自然言語処理　77－6
（１９９０．５．１８）

Recognition and Generation of Natural Language with Recurrent Neural Network

Ryotaro Kamimura
Information Science Laboratory
Tokai University
1117 Kitakaname Hiratsuka
Kanagawa 259-12, Japan

Abstract

In the present paper, we present the experimental results regarding the generation and recognition of the natural language by using the recurrent neural network with temporal supervised learning algorithm, developed by Williams and Zipser. The results can be summarized by the following three points. First, the network could recognize and generate the training sequence of English sentences, if the appropriate number of hidden units are given. Second, the network could also reproduce infinitely the grammatical sentences above the training interval, if the interval is sufficiently long. Third, the network showed the difficulty in estimating rare letters and the letters, when they are situated at the first position of a word or at the first part of a new sentence.

リカレントニューラルネットワークによる自然言語の認識と生成

上村　龍太郎
東海大学情報処理研究教育施設

概要

本論文では、リカレントニューラルネットワークを用いた自然言語の認識と生成に関する実験結果の報告を行う。結果は、次の三点に要約できる。第一に、適切な数の隠れユニットが与えられれば、ネットワークは自然言語の認識を行うことができる。次に、訓練のためのデータが十分に長ければ、ネットワークは訓練用のデータの範囲を越えて、自動的に適格文を生成し続けることができる。第三に、ネットワークは、語頭、文頭、及び頻度の少ない語に関して、推定の際に大きな誤差を示す。

1 Introduction

In the present paper, we are concerned with the recognition and the generation of natural language by using the recurrent network with the temporal supervised learning algorithm, developed by Williams and Zipser [12].

Multiple attempts to apply the neural network to the understanding of natural language have been already made. For example, the mechanism of sentence processing or several other topics are fully discussed in D.E. Rumelhart *et al.*[11]. However, as the neural network models have been developed so as to deal with static phenomena, it is impossible for them to be applied to the time-changing phenomena directly. The natural language is certainly not static, but dynamic. When the static network is applied to time-changing phenomena, one of the problems is that the network architecture must be unfolded for any time so as to represent the time-changing property, as is discussed in [10]. Thus, the network must prepare a great number of units, corresponding to any time, which prevents the network from dealing with the long time-changing sequences.

Recently, several attempts, for example [1], [2], [7], and [12], have been made, which aim mainly at the time-changing phenomena. Regarding the application of the recurrent network to natural language processing, the simple recurrent network developed by Elman [2] has been reported to be considerably powerful. Using the simple recurrent network, Elman has succeeded in showing the meanings of notion of "word". He also has shown the problem of the discovery of lexical classes from the word order. Moreover, he has attempted to demonstrate the performance of the recurrent network concerning the reference of pronouns using c-command. However, the architecture of the recurrent network used by Elman has been severely restricted. Thus, the generative capacity of natural language can not be simulated by using Elman's simple recurrent network.

The temporal supervised learning algorithm(TSLA), developed by Williams and Zipser, is considerably general. This means that we need not impose any restriction upon the network architecture. Since the exact properties of natural language remain to be clarified, the generality of the architecture is necessary. Moreover, the network need not be unfolded in time, which enables the network to be used on real-time mode and simplifies the computational procedure.

In the temporal supervised learning algorithm, the activity x_i at time $t+1$ is defined by the equation: $x_i(t+1) = f(\sum_j w_{ij} x_j(t))$, where w_{ij} is the connections strength from jth unit to ith unit, and f is the logistic function. The network is composed of two kind of units: hidden and visible unit. The visible units have their target values. The network is trained to decrease the difference between target values and actual values at the visible units. For further details regarding the computational methods concerning the temporal supervised learning algorithm, see Kamimura[5].

Section 2 will be devoted to the examination of the experimental results regarding the recognition and generation of natural language. In Subsection 2.1, we will present the learning process of the neural network. Subsection 2.2 will be concerned with the automatic generation of the sequence above the training interval. The analysis of errors after finishing the learning will be discussed in Subsection 2.3. We will discuss three problems regarding the application of the recurrent neural network to the natural language in Subsection 2.4. Finally, we will summarize and conclude the present paper in Section 3.

