

A High-Accurate Chinese-English NE Backward Translation System Combining Both Lexical Information and Web Statistics

Conrad Chen

Hsin-Hsi Chen

Department of Computer Science and Information Engineering, National
Taiwan University, Taipei, Taiwan

drchen@nlg.csie.ntu.edu.tw hhchen@csie.ntu.edu.tw

Abstract

Named entity translation is indispensable in cross language information retrieval nowadays. We propose an approach of combining lexical information, web statistics, and inverse search based on *Google* to backward translate a Chinese named entity (NE) into English. Our system achieves a high Top-1 accuracy of 87.6%, which is a relatively good performance reported in this area until present.

1 Introduction

Translation of named entities (NE) attracts much attention due to its practical applications in World Wide Web. The most challenging issue behind is: the genres of NEs are various, NEs are open vocabulary and their translations are very flexible.

Some previous approaches use phonetic similarity to identify corresponding transliterations, i.e., translation by phonetic values (Lin and Chen, 2002; Lee and Chang, 2003). Some approaches combine lexical (phonetic and meaning) and semantic information to find corresponding translation of NEs in bilingual corpora (Feng et al., 2004; Huang et al., 2004; Lam et al., 2004). These studies focus on the alignment of NEs in parallel or comparable corpora. That is called “close-ended” NE translation.

In “open-ended” NE translation, an arbitrary NE is given, and we want to find its corresponding translations. Most previous approaches exploit web search engine to help find translating candidates on the Internet. Al-Onaizan and Knight (2003) adopt language models to generate

possible candidates first, and then verify these candidates by web statistics. They achieve a Top-1 accuracy of about 72.6% with Arabic-to-English translation. Lu et al. (2004) use statistics of anchor texts in web search result to identify translation and obtain a Top-1 accuracy of about 63.6% in translating English out-of-vocabulary (OOV) words into Traditional Chinese. Zhang et al. (2005) use query expansion to retrieve candidates and then use lexical information, frequencies, and distances to find the correct translation. They achieve a Top-1 accuracy of 81.0% and claim that they outperform state-of-the-art OOV translation techniques then.

In this paper, we propose a three-step approach based on *Google* to deal with open-ended Chinese-to-English translation. Our system integrates various features which have been used by previous approaches in a novel way. We observe that most foreign Chinese NEs would have their corresponding English translations appearing in their returned snippets by *Google*. Therefore we combine lexical information and web statistics to find corresponding translations of given Chinese foreign NEs in returned snippets. A highly effective verification process, *inverse search*, is then adopted and raises the performance in a significant degree. Our approach achieves an overall Top-1 accuracy of 87.6% and a relatively high Top-4 accuracy of 94.7%.

2 Background

Translating NEs, which is different from translating common words, is an “asymmetric” translation. Translations of an NE in various languages can be organized as a tree according to the relations of translation language pairs, as shown in Figure 1. The root of the translating tree is the NE in its original language, i.e., initially de-

nominated. We call the translation of an NE along the tree downward as a “forward translation”. On the contrary, “backward translation” is to translate an NE along the tree upward.

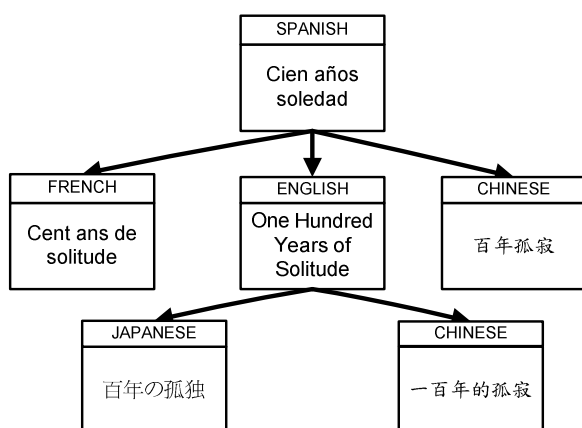


Figure 1. Translating tree of “Cien años soledad”.

