國立交通大學

資訊科學與工程研究所

碩 士 論 文

以網路為主之英對中專有名詞翻譯萃取

Empirical Approach to Resolving English to Chinese Named Entity Translation

研究生：張晨輝

指導教授：梁婷　博士

中 華 民 國　一 百 零一年 七 月

以網路為主之英對中專有名詞翻譯萃取

Empirical Approach to Resolving English to Chinese Named Entity Translation

研 究 生：張晨輝　　Student： Chen-Huei Jhang

指導教授：梁 婷 博士　　Advisor：Dr. Tyne Liang

國 立 交 通 大 學
資 訊 科 學 與 工 程 研 究 所
碩 士 論 文

A Thesis

Submitted to Institute of Computer Science and Engineering

College of Computer Science

National Chiao Tung University

in partial Fulfillment of the Requirements

for the Degree of

Master

In

Computer Science

July 2012

Hsinchu, Taiwan, Republic of China

中華民國一百零一年七月

以網路為主之英對中專有名詞翻譯萃取

研究生：張晨輝　　指導教授：梁　婷　博士

國立交通大學

資訊科學與工程研究所

摘要

專有名詞翻譯的成效影響許多自然語言處理的應用，例如跨語言資料檢索、機器翻譯、與自動問答系統等。由於網路資源豐富且更新迅速，近年專有名詞翻譯研究多利用搜索引擎回傳的網頁片段萃取翻譯候選詞，並根據候選詞與專有名詞在搜尋結果中的頻率、距離與二者的詞長比例等特徵，使用監督式學習模組或非監督式學習排選候選詞。有鑑於各領域的專有名詞有各自的命名規則，而先前研究較少考慮此點，因此本論文提出利用搜尋結果萃取翻譯候選詞並以命名規則協助搜尋詞擴展與候選詞評量。

在本論文中，我們考量四個領域的英對中譯名，分別是書名、電影名、醫藥名、和公司名等。所提的方法分三個階段進行:首先，我們使用 13 種特徵並以支援向量機模組(SVM)進行專有名詞領域辨識;然後，根據已定義好的領域命名規則做搜尋詞擴展;最後，我們利用制定好的表面樣式萃取候選詞，且依造頻率與命名規則排序候選詞。在實驗中，我們測試 3315 筆名稱，以排序第一的候選詞即為正確翻譯的機率可達到 82.3%。

關鍵字: 實體名稱翻譯、機器翻譯、網路、自然語言處理

Extracting English-to-Chinese Named Entity Translated Term Through Search Snippets

Student：Chen-Huei Jhang Advisor：Dr. Tyne Liang

Institute of Computer Science and Engineering
National Chiao Tung University

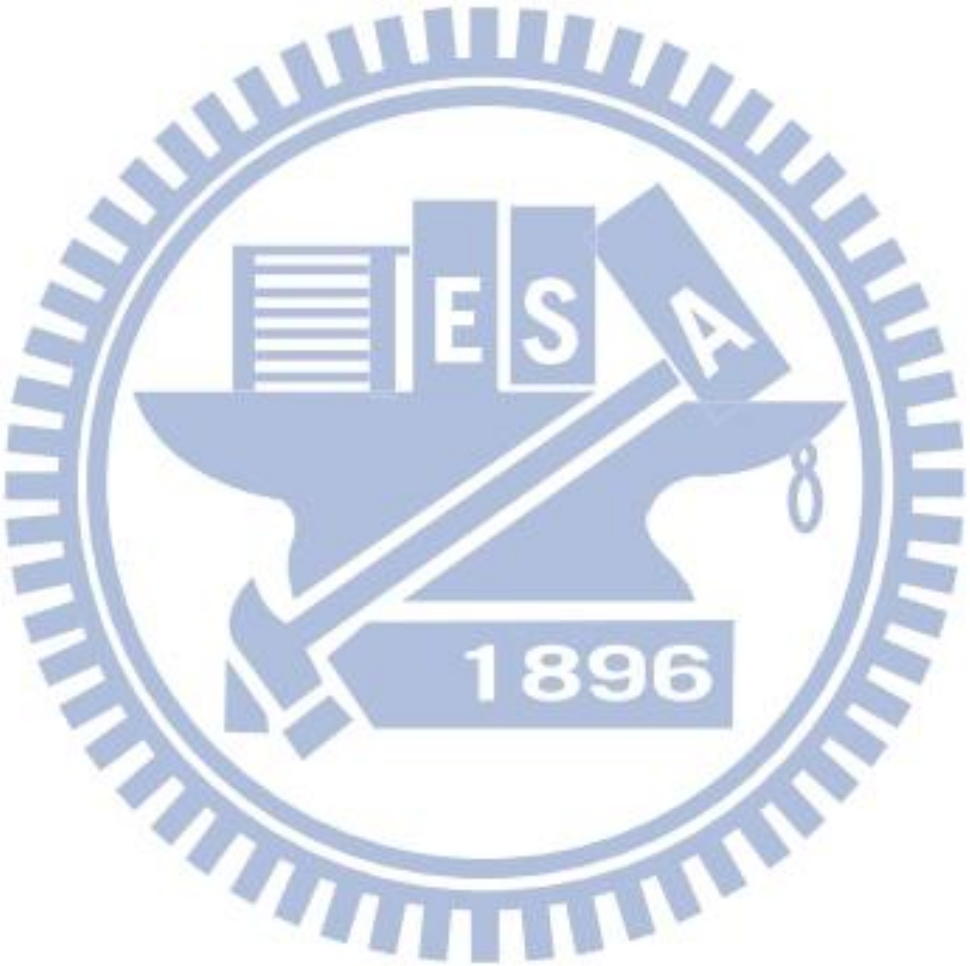
ABSTRACT

Name Entity translation plays an important role in many natural language processing (NLP) applications, such as machine translation, cross-language information retrieval, and question answering. With rich web information, many previous researches have employed with web resources, and search results. However, naming rules for the translating in domains are not concerned in most previous researches. In this thesis, we proposed an approach based on extracted translations from search results and considered naming rules for query expansion and translation candidate evaluation.

In this thesis, we extracted translations of name entities in four categories, namely, book, movie, medicine, and company. The proposed approach was implemented in three steps. We extracted features and identified name entities using support vector machine. Then, we applied pre-defined naming rules for different types of entities to expand queries with the purpose to require more relevant results. Finally, we extracted translation candidates by defined surface patterns and evaluated candidates. From the experiment results, the proposed approach yielded 82.3% accuracy of average top-1

inclusion rate.

Keyword: named entity translation, machine translation, web-based, natural language processing



