Automatic English-to-Chinese Postal Mail Address Translation System*

Xiao Tu[1,2], Yue Lu[1,2]
[1] Dept of Computer Science and Technology,
East China Normal University, Shanghai 200241, China
[2] Shanghai Research Institute of China Post Group,
Shanghai 200062, China
e-mail:[1]tree_tx@yahoo.com.cn, [2]ylu@cs.ecnu.edu.cn

Abstract

Making use of the character recognition technology and machine translation technology, the present system provides a feasible solution to supersede the manual labor work of translating English-written destination address of postal mails to Chinese. Due to the fact that errors in any OCR systems are unavoidable, high performance of translating the OCRed address texts with possible errors to their correct Chinese is a challenge issue in the system. We propose a rule-based address translation method employing the inexact word matching technology to automatically translate postal mails' destination address from English to Chinese. The present system has been implemented and applied in the letter sorting machine, which has demonstrated the proposed method is capable of effectively diminishing the influence of OCR errors.

Keywords:*Address translation, postal address, English-to-Chinese translation, OCRed text, inexact word matching*

1 Introduction

Machine Translation (MT) is a developing technology to automatically translate one natural language to another. Since the Georgetown University did the first machine translation experiment under the cooperation of IBM in 1954, there have been many researches and achievements in this field [1-2]. However, few of MT systems deal with translation of texts obtained by OCR. The OCR technology is practically applied, which achieves high recognition rate of over 95% with low error rate of less than 5%. Machine translation technology combined with OCR technology has a broad application prospect, especially in postal automation.

Increasing international social and economic activities results in a continual increase of postal mails sent from abroad to China. The English-written addresses of these postal mails have to be translated into Chinese by specially trained workers for the purpose of postal delivery. For example, Shanghai Postal Mail Sorting Center has to manually translate international mails of over 30, 000 every day. The processing volume still continues rising in recently years. To deal with the dramatic volume of mails, we have developed an automatic English-to-Chinese postal mail address translation system which employs the technologies of optical character recognition and machine translation to translate English-written destination addresses on postal mails to their corresponding Chinese.

1.1 System Overview

The system is composed of five units: Mail Feeder, Image Acquisition, Chinese Address Printing, Mail Sorting, and Address Translation Unit, as showed in Figure 1. Mail Feeder separates postal mails one by one to form a mail steam passing through the following units. Image Acquisition Unit captures each mail's image, as an example showed in Figure 2, and than sends it to the Address Translation Unit for English-to-Chinese translation. The Address Translation Unit detects the position of the mail's destination address (showed in Figure 3), recognizes all the characters by OCR, extracts the address information and translates the address to Chinese printed on mail envelopes. Finally the Mail Sorting Unit sends the mail to a particular stack according to its destination address.

1.2 Challenges

The following text was obtained by OCR technology from the image of the destination address area showed in

*The work is partially supported by the Science and Technology Commission of Shanghai Manicipality under research grant 09510701800.

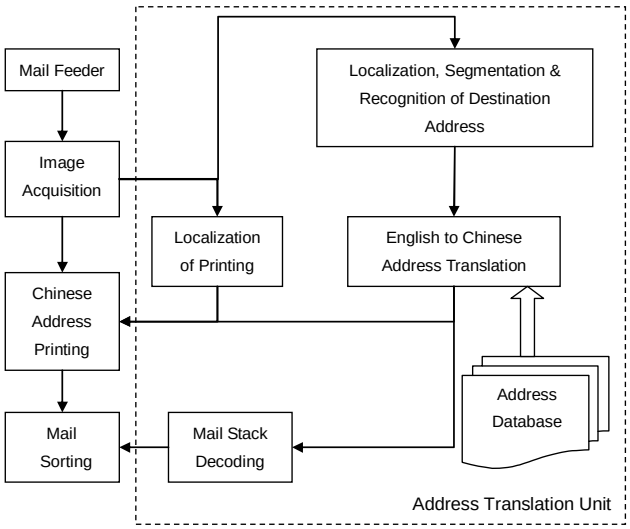


Figure 1: Diagram of Automatic English-to-Chinese Address Translation System.

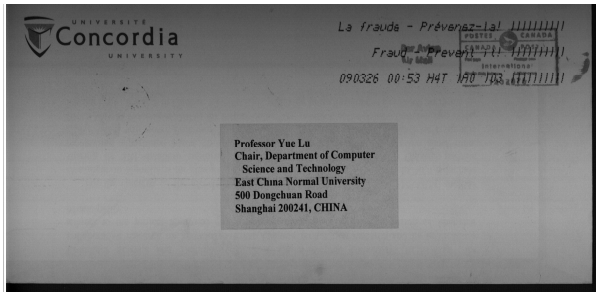


Figure 2: Image of International Post Mail.