Generally speaking, forward translation is easier than backward translation. On the one hand, there is no unique answer to forward translation. Many alternative ways can be adopted to forward translate an NE from one language to another. For example, “Jordan” can be translated into “喬丹 (Qiao-Dan)”, “喬登 (Qiao-Deng)”, “約旦 (Yue-Dan)”, and so on. On the other hand, there is generally one unique corresponding term in backward translation, especially when the target language is the root of the translating tree.

In addition, when the original NE appears in documents in the target language in forward translation, it often comes together with a corresponding translation in the target language (Cheng et al., 2004). That makes forward translation less challenging. In this paper, we focus our study on Chinese-English backward translation, i.e., the original language of NE and the target language in translation is English, and the source language to be translated is Chinese.

There are two important issues shown below to deal with backward translation of NEs or OOV words.

- Where to find the corresponding translation?
- How to identify the correct translation?

NEs seldom appear in multi-lingual or even mono-lingual dictionaries, i.e., they are OOV or unknown words. For unknown words, where can we find its corresponding translation? A bilingual corpus might be a possible solution. However, NEs appear in a vast context and bilingual corpora available can only cover a small proportion. Most text resources are monolingual. Can

we find translations of NEs in monolingual corpora? While mentioning a translated name during writing, sometimes we would annotate it with its original name in the original foreign language, especially when the name is less commonly known. But how often would it happen? With our testing data, which would be introduced in Section 4, over 97% of translated NEs would have its original NE appearing in the first 100 returned snippets by Google. Figure 2 shows several snippets returned by Google which contains the original NE of the given foreign NE.

[CEPS 思博網-- 文章書目;-1](#)
 篇名, 《老人與海》的象徵手法及作者的人生哲學. 並列篇名, Symbolic Means of the Author "The Old Man and the Sea" ... 摘要, 以象徵分析的方法對《老人與海》中老人、海、大魚等元素的象徵涵義進行了探索和解讀, 分析了海明威在小說中闡述的主題: “ ...
www.ceps.com.tw/ec/ecjnlarticleView.aspx?jnlcattype=1&jnlptype=4&jnltype=29&jnliid=1370&i... - 26k - [頁庫存檔](#) - [類似網頁](#)

[.:JSDVD Mall:. 世界名著-老人與海](#)
 世界名著-老人與海·太陽馬戲團-夢幻人生(DTS)·紐約放電俏姐妹·懷舊電影系列 16-秋決·艾瑪·奪命訓練班·新好男孩-電視演唱會·神鬼認證-特別版 ... 世界名著-老人與海. The Old Man and The Sea. 4715320115018, 我們提供的付款方式 ...
mall.jsdvd.com/product_info.php?products_id=3198 - 48k - [補充資料](#) - [頁庫存檔](#) - [類似網頁](#)

Figure 2. Several Traditional Chinese snippets of “老人與海” returned by Google which contains the translation “The Old Man and the Sea”.

When translations can be found in snippets, the next work would be identifying which name is the correct translation of NEs. First we should know how NEs would be translated. The commonest case is translating by phonetic values, or so-called transliteration. Most personal names and location names are transliterated. NEs may also be translated by meaning. It is the way in which most titles and nicknames and some organization names would be translated. Another common case is translating by phonetic values for some parts and by meaning for the others. For example, “Sears Tower” is translated into “西爾斯 (Xi-Er-Si) 大廈 (tower)” in Chinese. NEs would sometimes be translated by semantics or contents of the entity it indicates, especially with movies. Table 1 summarizes the possible translating ways of NEs. From the above discussion, we may use similarities in phonetic values, meanings of constituent words, semantics, and so

on to identify corresponding translations. Besides these linguistic features, non-linguistic features such as statistical information may also help use

well. We would discuss how to combine these features to identify corresponding translation in detail in the next section.

