Automatic Extraction of English-Chinese Transliteration Pairs using Dynamic Window and Tokenizer

Chengguo Jin
Dept. of Graduate School for Information Technology, POSTECH, Korea
chengguo@postech.ac.kr

Dong-Il Kim
Language Engineering Institute, YUST, China
dongil@ybust.edu.cn

Seung-Hoon Na
Dept. of Computer Science & Engineering
POSTECH, Korea
nsh1979@postech.ac.kr

Jong-Hyeok Lee
Dept. of Computer Science & Engineering
POSTECH, Korea
jhlee@postech.ac.kr

Abstract

Recently, many studies have been focused on extracting transliteration pairs from bilingual texts. Most of these studies are based on the statistical transliteration model. The paper discusses the limitations of previous approaches and proposes novel approaches called dynamic window and tokenizer to overcome these limitations. Experimental results show that the average rates of word and character precision are 99.0% and 99.78%, respectively.

1 Introduction

Machine transliteration is a type of translation based on phonetic similarity between two languages. Chinese Named entities including foreign person names, location names and company names, etc are usually transliterated from foreign words. The main problem of transliteration resulted from complex relations between Chinese phonetic symbols and characters. Usually, a foreign word can be transliterated into various Chinese words, and sometimes this will lead to transliteration complexity. In addition, dozens of Chinese characters correspond to each pinyin which uses the Latin alphabet to represent sounds in Standard Mandarin. In order to solve these problems, Chinese government published the “Names of the world's peoples”[12] containing 630,000 entries in 1993, which took about 40 years. However, some new foreign names still cannot be found in the dictionary. Constructing an unknown word dictionary is a difficult and time consuming job, so in this paper we propose a novel approach to automatically construct the resource by efficiently extracting transliteration pairs from bilingual texts.

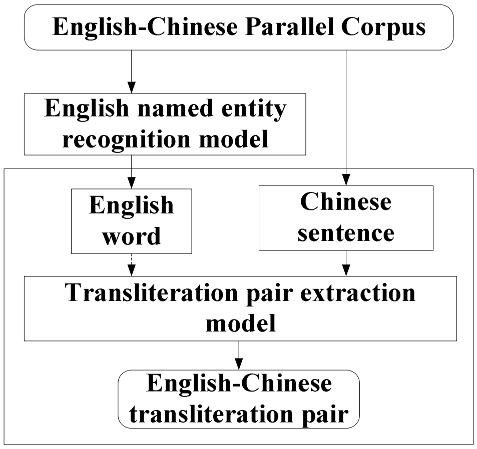
Recently, much research has been conducted on machine transliteration. Machine transliteration is classified into two types. One is automatic generation of transliterated word from the source language [6]; the other one is extracting transliteration pairs from bilingual texts [2]. Generally, the generation process performs worse than the extraction process. Especially in Chinese, people do not always transliterate foreign words only by sound but also consider the meanings. For example, the word ‘blog’ is not transliterated into ‘布劳哥’ (BuLaoGe) which is phonetically equivalent to the source word, but transliterated into ‘博客’(BoKe) which means ‘a lot of guests’. In this case, it is too difficult to automatically generate correct transliteration words. Therefore, our approach is based on the method of extracting transliteration pairs from bilingual texts.

The type of extraction of transliteration pairs can also be further divided into two types. One is extracting transliteration candidates from each language respectively, and then comparing the phonetic similarities between those candidates of two languages [2, 8]. The other one is only extracting transliteration candidates from the source language, and using the candidates to extract corresponding transliteration words from the target language [1]. In Chinese, there is no space between two words and no special character set to represent foreign words such as Japanese; hence the candidate extraction is difficult and usually results in a low precision. Therefore, the method presented in [2] which extracted transliteration candidates from both English and Chinese result in a poor performance. Compared to other works, Lee[1] only extracts transliteration candidates from English, and finds equivalent Chinese transliteration words without extracting candidates from Chinese texts. The method works well, but the performance is required to be improved. In this paper we present a novel approaches to obtain a remarkable result in extracting transliteration word pairs from parallel texts.

The remainder of the paper is organized as follows: Section 2 gives an overview of statistical machine transliteration and describes proposed approaches. Section 3 describes the experimental setup and a quantitative assessment of performance of our approaches. Conclusions and future work are presented in Section 4.

