Pergamon

Pattern Recognition, Vol. 30, No. 8, pp. 1339–1346, 1997

0031-3203/97 $17.00+.00

PII: S0031-3203(96)00154-9

A LANGUAGE MODEL BASED ON SEMANTICALLY CLUSTERED WORDS IN A CHINESE CHARACTER RECOGNITION SYSTEM

HSI-JIAN LEE* and CHENG-HUANG TUNG

Department of Computer Science and Information Engineering, National Chiao Tung University, Hsinchu, Taiwan 30050, R.O.C.

(*Received* 16 *July* 1996)

Abstract—This paper presents a new method for clustering the words in a dictionary into word groups. A Chinese character recognition system can then use these groups in a language model to improve the recognition accuracy. In the language model, the number of parameters we must train beforehand can be kept to a reasonable value. The Chinese synonym dictionary *Tong2yi4ci2 ci2lin2* providing the semantic features is used to calculate the weights of the semantic attributes of the character-based word classes. The weights of the semantic attributes are next updated according to the words of the Behavior dictionary, which has a rather complete word set. Then, the word classes are clustered to m groups according to the semantic measurement by a greedy method. The words in the Behavior dictionary can finally be assigned to the m groups. The parameter space for the bigram contextual information of the character recognition system is m^2. From the experimental results, the recognition system with the proposed model has shown better performance than that of a character-based bigram language model. © 1997 Pattern Recognition Society. Published by Elsevier Science Ltd.

Contextual postprocessing Language model Semantics Word group

1. INTRODUCTION

In a character recognition system, language models have been widely used for postprocessing to increase the recognition rate of the recognition system. In a language model, if the number of parameters used for describing contextual information is small, the ability for correcting the recognition errors in the character recognition stage will be insignificant.[1–3] For example, if the words in a dictionary are clustered into about 30 parts-of-speech, only a few recognition errors can be corrected by using the contextual information. In contrast, if the number of parameters used for describing the contextual information is very large, the training process for the parameters will be difficult, and the memory required will make the execution of a language model impractical.[4–7] For instance, if a Chinese language model adopts a word bigram to describe contextual information, the language model may consume all available memory.

In this paper, we propose a new method to cluster the words in the Behavior dictionary[8] into a reasonable number of groups; we have 800 groups in our experiments. Semantic information will be used to cluster the words in the Behavior dictionary. Anyway, the Behavior dictionary does not contain the semantic features. We will accomplish the clustering task by using the Chinese synonym dictionary *Tong2yi4ci2 ci2lin2*,[9] which provides the necessary semantic information.

* Author to whom correspondence should be addressed. Tel.: 886-35-711437; fax: 886-35-724176; e-mail: hjlee@csie.nctu.edu.tw.

In order to reduce the number of word classes, we first transform the words into a character-based word class. Assume that the character set *Ch_set* includes the 5401 frequently used Chinese characters Ch_i, which is denoted as $Ch\_set = \{Ch_1, Ch_2, \ldots, Ch_{5401}\}$. We define that the character-based word class Ch_i# includes the words with the prefix Ch_i and the postfix #, which represents a regular expression $\#=(Ch_1+Ch_2+\cdots+Ch_{5401})^*$. Similarly, the word class #Ch_i includes words with the suffix Ch_i, and the word class #Ch_i# includes words containing the character Ch_i. There are a total of 3×5401 word classes defined.

As the words in the Chinese dictionary *Tong2yi4ci2 ci2lin2* contain hierarchical semantic features, we can assign the words in the dictionary into 3×5401 word classes according to the semantics of the words, and obtain the semantic attributes of the word classes. For example, the word 印花稅 in *Tong2yi4ci2 ci2lin2* can be contained in one of the five word classes 印#, #印#, #花#, #稅#, and #稅. We will assign the word 印花稅 into one of the five word classes such that the semantic measurement among the semantic attributes of the word classes containing the word 印花稅 is minimal. After mapping all of the words in *Tong2yi4ci2 ci2lin2* into the word classes, the word classes are ranked according to the compactness of their semantic attributes, which will be defined later. After the word classes are ranked, the words in the Behavior dictionary will be clustered into the word classes. We cluster the word classes into a predefined number of m groups according to the semantic attributes. Then a language model based

