Université de Montréal

Filtering Parallel Texts to Improve Translation Model and Cross-Language Information Retrieval

Par

Jian Cai

Département d'informatique et de recherche opérationnelle
Faculté des arts et des sciences

Mémoire présenté à la Faculté des études supérieures
en vue l'obtention du grade de
Maitre en Informatique

December 2001

Université de Montréal

Faculté des études supérieures

Ce mémoire intitulé:

Filtering Parallel Texts to Improve Translation Model and Cross-Language Information Retrieval

Présenté par:

Jian Cai

A été évalué par un jury composé des personnes suivantes:

Sebastien Roy	(président-rapporteur)
Jian-Yun Nie	(directeur de recherche)
Philippe Langlais	(membre du jury)

Mémoire accepté le 11 mars 2002

Abstract

Cross-Language Information Retrieval (CLIR) aims to retrieve relevant document written in one language with a query written in another language. In addition to the common problems in monolingual information retrieval, query translation is crucial for CLIR. One of the approaches for query translation is the one that uses statistical translation models trained on a large set of parallel texts (a parallel corpus). Good retrieval results have been obtained using such an approach. However, for many languages, there is no large parallel corpus for model training. A recent research project has successfully constructed an automatic mining system - PTMiner - that mines parallel Web pages automatically. Several large sets of parallel texts have been obtained with this system, among them, a Chinese-English corpus.

However, the Web pages automatically mined are not always parallel. Their parallelism is far less perfect than manually constructed parallel corpora such as Canadian Hansards. As a consequence, the translation model trained from such raw corpus has poorer translation accuracy and leads to lower performance when used in CLIR. The current thesis studies the problem of how to improve the translation models, as well as the CLIR performance, by an additional filtering process on the parallel Web pages provided by PTMiner. Our study focuses on the Chinese-English case. However, this approach can also be applied to other language pairs.

The principle we use in the filtering process is based on the observation that non-parallel Web pages usually are ill-aligned. There is a large proportion of empty sentence alignments in them, i.e. the sentences that have no corresponding sentence in the other language. Therefore, a threshold is set on this proportion to filter out likely unparallel pairs of Web pages.

In addition, we also suggest several methods to improve the accuracy of sentence alignment, namely, by enhancing the correspondence of pairs of sentences whose length ratio is close to the standard length ratio of the two languages, the consideration of known translations that are stored in a bilingual dictionary. These approaches have brought some improvements to the parallelism of the corpus.

Three series of experiments have been conducted to test the filtering process. The first series examines the parallelism of the corpus after the filtering process; the second the translation accuracy of the resulting translation models; and the third examines CLIR performance. It is shown that the approach we suggest is effective. We have been able to create a cleaned corpus that results in more accurate translations (91.50% for Chinese-English, 87.50% for English-Chinese) and higher CLIR performance (28.11% for Chinese-English, 26.01% for English-Chinese). As the methods used in the filtering process are language independent, this method can also be used to other language pairs (possibly with some different values for parameters).

This study further confirms that automatically mined parallel Web pages are valuable resources for CLIR.

Key Word: Cross Language Information Retrieval, text corpus filtering, sentence alignment.

Résumé

La Recherche d'information Translinguistique (RIT) vise à retrouver les documents pertinents écrits en une langue à partir d'une requête écrite en une autre langue. A part les problèmes communs avec de la recherche d'information, RIT doit traiter le problème de la traduction de requête.

Une des approches utilisées pour la traduction de requêtes est basée sur l'utilisation des modèles de traduction statistiques, entraînés sur un grand ensemble de textes parallèles (un corpus parallèle). De bons résultats ont été obtenus en utilisant cette approche. Cependant, pour beaucoup de paires de langues, il n'existe pas de grands corpus d'entraînement. Un système de fouille automatique de pages Web parallèles - PTMiner - a été développé dans un projet de recherche récent. Plusieurs corpus de pages Web parallèles ont été constitués avec ce système, dont un corpus de chinois-anglais.

