A NEURAL NETWORK FOR 500 VOCABULARY WORD SPOTTING USING ACOUSTIC SUB-WORD UNITS

Ha-Jin Yu *Yung-Hwan Oh*
Dept. of Computer Science
KAIST (Korea Advanced Institute of Science and Technology)
Taejon 305-701 Korea
hjyu@bulsai.kaist.ac.kr

ABSTRACT

A neural network model based on a non-uniform unit for speaker-independent continuous speech recognition is proposed. The functions of the neural network model include segmenting the input speech into sub-word units, classifying the units and detecting words, and each of them is implemented by a module. The recognition unit we propose can includes arbitrary number of phonemes in a unit, so that it can absorb co-articulation effects which spread for several phonemes. The unit classifier module separates the speech into stationary and transition parts and use different parameters for them. The word detector module can learn all the pronunciation variations in the training data. The system is evaluated on a subset of TIMIT speech data.

1. INTRODUCTION

In speech recognition, neural network models have been considered because of the potential of providing massive parallelism and robustness in hardware faults. A number of neural network models have been used for isolated word recognition or speaker-dependent continuous speech recognition. Recently, neural network models have been also proposed for speaker-independent continuous speech recognition, but most of them are hybrid systems with hidden Markov models [1], and there are few pure connectionist approaches for that task. In our approach, all the functions from speech segmentation to word detection are implemented by neural networks.

The proposed system is based on segmental approach which has an advantage over frame-based techniques that by looking at a whole segment at once, we are able to take advantage of the correlation that exists among frames of a segment and be able to model the time dependence of spectral features. Most of the segmental approaches obtain segments from utterances by iterations [1], or shifting windows in time [2], so they require a large amount of computation time. In our approach, the utterance is segmented into acoustic segments by using a simple neural network structure without iterations and shifting windows during the recognition, by defining a non-uniform unit [3].

The non-uniform unit defined in this research has segmentation boundaries at stationary points, so that the segmentation result is relatively stable. A unit can have an arbitrary number of phonemes in itself, so it can absorb co-articulation effects which spread for several phonemes, and it has a transition part in the middle of the unit, so the system can model the temporal representation of the transition part.

The system is composed of three modules. The first module segments the utterance into non-uniform unit segments, and the segments are classified by the second module which uses different parameters for the transition and the stationary parts of the unit. The third module detects words from the classified unit series. The module can learn all the pronunciation variations of the words in the training data.

2. STRUCTURE OF THE SYSTEM

Figure 1 shows the structure of the system. The system consists of three modules: segmentation, unit classification, and word detector modules.

2.1. The Segmentation Module

The input feature vector stream is segmented by the segmentation module, which has three layers, REG, STM, and MIN layers. The REG layer calculates the regression coefficients of input vectors by shifting a time window. The output of mth neuron at time t, $r_m(t)$ is

$$r_m(t) = \sum_{n=-N}^{N} x_m[t+n] \cdot w_n^r, \qquad 1 \le m \le p \quad (1)$$

where the input $x_m[t]$ is mth component of mel-cepstral coefficients at time t; N = 3 decides the size of the window; p = 14 is the analysis order; and the weights are fixed to $w_n^r = n$.

The output value of the STM layer at time t, s(t) is the spectral transition measure [4].

