TR-I-0123

English Word Recognition Using HMM Phone Concatenated Training and NETgram

HMM音韻連結学習とNETgramを用いた英単語音声の認識

Katsuteru MARUYAMA, Masami NAKAMURA,
Takeshi KAWABATA and Kiyohiro SHIKANO
丸山 活輝　中村 雅己　川端 豪　鹿野 清宏

1989 . 11

Abstract

For realizing an interpreting telephony system, an accurate word recognition system is necessary. Because it is difficult to recognize English words using only their acoustical characteristics, an accurate word recognition system needs certain linguistic information. The NETgram, which is trained using Brown Corpus is applied to HMM English word recognition. NETgram can correct word recognition errors effectively. NETgram improved the word recognition rate from 81.0 to 86.9%. The performance is improved over that of traditional statistical trigram models.

ATR Interpreting Telephony Research Laboratories
ATR 自動翻訳電話研究所

Contents

1. Introduction 1

2. English Word Recognition
   Using HMM Phone Concatenated Training 2

3. Word Category Prediction Neural Network (NETgram) 5

4. Applying the NETgram to HMM English Word Recognition 10

5. Experiments 13

6. Results 13

7. Conclusion 16

References 18

List of Figure

Fig.1 HMM phone concatenated training 3

Fig.2 Word recognition by concatenating HMM phone models 4

Fig.3 Word category prediction using Brown Corpus text data 5

Fig.4 Basic bigram network for word category prediction 6

Fig.5 NETgram for word category prediction 7

Fig.6 How to train NETgram(Trigram model) 9

Fig.7 Applying the NETgram to HMM English word recognition 12

List of Table

Table 1 Word recognition rate (%) 14

Table 2 Examples of the improvement
(The numbers of recognition error) 15

1. Introduction

For realizing an interpreting telephony system, an accurate word recognition system is necessary. Because it is difficult to recognize English words using only their acoustical characteristics, an accurate word recognition system needs certain linguistic information.

There are statistical methods for predicting word categories using the appearance probabilities of the following word to correct word recognition errors in text[1][2]. However, traditional statistical approaches require huge training samples to estimate the probabilities of word sequence and considerable memory capacity to process the probabilities. Additionally, it is difficult to predict unseen data.

To solve these problems, the NETgram(neural network for word category prediction) was proposed[3]. The NETgram was applied to HMM English word recognition resulting in an improvement of its recognition performance. English phone models for HMM were trained using the forward-backward algorithm according to the pronunciation symbols of training data.

To compare the performances of NETgram and traditional statistical model, an experiment was carried out using the traditional statistical model.

2. English Word Recognition Using HMM Phone Concatenated Training

Improved HMM training was applied to English phone modeling without phone labeling, where the system concatenates phone models and finds their boundaries using the forward-backward algorithm according to the pronunciation symbols of training speech[4].

Figure 1 shows the concept of HMM phone concatenated training. Using this method, 57 phone models containing a silence model are trained. Word models are made by concatenating phone models according to word sequence of the training data using a word-phone table in which the word pronunciation symbols are recorded. At this time, the silence models are joined to automatically detect the first and last points of speech. After training HMM parameters using the forward-backward algorithm, the word model is decomposed into phone models.

Figure 2 shows the word recognition algorithm. 542 word models are made from phone models according to the pronunciation symbols of the reference words which appear in the speech database. HMM output probabilities of test data are calculated using all word models, and the word with the highest probability in their output probabilities is the recognition result.

In this recognition experiment by HMM, the separate vector quantization (SPVQ)[5] method or the multiple-codebook method is used. SPVQ is a method in which input data are represented by various VQ codes which correspond to different characteristics, such as spectra, power, and so on. In speech recognition, the high performance of SPVQ is well known. In this experiment, three parameters are utilized for SPVQ: WLR, power, and differenced cepstrum coefficients[6]. Differenced cepstrum coefficients are linear regression coefficients to measure the change in spectra.

The experiment task is to translate keyboard conversations which include 377 English sentences (2,834 words) uttered word by word by one male native speaker. The sentences are composed of 542 different words. HMM phone models are trained using the data without phone labels. In the speech data, 190 sentences (1,487 words) are training data, and 187 sentences (1,347 words) are test data.

