Microsegment-Based Connected Digit Recognition

John J. Godfrey[1], Aravind Ganapathiraju[2], C. S.Ramalingam[1], Joseph Picone[2]

[1]Texas Instruments Incorporated
PO Box 655303, MS 8374
Dallas, TX 75265

[2]Inst. for Signal and Information Processing
Mississippi State University
Mississippi State, MS 39762

ABSTRACT

By building acoustic phonetic models which explicitly represent as much knowledge of pronunciation in a small domain (the digits) as possible, we can create a recognition system which not only performs well but allows for meaningful error analysis and improvement. An HMM-based recognizer for the digits and a few associated words was constructed in accord with these principles. About 65 phonetic models were trained on 140 carefully labeled utterances, then iteratively trained on unlabeled data under orthographic supervision. The basic system achieved less than 3% word error rate on digit strings of unknown length from unseen test speakers, and 1.4% on 7-digit strings of known length. This is competitive with word-based models using the same HMM engine and similar parameter settings. As an R&D system, it allows meaningful analysis of errors and relatively straightforward means of improvement.

1. INTRODUCTION

In HMM-based speech recognition systems, the more phonetic knowledge about pronunciation is explicit in the data structures, the better their statistical power can deal with the many other sources of variation in speech. This simple principle is not so simple in practice, since our knowledge of phonetic variation is limited, its sources are numerous, and its acoustic cues resemble those due to speaker, sex, channel, pragmatic context, etc. Nevertheless in small applications, such as connected digit recognition, it is useful to test this concept by building a system which puts as much phonetic knowledge as possible into the models, their states, and their permitted sequences, and comparing its performance to more traditional approaches.

Large vocabulary recognition systems necessarily use subword acoustic units which are predictable from a lexical representation, such as phones, biphones, or triphones. Small vocabulary systems, especially for isolated words, generally use whole-word models to capture the coarticulatory and contextual variation within digits. But when continuous speech is involved, as in recognizing connected digit strings or a small number of set phrases, the number of word-based models may need to be multiplied to account for between-word effects, and a grammar added to manage sequential constraints. In these cases, it may make sense to invest the effort necessary to represent the details of pronunciation, both within and across words, explicitly in symbolic form, so that the statistical power of the HMM decoder can be used to explain the unpredictable acoustic differences among speakers, channels, etc. Bush and Kopec [1] described such a system in 1986 for a DTW-based recognizer, and similar ideas have been expressed by Erler and Deng [2] for HMMs in a research context.

Texas Instruments has developed several systems for small vocabulary applications, including connected digit recognition over the telephone, and comparisons of system performance as a function of recognition units and resource usage are frequently made. Doddington and Godfrey in 1988 experimented with a microsegment-based system and found that with carefully chosen models and expert labelling it could be competitive with a word-based system using the same basic HMM recognition engine, but they did not develop this further. Given extensive improvements to the underlying recognition system and better databases to work with, we decided to repeat and extend this experiment with several changes.

In the system described here, symbols are assigned to phonetic segments in sufficient detail to model a number of phonetic effects due to local context, coarticulation, dialect, speaking style, and even speaking rate. The initial labelling of speech data is designed so that it could support training phone or word models as well, by collapsing the marked microsegments into appropriate phoneme- or word-sized units. It is currently serving as a useful research tool for tracing recognition errors to the phonetic states that triggered them, and for measuring the impact of various computational resources on recognition performance.

2. METHOD

2.1 Recognition System

The Texas Instruments Hierarchical Grammar(HG) continuous speech recognizer has an LPC-based front end with mel-spaced filters and a Viterbi-Beam decoder. Since some of the models represent very short acoustic events, we used a 10ms frame period for the LPC analysis and a window duration of 20ms with 50% overlap. A 34-element parameter vector of spectral and energy measurements and differences is transformed into a 16-element feature vector for recognition. A pooled covariance metric was used for the acoustics in the experiments described here.

Since our purpose was not to optimize performance but to evaluate a tool for R&D use, the recognizer parameter settings for benchmarking the microsegment system were simple and standard ones. Except for frame size, the same parameter settings, training data and test data were used in the comparisons with other methods. Even relatively standard enhancements such as mixtures, full covariance, or discriminant training were avoided, and no preprocessing of the files was done. Other than the phonetic segments, only lipsmacks and audible breathing were modeled explicitly, using the same techniques. All other background sound was classified into two omnibus categories of noise and silence. The HMM model topology allowed up to one skip or two stays per state, except that two nonspeech models could stay indefinitely.

