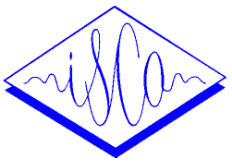

6th International Conference on Spoken Language Processing (ICSLP 2000)
Beijing, China
October 16-20, 2000

Multi-Path, Context Dependent SC-HMM Architectures for Improved Connected Word Recognition

Tzur Vaich and Arnon Cohen

DSP Group, Omarim 9, Industrial Park, Omer 84965 Israel
Tel.: +972-7-6900313 ; Fax: +972-7-6900314
tzurv@dsp.co.il

ABSTRACT

Connected Word Recognition (CWR) systems are needed for many consumer applications. In such applications cost is a major factor. Several architectures for HMM based CWR are examined to provide an optimal cost effective configuration.
Three architectures were examined: a simple LTR with no skips, LTR with skips (LTR-WS) and Multi-Path LTR. The LTR-WS and the MP-LTR were shown to be superior to the simple LTR. The MP-LTR exhibited the best results with long strings.

1. INTRODUCTION

Inexpensive Isolated Words Recognition (IWR) systems, implemented on fixed point CPU or DSP chips, are currently commercially available [1]. The market for such systems is limited. Continuous Speech Recognition (CSR) systems, though available on PC based machines, cannot yet be inexpensively implemented to cover the low-end consumer market. Connected Word Recognition (CWR) systems may provide a solution that is both inexpensively implemented and is user friendly. CWR are expected to open large consumer markets. Connected word speech posses a severe coarticulation problem that renders the recognition algorithm to be relatively complex and memory expensive. The goal of this work was to explore different HMMs architectures for CWR systems where the main concern is cost effectiveness. The database used for testing the system consisted of digits strings of various length.
When comparing different architectures for CWR, the cost factor is mainly determined by the amount of required memory. This is so since all systems require approximately the same DSP. A major saving in memory is achieved by using Semi-Continuous HMM (SC-HMM) rather than the conventional HMM. In SC-HMM a shared bank of N Gaussian functions is employed. Each model uses the common bank of Gaussians and thus requires only a matrix of linear combination coefficients.
A SC-HMM system, with Left-To-Right with no skips models using s states (each using q Gaussians), that employs a feature space of p dimensions, a Gaussian bank of N functions (with diagonal covariance matrices) and a dictionary of M models, requires 2pN parameters for the Gaussian bank and (2p-1+sq)M parameters for the models. For example a system with N=500 Gaussains, p=20 features, M=11 models with s=10 Left-to-Right states, with average of q=15 Gaussians in each state requires 2K parameters for the models and 20K parameters for the Gaussians bank. The number of Gaussians in the system has also a direct influence on the required MIPS.
Several HMM architectures were compared. The results of these architectures were investigated as a function of the number of Gaussians (cost). A relatively wide range of sizes of the Gaussian bank (500 to 2000 Gaussians) was used in the comparison.

2. MODEL STRUCTURES

The algorithm is based on Semi-Continuous Hidden Markov Models (SC-HMM). Three basic SC-HMM architectures were tested. The tests were performed with a connected digit database.

1. Left-To-Right model with no skips (LTR), (16 states).
2. Left-To-Right With Skips model (LTR-WS) (figure 1.a), (16 states).
3. Multi-Path Left-To-Right model MP-LTR (figure 1.b), (14 states).

The motivation behind MP-LTR was to establish a model that can better handle coarticulation effects. A similar approach was used by [2]. Each word model is divided into 3 parts: Beginning, Middle and End. Coarticulation is mainly presented in the

Fig. 1.a: 16 States HMM with one skip transition (LTR-WS).

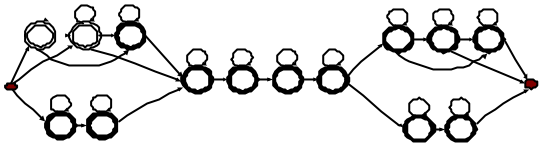

Fig. 1.b: 14 States HMM (MP-LTR).

beginning and end parts. The MP-LTR allows several "options" for the beginning and end sections of the word model.

