Proceedings of 1993 International Joint Conference on Neural Networks

Perception of Speech Signals Using Self-Organization on Linear Neuron Array

Cheng-Yuan Liou and Chwan-Yi Shiah
Department of Computer Science and Information Engineering
National Taiwan University

Correspondence address : Cheng-Yuan Liou, Department of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan, 10764, R.O.C. Tel.8862 3630231 ext. 3229, FAX. 8862 362 8167, e-mail: cyliou@csie.ntu.ed.tw

Keywords : speech recognition, neural network, self-organization, chinese speech recognition, perception.

Abstract :

A continuous speech recognition system with finite set of chinese words is devised for selected applications. With proper design of the self-organizing map for the speech signals, the precedence relations among the spectral patterns within a token period can be preserved by the topology preservations and the serious nonlinear time warping can thus be overcome. The one dimensional hierarchical relations among the sequential spectral patterns are able to be represented by the topology map developed on the linear array of neurons. We then devise two kinds of perception energies based on the trained map. One of the energies is dervied from properly fitting a precedence curve on the sequential excitation patterns of the map during a whole word period. The other energy is obtained from the accumulation of total excitations on the map during a word period. Thresholds for the perception energies are then designed experimentally. A set of 1309 linear array maps are used for representing the total 1309 standard chinese word pronunciations. Each linear array contains 100 equally spaced and linearly ordered neurons. A verification of the system on a personal computer with a modern DSP board has been performed and the result was quite satisfactory.

I. Introduction

Much effort has been expended in attempts to quantize feature parameters of speeches to accomplish the recognition tasks. Speech features have been estimated with varying degrees of success. The speech recognition is still really an unsolved problem. The current systems are just barely reaching an acceptable level of performance for selected applications, but are still woefully inadequate compared to people. Some of this effort has been motivated by recent active research and development of continuous speech recognition system using neural networks models [1,2].

The Hidden Markov Model [3,4] is a potential technique to model sequential relations of speech patterns in many designs such as [4,5] for English and [6] for Chinese. The nonlinear time warping problem [7] limits most designs for real applications. This problem is a little bit relieved by introducing a time delay variation in the vector quantization domain, such as [2]. Inspired by the success of using neural networks, such as NETtalk [5], time delay neural network (TDNN) [2], and neural phonetic typewriter [1], we attempt to explore the neural network models to build a chinese continuous speech recognition system without the nonlinear time warping limitations. Since the serious nonlinear time warping [7] and large whole phoneme warping exist frequently in continuous speeches, the number of various training patterns will increase vastly. And the large warping will drive the TDNN to an unexpected results. We devise a linear neuron array and apply the formal self-organization learning algorithm on the neurons. We can keep the precedence relation among patterns through topology prevervation. This is because the neighboring spectral patterns will have most similar structures. And the linear topology will link the patterns through their similarities. The exact time warping between patterns is not improtant in this precedence topology map. With some specialities of Chinese pronunciation, we will investigate the possibility of using this self-organization model to accomplish the continuous chinese speech recognition task in the following sections. We will depict the designed neural network model and preparation of speech signals plus their preprocession techniques in the next section. In section III, we will provide simulation evidences and recognition techniques to display the performance. Finally we will draw the conclusions in the last section.

II. Neural network model and preprocessing procedures

We will present the designed self-organization model and then discuss the preprocessing techniques in this section. To save the content we will use an example to illustrate the our method.

The model contain 1309 linear neural arrays. Each array is used for a different single chinese word. There are total 1309 standard different word pronunciations. These 1309 words compose the whole standard chinese speeches. We skip some detailed variations of pronunciations to speed the system and simplify the contents. Each linear neuron array contains 100 neurons which is linearly ordered and equally spaced with unit intervals. Each neuron has 15 weighting synapses. A linear neuron array is shown in Fig.1.

The neuron plus its 15 synapses serve as a basic unit. This basic unit in the array computes the weighted sum of its 15 inputs and then passes this sum through a sigmoid function [8].