2.2 Database

We chose Macrophone, a publicly available corpus of digitally collected telephone speech [3], for training and testing in all the experiments described here. All data were from male speakers saying strings of seven to fourteen digits, including occasional occurrences of “dash” and “area code.” All training was done on files from the designated training section, and all testing from the development test section of the corpus. The labeled data came from the first two directories of the training set. additional training data came from other directories in the training and development test set. Test data were 506 male digit strings from other directories in the development test set. These test files, more than the training data, were audited and purged of about a dozen problem files -- a few with children’s or women’s voices, a few misread items (“thirty”, “sixteen”, “excuse me”).

2.3 Model Creation

The phonetic labeling begins with a choice of symbols based on experience -- at least one symbol per phoneme in the standard lexical representation of each word. The goal is to provide a symbol for each acoustically identifiable phonetic state which may or must occur when the digit is spoken. For example, “one” has an optional u-like steady portion where the first two formants are essentially parallel (labeled “w1”), followed by an obligatory transitional portion in which both formants rise toward a central vowel (w2), an optional segment, not usually seen when the digit is spoken rapidly, where the vowel formants reach steady state (ah), and an obligatory final nasal segment. Depending on the following word and the style of speaking, the nasal segment may be an unreleased nasal stop (n), a geminate (nn as in “one nine”), a tap (nx as in “one oh” or “one eight”), a released stop (n - ax in very emphatic isolated pronunciation), or a labialized n (nw in “one one”) with a falling second formant during closure. There may also be epenthetic t or d before a following s or z. Some of these phonetic states occur elsewhere as well: the w1 between connected utterances of “oh oh”, the n variants in “seven” and “nine,” etc.

Some dialect differences may also be distinguished and captured symbolically. For example, three symbols are used for “oh” to distinguish common dialectal variants encountered in the Macrophone database. These occupy alternate paths in the pronunciation network, and can carry different transition probabilities, given enough training data. The steady state and offglide portions of “five” and “nine” each receive separate labels, and the pronunciation network has a path that skips the offglide. This matches the pronunciations of many southern speakers.

After labeling all the digits, “oh”, “area code”, and “dash,” as spoken by about 50 speakers, it was still necessary to search elsewhere in the corpus explicitly for examples of some of the low-frequency phonetic states (like tapped d in “eight eight”) which did not occur often enough in the labeled data. The 140 digit strings contained about 6000 labeled segment tokens. For each segment type, duration statistics were examined, and seed models were selected, excised, and trained over the labeled data as described below.

2.4 Pronunciation Grammar

A hierarchical grammar was created by hand to generate a finite state network of all the permitted sequences of segments (pronunciations), whether spoken as individual digits or in unconstrained strings. Since at least half of the segments are optional events, depending on styles of pronunciation, phrasing, tempo, etc., and since many of these occur at or near word boundaries, we chose a “most stable phonetic segment” for each digit, and wrote the low-level grammars in terms of mid-digit to mid-digit pairs which terminate and start at these points. Silence is a digit for the purposes of this grammar, and is the top-level

start and stop state for any complete string or substring in an utterance.

2.5 Training

The seed models were initially trained over the 140 phonetically marked utterances. Listening, examination of spectrograms, and recognition of the training data all suggested that the models were developing as expected. A few small changes in the segments and the grammars were made at this point: short and long versions of “oh” were introduced, certain low-frequency segments were deleted or simplified, a few segments were merged. A second pass of training was then run on another 400 unlabeled training files using only orthographic supervision, i.e., the recognizer chose the best-scoring sequence of microsegments along any pronunciation network path compatible with the digit sequence in the orthographic transcription. Non-speech events in the transcription, such as [\line_noise] or [cough], were ignored in this training.

In order to see how stable the acoustic models were, one recognition experiment was conducted at this point, after the two-stage training on a total of 540 digit strings. The results are in the last line of Table 1.

Finally, a third pass of training was run, in which the models trained in the first two stages were retrained over all the usable digit strings spoken by male speakers in the training section of Macrophone. This comprises approximately 4000 random 7-, 8-, 10- 12- and 14-digit strings (counting “dash,” “area,” and ”code” as digits).

3. RESULTS

3.1 Tests

To calibrate expectations for the microsegment-based system, word models were also run using the same HG recognizer. The word models were trained over all the digits in the training section of Macrophone (about 40000). Unfortunately, for historical reasons, the word models were only available in 20 millisecond frame lengths, and there was not enough time to change this.