Toutefois, les pages Web trouvées automatiquement ne sont pas toutes parallèles. Leur parallélisme est loin d'être comparable celle d'un corpus construit manuellement, tels que le Hansard. La conséquence de ceci est une précision de traduction moins élevée et une performance de RIT moins bonne en utilisant les modèles entraînés. La présente étude porte sur le filtrage de corpus de pages Web afin d'en améliorer la qualité. Cette étude porte en particulier sur le corpus de chinois-anglais. Cependant, les méthodes proposées dans ce mémoire peuvent être utilisées pour d'autres paires de langues.

Le principe utilisé dans notre filtrage s'appuie sur l'observation suivante : une paire de pages Web non parallèles est souvent mal alignée en phrase. Il y a beaucoup d'alignements vides, i.e. des phrases qui ne correspondent à aucune phrase dans l'autre langue. Ainsi, nous pouvons filtrer les paires avec un seuil sur la proportion des alignements vides.

De plus, nous tendons d'apporter certaines améliorations sur le processus d'alignement de phrases. Notamment, nous renforçons la correspondance entre deux phrases dont le rapport de longueur est proche du rapport standard entre les deux langues, et nous considérons les « traductions connues », i.e. celles stockées dans un dictionnaire bilingue. Ces approches ont apporté certaines améliorations sur le parallélisme du corpus.

Nous avons effectué trois séries de tests sur le processus de filtrage. La première série de tests examine le parallélisme du corpus nettoyé. La deuxième série examine la précision de traduction par les modèles entraînés avec le corpus nettoyé. La troisième série teste la performance de RIT en utilisant ces modèles. Ces tests montrent que l'approche que nous suggérons est efficace. Nous avons pu obtenir un corpus nettoyé qui aboutit à une meilleure précision de traduction (91.50% pour Chinois-Anglais, 87.50% pour Anglais-Chinois) et une meilleure performance de RIT (28.11% pour Chinois-Anglais, 26.01% pour Anglais-Chinois). Comme les méthodes utilisées sont indépendantes des langues, elles peuvent être utilisées pour filtrer de corpus parallèles par d'autres paires de langue.

Cette étude a contribué à confirmer que les pages Web parallèles obtenues automatiquement sont très utiles pour la RIT.

Mot de clé: Recherche d'information translinguistique, filtrage du corpus de textes, alignment des phrases.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In this thesis, we present our study on filtering raw parallel text corpus consisting of parallel Web pages, in order to improve the quality of the corpus and therefore the quality of translation models trained on them. This work is done in the context of Cross-language Information Retrieval (CLIR).

Traditional information retrieval (IR) systems have been well developed and widely used by libraries, governments, organizations, and corporations. But the emergence and fast growth of the Web in the last decade provided great demand and new challenges to IR systems. As Salton and Mcgill defined, "information retrieval is concerned with the representation, storage, organization, and accessing of information items. Items found in retrieval systems are characterized by an emphasis on narrative information. Such narrative information must be analyzed to determine the information content and to assess the role each item may play in satisfying the information needs of the system users." [SM83]. With the development of the Web, a huge amount of online resources is available. Various search engines have been put into service. Although these search engines look different from traditional IR systems, their basic functionality is the same. However, the document collection is dynamic and keeps changing. Another difference is the use in the different languages of documents and queries. Web users or Internet surfers, as well as search engines, are facing with the problem of CLIR. In a broad sense, CLIR refers to retrieving relevant documents in many languages from a given query. In a narrower sense, it means retrieving documents in one particular language other than that of the query. The problem focused by the narrow CLIR is fundamental, and we believe that before building a CLIR system in the broad sense, the narrow CLIR task should be solved first. In this thesis, CLIR is defined in the narrow sense.

Translation is the first problem of CLIR: to match a document and a query expressed in two different languages, either the document or the query should be translated. In general, we have three translation approaches to CLIR: translate documents into the query language [DL96]; translate query into the document language [Kwo99] [NSID99]; or translate both query and document into a third language. Intuitively, it is more feasible and easier to implement query translation. Although one may believe that document translation is more accurate because of more contextual information available, there is no solid experimental evidence supporting this. Therefore, we will focus on the query translation approach. Correspondingly, there are basically three groups of query translation approached: using a machine translation (MT) system, using a bilingual dictionary or a terminology database, and using a statistical translation model based on parallel texts.