Acknowledgement

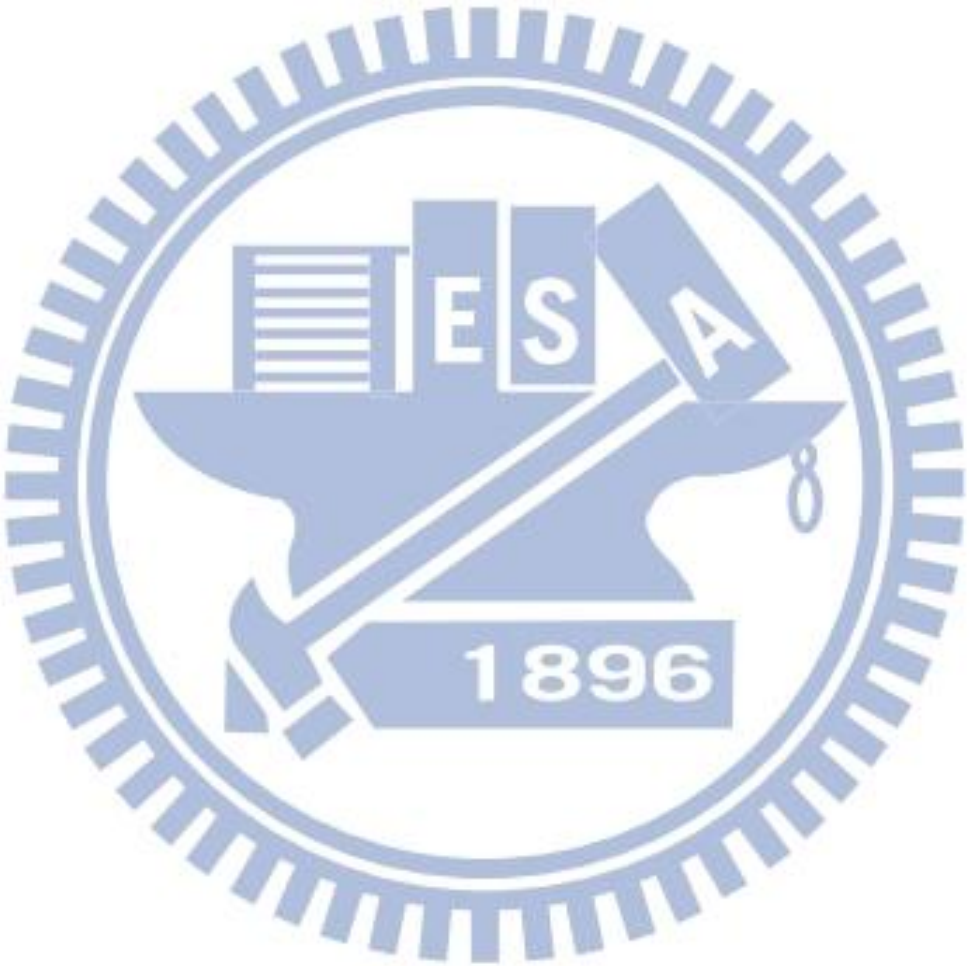
本論文能夠完成，首先要感謝梁婷老師，感謝老師這兩年的指導，在我有疑惑不得其解的時候，老師總能夠給我最確切的指導，使我能有突破，有所進展。也感謝冠熙學長，每當碰到疑惑時，跟學長討論後總能得到解答。另外還有笙權學長和晉榮同學時相討論。

也感謝家人們這兩年的支持，讓我在課業之外不用擔心生活，感謝父親、母親、大姊、二姊、三姊。

Table of Contents

List of Tables

List of Figures



Chapter1 Introduction

Proper noun translation plays an important role in machine translation (MT) and cross language information retrieval (CLIR). There are many approaches have been proposed and could be briefly categorized into three types, namely parallel/comparable corpus-based, dictionary-based, and web-based method. Corpus-based methods used statistical model to address name entity translation task [Prochasson and Fung, 2011]. Because translations come from corpus, performance is affected by domains and coverage of corpus. Dictionary-based methods use bilingual dictionaries to resolve term translation problem [Zhou et al., 2008], and have to resolve word sense disambiguation. Moreover, the dictionary-based approaches cannot resolve problems out-of-vocabulary (OOV).

On the other hand, web-based translation approaches employ the rich information available on web [Qu et al., 2011; Yang et al., 2009]. For example, there are approaches used the bilingual search results through a search engine. Compared to corpus-based and dictionary-based method, web-based methods save the cost in building up dictionaries and corpus. Figure 1 shows results containing translation pairs.

Figure 1. Results of querying “Clouds of witness” from Google

Previous researches have shown that term translations are annotated in a pair of parentheses before or after the term in web pages use to mine possible translations [Lin et al. 2006; Qu et al. 2011].

Web-based approaches are usually implemented as follows:

1. Data collection: bilingual search results are collected and used for candidate extraction.
2. Candidate extraction: when search results are collected, translation candidates are extracted from collected results.
3. Candidate ranking: candidates are ranked by evaluation function, and top candidates are represented as translations.

Using query expansion helps retrieve more results that contain term translation, thus improving performance.

In this thesis, we employ query expansion in order to acquire more search results. We concern four kinds of proper nouns. They are book names, movie titles, medicine names, and company names. The implementation can be summarized in the following steps:

1. Name Entity Recognition: retrieve source language results and recognize categories of name entities.
2. Query expansion: apply naming rules and clue word to prevent query drift
3. Candidate extraction: use pre-defined surface patterns.
4. Candidate evaluation: consider frequency and naming rule score.

The reminder of this thesis is organized as follows. Chapter 2 describes related work .Chapter 3 describes our approach in details. Chapter 4 describes experiments and experimental analysis. The conclusions and future work are in chapter 5.

Chapter2 Related Work

In this chapter, we will describe types of term translation methods. These methods could be categorized to three types:

1. Parallel/comparable corpus-based
2. Bilingual dictionary-based
3. Web-based method

In the following sections of this chapter, we will briefly describe previous researches.

2.1 Parallel/Comparable Corpus-based Method

Because translation pairs exist in corpus, corpus-based approaches extract the most possible correspondent translation from collected corpus. Since sentence pairs are hand-crafted, the advantage of corpus-based method is high quality of correspondent translation [Qu et al., 2011].

However, corpus-based method has some disadvantages. First, the cost of constructing a high quality parallel corpus is high, because it needs a lot of human effort. Second, the domain of a corpus is important. McNamee et al., [2009] verified that the domain of corpus is an important factor for bilingual translation retrieval, for example, a model was trained on sport domain may have difficulty to extract term translations from other domains. Third, new terms are generated every day, it is nearly impossible to construct corpus containing all novel terms, Therefore, corpus-based can not tackle OOV problem.

To avoid high cost of constructing a parallel corpus, Prochasson and Fung [2011] used comparable corpus. A comparable corpus is a collection of documents in the same domain with different languages. Prochasson et al., [2011] proposed a method that used context-vector and co-occurrence model to extract rare words translation. They used

Jaccard similarity to evaluate co-occurrence relation between words, and used Cosine similarity to compare two context-vectors. They trained a classifier by J48 decision tree algorithm in the *Weka* environment [Hall et al., 2009]. The experiment results showed that F-Measure reached 77% when they verified a Chinese-English translation on classifier trained by Spanish-French corpus. Vulic et al., [2011] proposed a method that used topic model to translate terms in comparable corpus. They utilized bilingual Latent Dirichlet Allocation (BiLDA) [Ni et al., 2009; Mimno et al., 2009; De Smet and Moens, 2009] to identify potential translation of a word. Their results showed that performance of combining information from word-topic distribution and similarity are the best.

2.2 Bilingual Dictionary-based Method

Dictionary-based methods translate words by looking up dictionaries. The methods usually encountered two problems. The first problem is ambiguity. Since there might be multiple translations for a word in dictionary, the difficulty was how to select the correct translation from dictionary. Liu et al., [2005] proposed a co-occurrence statistical model to resolve translation ambiguity, thus enhancing CLIR performance. Another problem was that there might have OOV words. An OOV words usually belongs to categories of compound words, proper noun, or technical terms. [Zhou et al., 2008]. Zhou et al., [2008] proposed a method to solve ambiguity problem by Graph-based model and OOV problem by pattern-based approach to enhance CLIR. Because terms are usually OOV, dictionary-based approaches are not appropriate for new generated term translation.

2.3 Web-based Method

Because both corpus-based method and dictionary-based method may suffer from OOV problem, web-based methods treated web resource as a big corpus. Researches that exploited these resources focused on two parts. Some researches aimed to construct a

bilingual term dictionary automatically by extracting translation pairs from large amount of web pages [Lin et al., 2008]. Another part of web-based researches exploited web resources for term translation. By submitting a term to search engine, the search engine returns bilingual search results of the term [Zhang et al., 2009; Qu et al., 2011].

Lin et al., [2008] extracted translation inside parentheses. They extracted patterns that matched Chinese string and followed by a pair of parentheses that contain English string. Second, inappropriate patterns are filtered out by pre-defined rules. Third, term boundaries were decided from vocabulary by training top 5 million most frequent Chinese queries. Fourth, they did word-alignment and computed link score for each pair. In the end, they extracted 26,753,972 translation pairs and verified correctness with terms existing in Wikipedia. Their results showed that the exactly match translation accuracy of extracted pairs was 36.4%. Jiang et al. [2009] proposed a method that adaptively generate pattern rule for each web documents that may contain translation pairs. In the first step, they parsed a web document to DOM nodes and filtered out documents that number of nodes which contained possible translation pair was lower than three. In the second step, they retrieved all patterns matched defined surface patterns. In the third step, they linked Chinese characters to English words in the pattern by looking up dictionary or by syllables matching, and they filter out patterns whose link number was lower than threshold. In the final step, they generalized patterns to regular expression rules, and they selected the most frequent rules to mine translation pairs from a web document. In the end, they extracted 12,610,626 pairs with 87.3% F-Score.