Figure 3: Destination Address of Post Mail.

Figure 3:
Professor␣Yue␣Lu
Chair,␣Department␣of␣Computer
␣Science␣and␣Technology
East␣Ch*m*a␣Norma*I*␣University
500␣Dong*e*huan Road
Shanghai␣200241,␣CHINA,
where a '␣' stands for a blank. The Italic-type letters are OCR errors. Obviously, 'in' in 'China' was misrecognized as 'm', the lowercase letter 'l' in 'Normal' was misrecognized as the uppercase letter 'I', and 'c' in 'Dongchuan' was misrecognized as 'e'. The errors are unavoidable in any OCR system, which results in a challenging issue for automatic translation of OCRed texts. In this paper, we propose a mail address translation method using the inexact word matching technology to reduce the influence caused by OCR errors.

At present most practical machine translation systems are rule-based[1,2], even Corpus-based MT systems need language rules. The statistic MT systems[3,4] and the example-based MT systems[5] can not work well without making full use of language rules. Since the rules vary among different MT systems, this paper will elaborate the related language rules that are employed in the present system. Considering that a Chinese address may be expressed in different ways in English and sometimes mixed with homonymic Chinese Pinyin, we specify the related language rules to translate the addresses of those international letters mailed to Shanghai. Based on the inexact word matching technology and the rules, we propose an address understanding method to convert an unstructured address text gotten from the OCR system to structured address information items. Then a unique Chinese address comes out by the comparison of structured address information items and records in the post address database.

The rest of the paper is organized as follows. In Section 2 the main idea of the address translation method based on inexact word matching is presented. Experimental results are given in Section 3. And Section 4 draws out some conclusions.

2 Proposed Address Translation Method

2.1 Address Understanding

A mail address contains a POSTCODE, a CITY, a DISTRICT, a ROAD, a ZONE, a BUILDING, NUMBERS (including a room number, a building number, a lane number and a floor number), a COMPANY and ADDRESSEES and so on. Each of them is called an Address Information Item. Based on the keywords such as 'Road', 'Room', 'Building' and 'University' in an address and the inexact word matching method, an unstructured address text is converted to a set of address information items before being translated into Chinese.

2.1.1 POS Tagging and Disambiguation

Each word's part of speech (POS) in an address text should be determinate before the conversion.

Tagset. We define a tagset of POS, a lexicon and the disambiguation rules. There are sixteen classes of POS and three subclasses of 'NoKeyword' in Table 1.

Table 1: Tageset

Part of Speech	
CityKeyword(CK)	CityName(CN)
DistrictKeyword(DK)	DistrictName (DN)
RoadKeyword(RK)	OrientalKeyword(OL)
BuildingKeyword (BK)	ZoneKeyword (ZK)
CompanyKeyword (CPK)	TitleKeyword (TK)
FloorKeyword (FK)	Postcode (PC)
Number (NB)	Punctuation (PN)
LetterString (LS)	
NoKeyword (NK)	
a. NoKeyword1 (NK1)	
b. NoKeyword2 (NK2)	
c. NoKeyword3 (NK3)	

Lexicon. In the lexicon, each lexeme is defined as $Lexeme_{No} < Word >< MatchingThreshold >< POS >< Rule_{No} >$.

$< MatchingThreshold >$ is a parameter used to verify whether a word is the same as the $< Word >$ in a lexeme. The computation of the similarity between two words plays an important role in our address translation system. Without loss of generality, the two words are considered as two character strings. For a string A of length m and a string B of length n, $V(i,j)$ is defined as the similarity value of the prefixes$[a_1, a_2, ..., a_i]$ and $[b_1, b_2, ..., b_j]$. The similarity of A and B is precisely the value $V(m,n)$. The similarity of two strings A and B can be computed by dynamic programming with recurrences[6]. The base conditions are:

$$V(i,j) = 0, 0 \leq i \leq m, 0 \leq j \leq n \tag{1}$$

The general recurrence relation is:
For $0 \leq i \leq m, 0 \leq j \leq n$

$$V(i,j) = \max \begin{cases} V(i-1,j-1) + \sigma(a_i, b_j), \\ V(i-1,j), \\ V(i,j-1), \end{cases} \tag{2}$$

where $\sigma(a_i, b_j)$ is the matching score between the two letters a_i and b_j, which is defined in the system as

$$\sigma(a_i, b_j) = \begin{cases} 2, a_i = b_j, \\ -2, a_i \neq b_j. \end{cases} \tag{3}$$

And the similarity between A and B is

$$Sim(A,B) = \frac{V(m,n)}{\tilde{V}_A}, \tag{4}$$

where$\tilde{V}_A = m \times \sigma(a_i, a_i) = 2m$ is the similarity between A and itself. If $Sim(A,B) \geq \theta$, where θ is a predefined threshold, B is considered the same as A.

