A New Postprocessing Algorithm Based on Statistical Language Model and Lexicon

CHAN Kwok Ping , CHEN Yong
Department of Computer Science, University of Hong Kong, Hong Kong
Email: {kpchan, yongchen}@csis.hku.hk

ABSTRACT

This paper presents a new kind of postprocessing algorithm for a Chinese OCR. In this algorithm, a statistical language model and a word dictionary are used as linguistic information to improve the performance of the OCR. The linguistic information is used to form the sentence candidates, to prune the unreasonable character combination and to evaluate the ultimate sentence candidates. Some experiments have been conducted to verify the algorithm. The experimental results show that the algorithm significantly improves the recognition rate.

1.INTRODUCTION

To improve the performance of the recognition system, researchers have developed and used many kinds of statistical language models for postprocessing. Some language models [1][2] were build up just like a N-th Markov model; others [3] [4] were build up based on language features, say, word or word class. For Markov language model, Viterbi algorithm is used to find out the optimal path. But, as N increases, the requirements on computation and storage will increase dramatically. For the word or word class language model, only word or word class information is used as linguistic information. The linguistic information contained in the sentence is not used fully. In this research, besides the word information, the conditional probability of two adjacent characters, as linguistic information, is also employed to further improving the postprocessing effect.

In this paper, we will create a new kind of language model. We will use it, together with a word dictionary, to implement the postprocessing algorithm. The statistical language model actually consists of probabilistic figures about which two Chinese characters can stand together and at what probability. The language model provides us with important information: first, if we randomly select two Chinese characters, the language model can tell us whether they can appear adjacently; second, the language model can further tell us at what probability they will stand together. The lexicon used in our system contains about 80,000 words. We will use the linguistic knowledge provide by the Statistical Language Model and the word dictionary to form the sentence candidates firstly, and then select a best sentence candidate from them as output of the system.

2. STATISTIC LANGUAGE MODEL

In our system, 3664 Chinese characters can be recognized in total. We conducted statistical analysis of all possible character pairs composed of the 3664 characters over a text corpus. Then we got over 570,000 entries. Each entry contains four elements. The first two elements are the two Chinese characters, which can stand together; the third element is usage frequency; the fourth element is the conditional probability about the Chinese character pair. Examples for entries in the Statistical Language Model are shown in Fig.1.

About the Statistical Language Model, there are two things that should be clarified. Firstly, the probability of one Chinese character

standing with itself should be included in the language model, because it is not unusual that one Chinese character stands with itself in a Chinese sentence. Secondly, there should be two probabilistic figures corresponding to one pair of Chinese characters, because there may be two kinds of sequence combinations of the two characters.

Character Pair	Usage Frequency	Conditional Probability
新　新	8	0.000327
新　税	263	0.010765
新　制	44	0.001801
新　运	5	0.000205
新　行	1	0.000573
新　到	6	0.000246
新　中	238	0.009742
新　国	19	0.000778

Fig .1 Entries in the Statistic Language Model

3. CHARACTER RECOGNIZER

The Character Recognizer, used to produce character candidate matrices for testing the algorithm, is a kind of Bayes classifier. To implement the classifier, we extracted the ET1 and ET2 of each training sample to form a 130-D feature vector. Then, the Karhunen-Loeve transformation was conducted to reduce the dimension to 64. Maximum Likelihood Estimator is used to find out the parameters. Finally, we use the classifier to recognize the testing samples to verify it.

The Character Recognizer supports a vocabulary of 3664 simplified Chinese characters. The vocabulary covers almost all of the frequently used Chinese characters. We have 280 character samples for each character. Among them, we use 240 samples to train the recognizer. The remaining 40 samples are earmarked for forming experimental sentences for testing our algorithm.

The purpose of our postprocessing algorithm is to pick up characters from a character candidate matrix to form a target sentence. If the correct character is not included into the candidate matrix, there is no way our algorithm can find the correct sentence. Hence, we have to include enough candidates such that the probability of including the correct candidates is high enough, say, greater than 99%. Hence, we should conduct experiment to prove that the Character Recognizer is capable of producing such character candidate matrix. We carried out the experiment over the earmarked 40 testing samples. We asked the recognizer to propose 10 candidates for each character and conducted statistical analysis on the recognition results. The analysis results are summarized in the Table 1. The results show that the recognizer is qualified.

4.POSTPROCESSING PROCESS

4.1 Principles of the Postprocessing algorithm

In this subsection, we first give out the criterion for evaluating sentence candidate. Then present explanation about it.

We build up the criterion for evaluating sentence as following formula

$$[(1-\alpha)*M+\alpha*P]^{1/N} \qquad (1)$$

Here:

M: the product of matching scores of all characters in the sentence. In calculating, we may take log of it, since the product may be too small to calculate.