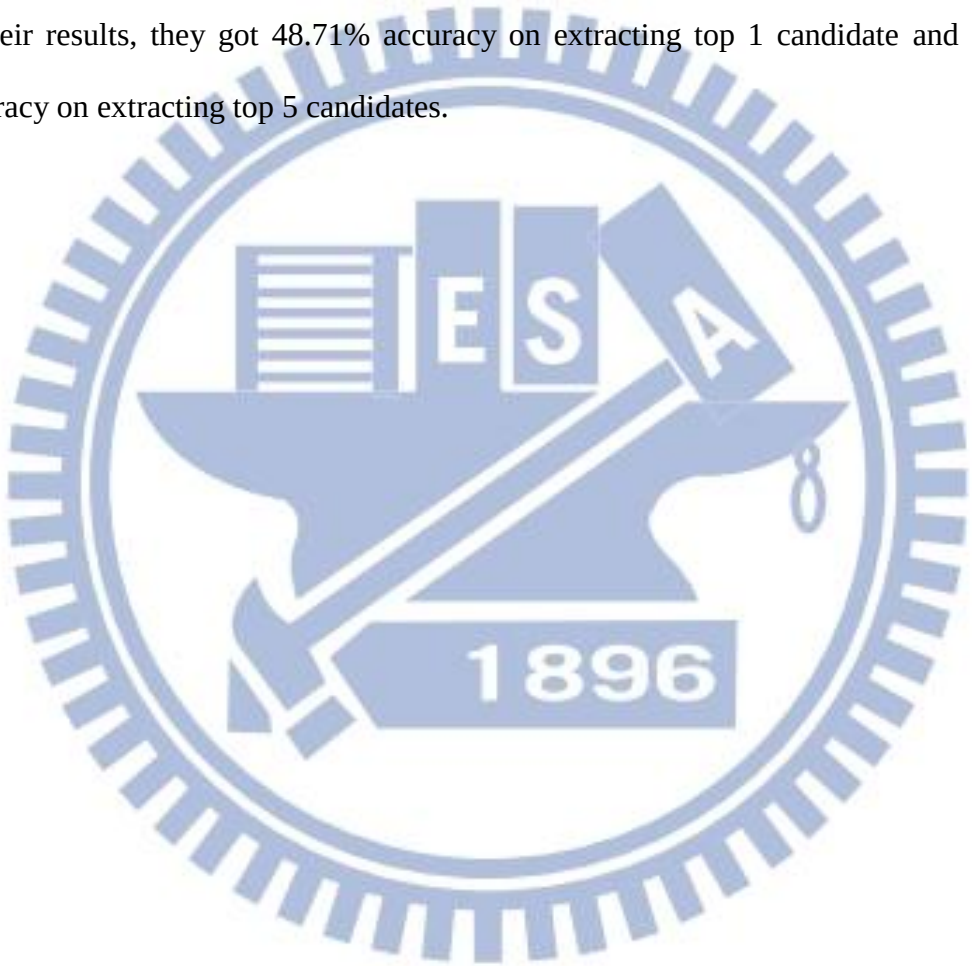
Previous researches have shown that translation of a term usually co-occurs together with the term [Cheng, Teng et al. 2004; Zhang, Huang et al. 2005; Udupa, K et al. 2009].

Zhang et al. [2009] used PAT-tree to extract translation candidates from search results, and SVM and Rank-SVM were employed as classifier and ordinal regression model. They evaluated their approach by extracting 300 English terms. In the end, they got 55.0% accuracy when they verified with top 1 candidate on Ranking-SVM. Qu et al. [2011] proposed a method to extract hybrid type translation. A hybrid type translation was defined as a translation contains different language. They extracted candidates by pre-defined regular expression rules and trained a Bayesian network for feature selection. To verify a candidate, they used Ranking-SVM to classify if a candidate is a translation. Their result showed that they reached 91.17% accuracy.

Some researches aimed to retrieve more results that contain translation to help extract candidates by adding query expansion with terms. Fang et al., [2006] proposed a Chinese-English term translation approach based on semantic prediction. In their approach, an unknown term was first translated word by word. For example, if input term is “三國演義”, the term will be translated to: {three, country, nation, act, practice, meaning, and justice}. Then each English translation was submitted to search engine with the correspondent Chinese character, and translation was filtered when returned result number was less than threshold. Each remained translation was submitted to search engine with the Chinese term one by one and extracted candidates from retrieved results. In the final step, candidates were ranked by frequency, pattern, and length of candidate. The experiment results show that the method reached 78.8% accuracy when they verified with top 1 candidate and reached 95.0% accuracy with top 10 candidates. However, this approach was time-consuming. Since it needed to request a search engine many times to translate a term, therefore, it was not able to translate massive amounts of terms.

Yang et al., [2009] proposed an approach to select most informative query expansion and extract translation of a Chinese organization name from web. In their

approach, they chunked Chinese organization name to four types by conditional random fields (CRFs) at first [J.Lafferty et al. 2001]. Second, they used a statistical machine translation (SMT) model to translate organization name. They counted mutual information (MI) value for each word. Third, they chose the word which had the highest MI value as query expansion and extracted translation of Chinese organization name from results by doing word-alignment on organization name and translation candidates. In their results, they got 48.71% accuracy on extracting top 1 candidate and 53.68% accuracy on extracting top 5 candidates.



Chapter 3 Web-based Term Translation Combining Naming Rules

In this chapter, we describe each steps of our approach in detail. Figure 2 shows the flow chart of our method. We will describe each steps of our approach in the subsection of this chapter.

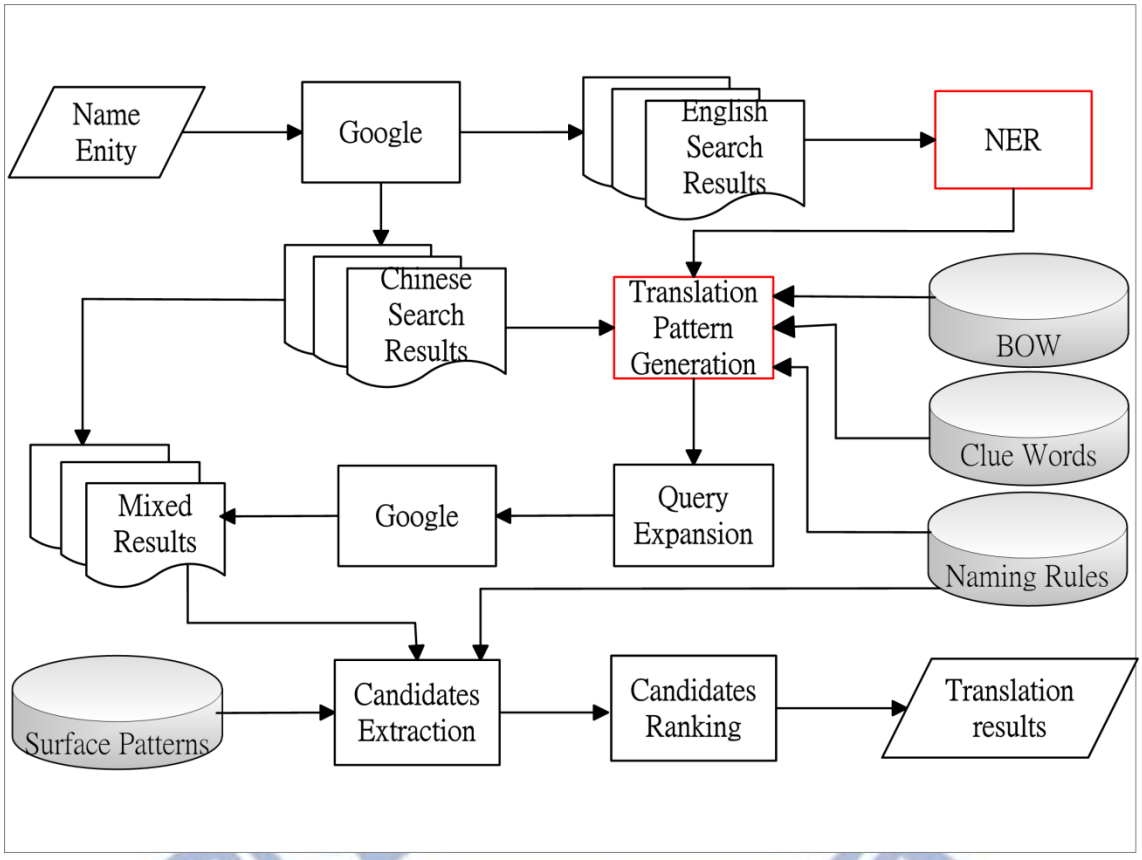


Figure 2. The flow chart of our approach

3.1 Named Entity Recognition

At first, we retrieve source language search results and use source language results for NER. In NER phase, because terms may belong to multiple categories, it is not suitable that we limit named entities only belonging to one category. For example, novels are usually made in to a film. To allow named entities belong to multiple categories, we trained a Boolean identifier for each category and identified named entities by these identifiers.