Two cases were analyzed for each one of the architectures: Context independent models and Context dependent models. Context independent models assume that one model is sufficient to describe all the various phonetic neighborhoods. The model is thus trained by all the occurrences of the particular word in the training database. In the context dependent case several models were used to model each word, to describe the particular word in different phonetic neighborhoods. The training of the model is performed with all the occurrences of the word in the desired phonetic neighborhood included in the training database.

We describe the phonetic neighborhood by means of: *w*- the word (digit) to be modeled, *n*- background noise, $j-$ any word in the dictionary (including background noise) and *f* - any word in the dictionary excluding background noise. The phonetic neighborhood *f - w - n* describes a partial connected string with the desired word *w* preceded by any one of the dictionary words and followed by background noise. This partial string usually describes the word *w* appearing at the end of the given string or in the middle before a pause.

In this paper we deal with two systems defined by the number and types of models:

- System A: 1 model/word (Context independent). The model is trained with all occurrences *j -w-j* .
- System B: 2 models/word. "End" model, trained with all the occurrences of *j - w–n*, and a second model trained with all occurrences of *j -w-f* .

A composite parallel model performs the recognition. In the case of system A the composite

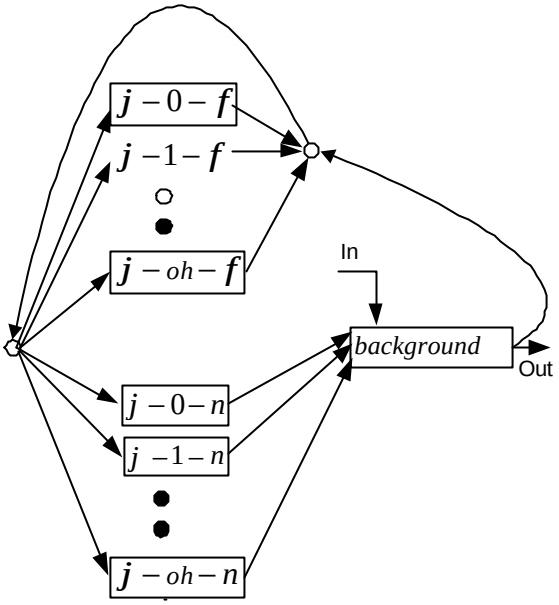

Fig. 2: Network composite model for system B.

model includes all word models (plus a background noise model) in parallel. The model of system B is shown in figure 2. The motivation for the model came from preliminary analysis of the results of system A that show relatively large errors at the end of the strings.

3. EXPERIMENTS

3.1 DATABASE

The TIDIGITS database includes 10,000 variable duration strings (1 to 7 digits) for training and 11,000 strings for test. The vocabulary consists of 11 words, the ten digits plus "oh". The data is sampled at 20KHz with 12 bits resolution. The strings were uttered by 111 males and 114 females. Since our interest is in the low end market (including telephony applications) the database was re-sampled at 8KHz in an office environment by sounding the original wave files using high quality speaker and recording with a directional microphone (C-400).

3.2 SYSTEM DESCRIPTION

The training and recognition was performed using the *HMMEng* ™ continuos speech recognition system (DSP Group's HMM-Engine). The Connected Word Recognition network described here does not use a language model, that is to say it assumes that words (including background noise or

"non-speech") appear in the given string in a statistically independent random manner. In most applications (such as telephone numbers or command strings) there is some a priori information given on sequence duration and the conditional probabilities of words, such that language models may be used. In this sense, the results presented in the paper are worst case results.

The end points of the string are estimated by means of a VAD algorithm described in [3].

