hand-labeled data, the type of categories, and the choice of features yielded smaller improvements. For the choice of features, the use of delta parameters and the use of 13 cepstral coefficients yielded a typically consistent, small improvement over the use of no delta features or 9 coefficients. The combination of PLP and MFCC features did not yield a noticeable improvement over the use of delta features with MFCC or PLP alone.

6. DISCUSSION

In these experiments, it was found that the grammar and duration limits had a greater effect on recognition accuracy than the output categories, cepstral features, or a 50% increase in the amount of training data. Despite theoretically-motivated beliefs to the contrary, the use of delta features and 13 cepstral coefficients usually did improve performance.

It is hypothesized that the reason for the PLP results being consistently slightly worse than the MFCC results is that the CMS subtraction was not pipelined, and therefore was able to use more data for noise compensation than the RASTA method. For implementing a real-time system, the pipelined CMS may yield results that are more similar to, or possibly worse than, RASTA results.

Finally, it should be noted that the run-time complexity of the final system is the same as for the baseline system (both run in approximately real-time). Training time has been increased, simply because more training data is used for forced alignment and the forward-backward method requires another cycle of network training.

For those who would like to replicate our results or try further experiments, both the Numbers corpus and the CSLU Toolkit can be downloaded from http://cslu.cse.ogi.edu/Toolkit (free for academic use).

7. ACKNOWLEDGEMENTS

The authors would like to thank Chris Covert, Ben Serridge, Johan Schalkwyk (CMS/RASTA comparison), Jacques de Villiers, and the CSLU member companies. This work was sponsored in part by the National Science Foundation (grant numbers GER-9354959 and IRI-9614217); the views expressed in this paper do not necessarily represent the views of the NSF.

8. REFERENCES

1. Cosi, P., Hosom, J.P., Shalkwyk, J., Sutton, S., and Cole, R.A., "Connected Digit Recognition Experiments with the OGI Toolkit's Neural Network and HMM Based Recognizers," IVTTA-ETWR-98, Turin, Sep. 1998, accepted for publication.

2. Cole, R.A., Fanty, M., Noel, M., and Lander, T., "Telephone Speech Corpus Development at CSLU," ICSLP-94, Yokohama, September 1994, pp. 1815-1818.

3. Wei, W. and Van Vuuren, S., "Improved Neural Network Training of Inter-Word Context Units for Connected Digit Recognition," ICASSP-98, vol. 1, Seattle, May 1998, pp. 497-500.

4. Boite, J.M., Bourlard, H., D'hoore, B., and Haesen, M., "A New Approach Towards Keyword Spotting," EUROSPEECH '93, Berlin, Sep. 1993, pp. 1273-1276.

5. Yan, Y., Fanty, M. and Cole, R., "Speech Recognition Using Neural Networks with Forward-Backward Probability Generated Targets," ICASSP-97, Munich, April 1997, pp. 3241-3244.