To identify an unknown named entity, we first submit the term to Google[1] and retrieve top 100 results in source language (English). After results are collected, html tags are cleared, and snippets are tagged part-of-speech (POS) tag by Natural Language Toolkit (NLTK[2]). Then the cleared and tagged results are sent to category identifier. Features which are used for identifier can be categorized to three types: syntactic features, word usage of intra-category features (WIC), and word distribution among inter-categories features (WAC). We describe each type in the following section.

3.1.1 Syntactic Features

Syntactic features exploit syntactic information of retrieved results, namely, part of speech (POS) tag patterns, relative position in titles, and relative position in snippets. The three features are shown as Table 1.

Table 1. Syntactic-related features focus on syntactic information of results

Syntactic	
POS(t)	$\sum_{pat \in p} p(pat\|c)$ p: set of POS tag patterns which co-occur near the named entity within 3 words, c:category (book, movie, medicine, company)
Tpos(t)	Related position of term in the title. 0 indicates at start of sentence, and 1 indicates at end of sentences
Spos(t)	Related position of term in the title. 0 indicates at start of sentence, and 1 indicates at end of sentences

To compute value of feature POS, we summed up probability of POS tag patterns which are around the named entity in three words. The probability of POS tag patterns were counted from collected results of training instances. The probability of a POS pattern belonging to category c was counted by following equation:

[1] http://www.google.com
[2] http://nltk.org/

$$\mathrm{p}(pat_i|\mathrm{c}) = \frac{frequency(pat_i, c)}{\sum_{pat_j \in p} frequncy\,(pat_j, c)} \quad (1)$$

where $\mathrm{frequency}(\mathrm{pat}, \mathrm{c})$ is the frequency of pattern pat appear in category c, and p represents the counts of all patterns appearing in category c.

Feature Tpos and Spos represent relative position of the named entity appear in search result titles and snippets respectively. We considered that positions of named entities might vary from categories, for example, a book title or a movie title usually occurs in first position of a snippet, and a company name usually occurs in middle of a snippet. The positions of a named entity occurred in titles or results were represented between 0 and 1, 0 represents at the start of the titles or snippets, and 1 represents at the end of the titles or snippets, then we averaged these position values.

3.1.2 Word usage of Intra-category

Consider that frequently used words vary between categories, so we exploited words usage information of retrieved results. And because usage of verbs and adjectives vary from categories too, we proposed five features that exploited information of word usage, verb usage in results, usage of verbs around the named entity, adjective usage, and usage of adjectives around the named entity. The features are shown as Table 2.

Table 2. Features of considering word usage inside category

Word usage of intra-category	
AWIC(t)	Usage of all words in retrieved results
AVIC(t)	Usage of all verbs in retrieved results
AJIC(t)	Usage of all adjectives in retrieved results
CVIC(t)	Usage of verbs around the named entity within three words
CJIC(t)	Usage of adjectives around the named entity within three words

AWIC feature considers the word distribution of a term. Weight of each extracted word was counted by equation as follows:

$$\sum_{x_i \in w} \frac{tf(w_i)}{\sum_{x_j \in w} tf(w_j)} \times prob(x_i|c) \quad (2)$$

where $\mathrm{tf}(w_i)$ represents term frequency of w_i in retrieved search results, w represents set of English vocabularies which appear in search result except stop words, $\mathrm{prob}(x_i|\mathrm{c})$ represents proportion of word x_i in category c. $\mathrm{prob}(x_i|\mathrm{c})$ is counted by equation (1) from search results of training instance. And

To consider verb usage in returned result, we considers verb usage in returned results in feature AVIC, and feature CVIC considers that verb appeared around the named entity in window size. Similar to features AVIC and CVIC, features AJIC and CJIC consider adjective usage in returned results. Feature AJIC considers usage of all adjectives in returned results, and feature CJIC considers usage of adjectives that appeared around the named entity in window size of 3 words.

3.1.3 Word Distribution among Inter-categories

These features aimed to complement word usage of intra-category (WIC) features. The disadvantage of WIC features were that they did not consider common words. For example, “do” is a frequently used word in all categories, since WIC features only compute proportion of a word inside a category, common words got high value in all categories, thus causing confusing to identify the named entity. To address this problem, we utilized word distribution among inter-categories (WAC) features to complete WIC. WAC features consider word distribution among categories, so if a word is a common word, it would get low value. The features are shown as Table 3.

Table 3. Features consider word distribution among categories

Word distribution among inter-categories	
AWAC(t)	Word distribution probabilities among categories
AVAC(t)	Distribution probabilities of verbs among categories
AJAC(t)	Distribution probabilities of adjectives among categories
CVAC(t)	Distribution probabilities of verbs around the named entity within 3 words
CJAC(t)	Distribution probabilities of verbs around the named entity within 3 words

The value of each feature was counted as equation (3):

$$\text{value} = \sum_{x_i \in w} \frac{tf(x_i)}{f_{max}} \times prob(c|x_i) = \sum_{x_i \in w} \frac{tf(x_i)}{f_{max}} \times \frac{average(x_i, c)}{\sum_{c \in C} average(x_i, c)} \quad (3)$$

Where value is the value of the feature, x_i is a extracted word in the feature, $\text{tf}(x_i)$ represents frequency of x_i, c represents the category, and $\text{average}(x_i, \text{c})$ represents occurrence frequency of x_i per named entity in category c.

Feature AWAC considers word usage of the retrieved results, we extracted one grams and two grams of English tokens from search results, and we counted value of AWAC by equation (3).

AVAC is similar to AWAC, the difference is that we care verbs distribution among categories instead of all English vocabularies. To compute value of feature AVAC, we extracted vocabularies which tagged as a verb from returned results. And value of AVAC is counted by equation (3).

AJAC focused on adjectives that appeared in search results. We consider that adjective usage is an important factor to judge categories. CVAC and CJAC focus on verbs and adjectives around the named entity in window size of 3 words. Since verbs or adjectives near the named entity might be modifier or action of the entity, they also

provide information for categorization.

3.2 Query Expansion

After a name entity is labeled, we retrieve Chinese search results and apply different query expansion strategies by its category label. Then search results with query expansion and search results without query expansion are used for extracting translation candidates.

We would like to retrieve more results which contain correct translations by query expansion. Because frequency is an important factor for our translation extraction method, it is helpful if the proportion of results which contain translation to all retrieved result is high. The best way of query expansion is adding translation with the named entity [Yang et al., 2009]. Since we do not know translation of the named entity, the alternative is adding part translation. For example, Figure 3 and Figure 4 show the difference between expanding query of "Clouds of Witness" and results without expanding query. It clearly shows that query expansion helps retrieve more results which contain translation. We will describe our approach of query expansion in detail in the following section.

Figure 3. Top five results without expanding query of clouds of witness

MWA最佳推理小說100選@ 呂仁茶社話推理:: 痞客邦PIXNET ::
lueren.pixnet.net/blog/.../27755183-mwa最佳推理小說100... - 頁庫存檔
2012年2月26日 – 077, 1927, Dorothy L. Sayers 桃樂絲・榭爾絲, Clouds of Witness 證言疑雲, 新星. 078, 1957, Ian Fleming 伊恩・佛萊明, From Russia, with Love ...

