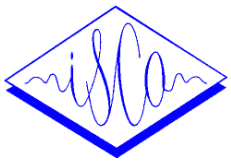

4th European Conference on
Speech Communication and Technology
EUROSPEECH '95
Madrid, Spain, September 18-21, 1995

RECURRENT NEURAL PREDICTION MODELS FOR SPEECH RECOGNITION

KyungMin NA, JeKwan RYU, Dong-Il CHANG, Soo-Ik CHAE and SouGuil ANN

Dept. of Electronics Engr., Seoul National University, KOREA
TEL: +82-2-880-7279, FAX: +82-2-882-3906
E-mail: CTLAB@KRSNUCC1.BITNET

ABSTRACT

This paper proposes recurrent neural prediction models (RNPM) for speech recognition which are recurrent neural networks trained as a nonlinear predictor of speech signals. Among various recurrent architectures, two well-known recurrent neural networks are tested here. The RNPM does not require any time alignment algorithm, which allows considerable reduction of computation time in recognition phase. Experiments on Korean digit recognition have shown that the performance of RNPM is a little better than that of other predictive neural networks.

1. INTRODUCTION

Though various neural networks have been successfully applied to many real world tasks, most of them are not suitable for modeling time-varying signals such as speech because their network parameters are fixed with respect to time. In recent years, various predictive neural networks such as hidden control neural network (HCNN), neural prediction model (NPM) and linked predictive neural network (LPNN) have been proposed to cope with this problem [1]-[3]. The predictive neural networks are basically a sequence of multilayer perceptron (MLP)-based nonlinear predictors, and dynamic programming techniques determine transitions between MLP predictors.

The HCNN was proposed by Levin [1] where hidden control input signals modulate the mapping of an MLP predictor according to time with the parameters of the MLP predictor fixed. Consider an N finite-state HCNN, and let $\mathbf{c}(t)$ be a hidden control input signal at time t which can take its value from a finite set $\{\mathbf{C}_1,\cdots,\mathbf{C}_N\}$. Then, its mapping function $F(\cdot)$ can be switched with time into one of N mapping functions $\{F_1(\cdot),\cdots,F_N(\cdot)\}$ by a proper choice of $\mathbf{c}(t)$ as follows:

$$\begin{aligned}\hat{\mathbf{x}}(t) &= F\big(\mathbf{x}(t-1),\cdots,\mathbf{x}(t-p),\mathbf{W},\mathbf{c}(t)=\mathbf{C}_i\big)\\ &= F_i\big(\mathbf{x}(t-1),\cdots,\mathbf{x}(t-p),\mathbf{W}\big),\end{aligned} \quad (1)$$

where $\hat{\mathbf{x}}(t)$ is a predicted output vector at time t, $\mathbf{x}(t-p)$ is the p-delayed vector of an input vector $\mathbf{x}(t)$ at time t, and $\mathbf{W}$ is a fixed network parameter matrix.

The NPM and LPNN were suggested by Iso and Watanabe [2], and by Tebelskis and Waibel [3] in which the mapping of an MLP predictor is changed according to time by changing the whole parameters of the MLP predictor. Consider an N finite-state LPNN or NPM, and let $\mathbf{w}(t)$ be a network parameter matrix at time t which can be one of N possible matrices $\mathbf{W}_1,\cdots,\mathbf{W}_N$. Then, $F(\cdot)$ can be changed with time into one of N mapping functions $\{F_1(\cdot),\cdots,F_N(\cdot)\}$ by a proper choice of $\mathbf{w}(t)$ as follows:

$$\begin{aligned}\hat{\mathbf{x}}(t) &= F\big(\mathbf{x}(t-1),\cdots,\mathbf{x}(t-p),\mathbf{w}(t)=\mathbf{W}_i\big)\\ &= F_i\big(\mathbf{x}(t-1),\cdots,\mathbf{x}(t-p)\big).\end{aligned} \quad (2)$$

Though the predictive neural networks have shown high performances in various tasks, they suffer from computational burden for dynamic programming techniques.

An alternative neural network approach to modeling time-varying signals is various recurrent neural networks (RNN), which can deal with time-varying input-output relations [4]-[8]. In general, RNNs are trained as a pattern discriminator with a proper time-varying target function [4]. Watrous and Shastri proposed a Gaussian function as time-

varying target function [4]. Elman presented a simple recurrent neural network in which delayed hidden unit outputs are fed back as supplementary input units [5]. Bourlard and Wellekens gave a brief description on various recurrent neural architectures for modeling speech dynamics [6]. Williams and Zipser provided the exact form of a gradient descent-based learning algorithm for completely recurrent neural networks running continually [7]. Lee *et al.* simplified a learning algorithm of Elman's recurrent neural networks for a practical use with application to speech recognition tasks [8]. Although RNN has been successful in limited applications such as phoneme recognition, it is difficult to find an appropriate time-varying target function and to apply it to various speech recognition tasks.