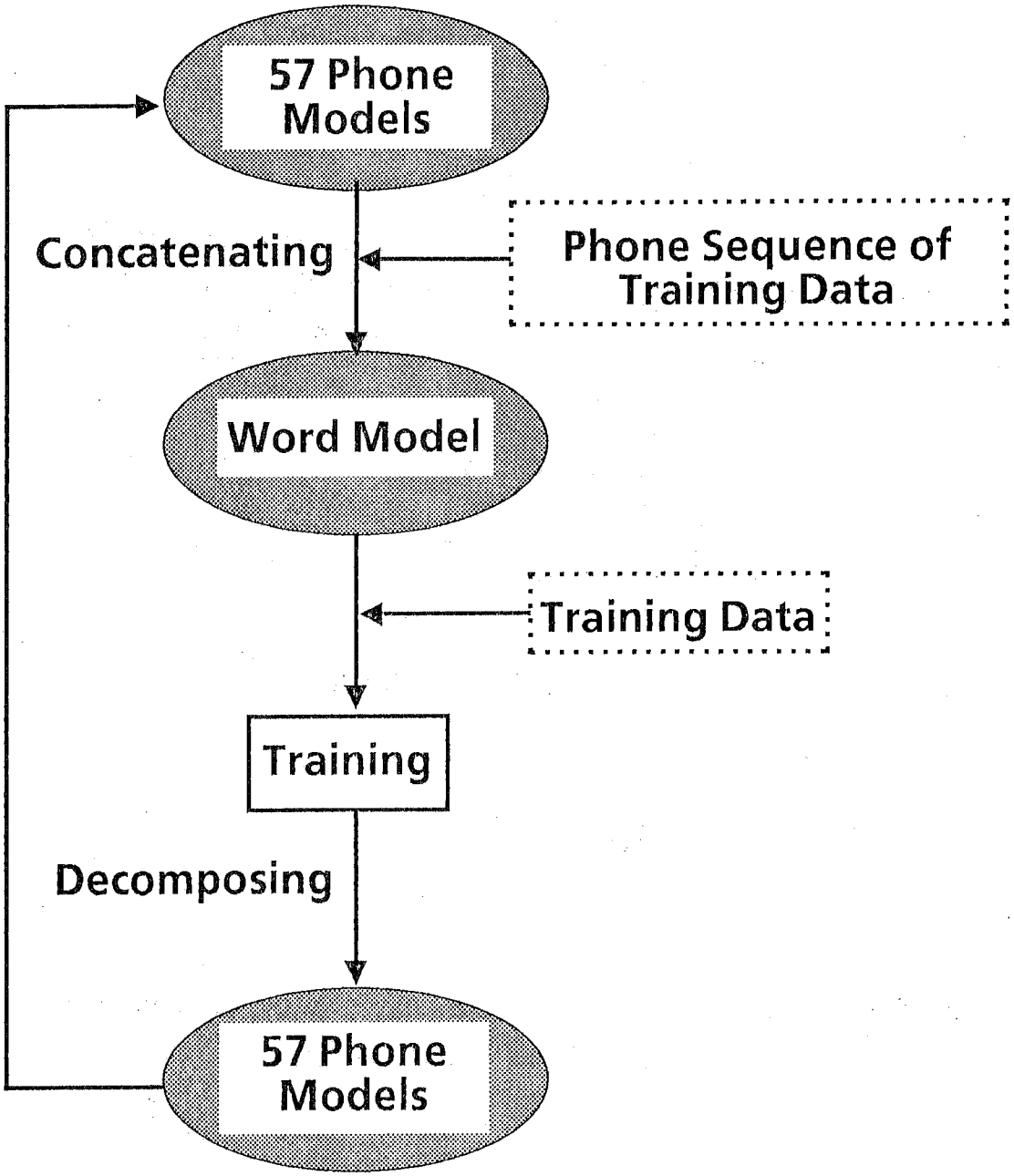

Fig.1 HMM phone concatenated training

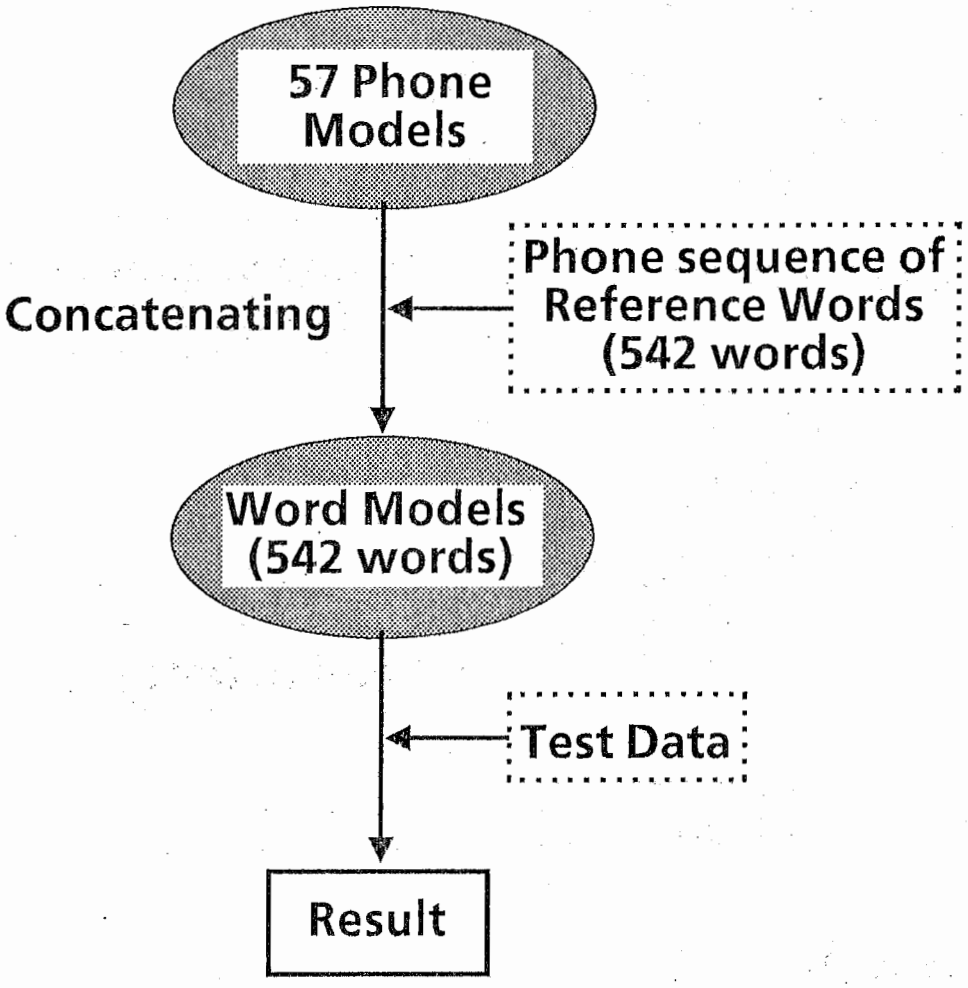

Fig.2 Word recognition by concatenating HMM phone models

3. Word Category Prediction Neural Network (NETgram)

There is a method of predicting word category using the appearance probabilities of the next word to correct word recognition errors in text [3]. The point of improving prediction ability is to use as much past word information as possible. However, by using this statistical approach, it is difficult to make an N-gram word prediction model because of the increased demand for sample data and parameters to learn probabilities.

To solve this problem, NETgrams, which were neural networks for N-gram word category prediction in text, were proposed. NETgrams can easily be expanded from Bigram to N-gram networks without exponentially increasing the number of free parameters.

The basic Bigram network is a 4-layer feed-forward network, as shown in Fig.4, which has 2 hidden layers so that each hidden layer obtains coarse-coded MFs (Micro Features) of the input or output word category. Because this network is trained for the next word category as the output for an input word category, hidden layers are expected to learn some linguistic structure from the relationship between one word category and the next in the text.