$$s(t) = \sum_{m=1}^{p} | r_m(t) | \cdot w_m^s \quad (2)$$

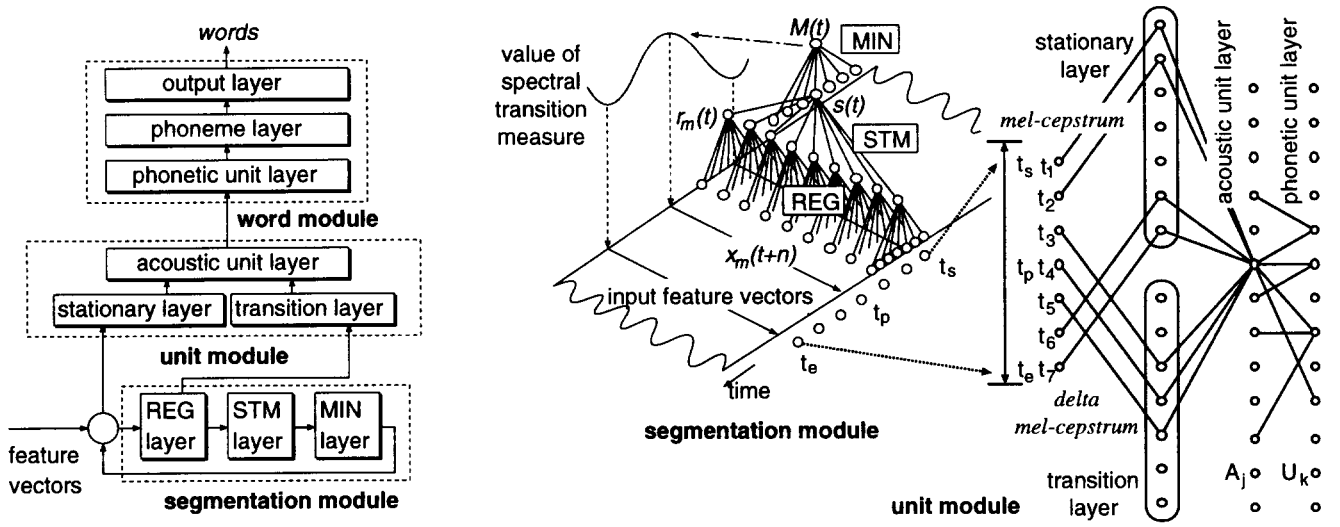
The MIN layer detects the points where the outputs of the STM layer are local minima. The output value M_t at time t is

$$M_t = f_h(\sum_{n=-N}^{N} f_h(s(t+n) - s(t)) - \theta_M) \quad (3)$$

where the function $f_h()$ is the hard limiting nonlinearity, and $\theta_M = 2N - 0.5$ is a threshold. The point where the output of the MIN layer is active becomes a boundary of a segment.

2.2. The Unit Classification Module

A segment is normalized to seven vectors by resampling it. One input vector is selected from t_pth frame, where the spectral transition measure has a peak value. Two vectors are selected at the two stationary frames t_s and t_e, which are at the ends of the segment. The rest of the vectors are



(a) Neural network structure

(b) The segmentation module and the unit module

Figure 1. Structure of the system

between those frames. That is, the input consists of the seven frames,

$$t_1 = t_s \quad (4)$$
$$t_2 = t_s + \lfloor (t_p - t_s)/3 \rfloor \quad (5)$$
$$t_3 = t_p - \lfloor (t_p - t_s)/3 \rfloor \quad (6)$$
$$t_4 = t_p \quad (7)$$
$$t_5 = t_p + \lfloor (t_e - t_p)/3 \rfloor \quad (8)$$
$$t_6 = t_e - \lfloor (t_e - t_p)/3 \rfloor \quad (9)$$
$$t_7 = t_e \quad (10)$$

The vectors close to the ends of a segment are fed to the stationary layer, and all seven vectors are fed to the transition layer. The features for the stationary layer are mel-cepstral coefficients, and those for the transition layer are delta mel-cepstral coefficients which can represent the dynamic nature of the utterance.

The activation A_j for jth output neuron of the acoustic unit layer is

$$A_j = f_h(\theta_U - \sum_{i=1}^{7} \|\vec{w_{ij}^s} - \vec{m_i}\| - \alpha \sum_{i=1}^{7} \|\vec{w_{ij}^t} - \vec{d_i}\|) \quad (11)$$

where $\vec{w_{ij}^s}, \vec{w_{ij}^t}$ are weight vectors between jth neuron of the acoustic unit layer and ith neuron of the stationary and the transition layers, respectively. $\vec{m_i}, \vec{d_i}$ are ith input vectors of mel-cepstrum and delta mel-cepstrum, respectively, and $\alpha = 7$. The neurons of the acoustic unit layer are linked to those of the phonetic unit layer in the word module.

2.3. The Word Detector Module

The word detector module plays the roll of the lexicon in this system. The module is trained by the phoneme transcriptions in the dictionary and by the training utterances as shown in Figure 2. Let a word w be composed of n phonemes $p_1, p_2, ..., p_n$ in the dictionary, and the word w be transcribed by the phonemes $q_k, q_{k+1}, ..., q_l$ $(0 < k \leq l \leq M)$ in a training sentence with phoneme transcription $q_1, q_2, ..., q_M$. A word output node has its own set of nodes which represents the n phonemes $p_1, p_2, ..., p_n$ in the phoneme layer.