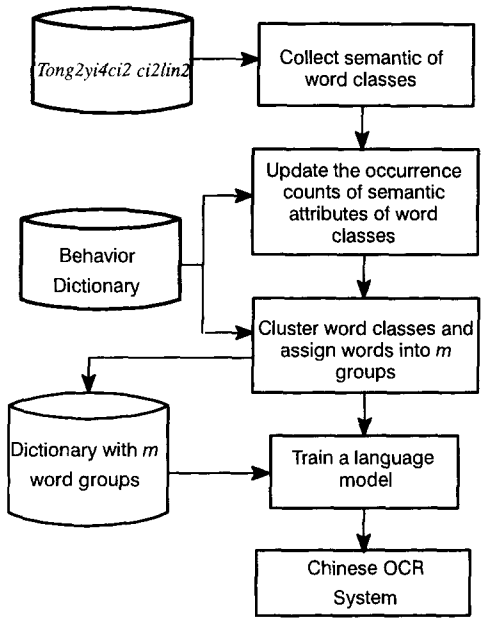

Fig. 1. The flow diagram for clustering the words in the Behavior dictionary into m groups.

on the m word groups can be constructed for postprocessing in a character recognition system.

The flow of our method is summarized in Fig. 1. First, we apply the synonym dictionary *Tong2yi4ci2 ci2lin2* to collect the semantic attributes of the character-based word classes and order the word classes. Second, the occurrence counts of the semantic attributes are updated according to the words in the Behavior dictionary. Third, the word classes are clustered to m groups according to their semantic measurement. According to the grouped word classes, the words in the Behavior dictionary can be assigned into the m groups. Fourth, a language model based on the grouped words can thus be constructed and used for postprocessing in a Chinese character recognition system.

2. CONSTRUCTION OF SEMANTIC ATTRIBUTES OF WORD CLASSES

The Chinese dictionary *Tong2yi4ci2 ci2lin2* contains more than 50,000 words classified into 12 major, 94 medium, and 1428 minor categories. The 12 major categories are listed in Fig. 2. Each major category contains some medium categories and each medium category contains several minor categories. Each word in *Tong2yi4ci2 ci2lin2* has a semantic entry of major, medium, and minor categories. For example, the word 宮殿 in *Tong2yi4ci2 ci2lin2* has the semantic entry Bn01, which represents *object* (=B), *architect* (=n), and *architect and building* (=01). In our system, we utilize the major and medium categories as the semantic entry. Thus the semantic entry of the word 宮殿 is Bn.

The semantic attributes of word classes are collected from that of the words in the dictionary *Tong2yi4ci2 ci2lin2*. Figure 3 gives a simple example with words and

A	Human
B	Object
C	Time and Space
D	Abstract
E	Characteristics
F	Action
G	Mental Activity
H	Activity
I	Phenomena and State
J	Association
K	Auxiliary
L	Honorary Words

Fig. 2. The 12 major categories of the semantics.

semantic entries. In Fig. 3(a), we assume that a dictionary contains only a total of 15 words, each of which has at least one semantic entry. If a word has several semantic entries, we distribute equally the weights among these semantic entries. For example, the words 擇友, 擇, and 善 have one, two, and three semantic entries, respectively. Figure 3(b) shows that the word 善 has distributed weight $\frac{1}{3}$ to its semantic entry, denoted as (Di,$\frac{1}{3}$), (Ee,$\frac{1}{3}$), and (Hj,$\frac{1}{3}$). Similarly, the word 擇友 has the semantic entry (Hj,1), and the word 擇 has semantic entries (Hi,$\frac{1}{2}$) and (Hj,$\frac{1}{2}$), respectively.

The *occurrence count* of the word class *Ch#* is the sum of the weights of the words with the prefix *Ch*. In general, the semantic attributes of the word $Class_i$ can be represented as $Class_i.att = (att_1, count_1), (att_2, count_2), \ldots))$, where att_i is the semantic entry, and $count_i$ is the occurrence count of the semantic entry. Figure 4(a) gives the semantic attributes of the word classes in Fig. 3(a).