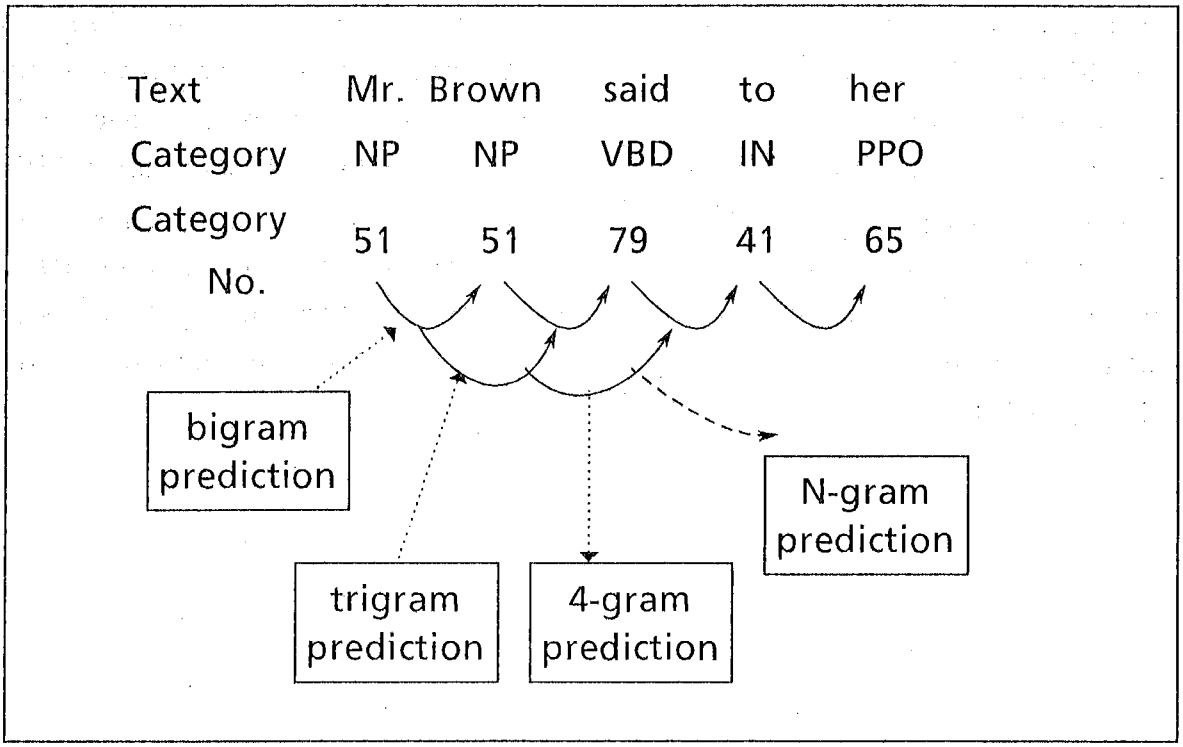

Fig.3 Word category prediction using Brown Corpus text data

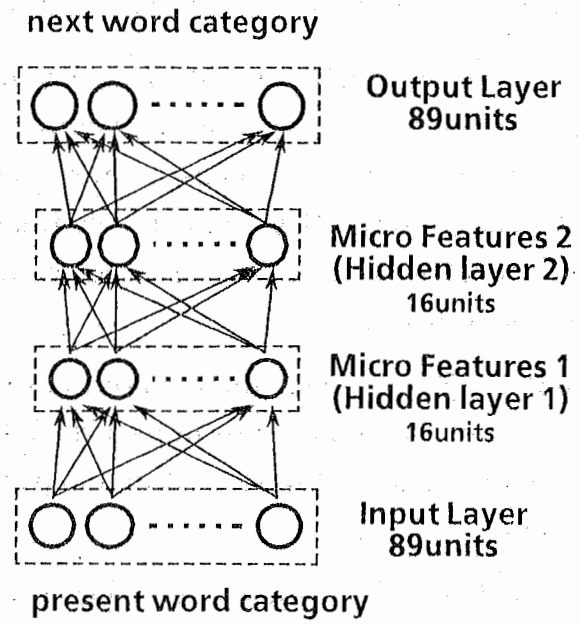

Fig.4 Basic bigram network for word category prediction

NETgram has a structure such that every new input block produced as the number of grams increases is fully connected to the lower hidden layer of one basic Bigram network with the link weight set w1'. Initial values of the link weight set w1' are all zero. Therefore, the training starts at the output values equal to the trained output values of the basic Bigram network. However, as all input word category information is compressed to one lower hidden layer (MF1), input word category information must, to some degree, be lost as input blocks increase. Therefore, when expanding from Trigram network to 4-gram network, one lower hidden layer block is added and the first and second input blocks are fully connected to one lower hidden layer block, and the second and third input blocks are fully connected to the other lower hidden layer block.