The 8KHz speech signal is processed with frame length of 256 samples (32 millisec.) with 50% overlapping. The feature vector consisted of a 10^{th} order mel-scale filter bank cepstral coefficients (MFCC) [4] and their 1^{st} order derivatives. Hence the feature space is of order 20.

3.3 RESULTS

In CWR systems it is very important to achieve full string correct recognition. We therefore report the results using three measures: Average of percent complete string recognition (*Av*), Average percent recognition of seven digits strings (*7d*) and Word Error Rate (*WER*) defined in equation 1.

$$WER = \frac{WordsNo - Sub - Del - Ins}{WordsNo} 100 \qquad (1)$$

Where *WordsNo* is the number of words in the string, *Sub* is the number of words substituted in the recognized string, *Del* is the number of words deleted in the recognized string and *Ins* is the number of words falsely inserted in the recognized string. The Levenshtein matching DP algorithm [5] was employed in order to determine the parameters of equation 1.

The results are presented in tables 1-4 and in figures 3-4. Tables 1-2 show the results of the performances of systems A and B with an inexpensive implementation of about 850 Gaussians.

Table 1:Recognition results, System A
"Inexpensive" mode (≈850 Gaussians)

	Gau. No.	System	*Av*	*WER*	*7d*
LTR	848	A	94.68	97.7	87.31
MP-LTR	864	A	95.8	98.3	90.38
LTR-WS	826	A	96.05	98.4	90.46

Table 2: Recognition results System B
"Inexpensive" mode (≈850 Gaussians)

	Gau. No.	System	*Av*	*WER*	*7d*
LTR	857	B	95.08	98.0	88.12
MP-LTR	875	B	96.14	98.5	90.95
LTR-WS	845	B	96.06	98.4	90.14

Tables 3-4 give the results for a relatively expensive system of about 1400 Gaussians.

Table 3:Recognition results, System A
"expensive" mode (≈1400 Gaussians)

	Gau. No.	System	*Av*	*WER*	*7d*
LTR	1410	A	95.1	97.9	88.12
MP-LTR	1100*	A	96.1	98.4	90.38
LTR-WS	1475	A	96.6	98.6	91.92

*results in the range 1100-1500 were almost constant.

Table 4:Recognition results, System B
"expensive" mode (≈1400 Gaussians)

	Gau. No.	System	*Av*	*WER*	*7d*
LTR	1422	B	96.05	98.35	90.46
MP-LTR	1493	B	96.7	98.7	92.4
LTR-WS	1393	B	96.8	98.7	92

Figure 3 shows the recognition results (*Av*) of the various tested systems as a function of the number of Gaussians (cost). Figure 4 shows the performance of system A as a function of cost. The performance is measured here by means of complete correct string recognition for each one of the string duration (1-7).

4. CONCLUSIONS

The recognition results of systems A and B were, in general, similar to other reports on Connected Digits Recognition [6]. Since the recognition rate is high, the absolute number of errors was low, making it difficult to perform statistical analysis on the distribution of errors. Nevertheless the following observations were made on the major error sources:

- Deletion of one "Oh" in strings with "Oh-Oh".

- Deletion of one "Eight" in strings with "Eight-Eight".
- Insertion of "Oh" mainly after the digit 'Two".
- Insertion of "Six". "Six" was inserted in many cases when the background noise was high. Similar insertion problem was reported at [7].
- Major substitutions were: "Four" as "Oh", "One" as "Nine" and "Five" as "Nine".

The results clearly demonstrate the superiority of the LTR-WS and MP-LTR on the simple LTR (with no skips). Apparently the LTR with no skips architecture does not possess sufficient flexibility to model the variations caused by coarticulation.
The LTR-WS architecture is shown to be superior to the MP-LTR when judged by the *Av* measure (see figure 3). This seems to contradict the basic assumption of flexibility since the MP-LTR architecture is more flexible than the LTR-WS. In our case however the LTR-WS has maximum 16 states while the MP-LTR has only 14. The LTR-WS has therefore an advantage in terms of better modeling the non-stationarities of the word.
The fact that under the strict measure *7d* the MP-LTR was shown to be superior to the LTR-WS (see tables 2 and 4) may suggest that it is indeed potentially a better architecture.
Tests with LTR-WS and MP-LTR with equal number of maximum states are now under investigation.