In order to evaluate the similarity of semantic attributes in a word class, we will define the compact measurement, *COMPACT*, between two semantic attributes (att_i,$count_i$) and (att_j,$count_j$). Each semantic attribute (att_i,$count_i$) can be further represented as $(C_{i,1}C_{i,2}, count_i)$, where $C_{i,1}$ and $C_{i,2}$ represent the major and medium semantic categories. The compactness between the two semantic attributes (att_i, $count_i$) and (att_j, $count_j$) is defined as

$$\begin{aligned} &COMPACT((att_i, count_i), (att_j, count_j)) \\ &= COMPACT((C_{i,1}C_{i,2}, count_i), (C_{j,1}, C_{j,2}, count_j)) \\ &= k \cdot count_i \cdot count_j, \end{aligned}$$

where k=2 if $C_{i,1} \neq C_{j,1}$, or k=1, otherwise. For example, the compactness between the two (Hi,1) and (Aj,4), which are the semantic attributes of the word class #友, is 8 (= 2 × 1 × 4). The average compactness of the semantic attributes in a word class can be defined as

$$AVG\_COMP(Class_i) = \begin{cases} \dfrac{\sum_j \sum_{k>j} COMPACT((att_j, count_j), (att_k, count_k))}{C(Class_i.count, 2)}, \\ \qquad \text{if } Class_i.count > 1, \\ 2, \qquad \text{if } Class_i.count = 1, \end{cases}$$

where $Class_i.count$ is the sum of the occurrence count of the semantic entries in $Class_i$. It is used as a normal-

擇	Hi, Hj	友	Aj	和善	Ed
擇友	Hj	戰友	Aj	改善	Ih
擇善	Hj	諍友	Aj	面善	Ed
擇偶	Hj	摯友	Aj	盡善盡美	Ed
擇鄰	Hj	善	Di, Ee, Hj	積善	Hi

(a)

擇友 : (Hj, 1)

擇 : (Hi, $\frac{1}{2}$), (Hj, $\frac{1}{2}$)

善 : (Di, $\frac{1}{3}$), (Ee, $\frac{1}{3}$), (Hj, $\frac{1}{3}$)

(b)

Fig. 3. A small example of words and semantic distributions. (a) The 15 words and their semantic entries. (b) The distribution of multiple semantic entries in a word.

#擇 #.att= 擇 #.att= ((Hi, $\frac{1}{2}$), (Hj, $\frac{9}{2}$))

#友 #.att= #友 .att= ((Hi, 1), (Aj, 4))

#善 #.att= ((Di, $\frac{1}{3}$), (Ed, 3), (Ee, $\frac{1}{3}$), (Hi, 1), (Hj, $\frac{4}{3}$), (Ih, 1))

#善 .att= ((Di, $\frac{1}{3}$), (Ed, 2), (Ee, $\frac{1}{3}$), (Hi, 1), (Hj, $\frac{4}{3}$), (Ih, 1))

#偶 #.att= #偶 .att= ((Hj, 1))

#鄰 #.att= #鄰 .att= ((Hj, 1))

.
.
.

(a)

AVG_COMP(#擇 #.att)= *AVG_COMP*(擇 #.att)=0.225

AVG_COMP(#友 #.att)= *AVG_COMP*(#友 .att)=0.8

AVG_COMP(#善 #.att)=1.49

AVG_COMP(#善 .att)=1.726

AVG_COMP(#偶 #.att)= *AVG_COMP*(#偶 .att)=2

AVG_COMP(#鄰 #.att)= *AVG_COMP*(#鄰 .att)=2

.
.
.

(b)

Fig. 4. (a) The semantic attributes of the word classes. (b) The average compactness of character-based word classes.

ization factor. For example, we have #友.*count*=5 and *AVG_COMP*(#友)=(8/*C*(5,2))=0.8. Note that the range of *COMPACT* is from 0 to 2. The word classes with similar semantic categories and large occurrence counts will have a small compactness. Since the class with a single count is generally not what we need, we assign it a rather large compactness. Figure 4(b) lists the average compactness for the character-based word classes.