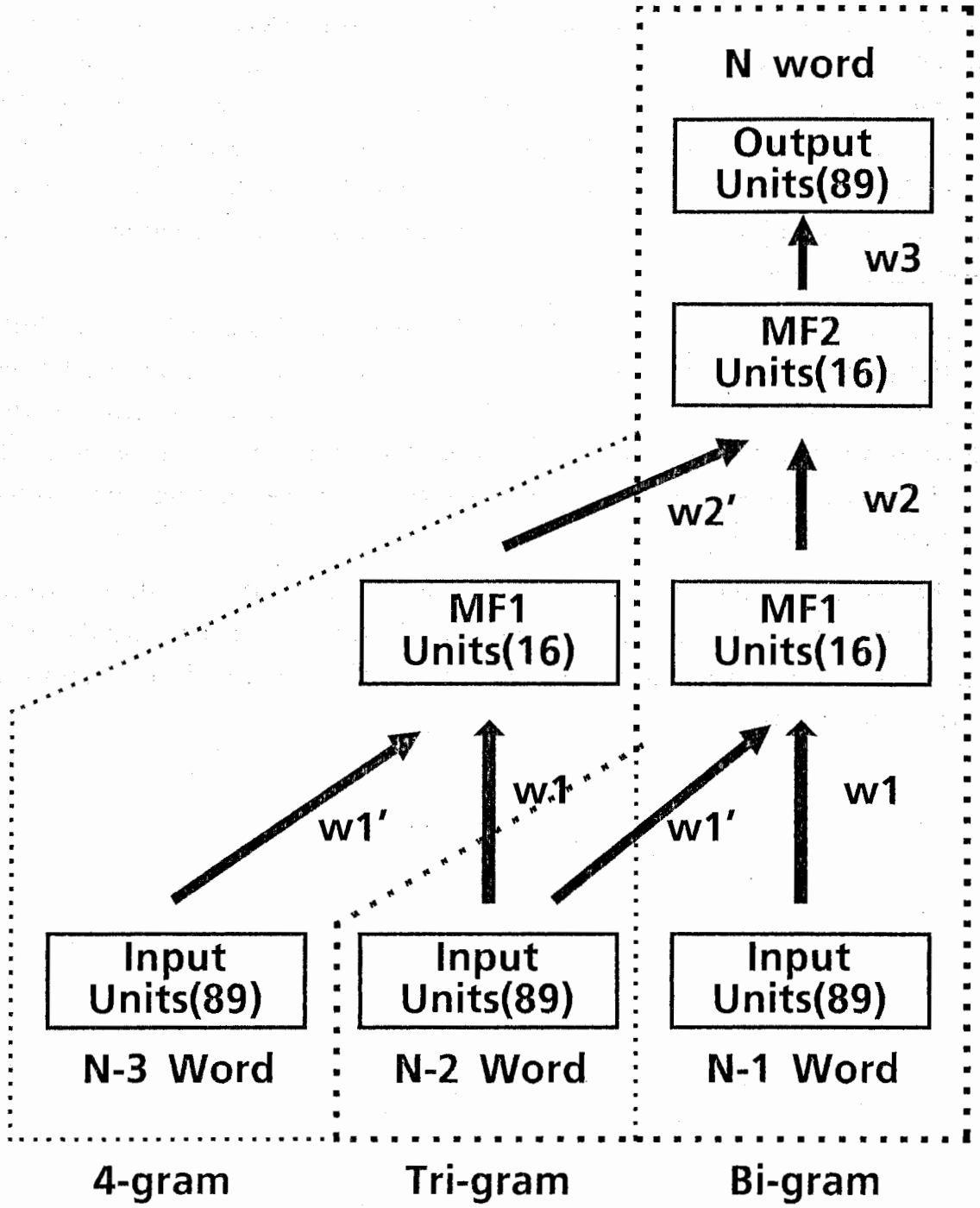

Fig.5 NETgram for word category prediction

Training a NETgram, e.g. a Trigram network, is shown in Fig.6. As the input data, word categories in the Brown Corpus text[7] are given in order from the first word in the sentence to the last word. In one input block, only one unit corresponding to the word category number is turned ON (1); The others are turned OFF (0). As output data, only one unit corresponding to the next word category number is trained by ON (1) ; The others are trained by OFF (0). The training algorithm is the Back-Propagation algorithm[8]. First, the basic Bigram network is trained, and next, the Trigram networks are trained with the link weight values trained by the basic Bigram network as initial values. 4-gram networks are trained in the same way.

The training results showed that the Trigram word category prediction ability of NETgrams was comparable to that of the statistical Trigram model and compressed information more than 130 times. Also, it was confirmed that NETgrams performed effectively for unseen data which never appeared in the training data, that is to say, NETgrams interpolate sparse training data just like the deleted interpolation.

The results of analyzing the hidden layer (Micro Features) after training showed that the word categories were classified into some linguistically significant groups, that is to say, a NETgram learns a linguistically significant structure naturally.