MT systems have been studied and developed for decades. It seems to be a good tool for CLIR. Although there are several commercial systems for a number of major language pairs, we are still limited by its availability for many languages since there is not high quality MT system for many language pairs. Moreover, the translation by current existing MT systems is not always reasonable, and to create new MT system does not sound reasonable and feasible in most cases. Therefore, MT systems are possible but costly means for query translation in CLIR. In addition, as stated in [NSID99], the current MT approaches are not completely compatible to CLIR requirements:

- They spend much effort to generate syntactically correct sentences. This effort is irrelevant to the current practice in IR which is mainly based on keywords;

- They choose only one translation in the target sentence even though there are several synonyms. While IR is interested in adding synonyms and related words into the query so that more relevant documents may be retrieved.

The approach based on bilingual dictionary or lexical database has been used in several CLIR experiments. It usually works in a simple way as follows: it looks into the dictionary to retrieve the translation. The problem with it is word ambiguity because all the meanings of the source words are mixed up in the translations. In general, a simple way of using such a resource leads to a low IR effectiveness.

The third approach investigates translation relationship from a large amount of parallel texts, and forms a statistical translation model. The rationale is: the more two-words co-occur in parallel sentences that are translation of each other, the more likely they are translation of each other. The experiments over English-French showed that the approach could achieve a high CLIR performance comparable to MT approach [NSID99]. Our work is a constitution of that of [NSID99], but is carried out for English-Chinese CLIR.

The premise of the statistical translation model approach is the availability of a large-scale parallel text corpus. The rapidly developing Web brings us a huge potential source of parallel text corpus. There are many parallel pages on the Web, most between English and another language. If we can collect these pages efficiently, we then may construct parallel corpora at low cost. Jiang Chen [Che00] had successfully developed a mining system, PTMiner, that can gather parallel Web pages automatically from the Web. He created a large parallel text corpus for English-Chinese; trained translation models from this corpus, and applied it to CLIR experiments. His work shows that it is possible to collect a large parallel corpus of Web pages automatically, and that such a corpus can help to translate queries. However, if we compare the CLIR performance between English-Chinese obtained by Chen and those obtained for English-French, we observe a large difference: the performance for English-Chinese is much lower. The large difference between the languages is certainly an important reason. Another reason may be the poor parallelism of the corpus used for model training. The corpus is noisy: a certain number of web pages are not truly parallel. The goal of this research is to try to improve the quality of the parallel corpus and the translation model. The principles include:

- Improving corpus parallelism: to remove those noisy pairs that are judged as non-parallel from the corpus by filtering criteria. These criteria include checking the empty sentence alignment proportion, examining the length ratio between two paired texts and adjusting the weight of predefined alignment anchors, etc.
- Improving sentence alignment results: in addition to the length criterion, we also use several other criteria, such as known translation anchors and HTML tag cognates, etc.

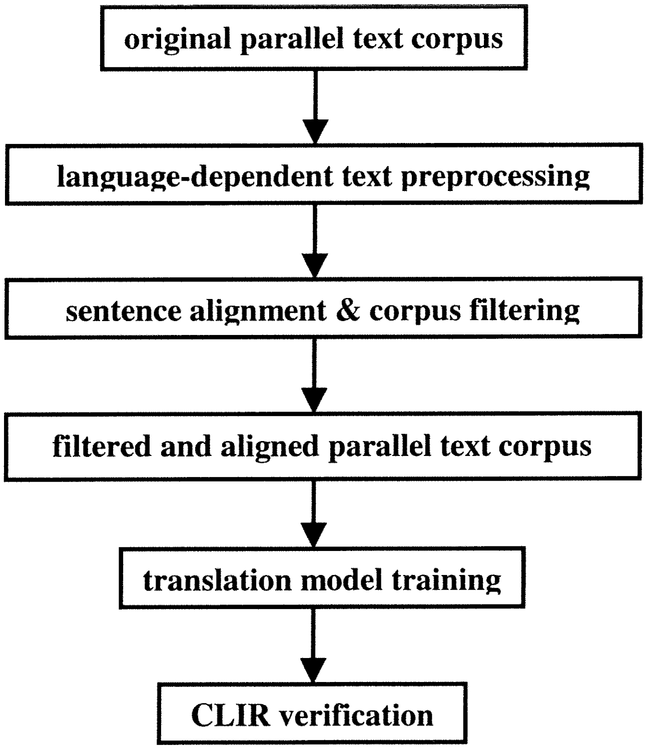
original parallel text corpus
language-dependent text preprocessing
sentence alignment & corpus filtering
filtered and aligned parallel text corpus
translation model training
CLIR verification