Among the word classes in Fig. 4(b), the word class 擇# has the minimal average compactness for the semantic attributes. Therefore, the words with prefix 擇 are grouped into the word class 擇#. Because the average compactness for the semantic attributes of the word class 擇# is the smallest, we assign the word class 擇# the first rank; that is, 擇#.*rank*=1. The words assigned into the word class 擇# and the attributes of the word class 擇# are shown in Fig. 5(a). After removing the assigned words, we perform the same process to obtain the semantic attributes and the average compactness for word classes. The final word classes extracted from the 15 words are shown in Fig. 6.

We perform the process to assign all words in the dictionary *Tong2yi4ci2 ci2lin2*. The word classes with semantic attributes are ordered sequentially. If a word class includes no words in *Tong2yi4ci2 ci2lin2*, the semantic attributes of the word class are set as null and the average compactness of the semantic attributes in the word class are defined to be infinite.

Since the Behavior dictionary has more complete words than the dictionary *Tong2yi4ci2 ci2lin2*, we will modify the occurrence count of the semantic entry according to the words in the Behavior dictionary. We ignore the word occurrence probability in this version.

擇 #.*rank*=1
擇 #.*att*= ((Hi,$\frac{1}{2}$), (Hj,$\frac{9}{2}$))
AVG_DIST(擇 #.*att*)=0.225

擇	Hi, Hj
擇友	Hj
擇善	Hj
擇偶	Hj
擇鄰	Hj

(a)

友	Aj	和善	Ed
戰友	Aj	改善	Ih
諍友	Aj	面善	Ed
摯友	Aj	盡善盡美	Ed
善	Di, Ee, Hj	積善	Hi

(b)

Fig. 5. (a) The words assigned into the word class 擇#. (b) The remaining words.

We attach each word class $Class_i$ a count $Class_i.Dict\_count$, initialized to zero, to record how many words in the Behavior dictionary are assigned to the word class, and then update the occurrence counts of the semantic features in word classes.

Let the words in the Behavior dictionary be represented as $W_1, \ldots, W_n$, where $n > 80,000$. Assume that a word W_i consists of characters $Ch_{i_1}, \ldots, Ch_{i_k}$. The word W_i will be assigned into the word class with the minimal rank among the word classes $\#Ch_{i_j}\#$, $j = 1, \ldots, k$, $Ch_{i_j}\#$, and $\#Ch_{i_k}$. For example, if the ranks of the word classes 電#, #電#, #視#, and 視# are 105, 502, 416, and 376, respectively, the word 電視 will be clustered into the character-based word class 電# with the best rank, and then the count 電#.*Dict_count* is increased by one.

After all words in the Behavior dictionary have been assigned into the word classes, we will modify the occurrence counts of the semantic attributes of the word classes. For each word class $Class_i$, the occurrence count $count_j$ of a semantic attribute att_j is updated as

$$count_j \cdot \frac{Class_i.Dict\_count}{Class_i.count}.$$

For instance, the original semantic attributes are ((Hi,$\frac{1}{2}$), (Hj,$\frac{9}{2}$)). If 擇#.*Dict_count* is 7, the new semantic attributes are 擇#.*att*=((Hi,$\frac{7}{10}$), (Hj,$\frac{63}{10}$)).

3. CLUSTERING WORD CLASSES INTO *m* GROUPS

After the occurrence counts of the semantic attributes in word classes have been modified, we will cluster the word classes into m groups. At the first step, we select m word classes $Class_1, Class_2, \ldots, Class_m$, with the largest $Class_j.count$, $j = 1, \ldots, m$, among all word classes. Each group G_j, $j = 1, \ldots, m$, is initialized as

$$G_j = \{(Class_{i_j}, (att_1, count_1), (att_2, count_2), \ldots)\}.$$

We define the size of a group G_j as

$$SIZE(G_j) = \sum_{Class_i \text{ in } G_j} Class_i.count.$$

To make the sizes of the m groups as similar as possible, we apply a greedy method to update the m groups. First, a group G_j with the minimal size is selected from the m groups. The semantic attributes of each unclustered word class $Class_i$ are combined with the semantic attributes in the group G_j. Then we measure the average compactness of the combined semantic attributes. For example, if the selected group G_j is G_j={(擇#,(Hi,$\frac{7}{10}$),(Hj,$\frac{63}{10}$))}, and the class $Class_i$=(#找#,(Hi,1),(Hj,1)) has not been clustered, we create the combined semantic attributes ((Hi,$\frac{17}{10}$),(Hj,$\frac{73}{10}$)). The average compactness of the semantic attributes is

$$\frac{Dist((\text{Hi},\frac{17}{10}),(\text{Hj},\frac{73}{10}))}{C(9,2)} = \frac{(\frac{17}{10})\cdot(\frac{73}{10})}{36} = 0.345.$$