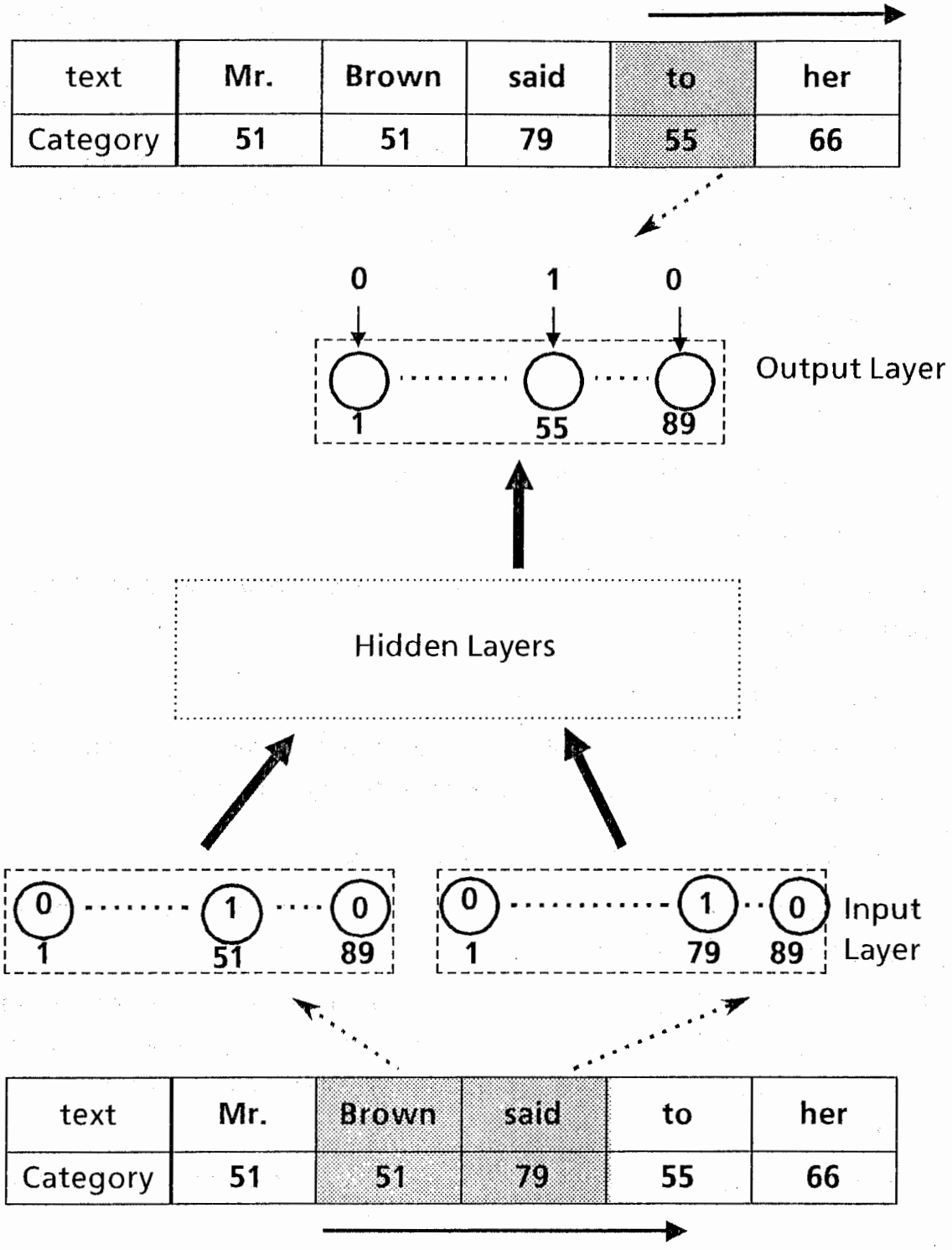

Fig.6 Training NETgram (Trigram model)

4. Applying the NETgram to HMM English Word Recognition

The algorithm for applying the NETgram to HMM English word recognition is shown in Fig.7.

Let w_i show a word just after w_{i-1} and just before w_{i+1}. Let C_i show one of the word categories to which the word w_i belongs. The same word belonging to a different category is regarded as a different word. The probability of w_i is calculated using the following approximations.

$$
\begin{aligned}
&P(w_i / w_{i-2}\, w_{i-1}) \\
&\quad \simeq P(w_i / C_{i-2}\, C_{i-1}) \\
&\quad = P(C_i / C_{i-2}\, C_{i-1}) \\
&\qquad \times \{ P(w_i / C_{i-2}\, C_{i-1}) / P(C_i / C_{i-2}\, C_{i-1}) \} \\
&\quad \simeq P(C_i / C_{i-2}\, C_{i-1}) \{ P(w_i) / P(C_i) \}
\end{aligned}
$$

Word trigram probabilities are approximated using category trigram probabilities as follows :

$$P(w_i / w_{i-2}\, w_{i-1}) \simeq P(w_i / C_{i-2}\, C_{i-1})$$

The probability of w_i is denoted by the preceding two-word sequence, w_{i-2}, w_{i-1}, and is approximated by their preceding two-category sequence.

$$P(w_i / C_{i-2}\, C_{i-1}) / P(C_i / C_{i-2}\, C_{i-1}) \simeq P(w_i) / P(C_i)$$

The probability ratio of w_i and C_i given by $C_{i-2}\, C_{i-1}$ is nearly equal to the total probability ratio of w^i and C_i.