We will denote the 15 weights of the ith neuron as w_{ij}, where $i = 1 \sim 100, j = 1 \sim 15$. The response (excitation) of the ith neuron to its input pattern $\underline{X} = [x_1, ..., x_{15}]^T$ is Y_i, where $Y_i = f\left(\sum_{j=1}^{15} x_j \times w_{ij}\right)$ and f is a sigmoid function. In all our method we use a linear f function to simplify the computation. These weights w_{ij} for a linear neuron array can be determined by the training algorithm as in [8] using a set of training patterns. The training parameters and algorithm are the same in [8]. There are 1309 training pattern sets. Each set is used for a different neuron array. In our design, each array represents a single word pronunciation.

The training set for each word pronunciation is obtained by the following procedures. Each word is pronounced repeatedly 100 times by the same speaker. We collect the sequence of preprocessed spectral envelope patterns.

The preparation of the spectral envelope patterns is similar to the method in [2]. Each pattern contains 15 normalized melscale coefficients. The overall pattern frame rate is 9ms. The sampling frequency is 8000Hz in our simulations. All spectral envelope patterns in the 100 tokens for each word are included in a training pattern set for one linear neuron array. All 130900 tokens are preprocessed in the same fashion.

The prepared training pattern set for each word is then used to train the 15 x 100 weights of a different linear neuron array by the self organization algorithm as in [8]. The training cycle is 500 which is selected experimently. For each cycle the all training patterns in the set are shuffled at random. Each training set contains 2500 to 5000 spectral patterns. The gain sequence $\alpha(t)$ is linearly decayed from 0.8 to 0 and the topological neighborhood $Nc(t)$ is linearly decayed from 50 to unit in the algorithm. The 1309 linear neuron arrays is ready to work after training. All the training parameters are selected experimentally in consideration of both saving computations and convergence accuracy. Fig.2(b) shows the trained pattern on a linear neural array using all 100 token templates of the word "#0-joan" . We plot the 100 x 15 trained weights of the 100 neurons from left to right in Fig.2(b). Comparing the Fig.2(a) and (b), we find (b) has an expanded form of (a). This expanded form is the key in our design.

III. Recognition method and Simulations

A 100 selected words for robot motion command will be used in our example to reduce the context length. The 100 words are listed in table I. The recognition is done by designing two perception energies on each neuron array. The first perception energy E_1 is the total excitations on each neuron array. The total excitations is obtained by accumulating the 100 neurons total excitations during a token period. Only the one third number of largest excitations during a whole token period are included in the accumulation of E_1 energy. Fig.3 shows the average E_1 of the 100 training tokens for each word when we input the 100 training tokens to its corresponding trained neuron array. When we input the training tokens to wrong neuron arrays this E_1 is much smaller. Fig.3 also shows the average of E_1 when wrong input is feed in.

The other perception energy is designed by the inverse of the fitting error. This fitting error is obtained by fitting a second order polynomial [9] to the sequential excitation patterns of the neuron array during a token period. Fig.4 shows the sequential excitation pattern when a training token for the word "#0-joan" is inputted to its corresponding neuron array. This fitting can be done by linear least square method except each data is weighted by its excitation strength. The strengths of excitations on each neuron are denoted by properly filled squares in the figure. Only the largest one third number of the excitations on the 100 neurons during whole token period are used in the fitting.

The ranges of curvature and slope of the curve is limited reasonably and experimentally in the fitting algorithm. After fitting the mean square of fitting errors is computed. The inverse of this fitting error is used as the second perception energy E_2.

The thresholds for the first energy E_1 is then experimentally designed with Fig.3. The middle value between the correct average and the wrong average is used as threshold of E_1 for each linear array or each word. The recognition is finished when a threshold is satisfied. If there are several thresholds are satisfied, the E_2 are further used to select the best fitting one which has largest E_2 value. For continuous speech, each token time can be easily identified by simultaneous monitoring and thresholding the appearance of accumulations of the 1309 E_1 values during a rough token period. Fig.5 shows the appearance of the partial E_1 during a sentence (#23-shiang #3-tzuoo #0-joan) means "turn left". The normalization factors are obtained from the all melscale coefficients within a successive 30 pattern frames. These factors are used to

normalize the melscale coefficients of the middle 10 pattern frames. The partial E_1 of the total excitations on the 100 neurons during a successive 5 pattern frames is then obtained for every 45ms (5 × 9ms). Fig.5 shows the cascade (or sequential) plot of this partial E_1. The accumulation of this partial E_1 every 45ms is then tested by the E_1 threshold. The recognition is done when a threshold is reached during a rough token period. The former word's E_1 is reset to zero when next threshold is reached. When multiple thresholds are satisfied during a rough token period, E_2 is then used for further identification.