Figure 1.1 Main process flow of our project

The goal of our project is to obtain a better translation model for CLIR from a raw and noisy parallel corpus composed of parallel Web pages. Before we train our translation model, we conduct some necessary processes on the original corpus. The whole process flow is illustrated in Figure 1.1.

Our experiments show that after a proper filtering of the raw parallel corpus, we can obtain better translation models and as a consequence, higher CLIR performance. This work shows that our filtering approach is effective. It further confirms that it is an interesting and competitive approach to use parallel Web pages for CLIR. In the following chapters, we will present the details of the whole work.

In chapter 2, we will first give an overview of the background and previous results of CLIR. We will describe the key techniques of corpus treatments, and sentence alignment algorithms. Then we will give a brief introduction to Chen's work on Web pages mining and his results.

We will describe the principle and implementation of our filtering approach in Chapter 3.

Chapter 4 will cover the experimentation of sentence alignment with respect to corpus filtering, using a set of reference benchmarks.

In chapter 5, we will describe the experiments and results of translation model training.

CLIR experiments with trained translation model will be presented in chapter 6.

Chapter 7 will be the general summary and conclusion of our work.

Chapter 2

CLIR and Related Technologies

The explosive growth of information offers potential solutions to more and more demands, but the solutions themselves also bring new problems. The efficiency and convenience of modern recording and communicating technologies provide opportunity to access all knowledge and achievements of human beings, no matter what kind of language they use. This also generates the demand to efficient information searching, retrieving and collecting tools across languages.

Cross-Language Information Retrieval (CLIR), which aims to retrieve all relevant information in different languages, is one of many such endeavors by which people search and collect relevant information from available information bank to solve specific problem.

In this chapter, we will first introduce the main technologies and achievements of CLIR, especially the approaches of parallel text alignment. Then as a specific example, we will describe the PTMiner system that automatically mines parallel Web pages on the Web, and the Chinese-English corpus results obtained by Chen using the PTMiner. Finally, because our research aims to find out an approach to purify parallel text corpus of English and Chinese texts, some special text treatment technologies for Chinese and English will also be explained.

2.1 Background of CLIR

Information retrieval (IR) works as follows: a user wants to get documents about a certain topic by providing a free textual description of it as a query; from this query, the information retrieval engine selects index terms, which are matched against previously indexed documents; then the best matched documents are returned to the user in a ranked list.

CLIR, as a natural extension and development of traditional IR that limits the query and information in the same language, is the general name of methods and technologies of IR when the languages of query and information are different. As we mentioned earlier, CLIR can be divided into broad sense and narrow sense. Because the methods of the narrow sense are also suitable or easily adaptable to that of the broad sense, most research has been focused on and obtained for the CLIR in the narrow sense.

As Grefenstette [Gre98] pointed out, CLIR has three problems to be solved. "The first problem ...is knowing how a term expressed in one language might be written in another. The second problem is deciding which of the possible translation should be retained. The third problem is deciding how to properly weight the importance of translation alternatives when more than one is retained." According to the direction of translation, we can categorize CLIR translation approaches into followings:

- Translate the query into the language of the information collection (documents);
- Translate the information into the language of query;
- Translate both the query and the information into a third language.

Obviously, the translation of information is a time-consuming work. On the contrary, translation of query is more feasible and easier. So most CLIR translations are processed with the query translation approach. Query translation can be realized with different approaches: it can be done by a machine translation (MT) system, with bilingual dictionaries, or by a statistical translation model trained from some existing parallel materials.