If the group G_j has the minimal compactness for the combined semantic attributes, the unclustered

擇 #.*rank*=1
擇 #.*att*= ((Hi,$\frac{1}{2}$), (Hj,$\frac{9}{2}$))
AVG_DIST(擇 #.*att*)=0.225
擇 Hi, Hj
擇友 Hj
擇善 Hj
擇偶 Hj
擇鄰 Hj

#友.*rank*=2
#友.*att*= ((Aj, 4))
AVG_DIST(#友.*att*)=0
友 Aj
戰友 Aj
諍友 Aj
摯友 Aj

#善 #.*rank*=3
#善 #.*att*= ((Di,$\frac{1}{3}$), (Ed, 3), (Ee,$\frac{1}{3}$), (Hi, 1), (Hj,$\frac{1}{3}$), (Ih, 1))
AVG_DIST(#善 #.*att*)=1.4
善 Di, Ee, Hj
和善 Ed
改善 Ih
面善 Ed
盡善盡美 Ed
積善 Hi

Fig. 6. The three words classes extracted from the 15 words.

word classes will be clustered into the selected group G_j. For example, if the unclustered class $Class_i$= (#找#,(Hi,1),(Hj,1)) combined with group G_j has the minimal compactness, we have a new group G_j= {(擇#,(Hj,12)),(#找#,(Hi,1),(Hj,1))}. The process is repeated until all word classes are clustered into m groups.

Because the words $W_1, \ldots, W_n$ in the Behavior dictionary have been assigned into the character-based word classes and the word classes have been clustered into m groups, the words $W_1, \ldots, W_n$ can thus be assigned into m groups.

4. MARKOV LANGUAGE BASED ON CLUSTERED WORDS

In this section, we will derive the language model based on the clustered words. In the procedure for training contextual information, a large training corpus is needed. Since the bigram POS (parts-of-speech) language model has shown high performance for word segmentation,[3] we apply the bigram POS language model to segment the sentences in a training corpus. Let the words in a segmented sentence be $w_1, w_2, \ldots, w_k$. Let $G(w_i)$ represent the group in which the word $G(w_i)$ has been clustered. We transform the segmented sentence into $G(w_1), G(w_2), \ldots, G(w_k)$ to train the transition probability, that is, the contextual information $P(G(w_i)|G(w_{i-1}))$. The conditional probability of the word w_i, $P(w_i|G(w_i))$, is trained similarly.

After a word transition graph is constructed, a new language model is applied for the contextual postprocessing, which is described as follows. Let $I = I_1 I_2 \ldots I_L$ be a sequence of character images, where I_i is the ith character image in the input sentence I, and L is the length of the sentence. Each character image I_i is recognized as M candidate characters $c_{i1}, c_{i2}, \ldots, c_{iM}$. Each candidate character c_{ik} has a matching score MS_{ik}. In the candidate character sets, there are M^L sentence hypotheses. The goal of the language model is to determine a sentence hypothesis $\hat{S} = c_{1i}c_{2j}\cdots c_{L,k}$ that has the maximum likelihood among all sentence hypotheses $\{S\}$. The occurrence likelihood of a sentence hypothesis $S = c_1 c_2 \cdots c_L$ is given by $P(S|I)$, where c_i is one of the candidate characters in the ith candidate set. Our goal can be represented as

$$P(\hat{S}|I) = \max_{S} P(S|I).$$

Since the basic syntax-meaningful unit in Chinese is a word, a sentence $S = c_1 c_2 \cdots c_L$ can be represented as $S = w_1 w_2 \cdots w_N$, where the word w_i is composed of one or more characters. Then the bigram contextual probability at the word level, $P(w_i|w_{i-1})$, can be modified as