2 Extraction of English-Chinese transliteration pairs

In this paper, we first extract English named entities from English-Chinese parallel texts, and select only those which are to be transliterated into Chinese. Next we extract Chinese transliteration words from corresponding Chinese texts. [Fig. 1] shows the entire process of extracting transliteration word pairs from English-Chinese parallel texts.

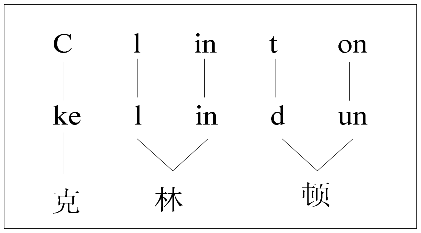

English-Chinese Parallel Corpus
English named entity recognition model
English word
Chinese sentence
Transliteration pair extraction model
English-Chinese transliteration pair

[Fig 1]. The process of extracting transliteration pairs from English-Chinese parallel corpus

2.1 Statistical machine transliteration model

Generally, the Chinese Romanization system pinyin which is used to represent the pronunciation of each Chinese character is adopted in Chinese transliteration related studies. For example, the Chinese word '克林顿' is first transformed to pinyin 'Ke Lin Dun', and we compare the phonetic similarities between 'Clinton' and 'KeLinDun'. In this paper, we assume that E is written in English, while C is written in Chinese, and TU represents transliteration units. So P(C|E), P(克林顿|Clinton) can be transformed to P(KeLinDun|Clinton). In this paper we define English TU as unigram, bigram, and trigram; Chinese TU is pinyin initial, pinyin final and the entire pinyin. With these definitions we can further write the probability, P(克林顿|Clinton), as follows:

$$P(\text{克林顿} \mid Clinton) \cong P(kelindun \mid Clinton)$$
$$\cong P(ke \mid C)P(l \mid l)P(in \mid in)P(t \mid d)P(un \mid on) \quad (1)$$



[Fig 2]. TU alignment between English and Chinese pinyin

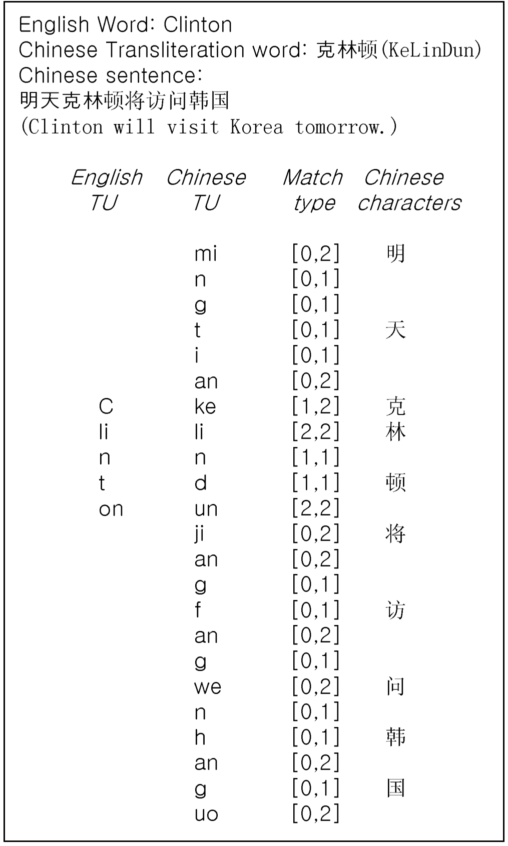
[Fig 2] shows the possible alignment between English word 'Clinton' and Chinese word '克林顿''s pinyin 'KeLinDun'.

In [1], the authors add the match type information in Eq. (1). The match type is defined with the lengths of TUs of two languages. For example, in the case of $P(ke \mid C)$ the match type is 2-1, because the size of Chinese TU ke is 2 and the size of English TU C is 1. Match type is useful when estimating transliteration model's parameters without a pronunciation dictionary. In this paper, we use the EM algorithm to estimate transliteration model's parameters without a pronunciation dictionary, so we applied match type to our model. Add Match type(M) to Eq.(1) to formulate as follows:

$$P(C \mid E) \approx \max_{M} P(C \mid M, E) P(M \mid E)$$
$$\approx \max_{M} P(C \mid M, E) P(M) \quad (2)$$

$$\log P(C \mid E) \approx \max_{M} \sum_{i=1}^{N} \left((\log P(v_i \mid u_i) + \log P(m_i) \right) \quad (3)$$

where u, v are English TU and Chinese TU, respectively and m is the match type of u and v.