5. REFERENCES

[1]D. Geller, M. Lieb, W. Budde, O. Muelhens and M. Zinke, "Speech recognition on SPERIC – an IC for commands and control application", Proc. EUROSPEECH97, Vol. 2, pp. 625-628, 1997.

[2] R. Chengalvarayan, "Robust energy normalization using speech/nonspeech discriminator for German connected digits recognition", Proc. EUROSPEECH99, pp. 61-64, 1999.

[3] H. Kremer, A. Cohen and T. Vaich, "Voice Activity Detector (VAD) for HMM based Speech Recognition", Proc. ICSPAT 99, 1999

[4] J.W. Picone, "Signal modeling techniques in speech recognition", Proceeding of the IEEE, Vol. 81, No. 9, pp. 1215-1247, Sep. 1993.

[5] M.H. Ackroyd, "Isolated word recognition using the weighted Levenshtein distance", IEEE Tran. on ASSP., Vol. ASSP-28, No. 2, pp. 234-244, Apr. 1980.

[6] L.R. Rabiner, J.G. Wilpon and F.K. Soong, "High performance connected digits recognition using hidden Markov models", IEEE Trans. On ASSP, Vol. 37, No. 8, pp. 1214-1225, Aug. 1989.

[7] J. Hakkinen, J. Suontausta, R. Hariharan, M. Vasilache and K. Laurila, "Improved feature vector normalization for noise robust connected speech recognition", Proc. EUROSPEECH99, pp. 2833-36, 1999.

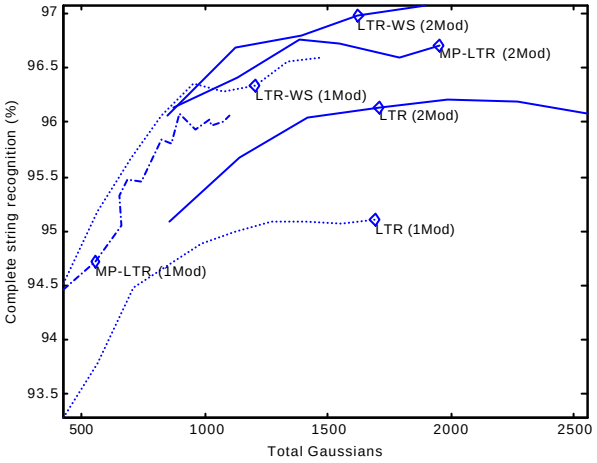

Fig. 3: Recognition Performance as function of Gaussian bank size

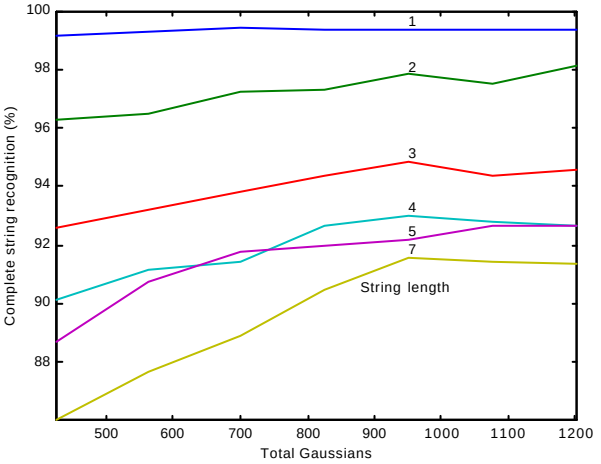

Fig. 4: Recognition performance as function of Gaussian bank size of system A with LTR-WS models.