A Chinese OCR Spelling Check Approach Based on Statistical Language Models*

Li Zhuang, Ta Bao, Xiaoyan Zhu
DCST, Tsinghua University, Beijing, China
State Key Laboratory of Intelligent Technology and Systems, Beijing, China

Chunheng Wang
Fujitsu R&D Center Co. Ltd., Beijing, China

Satoshi Naoi
Fujitsu Laboratories Ltd., Kawasaki, Japan

Abstract – *This paper describes an effective spelling check approach for Chinese OCR with a new multi-knowledge based statistical language model. This language model combines the conventional n-gram language model and the new LSA (Latent Semantic Analysis) language model, so both local information (syntax) and global information (semantic) are utilized. Furthermore, Chinese similar characters are used in Viterbi search process to expand the candidate list in order to add more possible correct results. With our approach, the best recognition accuracy rate increases from 79.3% to 91.9%, which means 60.9% error reduction.*

Keywords: OCR spelling check, post-processing, multi-knowledge based language model, n-gram, LSA, Chinese similar characters.

1 Introduction

OCR (Optical Character Recognition) means that a computer analyses character images automatically to achieve the text information, and OCR engine is an ICR (Independent Character Recognition) engine usually. It recognizes characters in every position of an image, and gives all the possible candidate results. In ICR recognition process, only image information is used, and the results contain many incorrect words. So the spelling correction approaches, which play the most important role in OCR post-processing, are required.

Familiar spelling check approaches are often based on language knowledge, and mainly include rule-based methods and statistic-based methods. Rule-based methods use rule sets, which describe some exact dictionary knowledge such as word or character frequency, part-of-speech information [7] and some other syntax or morphological features [16] of a language, to detect dubious areas and generate candidate words list. This kind of methods achieves significant success in some special domains, but it is difficult to deal with open natural language. On the other hand, statistic-based methods often use a language model that is achieved by using some language knowledge and analysing a huge of language phenomena on large corpus [8][12][14][15], so more context information is utilized, and this kind of methods is suitable for general domains. Moreover, by using language models, the candidate list can be automatically adjusted; thus we can get the correct resulta more exactly and quickly. Sometimes, rule-based methods and statistic-based methods are used together to achieve better performance [7].

According to the development of natural language processing, using more knowledge from language itself is required. That is to say semantic information must be introduced into language models. Since now the the most widely used language model for OCR spelling check is n-gram models, and no semantic knowledge is involved, in this paper, we construct a multi-knowledge based statistical language model with some semantic information introduced by using LSA language model [3] , and put it into a Chinese OCR spelling check task. Furthermore, we add some Chinese similar characters information in Viterbi search process to expand the candidate list in order to add more possible correct results. The experiment results show that after using the language model and Chinese similar characters information, the recognition accuracy rate increases from 79.3% to 91.9%, which means 60.9% error reduction.

In section 2 of this paper, we will review some related works, including the n-gram language model and the LSA language model. In section 3, we will introduce the multi-knowledge based language models and Chinese similar characters information we use in OCR post-processing. Experiment results and conclusion are in section 4 and 5, respectively.

2 Related works

2.1 N-gram language model

Statistical language models always use the product of conditional probabilities to compute the appearance probability of a sentence. Suppose s denotes a sentence, $w_1 w_2 ... w_t$ is the words sequence of s, and h_i is all history information before w_i, and then the appearance

probability of sentence s is

$$P(s) = P(w_1 w_2 ... w_t) = \prod_{i=1}^{t} P(w_i|h_i) \quad (1)$$

The most widely adopted language model is the n-gram language model [13]. It supposes the word w_i only has relations with the $n-1$ immediately preceding words, and so formula (1) can be rewritten as

$$P(s) = \prod_{i=1}^{t} P(w_i|h_i) = \prod_{i=1}^{t} P(w_i|w_{i-n+1} w_{i-n+2} ... w_{i-1}) \quad (2)$$

In n-gram models, n is often constrained to 2 or 3, which means bigram and trigram respectively.

N-gram language model only considers the sequence of a word string, but no meanings of the words. That is to say, it involves syntax information only, without semantic information.

2.2 LSA language model

LSA (Latent Semantic Analysis) method has been presented long ago [6], but only combined with statistical language model in recent years and now is a hot topic.

When to consider an occurrence probability of a word based on semantic meaning, the content should be understood and then be used to forecast the occurrence probability, which means using the degree of how congruous the word and history information are. The occurrence probabilities of a word in different documents are not the same, because the type of documents restricts the using of words in it. Some words that have relations always occur in the same type of documents. LSA language model is such a model that studies the relations between words and types of documents. These kinds of relations can be got in the model, and that shows the meaning of latent semantic analysis.