$$P(w_i|w_{i-1}) \approx P(G(w_i)|G(w_{i-1}))P(w_i|G(w_i)).$$

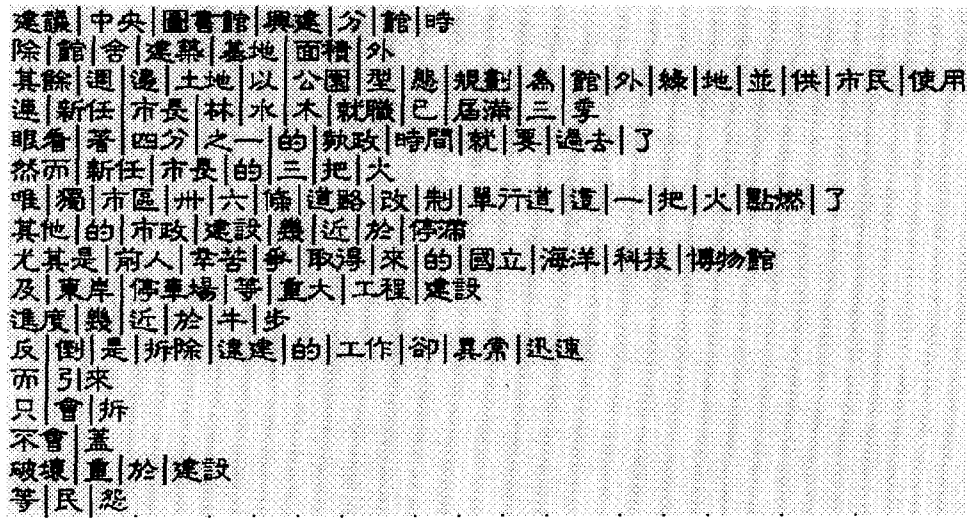

Fig. 7. The sentences segmented by the bigram POS language model.

The probability $P(S|I)$ can be computed as

$$P(S|I) \approx \prod_{i=1}^{N} P(G(w_i)|G(w_{i-1}))P(w_i|G(w_i)) \prod_{k} P(c_k|I_k).$$

The term $\prod_k P(c_k|I_k)$ is the matching score of word w_i, which is the product of the matching scores of the constituent characters c_k. After a word transition graph is constructed for all candidate characters of the sentence, the dynamic programming method for the Markov language model can be applied to find the most promising sentence hypothesis.

5. EXPERIMENTAL RESULTS

In the process of clustering the words, we defined the number of word groups m=800. In our experiments, the training corpus consisted of reports of local news. There were 178,027 sentences in the corpus, including a total of more than 2,000,000 Chinese characters. Some sentences in the corpus were segmented by the bigram POS language model for training contextual information, and the results are shown in Fig. 7.

In the following, we measure the performance of the language models based on character bigram, bigram POS, trigram POS, and semantically clustered word classes. An image file with 800 news sentences including 8136 characters are recognized by a Chinese character recognition system. The recognition rate is about 85.2%. After the character-based bigram language model is applied to perform contextual postprocessing, the recognition rate is increased to 89.2%. Similarly, the recognition rates for the bigram POS model and the trigram POS model are 87.3% and 87.5%, respectively, where the number of parts-of-speech is 30. When the language model based on clustered words is applied, the recognition rate is increased to 92.8%, which is higher than those for the other three language models.

Next, we discuss memory requirements of these language models. The memory required for contextual description in the bigram POS is 30^2 plus the number of words in the dictionary, and the memory required for the trigram POS language model is 30^3 plus the number of words in the dictionary. Their recognition rates increased by using these two models are relatively low. The largest memory requirement for contextual description the character-based bigram language model is 5401^2 and that for contextual description in our language model is the sum of 800^2 and the number of words in the dictionary, which is much less than 5401^2.