Translating Way	Description	Examples
Translating by Phonetic Values	The translation would have a similar pronunciation to its original NE.	“New York” and “紐約(pronounced as Niu-Yue)”
Translating by Meaning	The translation would have a similar or a related meaning to its original NE.	“紅(red)樓(chamber)夢(dream)” and “The Dream of the Red Chamber”
Translating by Phonetic Values for Some Parts and by Meaning for the Others	The entire NE is supposed to be translated by its meaning and the name parts are transliterated.	“Uncle Tom’s Cabin” and “湯姆(pronounced as Tang-Mu)叔叔的(uncle’s)小屋(cabin)”
Translating by Both Phonetic Values and Meaning	The translation would have both a similar pronunciation and a similar meaning to its original NE.	“New Yorker” and “紐約(pronounced as Niu-Yue)客(people, pronounced as Ke)”
Translating NEs by Heterography	The NE is translated by these heterographic words in neighboring languages.	“橫濱” and “Yokohama”, “鈴木一朗” and “Ichiro Suzuki”
Translating by Semantic or Content	The NE is translated by its semantic or the content of the entity it refers to.	“The Mask” and “摩登(modern)大(great)聖(saint)”
Parallel Names	NE is initially denominated as more than one name or in more than one language.	“孫中山(Sun Zhong-Shan)” and “Sun Yat-Sen”

Table 1. Possible translating ways of NEs.

3 Chinese-to-English NE Translation

As we have mentioned in the last section, we could find most English translations in Chinese web page snippets. We thus base our system on web search engine: retrieving candidates from returned snippets, combining both linguistic and statistical information to find the correct translation. Our system can be split into three steps: candidate retrieving, candidate evaluating, and candidate verifying. An overview of our system is given in Figure 3.

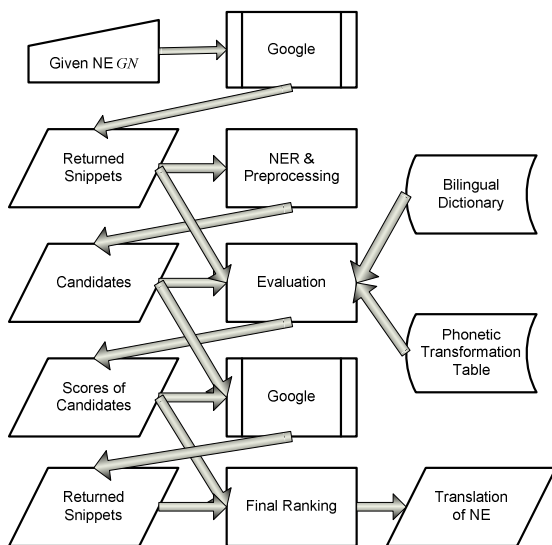


Figure 3. An Overview of the System.

In the first step, the NE to be translated, *GN*, is sent to *Google* to retrieve traditional Chinese web pages, and a simple English NE recognition

method and several preprocessing procedures are applied to obtain possible candidates from returned snippets. In the second step, four features (i.e., phonetic values, word senses, recurrences, and relative positions) are exploited to give these candidates a score. In the last step, the candidates with higher scores are sent to *Google* again. Recurrence information and relative positions concerning with the candidate to be verified of *GN* in returned snippets are counted along with the scores to decide the final ranking of candidates. These three steps will be detailed in the following subsections.

3.1 Retrieving Candidates

Before we can identify possible candidates, we must retrieve them first. In the returned traditional Chinese snippets by *Google*, there are still many English fragments. Therefore, the first task our system would do is to separate these English fragments into NEs and non-NEs. We propose a simple method to recognize possible NEs. All fragments conforming to the following properties would be recognized as NEs:

- The first and the last word of the fragment are numerals or capitalized.
- There are no three or more consequent lowercase words in the fragment.
- The whole fragment is within one sentence.

After retrieving possible NEs in returned snippets, there are still some works to do to make a

finer candidate list for verification. First, there might be many different forms for a same NE. For example, “Mr. & Mrs. Smith” may also appear in the form of “Mr. and Mrs. Smith”, “Mr. And Mrs. Smith”, and so on. To deal with these aliasing forms, we transform all different forms into a standard form for the later ranking and identification. The standard form follows the following rules:

- All letters are transformed into upper cases.
- Words consist “'”s are split.
- Symbols are rewritten into words.

For example, all forms of “Mr. & Mrs. Smith” would be transformed into “MR. AND MRS. SMITH”.

The second work we should complete before ranking is filtering useless substrings. An NE may comprise many single words. These component words may all be capitalized and thus all substrings of this NE would be fetched as candidates of our translation work. Therefore, substrings which always appear with a same preceding and following word are discarded here, since they would have a zero recurrence score in the next step, which would be detailed in the next subsection.