2 Results and Discussion

2.1 Learning Process

In this section, we will present three sequences generated by the network at three stages of learning process so as to show how the network learns a target sequence. The network used in the experiments was fully connected and composed of 5 visible units and 15 hidden units. The length of a sequence was 120 letters. This sequence was encoded in the binary number system with 5 bits. The learning was considered finished when the Hamming distance between target and observed sequence was zero. The learning method used the variable learning rate, Minkowski-r power metric (r=1.5), which are explained in Kamimura[5]. All the computations were performed on SX-1. For some other minor additional methods used to accelerate the learning time, see [3].

Figure 1 represents the variation of the mean absolute errors as a function of the number of learning cycles(iterations). It took 21715 iterations (about 5 hours) to reach this final point. As can be see in the figure, the variation of errors remains large until the final point.

After 1000 learning cycles, we had a sequence below:

> MISWLERGI GLQVDIS MYVLISW HASLISWLERGI WLEREY GLQVL ISVMIWLISWLAREI WLEREI GL QVEISGM WLISWHASLMQSMISWL EREI WLAVEI GLQVEISW.

As can be seen from the regenerated sequence, we can not find any meaningful word in the sequence.

In Figure 2, the weight matrix is reproduced. We explain briefly the way how to make the figure. After that, we will turn to the discussion of the meaning of weight matrix. As just stated, the network was composed of 5 visible units and 15 hidden units. Thus, the weight matrix forms a 20 × 20 matrix. In the matrix, the first 15 × 15 matrix represents the connections among hidden units. The other part of the matrix is for the connections among the visible units and also the connections from the visible units. If an element of the matrix is greater than 2.0, then $\diamond$ symbol was assigned in the figure. The + symbol was assigned to an element whose value is less than -2.0.

Now, the weight matrix represented in Figure 2 shows especially a sparse state in the connections among hidden units, that is, in the first 15 × 15 matrix. On the other hand, the other part of the matrix, especially in the connections from any unit to five visible units showed a considerably dense state. This means that at the early stages of learning the network can learn the target sequence by using the connections concerning the visible units. This tendency has been extensively observed in our experiments.

After 17500 learning cycles (iterations), we had the sequence:

> MISS MARTHA MEACHAM KEPT TMDS *ET EHS EHCG*HT*DMYS FMXU* E EKC JERTXAETJTYE JEXIOWZE*S EKTXOMSSUHCWTI RSUHS UPKTHE* MARTIE STET UHE DAUXEMPQTHT EPQ.

At this stage, several English words could be seen in the sequence. For example, "MISS", "KEPT", and a proper noun "MARTHA MEACHAM".

Finally, the network could generate the original sequence perfectly. The regenerated sequence is

> MISS MARTHA MEACHAM KEPT THE LITTLE BAKERY ON THE CORNER THE ONE WHERE YOU GO UP THREE STEPS AND THE BELL SOUNDS WHEN YO

which is a perfect original sequence. At this stage, the connections among hidden units and those between hidden units and visible units are further activated, and saturated, as can be seen in Figure 3.

2.2 Automatic Generation

In the present section, we will reproduce a regenerated sequence out of the training section. The objective of this attempt is to assure that the network acquire really some kind of structure in the sequence. We will reproduce here four sequences with 30, 50, 120 letters in the training sequences respectively. The regenerated sequence out of training section will be emphasized.

When the length of training sequence was 30 letters, we obtained the following sequence:

> MISS MARTHA MEACHAM KEPT THE L*EPT THE LEPT THQ KGSLARDHAESHAHEIMSS MARTHA* KEQ MARTISSALAIUIG*DH*DSX EIEMISS MEPTYWLARTHA IEQ EAEI.*

In this case, the network was composed of 5 visible units and 6 hidden units. It took 3152 iterations to reach a final state. As can be seen in the above sequence, the network could not reproduce the grammatical sentences above the training interval. The only English word we can see is a word 'THE".