IV. Conclusions

We propose a self-organization model to overcome the serious nonlinear time warping difficults which encounter in many existing systems. This time warping causes the increasing of the number of various training patterns vastly and large warping causes unexpected results.

The linear neuron array can keep the precedence topology relations among spectral patterns within a token. In particular, we devise two perception energies which accumulate the excitation informations on the neuron map during a token period in different way. The correct recognition is obtained when E_1 is satisfied and reconfirmed by E_2. The developed system performed in real time satisfactorily on a modern DSP processor based machine.

References

[1] Teuvo Kohonen, "The neural phonetic typewriter," *IEEE Computer*, vol.21, no.3, March 1988, pp.11-24.

[2] A.Waibel, T.Hanazawa, G.Hinton, K.Shikano, and K.J.Lang, "Phoneme recognition using time-delay neural networks, " *IEEE Trans. on ASSP*, vol.37, no.3, March 1989, pp.328 -339.

[3] A.B.Poritz, "Hidden Markov Models : A guided tour," in *Proc. IEEE Int. Conf. ASSP*, Apr.1988, pp. 7-13.

[4] Kai-Fu Lee, Hsiao-Wuen Hon, and Raj Reddy, "An overview of the SPHINX speech recognition system," in *IEEE Trans. on ASSP*, vol.38, no.1, January 1990 pp.35-45.

[5] T.J.Sejnowski and C.R.Rosenberg, " NETtalk : A parallel network that learns to read aloud," *Tech. Rep. JHU / EECS - 86 / 01*, Johns Hopkins Univ., June 1986.

[6] L.S.Lee, C.Y.Tseng, and M.Ouh-Young, "The synthesis rules in a chinese text-to-speech system, "*IEEE Trans. on Acoustics, Speech, and Signal Processing*, vol.37, no.9, Sept. 1989, page 1309 - 1320.

[7] A.Waibel and B.Yegnanarayana, "Comparative study of nonlinear time warping techniques in isolated word speech recognition systems ,"*Tech. Rep.* Carnegie-Mellon Univ., June 1981.

[8] Teuvo Kohonen,"Self-Organization and Associative Memory, " second edition Springer-Verlay,1988.

[9] Philip R.Bevington,"Data reduction and error analysis for the physical sciences,"1969.

Ø	1	2	3	4	5	6	7	8	9	1Ø	11	12	13	14	15	16	17	18	19
轉	上	下	左	右	前	後	一	二	三	四	五	六	七	八	九	停	聽	看	開
joan	shang	shiah	tzuoo	yow	chyan	how	i	ell	san	tsyh	wuu	liow	chi	ba	jeou	tyng	ting	kann	kai
2Ø	21	22	23	24	25	26	27	28	29	3Ø	31	32	33	34	35	36	37	38	39
關	倒	退	向	往	拾	度	百	仟	萬	公	尺	執	行	進	入	找	尋	檔	快
guan	daw	tuey	shiang	woang	shyr	duh	pae	chian	wann	gong	chyy	jyr	shyng	jinn	ruh	jao	shyn	daang	kuay
4Ø	41	42	43	44	45	46	47	48	49	5Ø	51	52	53	54	55	56	57	58	59
慢	點	案	紅	跟	我	去	來	重	覆	遍	線	黃	殺	躲	避	目	的	再	音
mann	dean	ann	horng	gen	woo	chiuh	lai	chorng	fuh	biann	liuh	hwang	sha	duoo	bih	muh	dih	tzay	in
6Ø	61	62	63	64	65	66	67	68	69	7Ø	71	72	73	74	75	76	77	78	79
波	探	測	外	內	線	大	小	超	速	記	住	指	令	命	中	順	序	環	投
bo	tann	tseh	way	ney	shiann	dah	sheau	chau	suh	jih	juh	jyy	linn	ming	jong	shuenn	shiuh	hwan	tour
8Ø	81	82	83	84	85	86	87	88	89	9Ø	91	92	93	94	95	96	97	98	99
等	叫	名	資	料	唔	不	懂	撞	牆	桌	鎮	傳	標	方	圓	動	作	始	檫
deeng	jyau	ming	tzy	liaw	niann	buh	doong	juang	chyang	juo	suoo	chwan	biau	fang	yuan	tong	tzuoh	shyy	tsa

Table1 The 100 words plus their Kwoyeu Romatzyh pronunciation used in our robot command example.