3.2 Evaluating Candidates

After candidate retrieving, we would obtain a sequence of m candidates, C_1, C_2, \dots, C_m . An integrated evaluating model is introduced to exploit four features (phonetic values, word senses, recurrences, and relative positions) to score these m candidates, as the following equation suggests:

$$Score(C_i, GN) = SScore(C_i, GN) \cdot LScore(C_i, GN)$$

$LScore(C_i, GN)$ combines phonetic values and word senses to evaluate the lexical similarity between C_i and GN . $SScore(C_i, GN)$ concerns both recurrences information and relative positions to evaluate the statistical relationship between C_i and GN . These two scores are then combined to obtain $Score(C_i, GN)$. How to estimate $LScore(C_i, GN)$ and $SScore(C_i, GN)$ would be discussed in detail in the following subsections.

3.2.1 Lexical Similarity

The lexical similarity concerns both phonetic values and word senses. An NE may consist of many single words. These component words

may be translated either by phonetic values or by word senses. Given a translation pair, we could split them into fragments which could be bipartite matched according to their translation relationships, as Figure 4 shows.

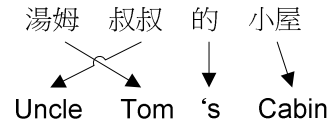


Figure 4. The translation relationships of “湯姆叔叔的小屋”.

To identify the lexical similarity between two NEs, we could estimate the similarity scores between the matched fragment pairs first, and then sum them up as a total score. We postulate that the matching with the highest score is the correct matching. Therefore the problem becomes a weighted bipartite matching problem, i.e., given the similarity scores between any fragment pairs, to find the bipartite matching with the highest score. In this way, our next problem is how to estimate the similarity scores between fragments.

We treat an English single word as a fragment unit, i.e., each English single word corresponds to one fragment. An English candidate C_i consisting of n single words would be split into n fragment units, $C_{i1}, C_{i2}, \dots, C_{in}$. We define a Chinese fragment unit that it could comprise one to four characters and may overlap each other. A fragment unit of GN can be written as GN_{ab} , which denotes the a th to b th characters of GN , and $b - a < 4$. The linguistic similarity score between two fragments is:

$$LSim(GN_{ab}, C_{ij}) = \text{Max}\{PVSIM(GN_{ab}, C_{ij}), WSSIM(GN_{ab}, C_{ij})\}$$

Where $PVSIM()$ estimates the similarity in phonetic values while $WSSIM()$ estimate it in word senses.

■ Phonetic Value

In this paper, we adopt a simple but novel method to estimate the similarity in phonetic values. Unlike many approaches, we don't introduce an intermediate phonetic alphabet system for comparison. We first transform the Chinese fragments into possible English strings, and then estimate the similarity between transformed strings and English candidates in surface strings, as Figure 5 shows. However, similar pronunciations does not equal to similar surface strings. Two quite dissimilar strings may have very similar pronunciations. Therefore, we take this strat-

egy: generate all possible transformations, and regard the one with the highest similarity as the English candidate.

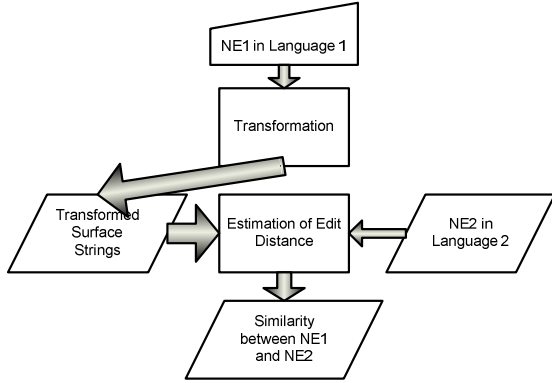


Figure 5. Phonetic similarity estimation of our system.