To calculate the above probability, the trigram probability of word category, $P(C_i / C_{i-2}\, C_{i-1})$, and word occurrence probability, $P(w_i)/P(C_i)$, are required. The word probability, $P(w_i) / P(C_i)$, is prestored in the dictionary of the word w_i for each word category.

To avoid the multiplication of probabilities, the log-likelihood, ST_i, is defined as :

$$ST_i = \log P(C_i / C_{i-2}\, C_{i-1}) + \log(P(w_i) / P(C_i))$$

The first term is retrieved from the trigram of word categories and the second term is retrieved from the word dictionary.

The maximum likelihood of a sentence, SW, is given by the sum of word likelihood values of an n-word sequence. The j-th word candidate in the i-th word of a sentence is denoted by $w_{i,j}$. The likelihood of $w_{i,j}$, $SW_{i,j}$, is defined as the sum of two types of likelihood which are the log-likelihood of HMM output probability, $SH_{i,j}$, and the trigram likelihood. Thus, the likelihood of $w_{i,j}$ is described as follows:

$$SW_{i,j} = (1\text{-}\omega) \cdot SH_{i,j} + \omega \cdot ST_{i,j}$$

where ω is the weighting parameter to adjust the scaling of two kind of likelihood.

The maximum sentence likelihood values are denoted by the following equations:

$$G_{0,j} = SW_{0,j} \qquad (i = 0)$$

$$G_{i,j} = max(SW_{i,j} + G_{i-1,j}) \qquad (i \neq 0)$$

When the length of a sentence is N, the maximum value of $G_{N-1,j}$ is regarded as the maximum likelihood of the word sequence. The back-tracing of $w_{i,j}$ gives the optimal word sequence.