English Word: Clinton
Chinese Transliteration word: 克林顿(KeLinDun)
Chinese sentence:
明天克林顿将访问韩国
(Clinton will visit Korea tomorrow.)

English TU	*Chinese TU*	*Match type*	*Chinese characters*
	mi	[0,2]	明
	n	[0,1]	
	g	[0,1]	
	t	[0,1]	天
	i	[0,1]	
	an	[0,2]	
C	ke	[1,2]	克
li	li	[2,2]	林
n	n	[1,1]	
t	d	[1,1]	顿
on	un	[2,2]	
	ji	[0,2]	将
	an	[0,2]	
	g	[0,1]	
	f	[0,1]	访
	an	[0,2]	
	g	[0,1]	
	we	[0,2]	问
	n	[0,1]	
	h	[0,1]	韩
	an	[0,2]	
	g	[0,1]	国
	uo	[0,2]	

[Fig 3]. The alignment of the English word and the Chinese sentence containing corresponding transliteration word

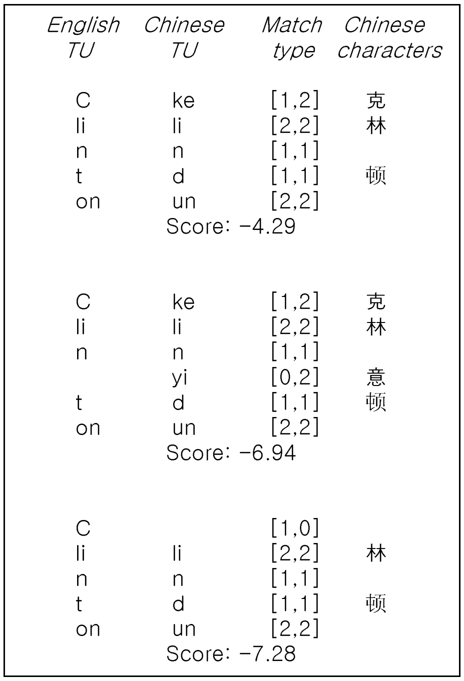
[Fig 3] shows how to extract the correct Chinese transliteration “克林顿”(KeLinDun) with the given English word “Clinton” from a Chinese sentence.

2.2 Proposed methods

When the statistical machine transliteration is used to extract transliteration pairs from a parallel text, the problems arise when there is more than one Chinese character sequence that is phonetically similar to the English word. In this paper we propose novel approaches called dynamic window and tokenizer to solve the problems effectively.

2.2.1 Dynamic window method

The dynamic window approach does not find the transliteration at once, but first sets the window size range according to the English word candidates, and slides each window within the range to find the correct transliterations.



English TU	*Chinese TU*	*Match type*	*Chinese characters*
C	ke	[1,2]	克
li	li	[2,2]	林
n	n	[1,1]	
t	d	[1,1]	顿
on	un	[2,2]	

Score: -4.29

English TU	*Chinese TU*	*Match type*	*Chinese characters*
C	ke	[1,2]	克
li	li	[2,2]	林
n	n	[1,1]	
	yi	[0,2]	意
t	d	[1,1]	顿
on	un	[2,2]	

Score: -6.94

English TU	*Chinese TU*	*Match type*	*Chinese characters*
C		[1,0]	
li	li	[2,2]	林
n	n	[1,1]	
t	d	[1,1]	顿
on	un	[2,2]	

Score: -7.28

[Fig 4]. Alignment result between English word “Clinton” and correct Chinese transliteration, add a character into correct Chinese transliteration, and eliminate a character from correct Chinese transliteration.

If we know the exact Chinese transliteration’s size, then we can efficiently extract Chinese transliterations by setting the window with the length of the actual Chinese transliteration word. For example, in [Fig 4] we do alignment between the English word “Clinton” and correct Chinese transliteration “克林顿”(KeLinDun), add a character into correct Chinese transliteration “克林意顿”(KeLinYiDun), and eliminate a character from correct Chinese transliteration “林顿”(LinDun) respectively. The result shows that the highest score is the alignment with correct Chinese transliteration. This is because the alignment between the English word and the correct Chinese transliteration will lead to more alignments between English TUs and Chinese TUs, which will result in highest scores among alignment with other Chinese sequences. This characteristic does not only exist between English and Chinese, but also exists between other language pairs.

However, in most circumstances, we can hardly determine the correct Chinese transliteration’s length. Therefore, we analyze the distribution between English words and Chinese transliterations to predict the possible range of Chinese transliteration’s length according to the English word. We present the algorithm for the dynamic window approach as follows:

Step 1: Set the range of Chinese transliteration's length according to the extracted English word candidate.