When the length of training sequence increased to 50 letters, we had one of the typical regenerated sequences below

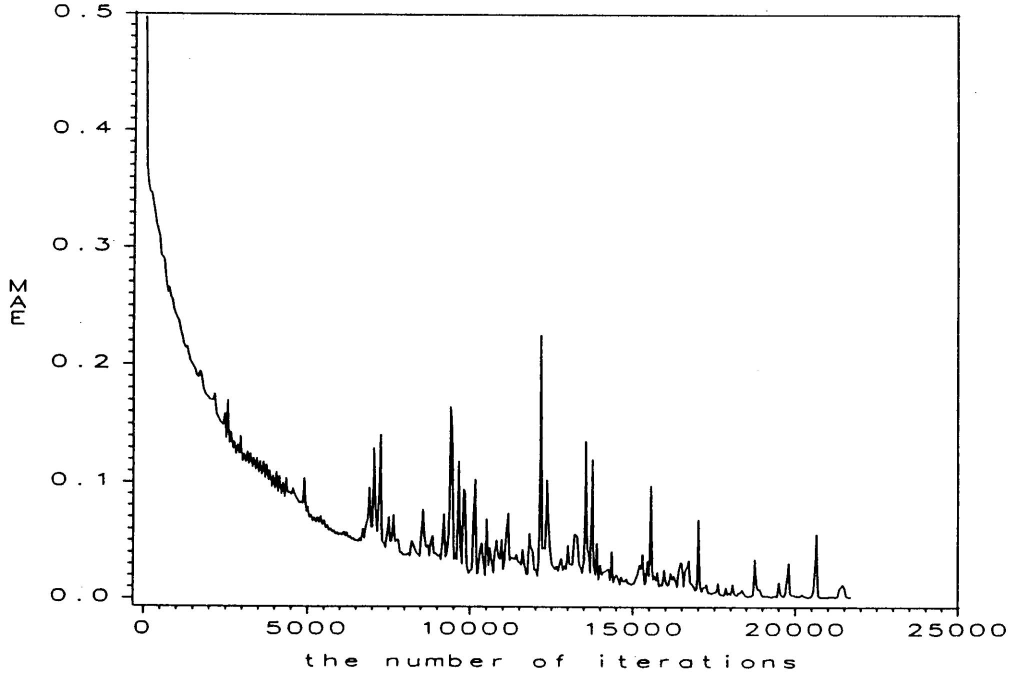

Figure 1: Mean absolute errors as a function of the number of learning cycles (iterations). The length of sequence was 120 letters.

+															◇				
																		+	
◇													+					+	+
								+								◇	+		
							+								+		◇		
																			+
							◇	+					◇			+	◇	+	
+						+	◇	◇					+		◇	+		+	◇
	◇		+	+		◇	+		◇				◇	◇		+	◇		
◇	◇	◇	◇		◇			+		◇	◇	+	+			◇	+	◇	
	+	+		+	+	+		◇	+	◇	+			+	+				◇
◇		+		+		+	◇	◇			◇	◇	◇		+		+		

Figure 2: Weight matrix obtained after 1000 learning cycles. Symbol + represents the inhibitory state and ◇ is for the activated state. For further details, see the text.

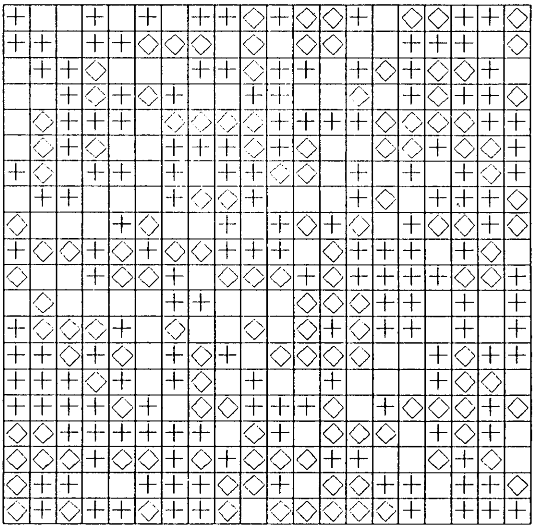

Figure 3: Weight matrix at the final stage of learning process.