Clouds of Witness [证言疑云]_惊悚悬疑_英文小说-原版英语学习网
www.en8848.com.cn › Thriller - 頁庫存檔 - 轉為繁體網頁
Clouds of Witness [证言疑云]. #. 小说大小：574 KB 推荐等级： 2级作 者：Dorothy L. Sayers 官方网址： http:// 整理时间： 2010-10-09. 内容简介……. 内容简介 温西勋爵 ...

Be Cool[黑道比酷] - 原版英语
www.en8848.com.cn › Thriller - 頁庫存檔 - 轉為繁體網頁
2009年1月20日 – [英文小说]The Hunt for Red October [猎杀红十月号] · [英文小说]The Franchise Affair [法兰柴思事件] · [英文小说]Clouds of Witness [证言疑云] ...

Figure 4. Top five results with expanding query of clouds of witness

3.2.1 Query Expansion of Book and Movie titles

We utilized retrieved Chinese search results to generate query expansions. Based on our observation, some vocabularies in the title were translated directly. For example, the word "Clouds" in "Clouds of Witness" is translated to "雲" (證言疑雲) directly, and the word "tomorrow" of "the day after tomorrow" is translated to "明天" (明天過後). Therefore, we would like to use this observation to help us find more useful search results.

Because translations usually co-occur with the named entity together, we extracted Chinese patterns near the named entity and aimed to extract part of translation from results by following steps:

1. We translate vocabularies in the title word by word by looking up BOW[3]. For example, "Clouds of Witness" was translated to {雲, 幻覺, 嫌疑,目擊者, 證人}.

2. We extract Chinese patterns near the title. For example, if snippet is "桃樂絲・榭爾絲, Clouds of Witness 證言疑雲", then 證言疑雲 and 榭爾絲 would be extracted,

[3]http://bow.sinica.edu.tw/

because they are the nearest Chinese patterns to title “Clouds of Witness”.

3. We sort extracted Chinese patterns by frequency.

4. We selected pattern which contain translation and has highest frequency. For example, we have patterns {介紹: 10 , 中文: 8 , 言疑雲 : 5, 疑雲:3}, because “言疑雲” contained “雲” and had highest frequency among patterns which contain translation, “言疑雲” was selected.

However, query drift would happen if we select an inappropriate query expansion term. To avoid query drift, we added clue words to prevent drift problem. A clue word is a Chinese pattern that often co-occurs with the named entity and translation of the named entity. To get clue words for each category, we submit training instances with its translation to Google. We then extract 2-gram to 4-gram of Chinese patterns from returned results and keep patterns whose average frequencies larger than 0.5 per named entity in the category. We select an appropriate clue word by following equation:

$$\frac{c}{f_{max}} \times prob(c|Ca) \quad (4)$$

where Ca is a category, c is a clue word, and f_{max} is the frequency of the most frequent clue word in the search result. After part of translation and clue word are extracted, we submit them with the title to get Chinese search results. In the above case, we submit “言疑雲” and “小說” with “clouds of witness” to Google and retrieve search results.

3.2.2 Query Expansion of Medicine Names

The query expansion method of book or movie titles is not suitable for medicine names. We observe that medicine names are usually transliterated, and Chinese characters which correspond to English syllables are fixed. Based on these observations, we proposed a naming rule based method to generate part translation. We collect English

syllables to Chinese character pairs from collected translation pairs of training instances by the following steps:

1. We split medicine name to English syllables, e.g. “Setazindol” → ”se”, “ta”, “zin”, “do”, “l”
2. Every syllables mapped to every Chinese characters in the correspondent translation, and thus generate $m \times n$ pairs, where m is number of syllables and n is number of Chinese characters. E.g. translation of “Setazindol” is “司他秦多”, and they generate pairs like (se, 司), (se , 他), (se , 秦), (se , 多), and so on. The number of generated pairs is 20 (5×4).
3. We retained pairs which have confidence larger than 0.125. Confidence was counted by following equation:

$$\frac{freq(e,c)}{freq(e) + freq(c) - fre(e,c)} \quad (5)$$

where e is a English syllable, c is a Chinese character, and freq represents frequency.

We expanded medicine named entities by following steps:

1. We split name to syllables and extracted correspondent Chinese character of these syllables with highest confidence. E.g. pairs of “Setazindol” with highest value are (“se”, “酶”, 0.192), (“ta”, “他”, 0.405), and (“do”, “多”, 0.325)
2. Selected the characters of two consecutive syllables with highest sum. E.g. the two consecutive syllables of “Setazindol” are “seta”, “tazin”, ”zindo”, and “dol”. Syllable “seta” has highest sum 0.597, so we selected “酶他” as query expansion.
3. If there did not exist any correspondent character, we select clue word by equation (4).

3.2.3 Query Expansion of Company Names

Expanding company names are similar to expanding medicine names. We find out that the proper nouns or location names of a company name are transliterated and the others are usually translated. Furthermore, we observe that some vocabularies are always translated to the same Chinese words. For example, “food” is usually translated to “食品”, and “trade” is usually translated to “貿易”. Therefore, we could generate appropriate translation part from English token to Chinese token pairs according to naming rule. A English token to Chinese token pair is like (“co ltd”, “有限公司”). It is obtained from company name in translation pairs of training instances. We obtain pairs like following steps:

1. English company name was split to one and two grams.
2. Correspondent translation was split to two to four grams.
3. Every English grams map to every Chinese grams as a pair, and thus generate $m \times n$ pairs, where m is number of English grams and n is number of Chinese grams. E.g. “JIUJIANG HUIYUAN FOOD STUFF CO., LTD” and “九江匯源食品飲料有限公司” generate 330 pairs.
4. We count confidence of each pair by equation (5) and retain pairs which confidence is larger than 0.01.

we describe query expansion method of company name as following steps:

1. We split English company name to one and two grams and retrieved correspondent pair with highest confidence. E.g. pairs of “九江匯源食品飲料有限公司” are {jiujiang：九江，food：食品，ltd：有限，co ltd：有限公司}
2. We selected Chinese pattern in the pairs which has highest frequency in the search result. E.g. The frequencies of each Chinese patterns are {九江: 15, 食品: 5, 有限:10, 有限公司: 8}, then we choose “九江” as query expansion.

3.3 Translation Candidate Extraction and Evaluation

In this section, we describe the method of extracting translation candidates and evaluating extracted candidates.

3.3.1 Candidate Extraction

We extract Chinese terms beside to the named entity in titles or in snippets as candidates, and define some surface pattern for categories respectively.

Figure 5 shows that translations were usually around the named entity, and were separated by white space or a comma. Based on this observation, we start extracting candidates when it encountered first Chinese character, and stop extracting candidates when it meets pre-defined stop symbols or encountered English token whose length large than 4. The pre-defined symbols are a white space, a comma, a period, or a parenthesis. These symbols are often used to separate two words in search results or web pages. The marked Chinese patterns in Figure 5 are translation candidates which are extracted by our method.

Figure 5. Example of extracted candidates in search results of “clouds of witness”

In addition to extracting Chinese patterns around the named entity, we defined two surface patterns to help extract candidates. For book and movie names, we extracted strings which are inside a pair of parentheses and have at least one Chinese character for book category and movie category. Moreover, we extracted Chinese patterns which

followed “企業名稱” or “中文名稱”.

3.3.2 Candidate Evaluation

Because translation often co-occurs with the named entity, our evaluation method considers two factors: frequency and score of naming rule. The score of naming rule represent the number of overlapping in a translation candidate and word translation set that used in query expansion. For example, suppose that “Clouds of Witness” was identified as a book name, and it was translated word by word to {雲, 幻覺, 嫌疑,目擊者, 證人}.The value of rulescore(證言疑雲) is one since it contains one translation (“雲”), and suppose that “Setazindol” was identified as a medicine name, the syllable to character pairs were collected as {se: 酶, ta: 他, zin: 嗪, do: 多}. Rule score of candidate “司他秦多” is two, since it contains “他” and “多”.

We calculate the score of each translation candidate *t* by equation (6):

$$\text{score(t)} = \sum_{1}^{n} 1 + rulescore(t) \qquad (6)$$

where t is a candidate, n is the frequency of t in retrieved research results.

Chapter 4 Experiments and Analysis

4.1 Experimental Setup

We collected English-Chinese name pairs of 1,405 book titles from multiple web sites, 3,810 movie titles from 開眼電影網[4], 7,131 medicine names from multiple web sites, and 219,309 company names from web. Table 4 shows some examples of English-Chinese pairs in each category. The average length of named entities and their correspondent translation are shown as Table 5.