LSA language model analyses the training data by constructing the relations between words and types of documents. First, the following matrix is generated.

$$W = \begin{pmatrix} w_{11} & w_{12} & \cdots & w_{1N} \\ w_{21} & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ w_{M1} & \cdots & \cdots & w_{MN} \end{pmatrix} \quad (3)$$

Each row of W is corresponding to a word, and each column of W is corresponding to a document in the training corpus. The value of the crossing of a row and a column is the occurrence times of the corresponding word in the corresponding document. For example, $W_{ij} = t$ means word i appears t times in document j.

The size of matrix W is $M \times N$, and it is rather huge for common training corpus. So SVD (Singular Value Decomposition) method is used to resolve this problem. For any W, there exists the following SVD [4]

$$W = U \sum V^T \quad (4)$$

where $U^T U = V^T V = I_{\min(M,N)}$, $\sum = diag(\sigma_1, \sigma_2, \cdots, \sigma_{\min(M,N)})$ and $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_{\min(M,N)}) \geq 0$.

For reduction of dimensions, we denote that $U = \{u_1, u_2, \cdots, u_{\min(M,N)}\}$ and $V = \{v_1, v_2, \cdots, v_{\min(M,N)}\}$, then let

$$\tilde{W} = \sum_{k=1}^{R} u_k \cdot \sigma_k \cdot v_k^T = \tilde{U} S \tilde{V}^T \quad (5)$$

$\tilde{W}$ is a matrix reconstructed by the R maximal singular values of W. Where $\tilde{U} = \{u_1, u_2, \cdots, u_R\}$, $\tilde{V} = \{v_1, v_2, \cdots, v_R\}$, and $S = diag(\sigma_1, \sigma_2, \cdots, \sigma_R)$. $\tilde{W}$ is the best rank-R approximation to W for any unitarily invariant norm

$$\min_{rank(A) \leq R} \|W - A\| = \left\|W - \tilde{W}\right\| = \sigma_{R+1} \quad (6)$$

where $\|\cdot\|$ refers to L_2 norm [4]. So we can use $\tilde{W}$ to substitute W, and the sizes of $\tilde{U}$, $\tilde{V}$, S are much smaller than W.

When we consider $\tilde{W} = (\tilde{U}S)\tilde{V}^T$, namely treat $\tilde{V}^T$ as a group of orthodoxy vectors of R-dimensional LSA space, and this decomposition is a projection of row vectors of $\tilde{W}$ onto the space. Each row of $\tilde{U}S$presents the coordinate of the corresponding row vector of $\tilde{W}$ in the space, and it is the coordinate of the corresponding word in the space, too. As the same, when we consider $\tilde{W} = \tilde{U}(S\tilde{V}^T)$, this decomposition is a projection of column vectors of $\tilde{W}$ onto the R-dimensional space that is constructed by orthodoxy vectors of $\tilde{U}$. Each column of $S\tilde{V}^T$ presents the coordinate of the corresponding document in the space . After SVD, every word has a corresponding coordinate in the space, and every document also has one. Thus the distance of two words or two documents can be calculated easily [1][2].

In the R-dimensional space of the corresponding word, the history information ($q-1$ appeared words) can be used to construct a vector $\vec{d}_{q-1}$ (n-dimensional vector), where each dimension is a value that indicates the times of the corresponding word appeared in the history. The corresponding coordinate of this vector in LSA space is

$$\dot{\vec{v}}_{q-1} = \vec{d}_{q-1}^T U \quad (7)$$

where $\dot{\vec{v}}_{q-1}$ presents the history in LSA space, and we can get the occurrence probabilities of words after the history by calculating the distance between words and $\dot{\vec{v}}_{q-1}$. Here the following formula is used:

$$K(w_q, \vec{d}_{q-1}) = \cos(u_q S^{\frac{1}{2}}, \dot{\vec{v}}_{q-1} S^{\frac{1}{2}})$$