The matching thresholds of the keywords are set as 1.0, while those of the others are set as 0.8. $< POS >$ lists $< Word >$'s all possible parts of speech. $< RuleNo >$ provides disambiguation rules to identify a word's POS in context. Given a word w, the similarity between w and the $< Word >$ in each lexeme is measured. Suppose that $Sim(w, w_i)$ is the maximum in all similarity values and w_i is the $< Word >$ in $Lexeme_i$, if $Sim(w, w_i)$ is larger than the $< MatchingThreshold >$ in $Lexeme_i$, the POS of w depends on $< POS >$ and $< Rule_{No} >$ in the same lexeme. And if there are two and more POSs in the $< POS >$ of $Lexeme_i$, the disambiguation rule with the $< Rule_{No} >$ is used to determinate the word's POS. Furthermore, those words that are not covered in the lexicon are 'LetterString'.

Disambiguation Rules. In the disambiguation rules, a word's POS depends both on the POSs of its previous word and its next one [7]. We represent the rule as
$< Rule_{No} >< Word >$
$< [Condition_1], [Condition_{1'}], POS_1 >$
$< [Condition_2], [Condition_{2'}], POS_2 >$
...
$< [Condition_m], [Condition_{m'}], POS_m >$
$< POS_0 >$,
where $[Condition_x]$ is a subset of the tagset, denoted by $POS_1|POS_2|...|POS_k$ (a '|' is a logic or). There are a couple of conditions $Condition_x$ and $Conditon_{x'}$ in each pair of angle brackets. If the previous word's POS is in the $[Condition_x]$ and the next word's POS is in the $[Condition_{x'}]$, the current word is POS_x. If not, consider the couple of conditions in the next pair of angle brackets. If all the couples of conditions fail, then the current word is POS_0.

2.1.2 Address Information Items

After every word's POS is tagged, an unstructured address text is converted to nine address information items by utilizing Deterministic Finite Automata (DFA). Here, address information items can be divided into two classes. One class is called Alphabet String, such as 'ADDRESSEES', 'COMPANY(or UNIVERSITY)', 'ZONE', 'BUILDING(or Department)', 'ROAD', 'POSTCODE', 'CITY' and 'DISTRICT'; the other is called Number String which is a ordered set of 'NUMBERS's.

(A) Alphabet String

Take 'ROAD' as an example. And there are six kinds of expression about a 'ROAD':

(a) LS LS ... LS RK e.g. Pu Dong Boulevard

(b) LS LS ... LS OL RK e.g. Beijing East Road

(c) LS LS ... LS NB RK e.g. Rui Jin 1 Lu

(d) OL LS LS ... LS RK e.g. West Zhong Shan Road

(e) LS LS ... LS RK OL e.g. Tian Shan Road West

(f) LS LS ... LS No NB RK OL e.g. Zhong Shan No 2 Road East

An alphabet string 'ROAD' consists of three parts, viz. a prefix, a road keyword and a suffix. A prefix is a sequcence of words before a road keyword in a 'ROAD' while a suffix is a sequcence of words after the keyword. Two DFAs are applied to determine the prefix and the suffix, showed as DFA1 and DFA2 in Figure 4 (a) and (b), respectively. An address text is separated into two sequcences of words by its road keyword, called S_1 and S_2 respectively. The longest subsequence of the S_1 in reverse order that DFA1 accepts is the prefix of a road in reverse order, while the longest subsequence of S_2 that DFA2 accepts is the suffix. Take 'Room 301 No 329 Nan Jing Road West Shanghai 200031 China'as an example of a OCRed address. 'Road' is a 'RoadKeyword', 'Jing Nan 329 No 301 Room'is S_1 in reserve order, and 'West shanghai 200031 China' is S_2. Obviously, DFA1 accepted 'Jing Nan' as the longest subsequence and DFA2 accepted 'West' as the longest one. Since that the prefix is 'Nan Jing' and the suffix is 'West', a 'Road' is 'Nan Jing Road West'.