An example which shows how a correct sentence hypothesis is selected from the candidate character sets

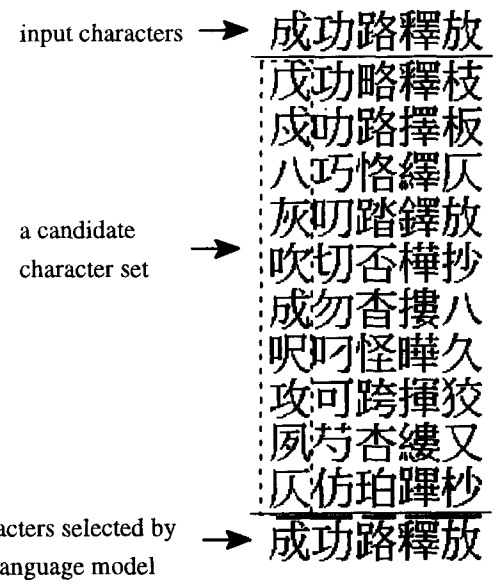

Fig. 8. An example showing how a correct sentence hypothesis is selected from the candidate character sets generated by a character recognition system.

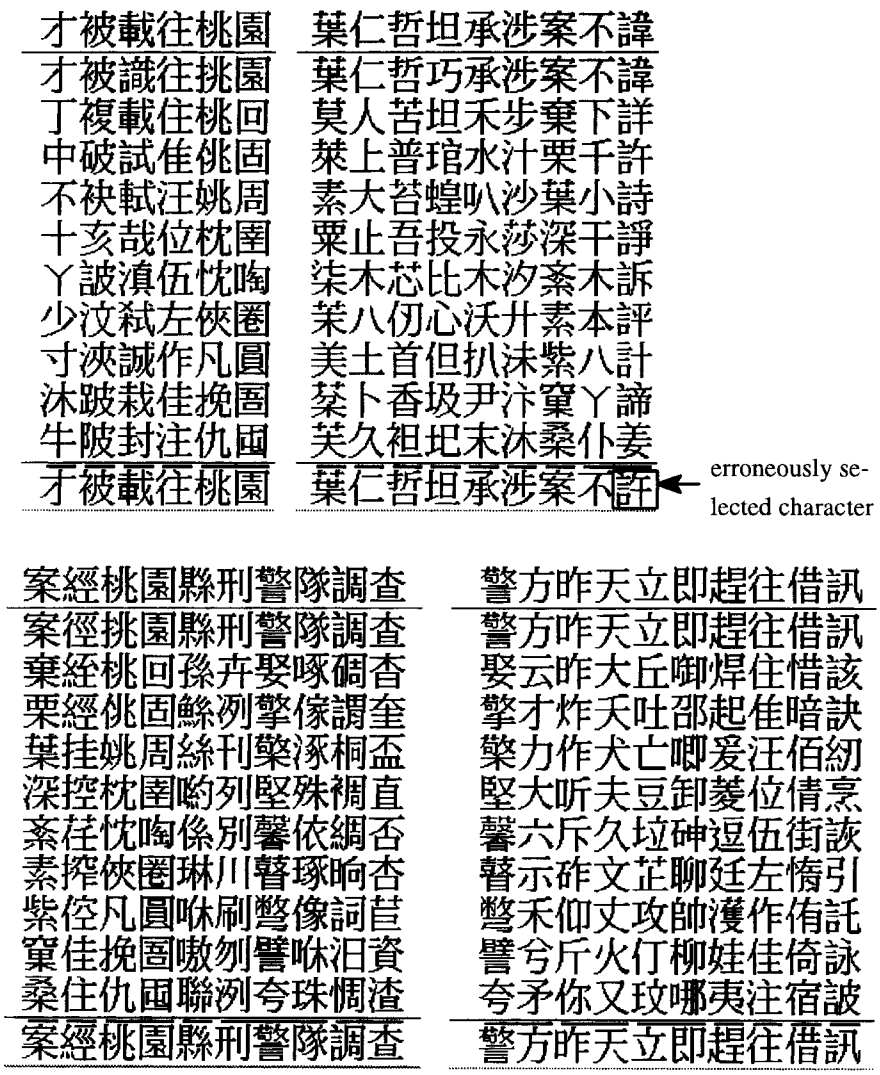

Fig. 9. Some more examples showing the performance of the language model.

is shown in Fig. 8. Each candidate character set containing 10 candidate characters is generated by a character recognition system. The example shows that our language model can select the candidate character from each candidate set correctly. Figure 9 gives more examples of contextual postprocessing. The character bounded by a box is selected incorrectly.