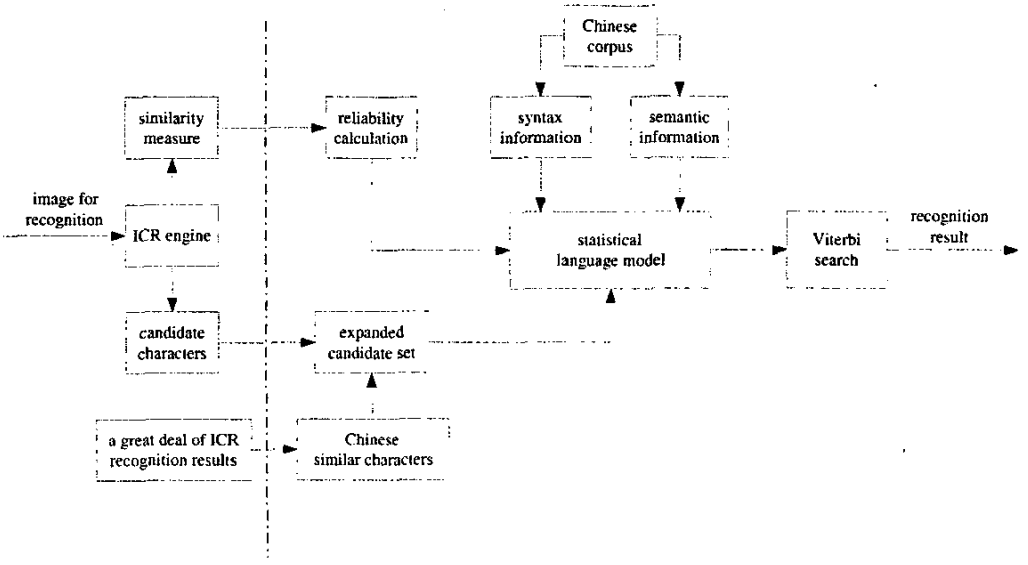

Figure 1: Our OCR system framework with statistical language model in post-processing

$$= \frac{u_q S \dot{\bar{v}}_{q-1}}{\left\| u_q S^{\frac{1}{2}} \right\| \left\| \dot{\bar{v}}_{q-1} S^{\frac{1}{2}} \right\|} \quad (8)$$

That the distance is closed to 1 means the word has a strong relation with the history, and closed to 0 means the word has a weak relation with the history. The occurrence probability of every word can be calculated by normalizing the sum of all distance between the words and the history. Furthermore, the distribution function of LSA language model can be presented as [5][9]:

$$P_{LSA}(w_q|h_1^{q-1}) = \frac{K(w_q,\bar{d}_{q-1}) - Min\{K(w,\bar{d}_{q-1})|w\in\varphi\}}{\sum_{w_q\in\varphi}(K(w_q,\bar{d}_{q-1}) - Min\{K(w,\bar{d}_{q-1})|w\in\varphi\})} \quad (9)$$

where h_1^{q-1} presents the history information of w_q .

In practical terms, the formula (9) can be modified as

$$P_{LSA}(w_q|h_1^{q-1}) = \frac{K(w_q,\bar{d}_{q-1}) - Min\{K(w,\bar{d}_{q-1})|w\in\varphi\} + \varepsilon}{\sum_{w_q\in\varphi}(K(w_q,\bar{d}_{q-1}) - Min\{K(w,\bar{d}_{q-1})|w\in\varphi\} + \varepsilon)} \quad (9')$$

where ϵ is a very small number added to avoid the phenomenon that for some w_q, $K(w_q,\bar{d}_{q-1}) - Min\{K(w,\bar{d}_{q-1})|w\in\varphi\} = 0$

3 The language models used in the OCR post-processing

3.1 Overview of our OCR system framework

Our OCR system framework can be shown as Fig.1. On the left of the dash dotted line is an OCR engine without post-processing, and the post-processing module is shown as the right part of the line. The kernel of the module is the statistical language model, which utilizes both local information (syntax) and global information (semantic).

3.2 A multi-knowledge based language model

The perplexity of LSA language model is much higher than that of n-gram language model. This fact indicates that though the idea of LSA language model is very outperform, the performance of a language model just using the global history information without the local history information is rather poor. That is because the difference of thousands kinds of words can express the same meaning using different sentence structures. So we build a multi-knowledge based language model that combines LSA language model using the global information and conventional n-gram language model using the local information together.

The following formula [3] is used to calculate the distribution function P_C of the combined language model:

$$P_C(w_q|h_1^{q-1}) = \frac{P_N(w_q|w_{q-n+1},\cdots,w_{q-1})P_{LSA}(\bar{d}_{q-1}|w_q)}{\sum_{w_i\in\varphi}P_N(w_i|w_{q-n+1},\cdots,w_{q-1})P_{LSA}(\bar{d}_{q-1}|w_i)} \quad (10)$$

where P_N is the distribution function of the n-gram language model, and P_{LSA} is the distribution function of the LSA language model.