Edit distances are usually used to estimate the surface similarity between strings. However, the typical edit distance does not completely satisfy the requirement in the context of translation identification. In translation, vowels are an unreliable feature. There are many variations in pronunciation of vowels, and the combinations of vowels are numerous. Different combinations of vowels may have a same phonetic value, however, same combinations may pronounce totally differently. The worst of all, human often arbitrarily determine the pronunciation of unfamiliar vowel combinations in translation. For these reasons, we adopt the strategy that vowels can be ignored in transformation. That is to say when it is hard to determine which vowel combination should be generated from given Chinese fragments, we can only transform the more certain part of consonants. Thus during the calculation of edit distances, the insertion of vowels would not be calculated into edit distances. Finally, the modified edit distance between two strings A and B is defined as follow:

$$\begin{aligned}
 ED_{A \rightarrow B}(0, t) &= t \\
 ED_{A \rightarrow B}(s, 0) &= s \\
 ED_{A \rightarrow B}(s, t) &= \min \left\{ \begin{array}{l} ED_{A \rightarrow B}(s, t-1) + Ins(t), \\ ED_{A \rightarrow B}(s-1, t) + 1, \\ ED_{A \rightarrow B}(s-1, t-1) + Rep(s, t) \end{array} \right\} \\
 Ins(t) &= \begin{cases} 0, & \text{if } B_t \text{ is a vowel} \\ 1, & \text{if } B_t \text{ is a consonant} \end{cases} \\
 Rep(s, t) &= \begin{cases} 0, & \text{if } A_s = B_t \\ 1, & \text{else} \end{cases}
 \end{aligned}$$

The modified edit distances are then transformed to similarity scores:

$$PVSim(A, B) = 1 - \frac{ED_{A \rightarrow B}(Len(A), Len(B))}{\max\{Len(A), Len(B)\}}$$

$Len()$ denotes the length of the string. In the above equation, the similarity scores are ranged from 0 to 1.

We build the fixed transformation table manually. All possible transformations from Chinese transliterating characters to corresponding English strings are built. If we cannot precisely indicate which vowel combination should be transformed, or there are too many possible combinations, we ignore vowels. Then we use a training set of 3,000 transliteration names to examine possible omissions due to human ignorance.

Word Senses

More or less similar to the estimation of phonetic similarity, we do not use an intermediate representation of meanings to estimate word sense similarity. We treat the English translations in the C-E bilingual dictionary (*reference removed for blind review*) directly as the word senses of their corresponding Chinese word entries. We adopt a simple 0-or-1 estimation of word sense similarity between two strings A and B , as the following equation suggests:

$$WSSim(A, B) = \begin{cases} 0, & \text{if } B \text{ is not a translation of } A \\ & \text{in the dictionary} \\ 1, & \text{if } B \text{ is a translation of } A \\ & \text{in the dictionary} \end{cases}$$

All the Chinese foreign names appearing in test data is removed from the dictionary.

From the above equations we could derive that $LSim()$ of fragment pairs is also ranged from 0 to 1. Candidates to be evaluated may comprise different number of component words, and this would result the different scoring base of the weighted bipartite matching. We should normalize the result scores of bipartite matching. As a result, the following equation is applied:

$$\begin{aligned}
 LScore(C_i, GN) &= \\
 \min \left\{ \frac{\sum_{\text{all matched pairs } GN_{ab} \text{ and } C_{ij}} LSim(GN_{ab}, C_{ij})}{\text{Total \# of words in } C_i}, \right. & \\
 \left. \frac{\sum_{\text{all matched pairs } GN_{ab} \text{ and } C_{ij}} LSim(GN_{ab}, C_{ij}) \cdot (b-a+1)}{\text{Total \# of characters in } GN} \right\} &
 \end{aligned}$$

3.2.2 Statistical Similarity

Two pieces of information are concerned together to estimate the statistical similarity: recurrences and relative positions. A candidate C_i might appear l times in the returned snippets, as $C_{i,1}, C_{i,2}, \dots, C_{i,l}$. For each $C_{i,k}$, we find the dis-

tance between it and the nearest GN in the returned snippets, and then compute the relative position scores as the following equation:

$$RP(C_{i,k}, GN) = \frac{1}{\lceil \text{Distance}(GN, C_{i,k}) / 4 \rceil + 1}$$