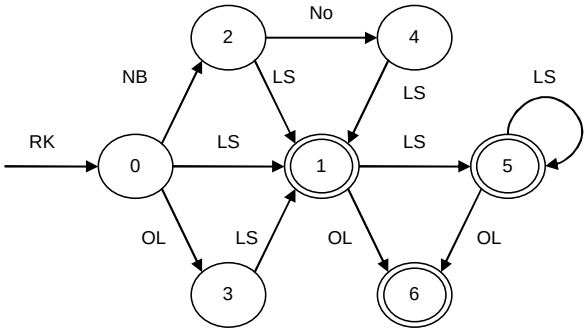
Other Alphabet Strings like 'ADDRESSEES', 'COMPANY', 'ZONE', 'BUILDING', 'POSTCODE', 'CITY' and 'DISTRICT' are also extracted by their corresponding DFAs.

(B) Number String

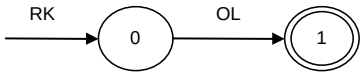
A number phase, such as 'Room 301' and 'No 329', consists of a 'NoKeyword' (or a 'FloorKeyword') and a number. And 'NUMBER's in an address text were sorted to form a Number String, such as '329\301'.

2.2 Address Translation

In the post address database, each address record has nine address information items with their corresponding Chinese expression. To translate an address, the similarity between its address understanding result and each record in the database is calculated.



(a) DFA1 for the Prefix(Reverse)



(b) DFA2 for the Suffix

Figure 4: Destination Address of Post Mail.

$AddrX(Sec_1, Sec_2, ..., Sec_9)$ is the address understanding result of an address, where $Sec_j(1 \leq j \leq 9)$ denotes a postcode, a city name, a district name, a road name, a zone name, a building name, numbers, a company name and addressee names, respectively. And $DB_k(Item_1, Item_2, ..., Item_9, CItem_1, CItem_2, ..., CItem_9)$ denotes a record in the database, where $1 \leq k \leq N$ (N is the total amount of the records in the database), $Item_j(1 \leq j \leq 9)$ denotes a postcode, a city name, a district name, a road name, a zone name, a building name, numbers, a company name and addressee names in the address record. And $CItem_j(1 \leq j \leq 9)$ is the Chinese expression of $Item_j$. The similarity value between $AddrX$ and DB_k is defined as

$$\varphi_k(AddrX, DB_k) = \frac{\sum_{j=1}^{9} Sim(Sec_j, Item_j)}{m} \quad (5)$$

where m is the number of nonempty address information items in $AddrX$ and $Sim(Sec_j, Item_j)$ is the similarity between Sec_j and $Item_j$ calculated by Eq(4).

Let$\varphi_i = \max(AddrX, DB_k)$. λ denotes a predefined address matching threshold, which ranges from 0 to 1. If $\varphi_i \geq \lambda$, $AddrX$ matches DB_i and $DB_i(CItem_1, CItem_2, ..., CItem_9)$ is the Chinese expression of $AddrX$, or there is no translation result for $AddrX$.

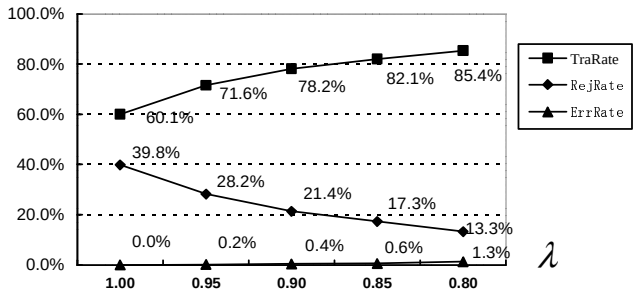


Figure 5: Comparison of the System Performance with Different λ.

3 System Implementation

We implemented the presented system with C++ on Windows 2000. The experiments with the proposed method were carried on 10,000 real envelop images. We respectively selected 1.00, 0.95, 0.90, 0.85 and 0.80 as the value of the address matching threshold λ .

Figure 5 illustrates the translation performance with different value of the threshold λ. Here, TraRate is short for translation rate which is a ratio of the amount of postal address with accurate Chinese expression to the total amount of images. RejRate, short for rejection rate, is a percentage of the address without translation result in all addresses. And ErrRate, short for error rate, is equal to the amount of postal address mistranslated divided by the total amount of images. It is obvious that the lower the value of threshold was, the higher translation rate the system achieved. And the error rate of the system increased with the decrease of the threshold value. The value of translation rate was higher at $\lambda = 0.80$ than the one at $\lambda = 0.85$, but the latter error rate was 1.3% more than double of the former one 0.6%. As such, $\lambda = 0.85$ was appropriate.