Approaches based on a machine translation system consists of submitting the source to some existing tools and get the target translation. It is a fixed translation approach, since the translation tool is fixed after they have been created, so they cannot reflect the changes of real world unless an upgrade version or a complete new one is created. Moreover, Nie et al. argued that "MT and IR have widely divergent concerns" [NSID99]. The most distinct one is MT concerns to give syntactically correct translation while IR only cares about single word translation. Secondly, MT system always picks up one translation from all possible translations that are synonyms or related words, which prevent it from benefiting the natural query expansion that may improve IR performance. And, MT system is not available for many language pairs and is difficult to build. Nie et al. also discussed the weakness of dictionary-based query translation that cannot disambiguate among many possible translations in different contexts, and a simple use of such a resource leads to poor IR performances.

The approach of using statistical translation model is to exploit translation information from existing parallel materials between two languages. A pair of parallel texts is two texts that are translation of each other. The two texts of a pair usually have the same textual structure and style, a specific word and its translation usually appear in pair in two texts. Thus, strong translation relationships between these two words can be extracted if we have sufficient parallel texts to do statistical analyses. This is the rationale of the translation approach based on parallel corpus. Comparing to the above two approaches, this translation approach does not require a manually edited bilingual dictionary or to construct a complete MT system. Furthermore, since we can generate a new translation model by changing the contents of the corpus, the obtained translation model will be sensitive to the parallel materials. In other words, if the materials are up-to-date, this translation approach should have following advantages:

- Dynamically obtain translations of new words or new meaning of existing words;
- Automatically generate the new translation model.
- Word translations are domain sensitive, as embodied by the training corpus.

The problem of statistical translation model approach is often the unavailability of large parallel text corpora for many language pairs. For French-English, there is the Hansard Corpus that contains a large set of parallel texts of the Canadian parliament debates. But for other language pairs such as Chinese-English, there is not parallel corpus of comparable size and coverage. Hong Kong Legislative Council has all its Records of Legislature in both Chinese and English, but they are in a narrow legal domain [http://www.legco.gov.hk/]. Basing on these records, Hong Kong University of Science and Technology created their HKUST English-Chinese Parallel Bilingual Corpus which contains in total 60MB of raw data in both languages [Wu94]. Because the quality of a statistical translation model depends on the parallel corpus, to obtain or create such a corpus is crucial to obtain an acceptable CLIR performance. This is why the PTMiner system has been developed. We will describe it in Section 2.3. Before then, let us assume that there is a large parallel corpus, and describe the treatments on such a corpus, namely sentence alignment and translation model training.

2.2 Sentence Alignment Approaches

The goal of training translation model is to determine the translation relationship of words. Text alignment is an important step to realize it, whose objective is to find the translation mapping between two units of parallel texts then leads to the mapping of words. According to the size of aligned unit, the alignment can be categorized into text, paragraph, sentence and word level alignment. The documents in a parallel corpus are aligned at text level, or are translation of each other for two texts without any further constraint. A word may match any word in the target text. A translation model trained in this way will lack focus in its translations: almost every word in the parallel text is a potential translation. It is possible to define the smaller elements as words, but in order to restraint the translation relationship within a narrow scope, we have to refine a pair of parallel texts into a set of parallel units that that are smaller than text.

Aligning two parallel texts at word level is difficult, because the quantity, position and frequency of a specific pair of words in two mapping sentences of two parallel texts may be very different. On the other hand, sentence level alignment is more feasible because sentence is the smallest meaningful word group that has a good correspondence in the languages: one sentence is often translated into one sentence in another language. Then sentence alignment is a good compromise between refinement of text unit and alignment feasibility.

Not every sentence exactly maps to one sentence in another language. The alignment results may not only include 1-to-1 mapping, but also m-to-n alignments such as 1-to-2, 2-to-1, 2-to-2, etc. Various sentence alignment algorithms based on different principles have been proposed. In the following subsections, we will describe some of them.