Table 4. Example of term pairs from four domains

Category	English term	Chinese term
Book title	Clouds of Witness	證言疑雲
Movie title	Pure Country	戀曲動我心
Medicine name	Dolcvmene	傘花羥
Company name	JIUJIANG HUIYUAN FOOD STUFF CO., LTD	九江匯源食品飲料有限公司

Table 5. Average length of collected term pairs

	English word Length	Chinese Character Length
Book title	3.93	6.44
Movie title	3.41	5.67
Medicine name	1.14	3.98
Company name	5.29	12.13

[4] http://www.atmovies.com.tw/home/movie_homepage.html

4.2 NER Experiments and Analysis

We employed LIBSVM[5] and CRF which is implemented by Monte[6] for testing performance of NER. We used 1315 book named entities, 2666 movie named entities, and 4000 named entities of medicine named entities and company entities respectively.

The proportion of testing instances to training instances is 3 to 7. In this setting, we used 983 book titles, 2,666 movie titles, 2800 medicine names, and 2,800 company names for training. And we used 415 book titles, 1,333 movie titles, 1,201 medicine names, and 1,199 company names for testing. The instances in testing data and instances in training data did not overlap.

We trained a Boolean identifier for each category by approximate number of positive instances and negative instances. The negative instances were collected from other categories equally. And identifiers were tested by approximate number of positive instances and negative instances, too. The result is shown as Table 6.

Table 6. Accuracy of SVM and NBC in each domain

Domain	SVM	CRF
Book title	80.5%	52.4%
Movie title	80.8%	52.5%
Medicine name	97.4%	73.7%
Company name	96.4%	69.7%

The resilts in Table 5 clearly shows that SVM model outperform CRF model when both used default settings, so we select SVM to be our category identifier. The accuracies of identifying book titles and movie titles are lower than identifying named entities of medicine category and company category. It was due to some book titles and

[5] http://www.csie.ntu.edu.tw/~cjlin/libsvm/
[6] http://montepython.sourceforge.net/

movie titles overlapped. For example, “Howards End” is a novel which was made into a film too. The retrieved search results were composed by both results of book category and movie category and this made the name entity difficult to identify. The results of “Howards End” are shown like Figure 6. Furthermore, if titles are common words, the search results would contain too much non-related contents. Figure 7 is the search results of a common word “Fire Ball” which is a movie name in our data. From Figure 7 we can observe that the title “Fire Ball” might be used in many categories such as music, dancing, and game industry. Therefore, we had difficulty to get enough information to identify its category.

Moreover, medicine names and company names are different from other categories. The wrong categorized instances of these two categories are that instances have rare results, so the number of retrieved results is not enough for identification, thus causing wrong identification.

Figure 6. Results of “Howards End”

Figure 7. Returned results of query "Fire Ball"

4.3 Query Expansion Results and Analysis

We describe experimental settings of generating these pairs and results of our query expansion method. Our result showed that naming rules helped retrieve more relevant results.

To evaluate the effectiveness of our query expansion method, we counted macro average proportion (MAP) of titles and snippets which have as least one translation to all titles and snippets respectively. MAP of each category is counted as following equation:

$$\frac{1}{N}\sum_{1}^{N}\frac{S_i}{N_i} \tag{7}$$

Where N represents number of name entities of the category, N_i represents number of results of i^{th} named entity, and S_i represents total number of results of i^{th} term. Result of query expansion method is shown as Table 7. Our results show that the proposed method enhanced query efficiency in all categories except titles in medicine category. It was due to that web pages which contain translation of a medicine name

usually did not show translation in their title.

Table 7. Result of expanding query and without expanding query

Domain and type	Without query expansion	With query expansion
Book (title)	13.6%	22.6%
Movie (title)	23.4%	29.4%
Medicine (title)	23.7%	19.1%
Company (title)	34.9%	36.1%
Book (snippet)	15.1%	39.9%
Movie (snippet)	30.3%	46.5%
Medicine (snippet)	33.0%	62.9%
Company (snippet)	53.7%	63.1%

Although MAP declines in titles of medicine category, but MAP increases significantly in snippets of medicine category, so the proposed method still helped retrieve more relevant results.

The other factor which impacts our query expansion performance is titles constitute of common word. We observed that our method might extract inappropriate query expansion when name entity has common word. For example, the book “Dead souls” was translated “死魂靈”. We expect to extract “死魂”, “魂靈”, or “死魂靈” as query expansion, However, the game whose name is “Yakuza: Dead Souls” also contain “Dead Souls” caused query drift. Since the word “靈魂” in its Chinese name, “如龍：死亡靈魂”, exactly matches the Chinese translation of “soul”, our query expansion method will choose “靈魂” as expansion term. Although we added clue word “英文” to prevent query drift, but “英文” related to game category too, so it did not work in this case. Table 8 shows some examples of extracting inappropriate expansion by our method.

Table 8. Examples of query drift from our method

Term	Translation	Expansion	Clue word
Dead Souls	死魂靈	靈魂	英文
East of Eden	伊甸之東	東方	英文
Gone for A Dance	花漾漫舞	跳舞	電影
prey for rock & roll	女聲搖擺	搖滾樂	電影
the ant bully	聯合縮小兵	螞蟻	電影

4.4 Translation Extraction Experiments

We utilized average top-n inclusion rate as evaluation metric of translation equivalent extraction. Average top-n inclusion rate is defined as the percentage of named entities whose translation can be found in the first *n* extracted translation candidates.

A candidate was judged as a correctly extraction only if it exactly matched our collected translation equivalent. For example, the translation of "Boss of Bosses" is "黑幫大帝國" (in Taiwan) or "老闆的老闆" (in P.R.C). We judged only "黑幫大帝國" and "老闆的老闆" as correct extraction, but "電影黑幫大帝國" or "DVD 老闆的老闆" would be judged as incorrect extraction.

In experiments of translation extraction, we used 4,15 book title pairs, 5,00 movie pairs, 1,201 medicine name pairs, and 1,199 company name pairs for evaluation.

4.4.1 Book Title Extraction

We compared our method with baseline method and baseline method with naming rule. The baseline method extracted translation from search result directly without query expansion and evaluates translation candidate only by frequency without considering naming rules. Table 9 shows results of book name entity translation extraction. *Base* means baseline model. *Base+NR* means baseline model with considering naming rule

for evaluation. *Base+QE+NR* is the proposed model which considers both naming rules for query expansion and evaluation. The result shows that naming rules are helpful. But query expansion was not as much helpful as naming rules in book category.

Table 9. Top 10 Inclusion Rate of Book Domain

	Top1	Top3	Top5	Top10
Base	67.7%	84.8%	89.4%	92.3%
Base+NR	72.5%	89.2%	92.3%	94.9%
Base+QE+NR	73.0%	88.2%	92.8%	94.5%

4.4.2 Movie Title Extraction and Medicine Name Extraction

Table 10 shows the extraction results of movie category. In movie category, considering naming rules was not as much helpful as book category. This was due to that movie titles often co-occur with their translation in search results. In this case, frequency was enough for identifying translation equivalents.

Table 10. Top 10 Inclusion Rate of Movie Domain

	Top1	Top3	Top5	Top10
Base	77.0%	91.6%	94.0%	96.0%
Base+NR	78.4%	93.4%	94.6%	96.6%
Base+QE+NR	79.8%	93.8%	95.8%	97.6%

4.4.3 Medicine Name extraction and Company Name extraction

The results show that naming rules help helps extract translation equivalent in medicine category and company category. The results are represented in Table 11 and Table 12.

Table 11 shows that Base+QE+NR approach increases 4.6% accuracy at top-1 inclusion rate when compared to baseline model. Query expansion helps slightly

medicine translation.

Table 11. Top 10 Inclusion Rate of Medicine Domain

	Top1	Top3	Top5	Top10
Base	81.2%	92.8%	95.0%	96.0%
Base+NR	85.8%	95.2%	96.0%	96.6%
Base+QE+NR	86.9%	95.1%	96.4%	97.6%

In company category, considering naming rule and query expansion increases accuracy more than only considering naming rule. The reason of query expansion increased performance is that retrieved results number was rare when we submit company name with its query expansion, so the proportion of correctly extracted translation to all extracted translation was high, so it helped rank correct translation in high order.