> MISS MARTHA MEACHAM KEPT THE LITTLE BAKERY
> ON THE *LADPLMNIT*D V DHE LIURMAK KSXKOVYUK-*
> *ERT VNER*YELPTYT LESZMMVYQKERP NNDPLISNEPQ*
> *KORY ON THE LADPLMNIT*T D.*

The number of hidden units of the network increased to 10 and it took 1699 iteration to converge. In this case, we can not see any grammatical sentences. The preposition "ON" and the article "THE" were generated by the network. The network could sometimes produce a prefect English sequence at this stage. However, the state was unstable and extremely dependent on the initial values.

Finally, when the length of sequence increased to 120 letters, the generated sequence was

> MISS MARTHA MEACHAM KEPT THE LITTLE BAKERY
> ON THE CORNER THE ONE WHERE YOU GO UP THREE
> STEPS AND THE BELL SOUNDS WHEN YO*U GO UP*
> *THREE STEPS AND THE BELL SOUNDS WHEN YOU GO*
> *UP THREE STEPS AND THE BELL SOUNDS WHEN YOU*
> *GO UP THREE STEPS AND THE BELL SOUNDS.*

At this stage, the network could reproduce infinitely a grammatical sequence. As can be seen in the regenerated sequence, a sequence "WHEN YOU GO UP THREE STEPS AND THE BELL SOUNDS" could be infinitely regenerated. The network estimated the sequence following "WHEN YO" to be a sequence "GO UP THREE STEPS". The correlation among words can span to about ten words.

2.3 Analysis of Error

In this section, we will attempt to show where the network has difficulty to infer a coming letter. Our main results are simply summarized by three points. First, the network produced the high error values, when it attempted to estimate the new words. Second, the word with high frequency could be easily estimated. Third, the network showed the difficulty, when it attempted to infer the first part of a new sentence.

Figure 4 represents the sum of the absolute errors over all outputs of the visible units. The interval (time interval) ranged from the first letter "M" to the 120th letter "O". The original data are given in the

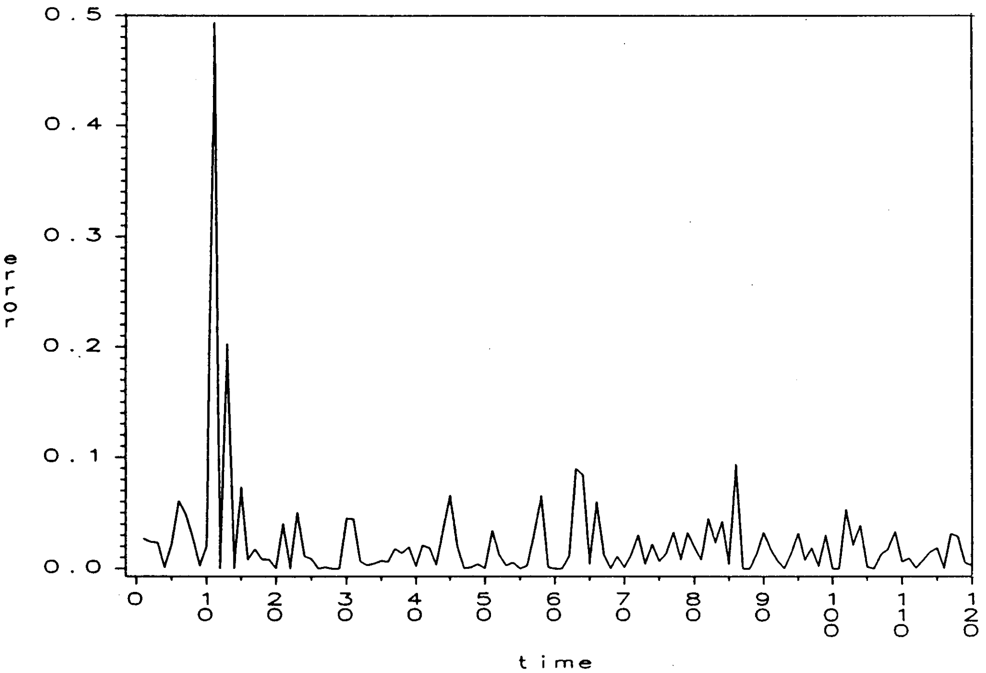

Figure 4: The sum of the absolute errors between the target letters and the outputs of five visible units as a function of time (the order of occurrence of letters).

subsection 2.1. As can be seen clearly, the first part of the sequence produced the extremely high error values. This is because the network must estimate the sequence of proper noun, that is, the name of a woman, which should not be used in the ordinary English sentences.