Step 2: Slide each window within the range to calculate the probability between an English word and a Chinese character sequence contained in the current window using Eq 3.

Step 3: Select the Chinese character sequence with highest score and back-track the alignment result to extract the correct transliteration word.

[Fig 5] shows the entire process of using the dynamic window approach to extract the correct transliteration word.

English Word	Ziegler
Chinese Sentence	齐格勒与意大利化学家居里奥共同获得了 1963 年诺贝尔化学奖。
English Sentence	Ziegler and Italian Chemist Julio received the Nobel prize of 1963 together.
Extracted transliteration without using dynamic window	家居里奥 (JiaJuLiAo)
Correct transliteration	齐格勒 (QiGeLe)

<table>
<tr><th colspan="3">Steps</th></tr>
<tr><td colspan="3">1. Set Chinese transliteration's range according to English word "Ziegler" to [2, 7] (After analyzing the distribution between an English word and a Chinese transliteration word, we found that if the English word length is ∟, then the Chinese transliteration word is between ∟/3 and∟.)</td></tr>
<tr><td colspan="3">2. Slide each window to find sequence with highest score.</td></tr>
<tr><td colspan="3">3 Select the Chinese character sequence with highest score and back-track the alignment result to extract a correct transliteration word.</td></tr>
<tr><td>Window size</td><td>Chinese character sequence with highest score of each window (underline the back-tracking result)</td><td>Score (normalize with window size)</td></tr>
<tr><td>2</td><td>奇格 (QiGe)</td><td>-9.327</td></tr>
<tr><td>3</td><td>齐格勒 (QiGeLe)</td><td>-6.290</td></tr>
<tr><td>4</td><td>齐格勒与 (QiGeLeYu)</td><td>-8.433</td></tr>
<tr><td>5</td><td>齐格勒与意 (QiGeLeYuYi)</td><td>-9.719</td></tr>
<tr><td>6</td><td>家居里奥共同 (JiaJuLiAoGongTong)</td><td>-10.458</td></tr>
<tr><td>7</td><td>齐格勒与意大利 (QiGeLeYuYiDaLi)</td><td>-10.721</td></tr>
</table>

[Fig 5]. Extract the correct transliteration using the dynamic window method

The dynamic window approach can effectively solve the problem shown in [Fig 5] which is the most common problem that arises from using statistical machine transliteration model to extract a transliteration from a Chinese sentence. However, it can not handle the case that a correct transliteration with correct window size can not be extracted. Moreover, when the dynamic window approach is used, the processing time will increase severely. Hence, the following approach is presented to deal with the problem as well as to improve the performance.

2.2.2 Tokenizer method

The tokenizer method is to divide a sentence with characters which have never been used in Chinese transliterations and applies the statistical transliteration model to each part to extract a correct transliteration.

There are certain characters that are frequently used for transliterating foreign words, such as"施(shi)，德(de)，勒(le)，赫(he) …". On the other hand, there are other characters, such as "是(shi)，的(de)，了(le)，和(he),…", that have never been used for Chinese transliteration, while they are phonetically equivalent with the above characters. These characters are mainly particles, copulas and non-Chinese characters etc., and always come with named entities and sometimes also cause some problems. For example, when the English word "David" is transliterated into Chinese, the last phoneme is omitted and transliterated into "大卫"(DaWei). In this case of a Chinese character such as "的"(De) which is phonetically similar with the omitted syllable 'd', the statistical transliteration model will incorrectly extract "大卫的"(DaWeiDe) as transliteration of "David". In [1], the authors deal with the problem through a post-process using some linguistic rules. Lee and Chang [1] merely eliminate the characters which have never been used in Chinese transliteration such as "的"(De) from the results. Nevertheless, the approach cannot solve the problem shows in [Fig 6], because the copula "是"(Shi) combines with the other character "者"(zhe) to form the character sequence "者是"(ZheShi) which is phonetically similar with the English word "Jacey", and is incorrectly recognized as a transliteration of "Jacey". Thus, in this case, although the copula "是"(Shi) is eliminated from the result through the post-process method presented in [1], the remaining part is not the correct transliteration. Compared with the method in [1], our tokenizer approach eliminates copula "是"(Shi) at pre-processing time and then the phonetic similarity between "Jacey" and the remaining part "者"(Zhe) becomes very low; hence our approach overcomes the problem prior to the entire process. In addition, the tokenizer approach also reduces the processing time dramatically due to separating a sentence into several parts. [Fig 6] shows the process of extracting a correct transliteration using the tokenizer method.