2.2.1 Length-based Sentence Alignment Algorithm

Alignment is the first stage in extracting structural information and statistical parameters from parallel corpus. Sentence alignment is "to identify correspondences between sentences in one language and sentences in the other language" of two parallel texts [GC91]. So we need to identify common characteristics of two sentences first. Because sentence is composed of strings of characters and/or spaces, no matter the language, sentences have a common property: length. Gale and Church found that the correlation degree between the length of a paragraph in characters and the length of its translation is very high (0.991), which suggests that length is a strong clue for sentence alignment. It is also found that there is a relatively stable length ratio between two parallel sentences of any two languages [GC91] [Wu94]. In the reverse order, if the length ratio of two relevant sentences in a parallel text couple satisfies some length criteria, then we can infer that they are translation of each other or parallel. The alignment algorithm based on length difference is called length-based sentence alignment algorithm.

General Principle

Given a pair of texts T_1 and T_2, S_1 and S_2 are sentence chunks of T_1 and T_2 respectively, a length-based alignment is to choose the alignment A that maximizes the probability over all possible alignments. Formally, as

$$A = \arg\max_A \Pr(A|T_1, T_2) \qquad (2.1)$$

where Pr indicates probability. Usually, two approximations are commonly made as follows: first, the probabilities of the individual aligned pairs within an alignment are independent, i.e., every match does not interfere with others; second, the probability of every match depends not on the entire texts, but only on the contents of the specific sentence chunks within the alignment, i.e., the context is ignored.

The maximization problem of the alignment probabilities can be converted into a minimum-sum problem, which suits to implement in a dynamic programming framework. Let $S_1 \Leftrightarrow S_2$ indicate that S_1 matches S_2, if $(S_1 \Leftrightarrow S_2) \in A$, $l_1 = \text{length}(S_1)$ and $l_2 = \text{length}(S_2)$, then length-based alignment A can be finally described as:

$$\begin{aligned} A &= \arg\max_A \Pr(A|T_1, T_2) \\ &\approx \Pi_A \Pr(S_1 \Leftrightarrow S_2 \mid T_1, T_2) \\ &\approx \arg\max_A \Pi_A \Pr(S_1 \Leftrightarrow S_2 \mid S_1, S_2) \\ &= \arg\min_A \Sigma_A (-\log \Pr(S_1 \Leftrightarrow S_2 \mid S_1, S_2)) \end{aligned}$$

$$\approx \arg\min_A \Sigma_A (-\log \Pr(S_1 \Leftrightarrow S_2 \mid l_1, l_2)) \qquad (2.2)$$

Then, the problem becomes to a minimize sum problem of -log $\Pr(S_1 \Leftrightarrow S_2 \mid l_1, l_2)$. It can be looked at as finding the "shortest distance" problem between aligned elements within two sequences, which can be solved using dynamic programming.

Gale-Church Algorithm

Basing on above principle and statistical investigation, Gale and Church proposed a typical length-based sentence alignment algorithm [GC91]. The rationale of Gale-Church algorithm is that "longer sentences in one language tend to be translated into longer sentences in the other language, and shorter sentences tend to be translated into shorter sentences". In detail, the algorithm can be described as the followings.

Gale-Church algorithm includes two alignment steps. First, the parallel texts are aligned at paragraph level; then the paired paragraphs are aligned at sentence level. In a specific paragraph pair, a probabilistic score is assigned to each possible correspondence of sentences, based on the ratio of lengths of the two sentences (in characters) and the variance of this ratio. Here the equation (2.2) is expressed as:

$$\arg\max_A \Pr(A|T_1, T_2)$$
$$\approx \arg\min_A \Sigma_A -\log \Pr(S_1 \Leftrightarrow S_2 \mid \delta(l_1, l_2)) = \min_A \Sigma_A d \qquad (2.3)$$

where δ is a function of length l_1, l_2, d is the distance $d = -\log \Pr(match|\delta)$. Applying Bayes Rule, we have:

$$\Pr(S_1 \Leftrightarrow S_2 \mid \delta) = \Pr(\delta|S_1 \Leftrightarrow S_2)\Pr(S_1 \Leftrightarrow S_2) / \Pr(\delta)$$