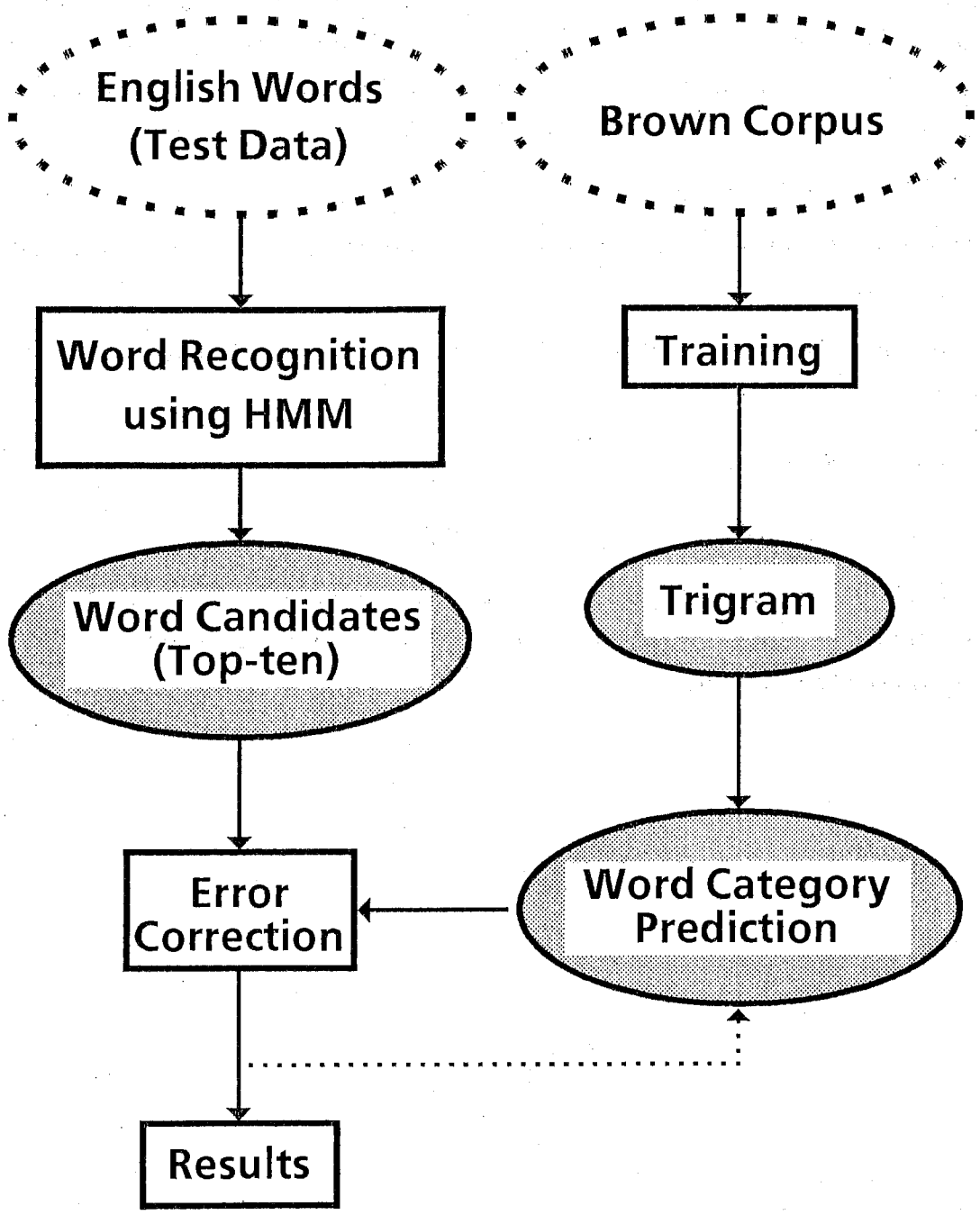

Fig.7 Applying the NETgram to
HMM English word recognition

5. Experiments

In this paper, best-ten candidates in the word recognition results by HMM are used. As the same word belonging to a different category is regarded as a different word, the word candidates are ten or more words.

To compare the performances of NETgram and a statistical model, both trigram models are used. NETgram and a statistical model are trained using the Brown Corpus Text Database[7] using 512, 1,024, 4,096 and 30,000 sentences.

6. Results

English word recognition results using HMM and the trigram models are shown in Table 1. The recognition rate in the the experiment using only HMM is 81.0%. Using NETgrams, the recognition rates have been improved about 5 or 6 %.

In the experiment results, examples of the improvement are shown in Table 2. The number of recognition errors decreases using NETgram.

On the other hand, in the case of the word "do", the recognition errors could not be corrected, because most examples of "do" in the test data appear in interrogative sentences such as, "Do you ~?". However, sentences in the Brown Corpus are based on the written language in which few interrogative sentences appear. Thus it is thought that the training of a particular conversation language structure is not enough.

In Table 1, as a result of the comparison between NETgram and a statistical trigram, the performance of NETgram is higher than that of the statistical trigram in the case of limited training data of about 4,000 sentences. The statistical model cannot learn word sequences which do not appear in the training data, thus the prediction value of that word sequence is zero. NETgram does not make such fatal mistakes.

Table 1 Word Recognition Rates (%)

Training Sentences	Word Prediction Model	
	NETgram	Statistical Trigram
512	86.3	85.5
1,024	86.9	85.4
4,096	86.9	86.6
30,000	87.2	87.7

Table 2 Examples of the improvements
(The number of recognition errors)

Prediction Model	Training Sentences	and	I	on	will
———	———	3	12	6	9
NETgram	512	0	7	5	2
	1,024	1	7	5	4
	4,096	1	5	5	3
	30,000	1	5	5	3
Statistical Trigram	512	2	10	4	2
	1,024	2	10	4	6
	4,096	1	9	3	2
	30,000	2	6	4	3

7. Conclusion

In this paper, the NETgram is applied to HMM English word recognition, and it is shown that the NETgram can effectively correct word recognition errors in text. The word recognition rate using HMM is 81.0%. The NETgram trained by 1,024 sentences improves the word recognition rate to 86.9%. And the NETgram performs better than the statistical trigram model.

Acknowledgment

The authors would like to thank Dr. Akira Kurematsu, president of ATR Interpreting Telephony Research Laboratories for his continuous support of this research. We are also indebted to the member of the Speech Processing Department at ATR.

References

[1] F.Jelinek, "Continuous Speech Recognition by Statistical Methods", Proceedings of the IEEE, Vol.64, No.4 (1976.4)

[2] K.Shikano, "Improvement of Word Recognition Results by Trigram Model", ICASSP 87, 29.2 (1987.4)

[3] M.Nakamura, K.Shikano, "A Study of English Word Category Prediction Based on Neural Networks", ICASSP 89, S13.10(1989.5)

[4] K.Maruyama, et al., "English Word Recognition Using HMM Phone Concatenated Training", IEICE SP88-119(1989.1)

[5] T.Hanazawa, et al., "Study of Separate Vector Quantization for HMM Phoneme Recognition", ASJ 2-P-8(1988.10)

[6] S.Furui, : "Speaker Independent Isolated Word Recognition Using Dynamic Feature of Speech Spectrum" , IEEE Trans.ASSP-34,1(1986)

[7] Brown University, "Brown Corpus", Tech. Report, Brown University (1967)

[8] D.E.Rumelhart et al., "Parallel Distributed Processing", M.I.T. Press (1986)