English Word	Jacey
Chinese Sentence	这本书的作者是佩尼娜汤姆森和杰西格雷厄姆。
English Sentence	The authors of this book are Peninah Thomson and Jacey Grahame.
Incorrectly extracted transliteration	者是(ZheShi)
Correct transliteration	杰西(JieXi)

Steps

1. Separate the Chinese sentence with characters, "这, 的, 是, 和" (including non-Chinese characters such as punctuation, number, English characters etc.), which have never been used in Chinese transliteration as follows:
本书的作者是佩尼娜汤姆森和杰西格雷厄姆

2. Apply statistical transliteration model to each part and select the part with highest score, and back-track the part to extract a correct transliteration.

No.	Chinese character sequence of each part (underline the back-tracking result)	Score (normalize with window size)
1	本书 (BenShu)	-24.79
2	作者 (ZuoZhe)	-15.83
3	佩尼娜汤姆森 (PeiNiNaTangMuShen)	-16.32
4	杰西格雷厄姆 (JieXi)	-10.29

[Fig 6]. Extracting the correct transliteration using the tokenizer method.

In conclusion, the two approaches complement each other; hence using them together will lead to a better performance.

3 Experiments

In this section, we focus on the setup for the experiments and a performance evaluation of the proposed approaches to extract transliteration word pairs from parallel corpora.

3.1 Experimental setup

We use 300 parallel English-Chinese sentences containing various person names, location names, company names etc. The corpus for training consists of 860 pairs of English names and their Chinese transliterations. The performance of transliteration pair extraction was evaluated based on precision and recall rates at the word and character levels. Since we consider exactly one proper name in the source language and one transliteration in the target language at a time, the word recall rates are the same as the word precision rates. In order to demonstrate the effectiveness of our approaches, we perform the following experiments: firstly, only use STM(Statistical transliteration model) which is the baseline of our experiment; secondly, we apply the dynamic window and tokenizer method with STM respectively; thirdly, we apply these two methods together; at last, we perform experiment presented in [1] to compare with our methods.

3.2 Evaluation of dynamic window and tokenizer methods

[table 1]. The experimental results of extracting transliteration pairs using proposed methods

Methods	Word precision	Character precision	Character recall
STM (baseline)	75.33%	86.65%	91.11%
STM+DW	96.00%	98.51%	99.05%
STM+TOK	78.66%	85.24%	86.94%
STM+DW+TOK	99.00%	99.78%	99.72%
STM+CW	98.00%	98.81%	98.69%
STM+CW+TOK	99.00%	99.89%	99.61%

As shown in table 1, the baseline STM achieves a word precision rate of 75%. The STM works relatively well with short sentences, but as the length of sentences increases the performance significantly decreases. The dynamic window approach overcomes the problem effectively. If the dynamic window method is applied with STM, the model will be tolerant with the length of sentences. The dynamic window approach improves the performance of STM around 21%, and reaches the average word precision rate of 96% (STM+DW). In order to estimate the highest performance that the dynamic window approach can achieve, we apply the correct window size which can be obtained from the evaluation data set with STM. The result (STM+CW) shows around 98% word precision rate and about 23% improvement over the baseline. Therefore, dynamic window approach is remarkably efficient; it shows only 2% difference with theoretically highest performance. However, the dynamic window approach increases the processing time too much.

When using tokenizer method (STM+TOK), only about 3% is approved over the baseline. Although the result is not considerably improved, it is extremely important that the problems that the dynamic window method cannot solve are managed to be solved. Thus, when using both dynamic window and tokenizer methods with STM (STM+DW+TOK), it is found that around 3% improvement is achieved over using only the dynamic window (STM+DW), as well as word precision rates of 99%.

[table 2]. Processing time evaluation of proposed methods

Methods	Processing time
STM (baseline)	5 sec (5751 milisec)
STM+DW	2min 34sec (154893 milisec)
STM+TOK	4sec (4574 milisec)
STM+DW+TOK	32sec (32751 milisec)

Table 2 shows the evaluation of processing time of dynamic window and tokenizer methods. Using the dynamic window leads to 27 times more processing time than STM, while using the tokenizer method with the dynamic window method reduces the processing time around 5 times than the original. Hence, we have achieved a higher precision as well as less processing time by combining these two methods.