In application, the importance of different language model should be considered. Because the local history information is more important than the global history information, n-gram language model is primary and

LSA language model is secondary. So formula (10) is modified as following with Bayes formula used:

$$P_C(w_q|h_1^{q-1}) = \frac{P_N(w_q|w_{q-n+1},\cdots,w_{q-1})^{\lambda}\cdot[\frac{P_{LSA}(w_q|\bar{d}_{q-1})}{P(w_q)}]^{(1-\lambda)}}{\sum_{w_i\in\varphi} P_N(w_i|w_{q-n+1},\cdots,w_{q-1})^{\lambda}\cdot[\frac{P_{LSA}(w_i|\bar{d}_{q-1})}{P(w_i)}]^{(1-\lambda)}} \quad (11)$$

where the parameter λ is used to control the weight of the two language models in the combination. it is trained via minimizing the perplexity of the model on a held-out corpus.

The computational complexity of training an LSA model is very high. In order to improve the efficiency, word clustering [11] is used. Then the conditional probability can be calculated with the following formula:

$$P(w_q|\bar{d}_{q-1}) = \sum_{k=1}^{K} P(w_q|C_k)P(C_k|\bar{d}_{q-1}) \quad (12)$$

where C_k denotes the kth class. In our application, the C-means clustering algorithm is used first to coarsely partition the words into a few classes, and then the bottom-up clustering algorithm is used in every class to get final clustering results. With word clustering used, the computational cost decreases significantly.

3.3 Chinese similar characters information

In Chinese, some characters are recognized as some other characters usually because their shapes are similar very much. Such characters are called Chinese similar characters.

Because an OCR engine depends mainly on image information, and the similar characters often produce confusion, in our application, Chinese similar characters information is introduced together. Here an MLE (Maximum Likelihood Estimation) based method [10] is used to generate the sets of similar characters. Let F denotes the first choice in the candidate list given by ICR engine, and C denotes the correct character, then the approach of similar characters generation can be described as following:

(1) Record $n(F,C)$ in the training corpus. $n(F,C)$ is the times that F is the first choice in the candidate list while the right character is C.

(2) Calculate the confusion probability $P(C|F) = n(F,C)/n(F)$, where $n(F)$ is the times that F is the first choice in the candidate list given by ICR engine.

(3) Sort all possible C according to the probability $P(C|F)$.

(4) Save the first M possible characters as the similar characters of F, and save their confusion probabilities.

Chinese similar characters are used in the Viterbi search process, where all the $M+1$ candidates, which include the first choice given by ICR engine and its M similar characters, are used in the Viterbi search process to expand the candidate list to increase the chance that the correct choice appears in the list.

4 Experiment and discussion

4.1 Experiment

The training corpus used here involves a lot of domains and contains about 100 million characters. Trigram model and LSA model are built respectively, and then the multi-knowledge based language model (λ = 0.88, 100 word classed) and the conventional n-gram language model are compared on the test corpus, which contains 1018 characters.

The results of the experiments, one is only for language models and the other is with Chinese similar characters used together on the test corpus, are shown as Tab.1 and Tab.2, respectively. Trigram and trigram+LSA (described in 3.2) are chosen for language models test. In the experiments only for language models test, the perplexities of trigram language model and trigram+LSA language model are 185.3 and 183.9, respectively.

We use the approach in 3.3 to generate Chinese similar characters and let M=5. That is to say, all the 6 characters for each candidate are added to the Viterbi search process.

The best recognition accuracy rate increases from 79.3% to 91.9%, and the recognition error rate decreases about 60.9%. That indicates the errors of OCR results can be efficiently reduced by using language model and similar characters information together.

4.2 Disccusion

The results of the experiments seem to show that the conventional n-gram language model outperforms our multi-knowledge based language model. We believe it is due to the fact that the semantic information is more sensitive to the content of the candidate list than the syntax information. So it is required that the candidate list includes more possible correct characters when the semantic information is used in a language model. From the results above, we can see that after adding the similar characters information, the absolute increment of recognition accuracy rate of trigram language model is 0.7%, while of trigram+LSA language model the absolute increment is 1.2%. That shows the validity of using similar characters to expand the candidate list, especially when using the semantic information. It can be expected that the multi-knowledge based language model outperforms the conventional n-gram language model if the candidate list includes more possible correct characters.