The word model and microsegment systems were compared on the same 506 “certified” test files from Macrophone described earlier. They were also compared on two subsets of these files, containing just the 7-digit and just the 10-digit strings. Each test was also run with and without using a grammar which knew the intended length(s); these are the constrained (C) and (U) unconstrained conditions of the first column of the table below.

System (constraints)	#in String	# Test Files	# Training files	% Word Err	% String Err
MicroSegment(C)	7	146	4000	1.4	7.5
Word (C)	7	146	4000	1.1	7.5
MicroSegment(U)	7	146	4000	2.8	16.4
Word (U)	7	146	4000	1.9	11.0
MicroSegment(C)	10	160	4000	2.7	15.0
Word (C)	10	160	4000	3.0	17.5
MicroSegment(U)	10	160	4000	2.7	19.4
Word (U)	10	160	4000	3.8	27.5
MicroSegment(C)	7-14	506	4000	2.4	13.2
Word (C)	7-14	506	4000	2.8	16.6
MicroSegment(U)	7-14	506	4000	2.9	20.5
Word (U)	7-14	506	4000	3.3	24.1
MicroSegment(U)	7-14	506	540	3.2	21.1

Table 1. Comparison of word and string error rates for the microsegment-based system and for word- and phoneme-based methods. (C) = constrained, (U) = unconstrained (see text).

3.2 Results

Table 1 gives the results for the two systems under several conditions: number of test files, amount of training, digit string length(s) in the test set, and whether constrained or not by a grammar for string length(s).

The microsegment method is clearly competitive with the whole-word method, achieving better performance in four of the six conditions. Word-based recognition will, of course, run faster and require less memory because of the pronunciation network. However, it is possible that the microsegment-based system could justify the extra resource requirement to be used outside the laboratory, if it turns out that the error rate can be driven significantly lower by error analysis and revisions, which word-based systems do not invite. And although word models achieve the lowest error scores on 7-digit strings, microsegments seem to outperform word models as the task gets harder (longer strings, less constraints.) Note also that the microsegment models only improved 10% (from 3.2% to 2.9% WER) when trained on 4000 vs. 540 digit strings. This suggests that more models or better features, rather than more data, are the key to improve performance. This is likely to be true for some of the traditionally troublesome combinations like connected pronunciations of "zero-oh", "two-oh", and "oh-oh" These do not occur very often in the training data, but need to be sought out and modeled with care, since they constitute both an insertion and a deletion problem.

Word models tend to fail because of inter-word pronunciation effects. What about microsegments? A quick glance at the errors showed that, out of 108 files with errors, 33 had an error in the absolute final position. Moreover, listening to the errors revealed that of the total of 158 errors, on strings averaging 10 digits in length, about 63 occur before a pause of some kind. Whether this is a result of the changes in tempo that occur at pauses, particularly the lengthening, pitch drop, and diminishing energy on the segments just before a pause, will be the subject of further study. Analysis will be carried out by comparing supervised vs. unsupervised recognition results on the error files, to see at what frames of what segments the missed identifications happened. Here the microsegment system is extremely useful and instructive.

4. CONCLUSION

The experiments described here were designed to address the specific question whether a system based on phonetically motivated microsegments can be competitive with other well-developed systems such as word- and phone-based systems. If so, it can be extended to other limited domains as well, provided enough data is available. The investment of time is a few weeks to a few months, the amount of training data required is reasonable, although speed is a problem. It provides a research tool for comparing and analyzing the performance of recognizers based on units from microsegments to words.

We conclude that a combination of phonetic analysis, training under expert supervision, training under mechanical supervision, and HMM decoding can result in an effective system for connected digit recognition. Future work will include refinement of the models, and extension of their coverage for dialects.

REFERENCES

[1] Bush, M. A. and Kopec,G. E., "Network-Based Connected Digit Recognition Using Explicit Acoustic-Phonetic Modeling", *Proc. IEEE ICASSP-86*, pp. 1097-1100, 1986.

[2] Erler, K. and L. Deng, "Hidden Markov model representation of quantized articulatory features for speech recognition," *Computer Speech and Language* 1993, Vol. 7, pp. 265-282.

[3] Bernstein, J., Taussig, K., and J. Godfrey, "Macrophone: An American English Telephone Speech Corpus for the Polyphone Project," *Proc. IEEE ICASSP-94*, pp. 81-84.