Table 12. Top 10 Inclusion Rate of Company Domain

	Top1	Top3	Top5	Top10
Base	71.1%	83.4%	87.5%	89.6%
Base+NR	83.3%	89.0%	89.8%	90.0%
Base+QE+NR	89.4%	92.8%	93.4%	93.9%

4.4.4 Model comparison

We also compared the our model with the approach [Zhang et al., 2009]. We compared accuracy medicine category Because Zhang's approach could not address translations of long token length, and length of translations in medicine category are usually less than

four Chinese characters. The results in Table 13 show that our approach outperformed Zhang's from top 1 inclusion rate to top 10 inclusion rate.

Table 13. Results of comparison to Chang's approach

	Top 1	Top 3	Top 5	Top 10
Zhang et al. 09	71.8%	81.9%	82.5%	82.5%
Our approach	86.9%	95.1%	96.4%	97.6%

4.4.5 Analysis of Translation Extraction Method

Our extraction method could not extract translation correctly in following conditions:

1. Our approach can't not extract translation which contains English token that longer than four characters. For example, movie "Tamagotchi: The Movie" was translated to "塔麻可吉 DOKI!DOKI!星球大暴走!?". But we could only extract "塔麻可吉" and "DOKI!星球大暴走".
2. Our approach could only extract part of translation if the translation contains punctuation marks that were defined as stop symbols in our method. For example, "Africa, How Are You With Pain?" is translated to "非洲，你還好嗎？". Because there is a comma in the translation, we can only extracted "你好嗎" as a candidate.
3. Some named entities were common words or are often used in different categories. In these two situations, our approach extracts many irrelevant candidates, thus declining accuracy.
4. Some named entities may have no search results, so translation of the named entity could not be extracted from web.

Chapter 5 Conclusion and Future Work

5.1 Conclusion

In this thesis, we proposed a web-based approach to address name entity translation by extracting translation equivalents from search results. In our approach, we label category types of an unknown named entity at first. Then we expand query to retrieve more results that contain translations according to category label. Finally, we extract translation candidates by defined surface patterns and evaluate candidates by frequency and naming rule.

From the results of our experiments, we reach high accuracies when we identify medicine names and company names, and we reach about 80% accuracies when we identify book titles and movie titles. Our query expansion method increased 11.4% of MAP when compared to search results without query expansion. And we reached 82.3% accuracy of average top-1 inclusion rate. The findings of this thesis could be summarized to the following points:

1. Applying query expansion helped retrieve more results that contain translation, thus enhancing extraction.
2. Extracting translation candidates by matching simple surface pattern is useful.
3. Using search results is not enough for extracting translation of name entities that are also common words. To address this problem, we find out that content of whole web page may be helpful.
4. Naming rules for different category helped evaluate translation candidates, but an elaborated design evaluation method is also necessary.

5.2 Future Work

In the future, we could improve the term translation by studying the following issues:

1. Collect web pages or document in advance and index the document. It helps decrease system execution time and can exploit information of whole pages, thus solving problem of term names which are common words.
2. Apply more surface patterns for each category to improve translation extraction.
3. Apply more naming rules for each category to enhance the system performance.
4. Design a more elaborated evaluation function to rank translation candidates.

Reference

Yang, F., Zhao, J., and Liu, K., "A Chinese-English Organization Name Translation System Using Heuristic Web Mining and Asymmetric Alignment." *In Proceedings of the 47th Annual Meeting of the ACL and the 4th IJCNLP of the AFNLP*, pp. 387-395. Suntec, Singapore, 2009.

Liu, L., GE, Y. D., Yan, Z. X., Yao J. M., "A CLIR-Oriented OOV Translation Mining Method From Bilingual Webpages." *In Proceedings of the 2011 International Conference on Machine Learning and Cybernetics*, pp. 1872-1877, Guilin, 2011.

Liu, Y., Jin, R., Chai, J. Y., "A Maximum Coherence Model for Dictionary-based Cross-language Information Retrieval." *The 28th Annual International ACM SIGIR Conference on Research and development in information retrieval*, Salvador, Brazil, 2005.

Gao, J. F., and Nie, J. Y., "A Study of Statistical Models for Query Translation: Finding a Good Unit of Translation." *In Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, Seattle, Washington, USA, 2006.

Fang, G., Yu, H., and Nishino, F., "Chinese-English Term Translation Mining Based on Semantic Prediction." In *Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions*, pp. 199-206, Sydney, 2006.

Zhang, Y., Wang, Y. and Xue, X., "English-Chinese Bi-Directional OOV Translation based on Web Mining and Supervised Learning." *In Proceedings of the ACL-IJCNLP 2009 Conference*, pp. 129-132, Suntec, Singapore, 2009.

Jiang, L., Yang, S., Zhou, M., Liu, X., and Zhu, Q., "Mining Bilingual Data from the Web with Adaptively Learnt Patterns." *In Proceedings of the 47th Annual Meeting of the ACL and the 4th IJCNLP of the AFNLP*, pp. 870-878, Suntec, Singapore, 2009.

Lin, D., Zhao, S., Durme, B. V., Marius Paşca, M., “Mining Parenthetical Translations from the Web by Word Alignment.” *In Proceedings of ACL-08: HLT*, pp. 994-1002, Columbus, Ohio, USA, 2008.

Hermjakob, U., Knight, K., and Hal Daum´e III, “Name Translation in StatisticalMachine Translation Learning When to Transliterate.” *In Proceedings of ACL-08: HLT*, pp. 389-397, Columbus, Ohio, USA, 2008.

Emmanuel Prochasson and Pascale Fung, “RareWord Translation Extraction from Aligned Comparable Documents.” *In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, pp. 1327–1335, Portland, Oregon, 2011.

Lin, W. P., Snover, M., and Ji, H., “Unsupervised Language-Independent Name Translation Mining from Wikipedia Infoboxes.” *In Proceedings of EMNLP 2011, Conference on Empirical Methods in Natural Language Processing*, pp. 43–52, Edinburgh, Scotland, UK, 2011.

Qu, J., Minh, N. L., and Shimazu, A., “Web based English-Chinese OOV term translation using Adaptive rules and Recursive feature selection.” *In 25th Pacific Asia Conference on Language, Information and Computation*, pp. 1-10, Singapore, 2011.

Yang, Y., Zhao, T., Lu, Q., Zheng, D., and Yu, H., “Chinese Term Extraction Using Different Types of Relevance”, *In Proceedings of the ACL-IJCNLP 2009 Conference*, pp. 213-216, Suntec, Singapore, 2009.