5 Conclusion

In our approach for Chinese OCR spelling check, we construct a multi-knowledge based language model in

Table 1: Results of the experiment only for language models on the test corpus

Model	Trigram	Trigram+LSA
Former correct chars	807	807
Former correct rate	79.3%	79.3%
Correct chars with LM	928	920
Correct rate with LM	91.2%	90.3%
Improvement of absolutely correct rate	11.9%	11%

Table 2: Results of the experiment with Chinese similar characters used on the test corpus

Model	Trigram + Similar characters	Trigram + LSA + Similar characters
Former correct chars	807	807
Former correct rate	79.3%	79.3%
Correct chars with LM	935	931
Correct rate with LM	91.9%	91.5%
Improvement of absolutely correct rate	**12.6%**	12.2%

order to use more knowledge from language itself. We use conventional n-gram language model to introduce local information (syntax), and LSA language model to introduce global information (semantic), then compare our multi-knowledge based model with the conventional n-gram model. Though the experiment results seem to show that the n-gram model outperforms, we could expect our multi-knowledge based language model performs better if the candidate list includes more possible correct characters.

In our approach, Chinese similar characters are used in the Viterbi search process. They expand the candidate list and add some more correct characters in it. Similar characters information makes an efficient improvement in the recognition accuracy rate, and it is very important when the semantic information is used in a language model.

After using language models and Chinese similar characters information in an OCR spelling check task, the best recognition accuracy rate increases form 79.3% to 91.9%. That means 60.9% error reduction is achieved.

6 Acknowledgments

This work was jointly supported by the Natural Science Foundation of China (Grants No. 60272019 and 60321002).

References

[1] Jerome R. Bellegarda, "A multispan language modeling framework for large vocabulary speech recognition", IEEE Trans. On Speech Audio Processing, September 1998, vol. 6, pp. 456-467.

[2] Jerome R. Bellegarda, "Large vocabulary speech recognition with multispan statistical language models", IEEE Trans. On Speech And Audio Processing, January 2000, Vol. 8, pp. 76-84.

[3] Jerome R. Bellegarda, "Exploiting latent semantic information in statistical language modeling", Proceedings of the IEEE, 2000, 88(8): 1279-1296.

[4] Berry MW, Dumais ST, et al, "Using linear algebra for intelligent information retrieval", SIAM Rev, 1995, 37(4): 573-595.

[5] Noah Coccaro, Daniel Jurafsky, "Towards Better Integration Of Semantic Predictors In Statistical Language Modeling", Proceedings of ICSLP-98, Sydney.

[6] S. Deerwester, S. T. Dumais, et al, "Indexing by latent semantic analysis", J. Amer. Soc. Inform. Sci., vol. 41, pp. 391-407, 1990.

[7] Andrew R. Golding, Y. Schabes, "Combining trigram-based and feature-based methods for context-sensitive spelling correction", Proceedings of the 34th Annual Meeting of the ACL, Santa Cruz, CA.

[8] Rong Jin, Alex G. Hauptmann, ChengXiang Zhai, "A content-based probabilistic correction model for OCR document retrieval", The 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval Workshop Program (SIGIR 2002), Tampere, Finland, August 11-15, 2002.

[9] Kilyoun Kim, Key-Sun Choi, "Dimension-reduced estimation of word co-occurrence probility", Proceeding of the Conference on ACL2000.

[10] Y.X. Li, X.Q. Ding, et al, "Post-processing study of Chinese document recognition based on HMM", Journal of Chinese Information Processing, 1999, 13(4): 29-34. (in Chinese)

[11] Sven Martin, Jorg Liermann, Hermann Ney, "Algorithms for bigram and trigram word clustering", Speech Communication, 1998, vol. 24, pp. 19-37.

[12] Surapant Meknavin, "Combining trigram and winnow in Thai OCR error correction", Proceedings of the Seventieth International Conference on Computational Linguistics (COLING-ACL '98), Montreal, Canada, August, pp. 836-842.

[13] Gerasimos Potamianos, Frederick Jelinek, "A Study of n-gram and decision tree letter language modeling methods", Speech Communication 24 (1998) pp. 171-192.

[14] X. Tong, D. A. Evans, "A statistical approach to automatic OCR error correction in context", Proceedings of the Fourth Workshop on Very Large Corpora (WVLC-4).

[15] X. Tong, ChengXiang Zhai, et al, "OCR correction and query expansion for retrieval on OCR data–CLARIT TREC-5 Confusion Track Report", Proceeding of the fifth Text Retrieval Conference TREC-5, NIST Special Publication 500-238, 1996, pp. 341-345.

[16] Pornchai Tummarattananont, "Improvement of Thai OCR error correction using overlapping constraints to correction suggestion", The Fifth Symposium on Natural Language Processing 2002 and Oriental COCOSDA Workshop 2002.