Figure 5 also represents the sum of the absolute errors between target letters and the outputs of visible units. However, the time interval is restricted to that ranging between 26th letter and 61th letter, so as to show clearly the behaviors of the network. Four salient peaks in the figure are observed at time 30, 45, 51, and 58 respectively. The time 30 and 51 are assigned to the first letter of a word "LITTLE" and "CORNER". We can also a high peak at the beginning of a word "BAKERY" at time 37. From these results, we can say that the network has difficulty in estimating the first letter of a new word. In other words, the information contained in the first letter is higher than any other letter in a word. This is completely reasonable from the information theoretical point of view. For the experimental results regarding the information contained in words from the information theoretical point of view, see Petrova *et al.* [8].

At time 58, we also see a peak of error for the first letter of a word "THE". This peak can be explained by the fact that a new construction begins at this time. On the other hand, the word "THE" at the beginning of the graph, that is, from the time 26 to 28, does not produce a large error. This is because the word "THE" is a word with the highest frequency in the sequence and the network can easily estimate the words with a high frequency.

2.4 Discussion

In the above sections, we have demonstrated that the recurrent neural networks can recognize and generate automatically the English sequence. In this section, we will discuss three problems: the computational requirement, the weight matrix, and the fundamental unit.

First, the computational requirement is enormously large, when the network size is relatively large and the sequence is long. We have observed that the number of hidden units necessary for the recognition of a given sequence is not in proportion to the length of sequence but remains small, compared with the length of the sequence. For further details regarding the minimum number of hidden units in the network, see [5].

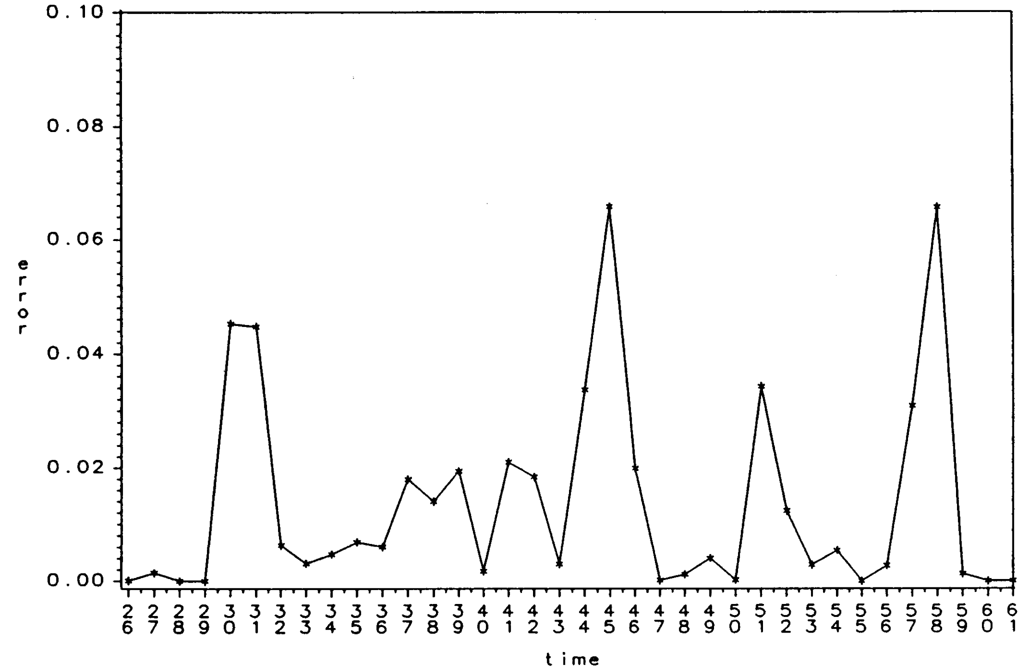

Figure 5: The sum of the absolute errors between target letters and the output of five visible units as a function of time. The time interval ranges from the 26th letter to 61th letter, that is, "THE LITTLE BAKERY ON THE CORNER THE ".