In other words, if the candidate is adjacent to the given NE, it would have a relative position score of 1. Relative position scores of all $C_{i,k}$ would be summed up to obtain the primitive statistical score:

$$PSS(C_i, GN) = \sum_k RP(C_{i,k}, GN)$$

As we mentioned before, since the imprecision of NE recognition, most substrings of NEs would also be recognized as candidates. This would result a problem. There are often typos in the information provided on the Internet. If some component word of an NE is misspelled, the substrings constituted by the rest words would have a higher statistical score than the correct NE. To prevent such kind of situations, we introduce entropy of the context of the candidate. If a candidate has a more varied context, it is more possible to be an independent term instead of a substring of other terms. Entropy provides such a property: if the possible cases are more varied, there is higher entropy, and vice versa. Entropy function here concerns the possible cases of the most adjacent word at both ends of the candidate, as the following equation suggests:

$$\text{Entropy}(\text{Context of } C_i) = \begin{cases} 1 & , \text{ while \# of possible context} = 1 \\ -\sum_{CT_r} NCT_r / NC_i \cdot \log_{NPT_i} NCT_r / NC_i, & \text{else} \end{cases}$$

Where NCT_r and NC_i denote the appearing times of the r th context CT_r and the candidate C_i in the returned snippets respectively, and NPT_i denotes the total number of different cases of the context of C_i . Since we want to normalize the entropy to 0~1, we take NPT_i as the base of the logarithm function.

While concerning context combinations, only capitalized English word is discriminated. All other words would be viewed as one sort "OTHER". For example, assuming the context of "David" comprises three times of (Craig, OTHER), three times of (OTHER, Stern), and six times of (OTHER, OTHER), then:

$$\begin{aligned} \text{Entropy}(\text{Context of "David"}) = \\ -\left(\frac{3}{12} \log_3 \frac{3}{12} + \frac{3}{12} \cdot \log_3 \frac{3}{12} + \frac{6}{12} \cdot \log_3 \frac{6}{12}\right) = 0.946 \end{aligned}$$

Next we use $\text{Entropy}(\text{Context of } C_i)$ to weight the primitive score $PSS(C_i, GN)$ to obtain the final statistical score.:

$$\begin{aligned} \text{SScore}(C_i, GN) = \\ \text{Entropy}(\text{Context of } C_i) \cdot PSS(C_i, GN) \end{aligned}$$

3.3 Verifying Candidates

In evaluating candidate, we concern only the appearing frequencies of candidates when the NE to be translated is presented. In the other direction, we should also concern the appearing frequencies of the NE to be translated when the candidate is presented to prevent common words getting an improper high score in evaluation. We perform the *inverse search* approach for this sake. Like the evaluation of statistical scores in the last step, candidates are sent to *Google* to retrieve Traditional Chinese snippets, and the same equation of $\text{SScore}()$ is computed concerning the candidate. However, since there are too many candidates, we cannot perform this process on all candidates. Therefore, an elimination mechanism is adopted to select candidates for verification. The elimination mechanism works as follows:

1. Send the Top-3 candidates into *Google* for verification.
2. Count $\text{SScore}(GN, C_i)$. (Notice that the order of the parameter is reversed.) Re-weight $\text{Score}(C_i, GN)$ by multiplying $\text{SScore}(GN, C_i)$
3. Re-rank candidates
4. After re-ranking, if new candidates become the Top-3 ones, redo the first step. Otherwise end this process.

The candidates have been verified would be recorded to prevent duplicate re-weighting and unnecessary verification.

There is one problem in verification we should concern. Since we only consider recurrence information in both directions, but not co-occurrence information, this would result some problem when dealing rarely used translations. For example, "Peter Pan" can be translated into "彼得潘" or "彼德潘" (both pronounced as Bi-De-Pan) in Chinese, but most people would use the former translation. Thus if we send "Peter Pan" to verification when translating "彼德潘", we would get a very low score.

To deal with this situation, we adopt the strategy of disbelieving verification in some situa-

tions. If all candidates have scores lower than the threshold, we presume that the given NE is a rarely used translation. In this situation, we use only $Score(C_n, GN)$ estimated by the evaluation step to rank its candidates, without multiplying $SScore(GN, C_i)$ of the inverse search. The threshold is set to 1.5 by heuristic, since we consider that a commonly used translation is supposed to have their $SScore()$ larger than 1 in both directions.