In this paper, we propose a new kind of predictive neural networks, recurrent neural prediction models (RNPM). The RNPM is layered recurrent neural networks trained as a nonlinear predictor of speech feature vectors. Training algorithms are derived by applying a gradient descent method to an accumulated prediction error. In recognition phase, the model that scores the smallest accumulated prediction error is selected as a recognition result. With all merits of other predictive neural networks [2], RNPM does not require any time alignment algorithm which reduces computation time considerably in recognition phase.

Section 2 presents two well-known RNN-based RNPM and their training algorithms. Section 3 gives experimental results, and finally Section 4 contains conclusions.

2. RECURRENT NEURAL PREDICTION MODELS

Unlike RNNs trained as a discriminator, RNPM models each word class independently, and does not need a time-varying target function since it is trained as a predictor. Moreover, since recurrent connections provide a short-term memory for absorbing temporal variations, RNPM does not adopt dynamic programming processes for time alignment, which allows to significantly alleviate computational burden in recognition phase.

Recurrent input signals of RNPM modulate its mapping with time just as hidden control input signals of HCNN do. Let $\mathbf{r}(t)$ be a recurrent input signal at time t, the mapping function $F(\cdot)$ of PRNN is time-varying as follows:

$$\hat{\mathbf{x}}(t) = F\big(\mathbf{x}(t-1),\cdots,\mathbf{x}(t-p),\mathbf{r}(t-1),\mathbf{W}\big). \quad (3)$$

For speech recognition, HCNN, LPNN and NPM are designed as a left-to-right finite-state model with a forced trajectory. However, RNPM is a more flexible finite-state model since recurrent input signals are generated from internal representation of the speech sequential information.

Though there can be many recurrent neural network architectures, we consider here two types of RNNs, a single-layer RNN-based RNPM (SRNPM) [7] and Elman's RNN-based RNPM (ERNPM) [5], [8] for convenience.

2.1 Single-layer RNN-based RNPM (SRNPM)

Since RNPM is trained as a nonlinear predictor, the output units contain a linear activation function while the hidden units contain a sigmoid activation function. To satisfy a stable condition in training, we remove the recurrent connections from the output units. The structure of SRNPM is given in Fig. 1.

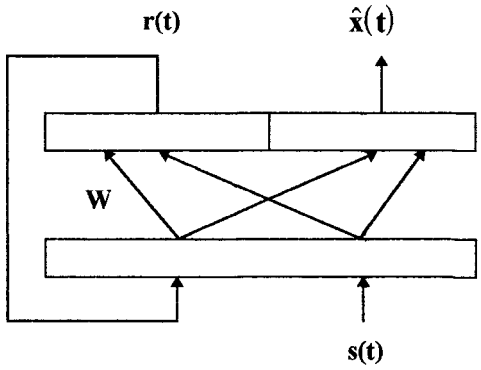

Figure 1. Structure of SRNPM.

The SRNPM outputs a predicted vector $\hat{\mathbf{x}}(t)$ using $\mathbf{x}(t-p),\cdots,\mathbf{x}(t-1)$, and $\mathbf{r}(t-1)$ as input vectors. In Fig. 1, $\mathbf{s}(t)$ denotes a concatenation of $\mathbf{x}(t-p),\cdots,\mathbf{x}(t-1)$. The input-output relation is given as

$$[\mathbf{r}(t),\hat{\mathbf{x}}(t)] = [\mathbf{r}(t-1),\mathbf{s}(t)]\mathbf{W}, \quad (4)$$

where [·] denotes the concatenation operation.

The accumulated prediction error is given as follows:

$$D = \sum_t ||\hat{\mathbf{x}}(t) - \mathbf{x}(t)||^2 . \quad (5)$$

The training algorithm for SRNPM is derived by applying gradient descent method to (4) just as Williams and Zipser did [7]. As a result, we modify the real-time recurrent learning (RTRL) algorithm for SRNPM [7].

2.2 Elman's RNN-based RNPM (ERNPM)

We consider here three-layer, Elman-type recurrent neural networks only for ERNPM. The structure of ERNPM is given in Fig. 2.

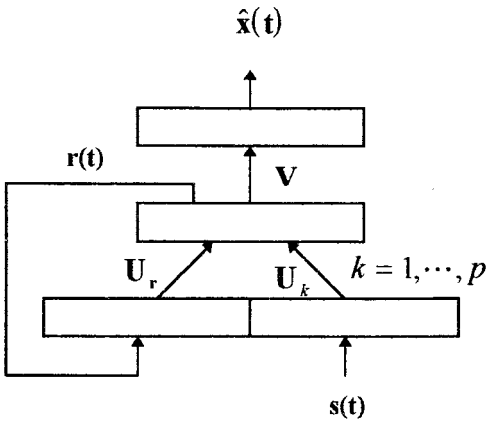

Figure 2. Structure of ERNPM.