3.3 Comparing experiment

In order to compare with previous methods, we perform the experiment presented in [1]. Table 3 shows using the post-processing method presented in [1] achieves around 87% of word precision rates, and about 12% improvement over the baseline. However, our methods are 11% superior to the method in [1].

[Table 3] Comparing experiment with previous work

Methods	Word Precision	Character Precision	Character Recall
STM (baseline)	75.33%	86.65%	91.11%
STM+DW+TOK	99.00%	99.78%	99.72%
STM+[1]'s method	87.99%	90.17%	91.11%

4 Conclusions and future work

In this paper, we presented two novel approaches called dynamic window and tokenizer based on the statistical machine transliteration model. Our approaches achieved high precision without any post-processing procedures. The dynamic window approach was based on a fundamental property, which more TUs aligned between correct transliteration pairs. Also, we reasonably estimated the range of correct transliteration's length to extract transliteration pairs in high precision. The tokenizer method eliminated characters that have never been used in Chinese transliteration to separate a sentence into several parts. This resulted in a certain degree of improvement of precision and significantly reduction of processing time. These two methods are both based on common natures of all languages; thus our approaches can be readily port to other language pairs.

In this paper, we only considered the English words that are to be transliterated into Chinese. Our work is ongoing, and in near future, we will extend our works to extract transliteration pairs from large scale comparable corpora. In comparable corpora, there are many uncertainties, for example, the extracted English word may be not transliterated into Chinese or there may be no correct transliteration in Chinese texts. However, with large comparable corpora, a word will appear several times, and we can use the frequency or entropy information to extract correct transliteration pairs based on the proposed perfect algorithm.

Acknowledgement

This work was supported by the Korea Science and Engineering Foundation (KOSEF) through the Advanced Information Technology Research Center (AITrc), also in part by the BK 21 Project and MIC & IITA through IT Leading R&D Support Project in 2007.

Reference

[1] C.-J. Lee, J.S. Chang, J.-S.R. Jang, Extraction of transliteration pairs from parallel corpora using a statistical transliteration model, in: Information Sciences 176, 67-90 (2006)

[2] Richard Sproat, Tao Tao, ChengXiang Zhai, Named Entity Transliteration with Comparable Corpora, in: Proceedings of the 21st International Conference on Computational Linguistics. (2006)

[3] J.S. Lee and K.S. Choi, "English to Korean statistical transliteration for information retrieval," International Journal of Computer Processing of Oriental Languages, pp.17–37, (1998).

[4] K. Knight, J. Graehl, Machine transliteration, Computational Linguistics 24 (4), 599–612, (1998).

[5] W.-H. Lin, H.-H. Chen, Backward transliteration by learning phonetic similarity, in: CoNLL-2002, Sixth Conference on Natural Language Learning, Taipei, Taiwan, (2002).

[6] J.-H. Oh, K.-S. Choi, An English–Korean transliteration model using pronunciation and contextual rules, in: Proceedings of the 19th International Conference on Computational Linguistics (COLING), Taipei, Taiwan, pp. 758–764, (2002).

[7] C.-J. Lee, J.S. Chang, J.-S.R. Jang, A statistical approach to Chinese-to-English Backtransliteration, in: Proceedings of the 17th Pacific Asia Conference on Language, Information, and Computation (PACLIC), Singapore, pp. 310–318, (2003).

[8] Jong-Hoon Oh, Sun-Mee Bae, Key-Sun Choi, An Algorithm for extracting English-Korean Transliteration pairs using Automatic E-K Transliteration In Proceedings of Korean Information Science Socieity (Spring). (In Korean), (2004).

[9] Jong-Hoon Oh, Jin-Xia Huang, Key-Sun Choi, An Alignment Model for Extracting English-Korean Translations of Term Constituents, Journal of Korean Information Science Society, SA, 32(4), (2005)

[10] Chun-Jen Lee, Jason S. Chang, Jyh-Shing Roger Jang: Alignment of bilingual named entities in parallel corpora using statistical models and multiple knowledge sources. ACM Trans. Asian Lang. Inf. Process. 5(2): 121-145 (2006)

[11] Lee, C. J. and Chang, J. S., Acquisition of English-Chinese Transliterated Word Pairs from Parallel-Aligned Texts Using a Statistical Machine Transliteration Model, In. Proceedings of HLT-NAACL, Edmonton, Canada, pp. 96-103, (2003).

[12] Xinhua Agency, Names of the world's peoples: a comprehensive dictionary of names in Roman-Chinese (世界人名翻译大辞典), (1993)