4 Experiments

To evaluate the performance of our system, 15 common users are invited to provide 100 foreign NEs per user. These users are asked to simulate a scenario of using web search machine to perform cross-lingual information retrieval. The proportion of different types of NEs is roughly conformed to the real distribution, except for creation titles. We gather a larger proportion of creation titles than other types of NEs, since the ways of translating creation titles is less regular and we may use them to test how much help could the web statistics provide.

After removing duplicate entries provided by users, finally we obtain 1,119 nouns. Among them 7 are not NEs, 65 are originated from Oriental languages (Chinese, Japanese, and Korean), and the rest 1,047 foreign NEs are our main experimental subjects. Among these 1,047 names there are 455 personal names, 264 location names, 117 organization names, 196 creation titles, and 15 other types of NEs.

Table 2 and Figure 5 show the performance of the system with different types of NEs. We could observe that the translating performance is best with location names. It is within our expectation, since location names are one of the most limited NE types. Human usually provide location names in a very limited range, and thus there are less location names having ambiguous

translations and less rare location names in the test data. Besides, because most location names are purely transliterated, it can give us some clues about the performance of our phonetic model.

Our system performs worst with creation titles. One reason is that the naming and translating style of creation titles are less formulated. Many titles are not translated by lexical information, but by semantic information or else. For example, “Mr. & Mrs. Smith” is translated into “史密斯任務(Smiths’ Mission)” by the content of the creation it denotes. Another reason is that many titles are not originated from English, such as “Ie Nozze di Figaro”. It results the C-E bilingual dictionary cannot be used in recognizing word sense similarity. A more serious problem with titles is that titles generally consist of more single words than other types of NEs. Therefore, in the returned snippets by *Google*, the correct translation is often cut off. It would results a great bias in estimating statistical scores.

Table 3 compares the result of different feature combinations. It considers only foreign NEs in the test data. From the result we could conclude that both statistical and lexical features are helpful for translation finding, while the inverse search are the key of our system to achieve a good performance.

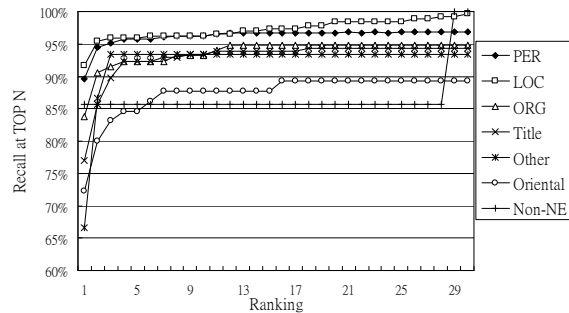


Figure 5. Curve of recall versus ranking.

	Total	Top-1		Top-2		Top-4		Top-M	
		Num	Recall	Num	Recall	Num	Recall	Num	Recall
PER	455	408	89.7%	430	94.5%	436	95.8%	443	97.3%
LOC	264	242	91.7%	252	95.5%	253	95.8%	264	100.0%
ORG	117	98	83.8%	106	90.6%	108	92.3%	114	97.4%
TITLE	196	151	77.0%	168	85.7%	181	92.3%	189	96.4%
Other	15	10	66.7%	13	86.7%	14	93.3%	15	100.0%
All NE	1047	909	87.6%	969	92.6%	992	94.7%	1025	97.9%
Oriental	65	47	72.3%	52	80.0%	55	84.6%	60	92.3%
Non-NE	7	6	85.7%	6	85.7%	6	85.7%	7	100.0%
Overall	1119	962	86.0%	1027	91.8%	1053	94.1%	1092	97.6%

Table 2. Experiment results of our system with different NE types.

	Top-1		Top-2		Top-4	
	Num	Recall	Num	Recall	Num	Recall
<i>SScore</i>	540	51.6%	745	71.2%	887	84.7%
<i>LScore</i>	721	68.9%	789	75.4%	844	80.6%
<i>SScore + LScore</i>	837	79.9%	916	87.5%	953	91.0%
+ <i>Inverse Search</i>	909	87.6%	969	92.6%	992	94.7%

Table 3. Experiment results of our system with different feature combinations.