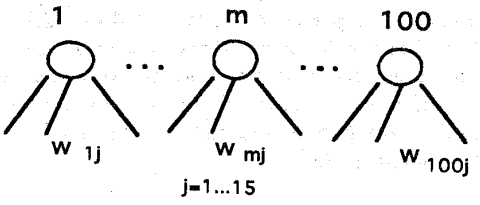

Fig.1 The linear neuron array for each word. Neurons are linearly ordered with equal unit intervals.

Time elapse⟶(9ms interval)

15 normalized melscale coefficients (spectral pattern)

Fig.2(a) templates within one token of the word "#0-joan".

Neuron number from 0 ⟶ 99

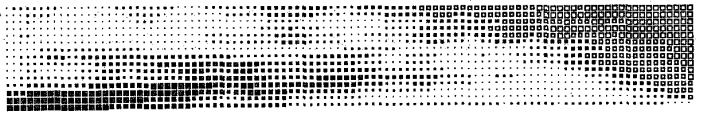

Fig.2(b) The trained linear neuron array using the all 100 token templates of the word "#0-joan". We plot the 15 weights for each neuron from left to right. (b) has an expanded form of (a).

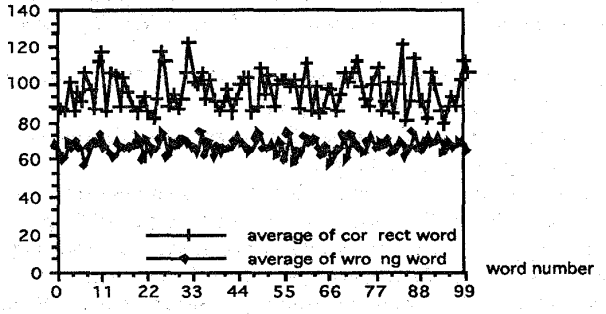

Fig.3 The average of E_1 when we input correct and incorrect templates during a token period.

Neuron number from 0 ⟶ 99

time elapse ⇓

9ms interval

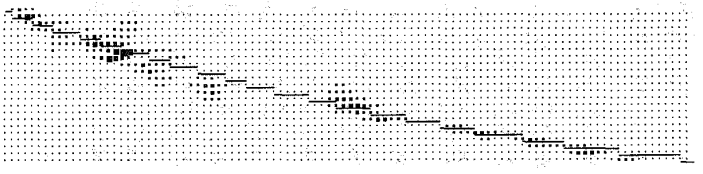

Fig.4 Fitting a second order polynomial to the excitation pattern on the trained linear neuron array using the correct word "#0-joan" during a token period. In this case $cx^2+bx+a=y, a=-0.709, b=-0.345, c=0.001$.

Word number from 0 ⟶ 99

time elapse ⇓

45ms interval

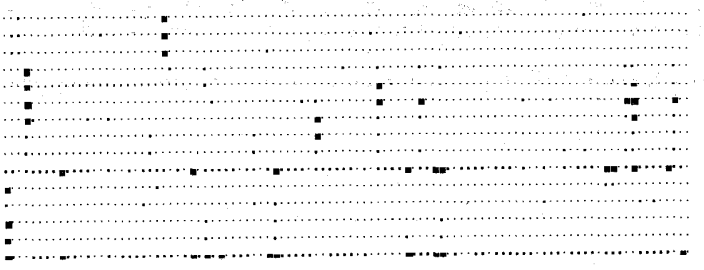

Fig.5 The sequence of partial E_1 during every 45ms ($5 \times 9ms$) for every one of the 100 linear neuron arrays. The pronounced robot command "#23-shiang #3-tzuoo #0-joan" can be clearly identified by thresholding of the accumulation of partial E_1 during a rough token period.