However, we need the new powerful algorithm which enables the network to converge significantly faster.

Second, the weight matrix has shown dauntingly an obscure characteristic. We have confirmed that the weight matrix shows an explicit regularity, when the network deals with simple tasks such as the recognition of sine functions. Since the information regarding the sequence should be compressed in the connections (weight matrix), a new method to clarify the meaning of weight matrix will be needed.

We have considered the letter as the fundamental units in the experiments to simplify the experiments and to evade the problems of the definition of the fundamental unit in the natural language. In making actually automatic generator of natural language, it is better to use so-called "word" as the fundamental unit. The experiments using the words or the grammatical units has been under investigation.

3 Conclusion

In the present paper, we have demonstrated the performance of recurrent neural network with temporal supervised learning algorithm, when applied to the recognition and generation of natural language. We have observed that the network can estimate the unknown sentences, when a sequence with an appropriate length is given. The recurrent network used by our experiment had fully connected general architecture. Thus, any restriction has not been imposed upon the network so as to incorporate syntactic or semantic constraints in the later studies. This generality has not been shared by other networks, for example, the Markov models [6] and the simple recurrent neural network [2]. Thus, we can expect that a new structure will emerge among the connections of the recurrent neural networks.

References

[1] A. Cleeremans, D. Servan-Schreiber, J. L. McClelland, "Finite state automata and simple recurrent networks," *Neural Computation*, Vol.1, No.3, pp.372-381, 1989.

[2] J. L. Elman, "Finding structure in time," *CRL Technical Report*, University of California, San Diego, No.8801, 1988.

[3] S. E. Fahlman, "Faster-learning variations on back-propagation: an empirical study," in *Proceedings of the 1988 Connectionist Models* , Carnegie Mellon University, pp. 38-51, 1988.

[4] S. J. Hanson and D. J. Burr, "Minkowski-r back-propagation: learning in connectionist models with non-Euclidian signals", in *Neural Information Processing Systems*, D. Z. Anderson, Ed., New York: American Institute of Physics, pp.348-357,1988.

[5] R. Kamimura, "Application of temporal supervised learning algorithm to generation of natural language," to appear in *Proceedings of International Joint Conference on Neural Newman*, San Diego, 1990.

[6] S. E. Levinson, L. R. Rabiner, and M. M. Sondhi, "An introduction to the application of the theory of probabilistic functions of a Markov process to automatic speech recognition," *The Bell System Technical Journal*, Vol.62, No.4, pp.1035-1074, 1983.

[7] B. A. Pearlmutter, "Learning state space trajectories in recurrent neural networks," *Neural Computation*, Vol.1, No.2, pp.263-269, 1989.

[8] N. Petrova, R. Piotrowski, and P. Guiraud, "Charactéristique informationelles du mot français, "Bulletin de la société de Linguistique de Paris", Vol.65, pp14-28, 1971.

[9] F. J. Pineda, "Recurrent backpropagation and the dynamical approach to adaptive neural computation," *Neural Computation*, Vol.1, No.2, pp. 161-172, 1989.

[10] D. E. Rumelhart, G. E. Hinton and R. J. Williams, "Learning internal representations by error propagation," in *Parallel distributed processing* . D. E. Rumelhart, J. L. McClelland and the PDP research group, Ed., Cambridge, Massachusetts: The MIT Press, Vol.1, pp.318-362, 1986.

[11] D. E. Rumelhart, G. E. Hinton and R. J. Williams, *Parallel distributed processing* . D. E. Rumelhart, J. L. McClelland and the PDP research group, Ed., Cambridge, Massachusetts: The MIT Press, Vol.2, 1986.

[12] R. J. Williams and D. Zipser, "Experimental analysis of the Real-time Recurrent Learning algorithm," *Connection Science*, Vol.1, No.1, pp.87-111,1989