The phonemes p_i $(0 < i \leq n)$ and q_j $(k \leq j \leq l)$ are matched by dynamic programming as in Figure 2 (a). After segmentation, a segment s is labeled as $[q_s, q_{s+1}, ..., q_t]$ $(0 < s < l$ and $k < t)$ by the phonemes included in the segment as in (b). If a phoneme $q_j (k \leq j \leq l)$ in the segment s is matched to a phoneme p_i, then a link is added between the node for the segment s in the phonetic unit layer and the node for the phoneme p_i in the phoneme layer as in (c). For more training data with the same words, the new unit nodes are linked to the phoneme layer nodes by the same procedure.

A neuron in the phonetic unit layer is activated when the corresponding neurons in the acoustic unit layer are activated. The phonetic units are classified by the phonemes included in the units, and the acoustic units are distinguished by the acoustic distances between the units. The mapping is needed because the words can be described by phonetic units, and the input signal can be easily transformed to acoustic units. The value of ith neuron $P_i^w[t]$ in the phoneme layer of the word w according to the input from the phonetic unit layer at time t is

$$P_i^w[t] = \begin{cases} \max(P_i^w[t-1], \max_k(U_k[t])) & \text{if } B_i^w = 1 \text{ and } F_i^w = 1 \\ \max(U_k[t]) & \text{if } B_i^w = 1 \text{ and } F_i^w = 0 \\ P_i^w[t-1] - d & \text{if } B_i^w = 0 \text{ and } F_i^w = 0 \\ & \text{and } P_i^w[t-1] > d \\ P_i^w[t-1] - P_i^w[t-1] & \text{otherwise.} \end{cases} \quad (12)$$

$$B_i^w = \begin{cases} 1, & \text{if } P_{i-1}^w[t-1] > 0 \text{ or } i = 0, \\ 0, & \text{otherwise.} \end{cases} \quad (13)$$

$$F_i^w = \begin{cases} 1, & \text{if } P_{i+1}^w[t-1] > 0 \text{ or } i = 0, \\ 0, & \text{otherwise.} \end{cases} \quad (14)$$

where $U_k[t]$ is kth neuron in the phonetic unit layer at time t, which is linked to P_i^w, $1 \leq i \leq L_w$, L_w is the number of phonemes in the word w, and d is a constant. The neurons in the phoneme layer are activated only when they are activated in order. Otherwise, the value of P_i^w is decreased by the constant d.

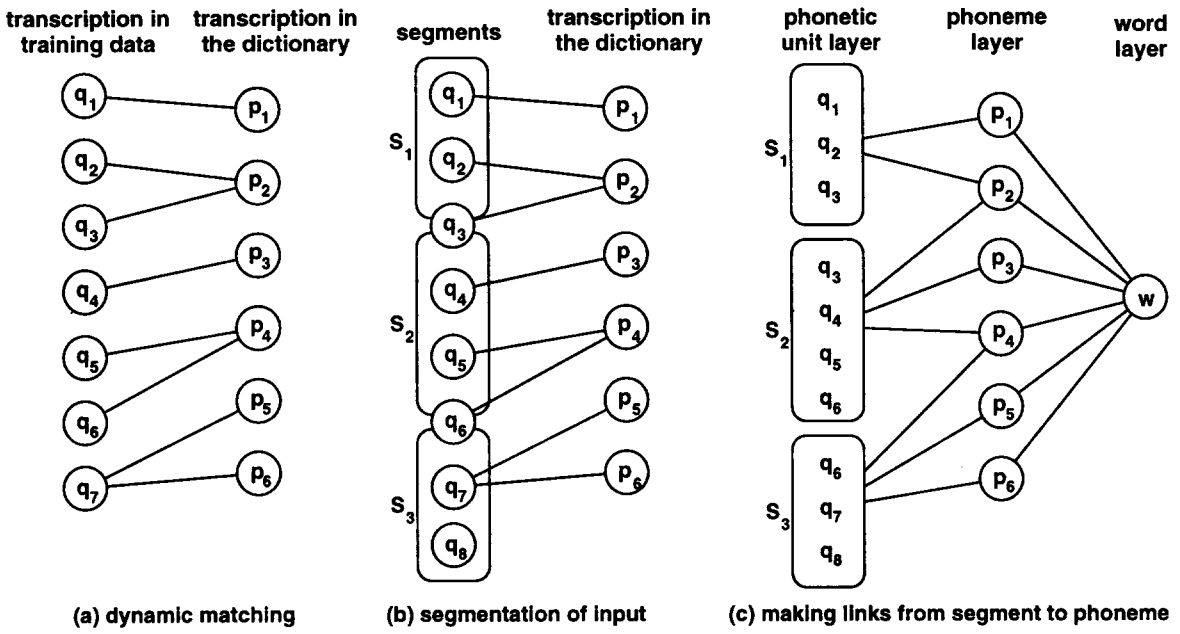


Figure 2. Training the word detector module

Finally, the output of the word module $O^w[t]$ which corresponds to the word w at time t is