Number of candidates(n)	Probability for including the true character in the top-n candidates
1	0.867
2	0.953889
3	0.971389
4	0.977500
5	0.981667
6	0.9850
7	0.985556
8	0.987778
9	0.989167
10	0.990278

Table 1. Probability for the Correct Character to Be Included in the n-best Candidates

P: the product of conditional probabilities of all character pairs contained in the sentence. In calculating, we may take log of it, since the product may be too small to calculate.

N: the number of words contained in the sentence

α : the parameter used to balance the impacts of the two terms on the criterion

In the formula, there are two terms. The value of the first term can be obtained from the Character Recognizer. The value of the second term is linguistic information, obtained from the statistical language model.

Assume that a sentence comprises a sequence of character sub-images, $I=I_1I_2.....I_L$. For each sub image I_i, there are n-best candidates (in our case, n=10), $C_{i1},C_{i2},.....C_{in}$ with matching score $d_{i1},d_{i2}.....d_{in}$.

For a character recognizer without linguistic information, the parameter α in the criterion will be zero. The input sentence will be just recognized as C_{11} $C_{21}.....C_{L1}$. In this case, the characters in the sentence are treated separately. The linguistic information between characters is ignored. Actually, when the Character Recognizer faces a sentence instead of an individual character, the linguistic information in the sentence can provide a great help in correctly recognizing the characters in the sentence. Hence, the Character Recognizer should be equipped with language knowledge. In our research, the conditional probability is provided with the Character Recognizer. In this case, the parameter, α, will not be zero. We may use it to balance the impacts of the linguistic information and the matching score on the whole criterion. It should be adjusted in experiments to enable the criterion to be optimal.

Compared to the Character Recognizer without linguistic information, the Character Recognizer with linguistic information can select the best sentence candidate from a wider range—all possible strings {Si} constructed by characters in the candidate matrix.

The conditional probability reflects the probability of two characters occur simultaneously and adjacently. The reason that we select and use the conditional probability in the criterion is based on following features of Chinese language.

From the Statistical Language Model, We found that character pairs which are two-character words usually have much higher usage frequency and conditional probability than those which are not two-character words. Some examples are listed in the Fig.2. Our statistical analysis over the Statistical Language Model shows that the average usage frequency of character pairs which are words is 366, while the average usage frequency of character pairs which are not words is around 45. Also, In Chinese language, all sentences are constructed by words. So, Among the sentence candidates produced from a candidate matrix, only those

sentence candidates that have higher conditional probabilities and contain more words will be the most likely candidates for the true sentence. In fact, in the candidate matrix, only a small part of characters listed in two adjacent columns can form two-character words or a part of multi-character words (containing over two characters); remaining characters form character pairs of no meanings in semantics. For example, in the Fig.3, there are only two words, “一些” (some), and “必然” (necessarily).

Words	usage frequency	cond. prob.	non-words	usage frequency	cond. prob.
人 民	9167	0.126373	人 程	3	0.000041
祖 国	552	0.516370	祖 地	1	0.000935
大 学	2416	0.038207	学 育	3	0.000111
学 习	2152	0.079624	祖 在	7	0.006548
工 作	19748	0.317237	工 错	1	0.000016

Fig. 2 Character Pairs to be or not to be Words

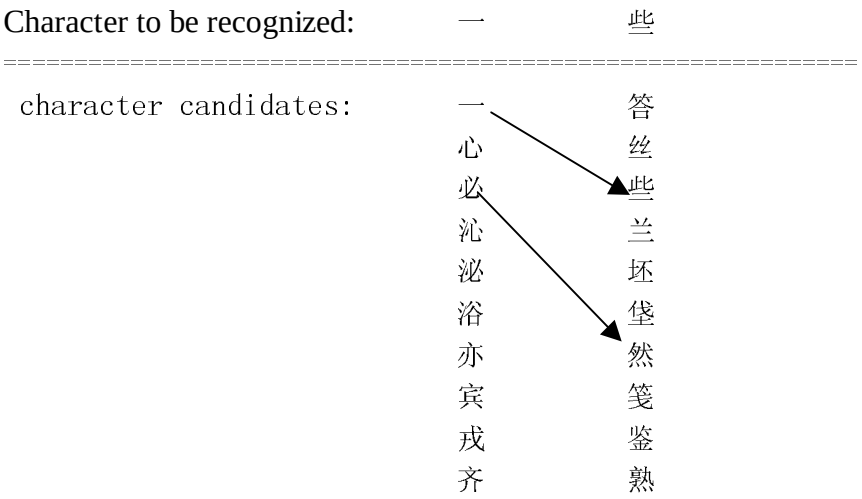

Fig.3 Few Characters Listed in Two Adjacent Columns Can Form Words

4.2 Implementation of the Postprocessing Algorithm

There are two major steps in implementing the process. In the first step, we will form the sentence candidates. In the second step, we will select the best sentence candidate as the postprocessing result.