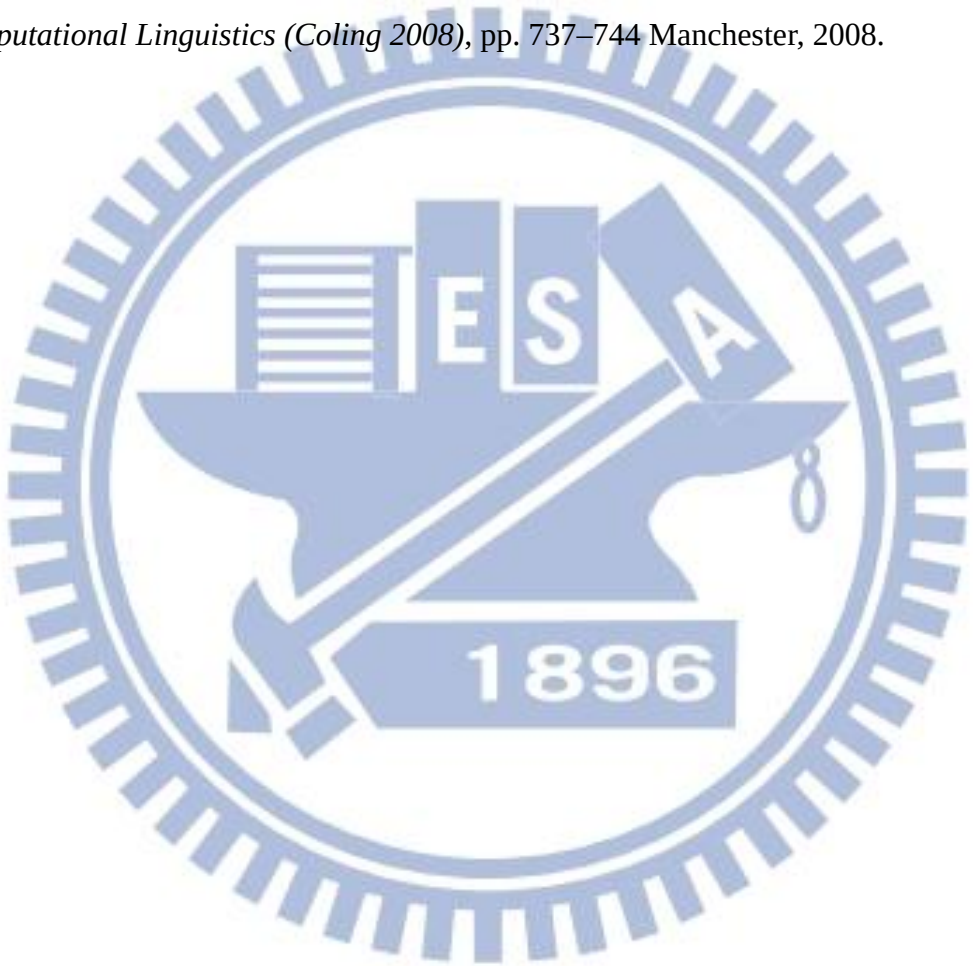
Chih-Chung Chang and Chih-Jen Lin, LIBSVM : a library for support vector machines. ACM Transactions on Intelligent Systems and Technology, 2:27:1--27:27, 2011. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm

Scikit-learn: Machine Learning in Python, Pedregosa et al., JMLR 12, pp. 2825-2830, 2011.

Hui Fang, “A Re-examination of Query Expansion Using Lexical Resources” *In Proceedings of ACL-08: HLT*, pp. 139–147, Columbus, Ohio, USA, 2008.

Cao, G., Robertson, S., and Nie, J. Y., “Selecting Query Term Alterations for Web Search by Exploiting Query Contexts.” *In Proceedings of ACL-08: HLT*, pp. 148–155, Columbus, Ohio, USA, 2008.

Riezler, S., Liu, Y., and Vasserman, A., “Translating Queries into Snippets for Improved Query Expansion.” *In Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pp. 737–744 Manchester, 2008.



Appendix A: Term Pairs

Book Pairs	
Clouds of Witness	證言疑雲
Crocodile on the Sandbank	沙丘上的鱷魚
Diary of a Pilgrimage	朝聖路日記
Desiree's Diary	德希蕾日記
Death on the Nile	尼羅河上的慘案
Different Seasons	四季奇談\不同的季節\四季奇譚
Digital Fortress	數字堡壘\數字城堡\數位密碼
Death Comes for the Archbishop	大主教之死
Doctor Zhivago	日瓦格醫生
Don Juan	唐璜
Daisy Miller	黛茜米勒
Movie Pairs	
Chopin: Desire for Love	蕭邦—琴戀喬治桑
Pure Country	戀曲動我心
Deep Blue	深藍
Treasure Planet	星銀島
The Little Mermaid II: Return to the Sea	小美人魚２：重返大海\小美人魚 2 重返大海
Zeiten andern Dich	時代改變你
The Horse Boy	孤老而終\遠山遠處\馬背上的男孩
Dragon Blade	龍刀奇緣
Oldman Z	老人 Z
Hip-Hop Storm	街舞狂潮
The Days of Noah 2: Apocalypse	挪亞方舟驚世啟示 2

Medicine Pairs	
Oleandomycin	竹桃黴素
Auranofin	金諾芬
Aspoxicillin	阿撲西林
Acetiamine	乙酰硫胺
Ethylnoradrenaline	乙諾那林
Mecysteine	美司坦
Alonimid	阿洛米酮
Espatropate	艾帕托酯
Traxanox	曲呫諾
Thiohexital	硫己比妥
Dolcvmene	傘花羥
Company Pairs	
SHENZHEN OUAOYA IMPORT EXPORT CO.,LTD.	深圳市歐澳亞進出口有限公司
JIUJIANG HUIYUAN FOOD STUFF CO., LTD	九江匯源食品飲料有限公司
ZHENGZHOU TECHSENSE CHEMICAL CO.,LTD.	鄭州科信化工有限公司
GEM—KAI INTERNATIONAL TRADING （SHANGHAI） CO.,LTD.	上海晉凱國際貿易有限公司
SWEETISLAND ENTERPRISES DEVELOPMENT LTD	廣州瑞蘭企業發展有限公司
Hang Zhou Green Home Foods co.,ltd	杭州綠家食品有限公司
JINHAI JOINT TRADE CO,.LTD.	珠海經濟特區津海聯合貿易有限公司
QINHUANGDAO HUIZE HANDICRAFT CO.,LTD	秦皇島惠澤工藝品有限公司

Appendix B: Extracted Clue Words

Book	Movie	Medicine
英文	電影	名稱
電子	片名	英文
英語	影片	文名
子書	海報	文名稱
電子書	上映	中文
原著	劇照	英文名
文原	日期	別名
英文原	英文	英文名稱
文原著	映日	翻譯
英文原著	映日期	化工
免費	上映日	產品
內容	文片	藥品
小說	上映日期	詞典
資料	文片名	通用
作者	中文	藥物
故事	片長	中文名
資源	介紹	文別
圖書	導演	文別名
閱讀	內容	供應
書下	類似	用名
書下載	類似內	文名稱
文件	似內容	通用名
文學	似內	生物

Appendix C: English to Chinese Pairs

English Syllable to Chinese Character Pairs			
English Syllable	**Chinese Character**	**Confidence**	**Frequency**
set	瓊	0.857143	18
vin	春	0.774194	24
ges	孕	0.745455	41
u	烏	0.695652	16
ros	前	0.686567	46
b	單	0.674419	29
yci	黴	0.673913	93
qui	喹	0.650602	108
zo	唑	0.645403	344
gua	胍	0.625	25
dex	右	0.615385	24
b	抗	0.604167	29
ylli	茶	0.592593	32
en	恩	0.585366	24
vin	長	0.55814	24
da	達	0.557447	131
e	依	0.542017	129
fu	呋	0.540541	60
za	扎	0.536913	80
bu	布	0.534413	132
zi	嗪	0.52514	188
spi	螺	0.525	21
cai	因	0.518182	57
ser	舍	0.517241	15
niu	鈕	0.516304	95
mo	莫	0.515723	164
phi	啡	0.5	21
c	酸	0.489888	218
pen	噴	0.483871	45
io	碘	0.467105	71
vi	韋	0.463768	32
pol	聚	0.44186	19
tas	鉀	0.44	11

English Token to Chinese Token Pairs			
sheqi	社旗	0.117188	30
sheqi	社旗縣	0.117188	30
sheqi	旗縣	0.117188	30
xiangcheng	襄城縣	0.113764	81
cable	纜廠	0.111317	60
xiangcheng	襄城	0.110821	85
farm	農場	0.107292	103
xinye	新野縣	0.107212	55
jurong	容市	0.107143	30
jurong	句容市	0.107143	30
yanling	鄢陵縣	0.10625	68
xiangxiang	湘鄉	0.105455	29
laiyang	萊陽市	0.104762	44
tianchang	天長市天	0.10473	31
tianchang	長市天	0.10473	31
ltd.	電腦有限	0.104634	70
ltd.	電腦有	0.104634	70
boai	愛縣	0.104444	47
boai	博愛縣	0.104444	47
shenzhen	貿易行	0.104389	176
shenzhen	易行	0.104389	176
shanghai	上海明	0.104377	31
rubber	橡膠廠	0.104167	45
factory	橡膠廠	0.104167	45
grain	糧庫	0.103858	35
shangcai	上蔡	0.103757	58
hebei	河北天	0.103734	25
shangcai	上蔡縣	0.103647	54
co.,	電腦有限	0.103139	69
co.,	電腦有	0.103139	69
shenzhen	深圳市森	0.102837	58
shenzhen	圳市森	0.102837	58
dalian	大連利	0.102804	66
dalian	連利	0.102804	66
co., ltd.	電腦有	0.101644	68