From the result we could also find that our system has a high recall of 94.7% while considering top 4 candidates. If we only count in the given NEs with their correct translation appearing in the returned snippets, the recall would go to 96.8%. This achievement may be not yet good enough for computer-driven applications, but it is certainly a good performance for user querying.

5 Conclusion

In this study we combine several relatively simple implementations of approaches that have been proposed in the previous studies and obtain a very good performance. We find that the Internet is a quite good source for discovering NE translations. Using snippets returned by *Google* we can efficiently reduce the number of the possible candidates and acquire much useful information to verify these candidates. Since the number of candidates is generally less than processing with unaligned corpus, simple models can perform filtering quite well and the over-fitting problem is thus prevented.

From the failure cases of our system, (see Appendix A) we could observe that the performance of this integrated approach could still be boosted by more sophisticated models, more extensive dictionaries, and more delicate training mechanisms. For example, performing stemming or adopting a more extensive dictionary might enhance the accuracy of estimating word sense similarity; the statistic formula can be replaced by more formal measures such as co-occurrences or mutual information to make a more precise assessment of statistical relationship. These tasks would be our future works in developing a more accurate and efficient NE translation system.

Reference

- Al-Onaizan, Yaser and Kevin Knight. 2002. Translating Named Entities Using Monolingual and Bilingual Resources. *ACL* 2002: 400-408.
- Cheng, Pu-Jen, J.W. Teng, R.C. Chen, J.H. Wang, W.H. Lu, and L.F. Chien. Translating unknown queries with web corpora for cross-language information retrieval. *SIGIR* 2004: 146-153.

Feng, Donghui, Lv Y., and Zhou M. 2004. A New Approach for English-Chinese Named Entity Alignment. *EMNLP* 2004: 372-379.

Huang, Fei, Stephan Vogel, and Alex Waibel. 2003. Improving Named Entity Translation Combining Phonetic and Semantic Similarities. *HLT-NAACL* 2004: 281-288.

Lam, Wai, Ruizhang Huang, and Pik-Shan Cheung. 2004. Learning phonetic similarity for matching named entity translations and mining new translations. *SIGIR* 2004: 289-296.

Lee, Chun-Jen and Jason S. Chang. 2003. Acquisition of English-Chinese Transliterated Word Pairs from Parallel-Aligned Texts. *HLT-NAACL* 2003. Workshop on Data Driven MT: 96-103.

Lin, Wei-Hao and Hsin-Hsi Chen. 2002. Backward Machine Transliteration by Learning Phonetic Similarity. *Proceedings of CoNLL-2002*: 139-145.

Lu, Wen-Hsiang, Lee-Feng Chien, and Hsi-Jian Lee. 2004. Anchor Text Mining for Translation of Web Queries: A Transitive Translation Approach. *ACM Transactions on Information Systems* 22(2): 242-269.

Zhang, Ying, Fei Huang, and Stephan Vogel. 2005. Mining translations of OOV terms from the web through cross-lingual query expansion. *SIGIR* 2005: 669-670.

Zhang, Ying and Phil Vines. 2004. Using the web for automated translation extraction in cross-language information retrieval. *SIGIR* 2004: 162-169.

Appendix A. Some Failure Cases of Our System

GN	Top 1	Correct Translation	Rank
海珊	CBS	SADDAM HUSSEIN	2
紐澤西	JERSEY	NEW JERSEY	2
天方夜譚	ONLINE	ARABIAN NIGHTS	2
勞斯萊斯	ROYCE	ROLLS ROYCE	2
朱利斯厄文	NBA	JULIUS ERVING	2
艾薇兒	LAVIGNE	AVRIL LAVIGNE	2
羅琳	JK	JK. ROWLING	2
塞爾蒂克	RICKY DAVIS	CELTICS	8
印象日出	MONET	IMPRESSION SUNRISE	9
蘇聯	TUPOLEV TU	USSR	33
梅德維登科	NBA	MEDVENDENKO	N/A
命運交響曲	TOS	SYMPHONY NO. 5	N/A
愛的教育	AROUND03	CUORE	N/A
民主黨	JACK LAYTON	DEMOCRATIC PARTY	N/A