In the first step, we will form the sentence candidates from the candidate matrix, which can denoted by the notation $[C_{ij}]_{n*m}$. It means that the there are n characters in the sentence, m candidates for each character, which, in our case, is 10.We start with the first candidate of the first character C_{11}, match it with the first candidate of the second character C_{21}, and determine whether they can stand together. If there is a non-zero conditional probability in the language model corresponding to the two characters, then they can stand together to form a character combination. And, we will keep the character combination in an argument array; if not, we will simply discard this combination and proceed to the second character candidate of second character C_{22}. This process will continue until all candidates of the second character are matched with C_{11}. Then, above procedure is repeated with the second candidate of the first character C_{12} and all the candidates of the second character. After each candidate of the first character is matched with each character candidates of the second character, we will get all possible character combinations of the first two columns of character candidates. After that, we can continue the above steps with the newly-created character combinations and with the candidates of the third character. Eventually, step by step, we can reach the last character in the sentence. All complete sentence candidates

are available. Once all the sentence candidates are formed, we will evaluate them with the formula (1), to select the best sentence candidate.

5. PRUNING OF THE UNREASONABLE CHARACTER COMBINATIONS

We note that the number of character combinations will dramatically increase, as we go through the candidate matrix. To prevent the dramatic increase of character combinations and to minimize the processing time, we have to prune the unreasonable character combinations on the way to the last character in the sentence. To achieve this goal, the dynamic programming method is used.

In this method, the sum of the conditional probability and the matching score is used as cost of the transition from current candidate(node) to the next candidate(node). For each step from the first character to the last character, we will preserve top 10 character combinations with largest cost, which finally form 10 sentence candidates. We will select a best one from them as output in term of formula (1).

6. EXPERIMENTAL RESULTS

Up to now, we have conducted experiments over 2000 sentences of various length to test the algorithm, and got following results:

The Character Recognizer without using our algorithm can successfully recognize 780 correct sentence, while the Character Recognizer equipped with our algorithm can successfully recognize up to 1465 correct sentence, meanwhile, reduce number of incorrect characters in 176 wrong sentences.

The total number of characters in the 2000 sentences is 20661. The number of characters correctly recognized by the recognizer without postprocessing algorithm is 17913, with a recognition rate of 86.70%. The number of characters correctly recognized by the recognizer with postprocessing algorithm is 19586, with a recognition rate of 94.80%. The number of characters corrected by the algorithm is 1842. The number of wrong characters caused by the algorithm is 169.

7.FURTHER RESEARCH AND IMPROVEMENT

From the experimental results, we know that current method based on statistical language model and lexicon is not the best one in controlling the number of unreasonable character combinations in the middle of forming complete sentence candidates, and in evaluating the ultimate sentence candidates. To improve performances, the syntactic and semantic technologies will be used in the future research. The syntactic and semantic technologies are more suitable than current method to deal with pruning and evaluating work. Following example illustrate the strength of syntactic method over current method.

大厅入口(hall entrance)

In this short sentence, the Chinese character “入” (enter) may be very easily recognized as “人” (people) due to the similarity between them. If this case occurs, and meanwhile all other three characters are correctly recognized, there are will be two sentence candidates.

大厅入口(hall entrance)

大厅人口(hall population)

For the current method, it has difficulty in telling the correct answer based on statistical language model and word dictionary, because "入口"(entrance) and "人口" (population) both are words in Chinese language, have high joint probabilities and frequently appear in Chinese text. However, from the perspective of syntactic and semantic technologies, it is easy to determine the second sentence candidate is wrong, since in Chinese language the word "大

厅" (hall) followed by the word "人口" (population) is impossible, they are absolutely not compatible in syntax.

8.REFERENCES

[1] F.Jelinek, “Continuous speech recognition by statistical method” Proceedings of the IEEE, 64(4), April 1976.

[2] L.R.Bahl, F.Jelinek and R.L.Mercer., “A maximum likelihood approach to continuous speech recognition” IEEE Trans. On PAMI, PAMI-5(2):179-190.

[3] Hsi-Jian and Cheng-Huang Tung., “A language model based on semantically clustered words in a Chinese character recognition system” Proceedings of the Third International Conference on Document Analysis and Recognition Vol.1 1995.

[4] Pak-Kwong Wong and Chorkin Chan, “Postprocessing statistical language model for a handwritten Chinese character recognizer” IEEE Trans on SMC-part B: Cybernetics. Vol.29 No.2, April 1999

[5] Sun, S.-W. “A contextual postprocessing for optical Chinese character recognition” IEEE International Sympoisum on Circuits and Systems, Vol.5 1991.