6. CONCLUSIONS

In this paper, we have proposed a new method to cluster the words in the Behavior dictionary into a reasonable number of m groups. We performed the clustering task by applying the dictionary *Tong2yi4ci2 ci2lin2* to train the semantic attributes of character-based word classes. The occurrence counts of the semantic attributes of word classes are updated by counting the words in the Behavior dictionary. The updated word classes are grouped into m groups according to the semantic measurement, and then the words in the Behavior dictionary are clustered into m groups. The parameter space for the bigram contextual information can be reduced to m^2. From the experimental results, we have shown that the language model based on the grouped word classes has better performance than a character-based bigram language model. For further improvement, we can modify the clustering criterion by considering the word occurrence probabilities.

REFERENCES

1. L. F. Chien, Some new approaches for language modeling and processing in speech recognition applications, Ph.D. Thesis, Institute of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan (1991).
2. N. C. Wang, A handwritten Chinese text recognition system with a contextual postprocessing module, Master Thesis, Institute of Computer Science and Information Engineering, National Chiao Tung University, Hsinchu, Taiwan (1991).
3. H. J. Lee and C. H. C. Chien, A Markov language model in handwritten Chinese text recognition, *Proc. 2nd ICDAR*, 72–75 (1993).

4. B. Merialdo, Multilevel decoding for very-large-size-dictionary speech recognition, *IBM J. Res. Develop.* **32**, 227–237 (1988).
5. R. M. K. Sinha, Rule based contextual post-processing for Devanagari text recognition, *Pattern Recognition* **20**, 475–485 (1987).
6. E. J. Yannakoudakis, I. Tsomokos and P. J. Hutton, *n*-grams and their implication to natural language understanding, *Pattern Recognition* **23**, 509–528 (1990).
7. C. H. Tung and H. J. Lee, Increasing character recognition accuracy by detection and correction of erroneously-identified characters, *Pattern Recognition* **27**, 1259–1266 (1994).
8. *Behavior Electronic Dictionary.* Dehavior Design Corp., Taiwan (1994).
9. J. J. Mei, Y. M. Chu, Y. Q. Gau and H. X. Yin, *Tong2yi4ci2 ci2lin2—Chinese Synonym Dictionary* (in Chinese). Shianghai Publishing Co., Shianghai (1983).

About the Author — HSI-JIAN LEE received the B.S., M.S., and Ph.D. degrees in Computer Engineering from National Chiao Tung University, Hsinchu, Taiwan, in 1976, 1980, and 1984, respectively. From 1981 to 1984, he was a lecturer in the Department of Computer Engineering, National Chiao Tung University, and from 1984 to 1989 an associate professor in the same department. Since August 1989, he has been with National Chiao Tung University as a professor. He is at present Chairman of the Department of Computer Science and Information Engineering, National Chiao Tung University. He has been a member of the Government Board of the Chinese Language Computer Society, a member of the Executive Committee of the Chinese Society on Image Processing and Pattern Recognition, and a member of the Executive Committee of R.O.C. Computational Linguistic Society. He is now the president of the Chinese Language Computer Society (CLCS), the Editor-in-Chief of the International Journal of Computer Processing of Oriental Languages (CPOL), and an Associate Editor of the International Journal of Pattern Recognition and Artificial Intelligence. He was responsible for the 1992 R.O.C. Computational Linguistic Workshop and 1993 R.O.C. Conference on Computer Vision, Graphics, and Image Processing. He was the program chair of the 1994 International Computer Symposium and the Fourth International Workshop on Frontiers in Handwriting Recognition (IWFHR). In 1992–1994, he was a winner of outstanding researchers of the National Science Council, R.O.C. His current research interests include document analysis, optical character recognition, image processing, pattern recognition, and artificial intelligence. He is a member of Phi Tau Phi.

About the Author — CHENG-HUANG TUNG was born in Tainan city, Taiwan, R.O.C., on 28 May 1967. He received the B.S. and Ph.D. degrees in Computer Science and Information Engineering from the National Chiao Tung University, Hsinchu, Taiwan, in 1989 and 1994. His research interests are in the areas of pattern recognition, artificial intelligence, and natural language processing.