$$O^w[t] = f_h(\sum_{i=1}^{L_w} P_i^w[t] \cdot w_i^w - \theta_w) \qquad (15)$$

where θ_w is the threshold for the word w, and $w_i^w = 1/L_w$. The activation of an output neuron means the detection of the word w ending at time t.

2.4. Training the network

The modules are trained by supervised Hebbian learning using phoneme-labeled speech data, except the segmentation module which is fixed. After a recognition process, if a word is not recognized correctely, the segments that caused errors are used for training. A segment is defined as a unit by the phonemes included in it. If the unit corresponding to the segment has not been defined, a node for the unit is added to the acoustic unit layer. If the unit is already trained with other utterances, the links are updated. A node for a word is added whenever a new word presents in the training utterance.

3. EXPERIMENTAL RESULTS

The system is simulated on a workstation. A 16 msec Hamming window is applied to the speech for every 8 msec, and 14th-order LPC cepstral coefficients are extracted. Table 1 shows the experimental conditions and a part of the results. A subset of TIMIT (SX set) is used to test the system. We select two test sets with the vocabulary size of 520 and 1000 each. The 520 words set is used to investigate the properties of the system, and the 1000 words set is used to compare the performance with other systems.

As a result, 5068 units are defined in the training of 520 words set, and 59.3% (2413) of the total 4077 units in the test data are classified correctly as one of the ten candidates. Figure 3 shows the receiver operating characteristic (ROC) curve for the 520 words spotting experiment. When we select only the first one word candidate at any point to get a sentence recognition result with no grammar, that is,

Table 1. Experimental conditions and some results

vocabulary		520 words		1000 words
segmentation		transition	stationary	stationary
No.of sentences	training	900	900	1782
	test	150	150	381
test keywords		594	594	1213
No.of speakers (female + male)	training	368	368 (72+296)	623 (190+433)
	test	103	103 (9+94)	215 (53+162)
No.of units	defined	5339	5068	7902
	test	5710	4077	7986
% unit correct		48.3	59.3	57.5
% word correct		51.2	77.8	65.5

when the perplexity is 520, the word detection rate is 77.6% at 23.6 fa/kw/hr. 10.1% (411) of the units in the test data are unseen units, and 13.5% (80) of the words are missed because of the unseen units. This indicates that 59% of the errors are due to the unseen units. The system can be improved by training with the unseen units.

We also evaluate the system by using the points where the values of the spectral transition measure are peaks rather than valleys for the segmentation boundaries, to show that using stationary points is more effective. 5339 units are defined and 48.3% of a total of 5710 units are classified correctly. Figure 3 shows the comparison of the two cases. The result shows that by using the valley of the spectral transition we can achieve a detection performance rate 50 % higher than that by using the peaks.

The system is compared with dynamic time warping [5]

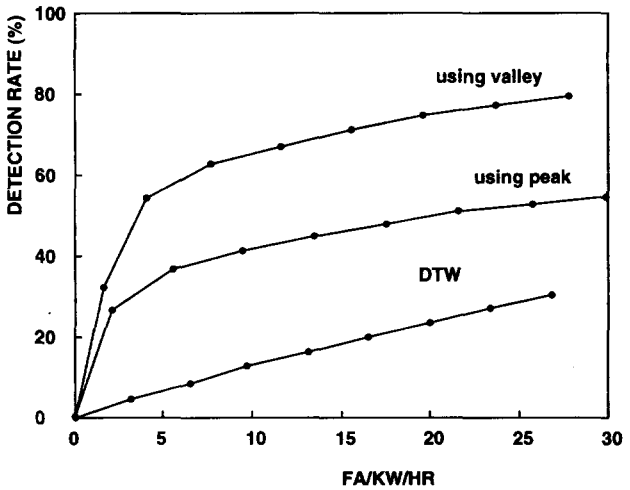


Figure 3. Word detection result (ROC curve)