The address understanding result of Figure3 is: AddrX (‘200241’, ‘Shanghai’, ‘Dongehuan Road’, ‘500’, ‘Department of Computer Science and Technology’, ‘East Chma NormaI University’, ‘Professor Yue Lu’). And the record with the greatest similarity value is DB_i, whose English Items are ‘200241’, ‘Shanghai’, ‘Dongchuan Road’, ‘500’, ‘Department of Computer Science and Technology’, ‘East China Normal University’, ‘Professor Yue Lu’.

Each value of the similarity between $AddrX$’s Sec_j and DB_i’ s $Item_j$ is in Table 2.

The similarity between them is $\varphi_i = 0.9515 \geq \lambda$, so $AddrX$ and DB_i are matched. The Chinese expression of DB_i items, as listed in Table 3, are the translation results of $AddrX$.

To make delivery possible, just the road and the number were printed on the mail envelope, as showed in Figure 6.

Table 2: Similarity between $AddrX$ and DB_i

Items	$AddrX$	DB_i	Similarity value
POSTCODE	200241	200241	1.0000
CITY	Shanghai	Shanghai	1.0000
ROAD	Dongehuan Road	Dongchuan Road	0.8571
NUMBERS	500	500	1.0000
UNIVERSITY	East Chma NormaI University	East China Normal University	0.8036
DEPARTMENT	Department of Computer Science and Technology	Department of Computer Science and Technology	1.0000
ADDRESSEE	Professor Yue Lu	Professor Yue Lu	1.0000

Table 3: DB_i’s Items and CItems

Items	CItems
200041	200041
Shanghai	上海
Dongchuan Road	东川路
500	500
Department of Computer Science and Technology	计算机科学与技术系
East China Normal University	华东师范大学
Professor Yue Lu	吕岳教授

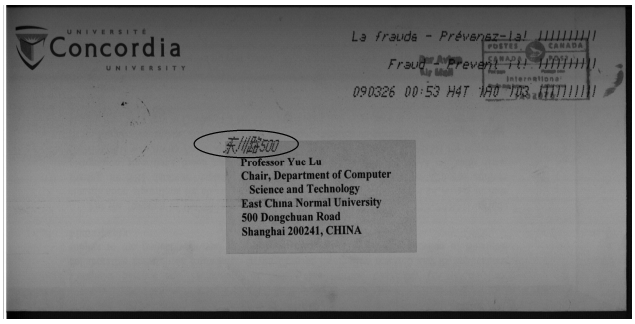


Figure 6: Chinese Address Printed on Envelope.

4 Conclusion

The present Automatic English-to-Chinese Postal Mail Address Translation System is a novel attempt in both machine translation and postal automation. To solve the influence caused by OCR errors, we propose an address translation method based on the inexact word matching technology. The experimental results have showed that the system has achieved a translation rate of 82% with a low error rate and proved the proposed method is effective and feasible. The system has been already put into application in Shanghai Post Office. In further work, entropy and relationship of each address information item will be taken into account to further improve the robustness of address understanding.

References

[1] Z. W. Feng, *Studies of Sic-tech Translation*, China Translation and Publishing Corporation, Beijing,2004.

[2] T. J. Zhang, *Theories of Machine Translation*, Harbin Industrial University Press, Harbin,2000.

[3] P. F. Brown, S.D.Pietra, et al., 'The Mathematics of Statistical Machine Translation: Parameter Estimation', *Computational Linguistics*, 1993, vol 19, No.2, pp.263-311.

[4] F. J. Och, H. Ney, 'Discriminative Training and Maximum Entropy Models for Statistical Machine Translation', In: *Proc. of the 40th Annual Meeting of the Association for Computational Linguistics*, Philadelphia, PA, 2002, pp.295-302.

[5] M. Nagao, 'A framework of a mechanical translation between Japanese and English by analogy principle', *Artificial and Human Intelligence*, NATO Publications,1984, pp.173-180.

[6] P. Sellers, 'The Theory and Computation of Evolutionary Distance', *Pattern Recognition, Journal of Algorithms*,1980, vol 1, pp.359-373.

[7] Z. P. Du, B. M. Wu, L. H. Zhang, et al, 'Rule-Based POS Tagging in English-Chinese MT System', *Journal of Information Engineering University*, 2003, vol 4, No.3, pp.89-92.