Let an output vector of hidden units at time t be $\mathbf{h}(t)$, and $\mathbf{U}_1, \cdots, \mathbf{U}_p, \mathbf{U}_r, \mathbf{V}$ be weight matrices. Then, the ERNPM can be described by the following vector equations:

$$\mathbf{h}(t) = \mathbf{f}\left(\sum_{k=1}^{p} \mathbf{x}(t-k)\mathbf{U}_k + \mathbf{r}(t-1)\mathbf{U}_r \right), \quad (6)$$

$$\hat{\mathbf{x}}(t) = \mathbf{h}(t)\mathbf{V}, \quad (7)$$

where the vector function $\mathbf{f}(\mathbf{a})$ gives a vector by applying a sigmoid function to each element of $\mathbf{a}$, and $\mathbf{r}(t-1) = \mathbf{h}(t-1)$.

The training algorithm is derived by applying gradient descent method to this accumulated prediction error (5), and is simplified for a practical use just as Lee *et al.* did [8]. In recognition phase, the model that scores the smallest accumulated prediction error is selected as a recognition result both for SRNPM and ERNPM.

3. SPEECH RECOGNITION EXPERIMENTS

We have evaluated RNPM on a database of isolated Korean digit with two versions of each digit pronounced by 25 male speakers. The Korean digits consist of monosyllables, and contain confusible words such as {3 ("sam"), 4 ("sa")}, {1 ("il"), 2 ("I"), 7 ("chil")} and {0 ("gong"), 5 ("oh"), 9 ("gu")}.

As an input feature vector for each frame, 10 LPC-cepstral coefficients (excluding the 0-th order coefficient) were derived. Among 500 data, 150 data from the first version of 15 speakers (Data A) were used for training while 150 data from the second version of the same 15 speakers (Data B) were used for closed test, and 200 data from the remaining two versions of 10 speakers (Data C) were used for open test.

The order of predictor p is chosen as 3, and total number of training epochs is 1000. The learning rate is 0.001, and the number of hidden units is 5 for SRNPM and 11 for ERNPM. For comparison, NPM and HCNN with 8 states and the learning rate of 0.001 were tested. The results are given in Table 1. ERNPM has shown a little higher recognition rate than NPM and HCNN. All recognition errors belonged to the confusible word set, some of which may be corrected by discriminative training algorithms [9]-[10].

4. CONCLUSIONS

In this paper, we proposed a new speech recognition model, recurrent neural prediction models (RNPM) which are recurrent neural networks trained as a nonlinear predictor of speech feature vectors. Two types of RNN were designed as RNPMs, and their structures and training algorithms were described.

The RNPM has all merits that other predictive neural networks have, and does not require any time alignment algorithm such as dynamic programming and Viterbi algorithm. Therefore, RNPM can reduce computational burden in recognition phase in comparison with other predictive neural networks.

Though experimental results have shown the effectiveness of the RNPM, it suffers from nondiscriminative training algorithm since it models each word class independently. To improve the performance of the RNPM, we are developing the generalized probabilistic descent (GPD)-based discriminative training algorithm [9]-[10].

Another possibility of improving the performance of RNPM-based speech recognizers is to find a new efficient recurrent architecture which can model both the temporal variation and spectral variation of speech signals. Other applications such as hand-written character recognition will be possible, too. Further study on these may be interesting.

Table 1. Recognition results.

MODEL	DATA A	DATA B	DATA C
SRNPM	100.0 %	95.3 %	87.5 %
ERNPM	100.0 %	97.3 %	91.0 %
NPM	100.0 %	97.3 %	90.5 %
HCNN	100.0 %	96.0 %	89.0 %

References

[1] E. Levin, "Word recognition using hidden control neural architecture," *Proc. ICASSP-90*, pp. 433-436, 1990.

[2] J. Tebelskis and A. Waibel, "Large vocabulary recognition using linked predictive neural networks," *Proc. ICASSP-90*, pp. 437-440, 1990.

[3] K. Iso and T. Watanabe, "Speaker-independent word recognition using a neural prediction model," *Proc. ICASSP-90*, pp. 441-444, 1990.

[4] R. L. Watrous and L. Shastri, "Learning phonetic features using connectionist networks: an experiment in speech recognition," *Proc. IJCNN-87*, vol. 4, pp. 381-388, 1987.

[5] J. L. Elman, "Finding structure in time," *CRL Technical Report 8801*, University of California, 1988.

[6] H. Bourlard and C. J. Wellekens, "Speech dynamics and recurrent neural networks," *Proc. ICASSP-89*, pp. 33-36, 1989.

[7] R. J. Williams and D. Zipser, "A learning algorithm for continually running fully recurrent neural networks," *Neural Computation*, vol. 1, pp. 270-280, 1989.

[8] S. J. Lee, K. C. Kim, H. S. Yoon and J. W. Cho, "Application of fully recurrent neural networks for speech recognition," *Proc. ICASSP-91*, pp. 77-80, 1991.

[9] B. H. Juang and S. Katagiri, "Discriminative learning for minimum error classification," *IEEE Trans. Signal Processing*, vol. 40, no. 12, pp. 3043-3054, 1992.

[10] K. M. Na, J. Y. Rheem and S. G. Ann, "A discriminative training algorithm for predictive neural network models," *Proc. ISCAS-94*, pp. 431-434, 1994.