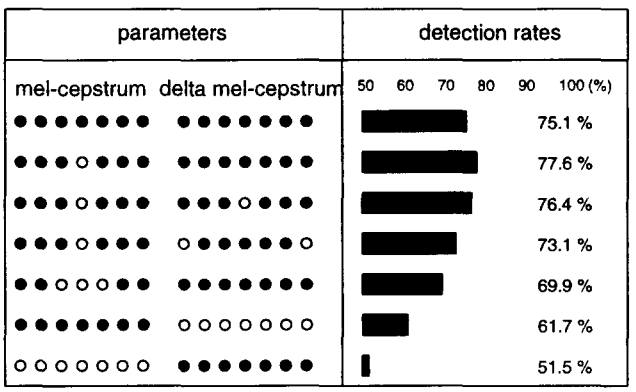


Figure 4. Recognition performances with various input configurations

where words are basic units. All the frames in the test sentences are considered as start points, and the frames from 1/2 to 3/2 of the length of a reference pattern from the start points are considered as end points. Six utterances of each word are used as references. Figure 3 shows that the experiment using dynamic time warping gives much lower performance than those using the proposed neural networks.

The units defined in this research have two parts, stationary and transition. To show the effect of each parameter on each part of the unit, we change the input range of each parameter and evaluate the performance. Figure 4 shows the different configurations of the input vectors and the recognition results. The seven circles in a row indicate the seven vectors taken from each segment, and the black circles represent inputs to the network. As a result, mel-cepstral coefficients at the center of a unit, where the spectral transition peaks, have a negative effect on the recognition performance. However, delta mel-cepstral coefficients are effective at the stationary points as well as at the transition points.

In the simulation, the units are pre-selected by using only the two vectors at the ends of the segments, because it takes too much time to evaluate all the activation values of the acoustic unit layer. The number of neurons in the acoustic unit layer that need full calculation can be reduced to about 10% of the total number of neurons, with the word detection rate unchanged.

Table 2 shows the execution time of the system, when it

Table 2. Execution time

vocabulary		520 words	1000 words
training data	data	2569 sec (43 min)	6870 sec (114 min)
	cpu time	1802 sec (18 min)	4260 sec (71 min)
		0.7 real time	0.6 real time
test data	data	429 sec (7 min)	1127 sec (19 min)
	cpu time	437 sec (7 min)	1868 sec (31 min)
		1.0 real time	1.6 real time

is simulated on an Ultra Sparc Workstation. As the table shows, the recognition time in 520 words task is 437 seconds for the speech data of 429 seconds long, so we can say that the system works in real time for the task. In 1000 words task, the system required 1.6 times of the real time.

The training time for both the tasks is less than real time, because the training process is *incremental learning*, that is a new class is added to the system only when there are errors during the test with the training data. In the beginning of the training process, the number of word classes and unit classes is small, so it takes only a small amount of time for testing. The number of classes increases as the training speech data is added in the training process.

4. CONCLUSIONS

We have presented a neural network for speaker-independent continuous speech recognition based on non-uniform units. The network segments the utterance into acoustic segments, classifies the segments into unit classes, and detects words. All the functions are implemented by neural network modules with simple structures. Experimental results demonstrate that using the stationary part of the speech as a segmentation boundary is more effective than using the transition part. Using different parameters for stationary and transition parts of the unit is also shown to be effective. The system can be trained and can recognize continuous speech in real time by virtue of the simple structure.

REFERENCES

[1] G. Zavaliagkos, Y. Zhao, Ro Schwartz and J. Makhoul, "A hybrid segmental neural net/hidden Markov model system for continuous speech recognition," IEEE Transactions on speech and audio processing, vol. 2, No. 1, Part II, 151-160, January 1994

[2] A. Waibel, T. Hanazawa, G.E. Hinton, K.Shikano and K.J.Lang, "Phoneme recognition using time-delay neural networks," IEEE Transactions on Acoustic, Speech, and Signal Processing, Vol. 37(3), pp.328-339, March 1989

[3] Ha-Jin Yu, Yung-Hwan Oh, "A Neural Network using Acoustic Sub-word units for Continuous Speech Recognition" ICSLP96, pp.506-509, October 1996

[4] S. Furui, "On the Role of Spectral Transition for Speech Perception," Journal of Acoustic Society of America 80(4), pp. 1016-1025, 1986

[5] Harvey F. Silverman and David P. Morgan, "The application of dynamic programming to connected speech recognition," IEEE ASSP Magazine, pp. 6-25, July 1990