where $\Pr(\delta)$ is a normalizing constant that can be ignored during minimization because it is the same for all proposed matches. The other two distributions are estimated as follows. First, the conditional probability is estimated by

$$\Pr(\ \delta|S_1 \Leftrightarrow S_2\ ) = 2\ (1-\Pr(|\delta|)\ )$$

where $\Pr(|\delta|)$ is the probability with δ defined as $(l_2 - l_1\ c)\ /\sqrt{l_1 s^2}$ so that it has a normal distribution with mean zero and variance one, where c is the expected number of characters in target language S_2 per character in source language S_1; s^2 is the variance of the number of characters in S_2 per character in S_1.

The prior probability of *match*, $\Pr(\ S_1 \Leftrightarrow S_2\ )$, is obtained from statistical investigation result Table 2.1, in which six types of matching are defined and their probabilities derived from the statistical matching proportions over the manual aligned trilingual USB bank report corpus are also given.

Matching category	***Matching proportion Pr($S_1 \Leftrightarrow S_2$)***
1-to-1	89%
1-to-0 or 0-to-1	0.99%
2-to-1 or 1-to-2	8.9%
2-to-2	1.1%
Total	100%

Table 2.1 Sentence matching probability of USB corpus [GC91]

Moreover, if the distance function d is defined in a general way to reflect the influence of insertion, deletion, substitution, contraction, expansion and merging. Then d is defined in equation (2.3) but takes four arguments: x_1, y_1, x_2, y_2 as:

- $d(x_1, y_1; 0, 0)$, cost of substituting x_1 with y_1.
- $d(x_1, 0; 0, 0)$, the cost of deleting x_1.
- $d(0, y_1; 0, 0)$, the cost of insertion of y_1.
- $d(x_1, y_1; x_2, 0)$, the cost of contracting x_1 and x_2 to y_1.
- $d(x_1, y_1; 0, y_2)$, the cost of expanding x_1 to y_1 and y_2.
- $d(x_1, y_1; x_2, y_2)$, the cost of merging x_1 and x_2 and matching with y_1 and y_2.

Finally, to search for the best alignment among many possible alignments, a dynamic programming framework is used in the following recursion equation. Let s_i, $i = 1 \ldots m$, be the sentences of language S_1; t_j, $j = 1 \ldots n$, be the translation in language S_2; d be the distance function; and $D(i, j)$ be the minimum distance between sentences $s_1 \ldots s_i$ and their translation $t_1 \ldots t_j$, under the maximum likelihood alignment. Then, the distance $D(i, j)$ is computed by minimizing over above six cases and the probabilities of supposed match sentences. That is, $D(i. j)$ is defined by following recurrence with the initial condition $D(i, j) = 0$.

$$D(i,j) = \min \begin{cases} D(i, j\text{-}1) + d(0, t_j; 0, 0) \\ D(i\text{-}1, j) + d(s_i, 0; 0, 0) \\ D(i\text{-}1, j\text{-}1) + d(s_i, t_j; 0, 0) \\ D(i\text{-}1, j\text{-}2) + d(s_i, t_j; 0, t_{j\text{-}1}) \\ D(i\text{-}2, j\text{-}1) + d(s_i, t_j; s_{i\text{-}1}, 0) \\ D(i\text{-}2, j\text{-}2) + d(s_i, t_j; s_{i\text{-}1}, t_{j\text{-}1}) \end{cases}$$

A series of evaluation experiments have been performed on a trilingual corpus in English, French and German, and on the bilingual Canadian Hansards of French and English. Gale-Church algorithm had been proven to be quite accurate, and fairly language-independent.

Other Length-based Algorithm

Rather than Gale-Church algorithm, Brown et al. also proposed another typical length-based algorithm [BLM91]. Brown algorithm uses the same rationale of Gale-Church algorithm, aligning texts only by counting the length of parallel documents. The difference of them is that Gale-Church algorithm counts length by character while Brown algorithm by word. Because there is not comparison between words in several Asian languages, Brown algorithm basing on words is less applicable to these languages than the Gale-Church algorithm.

2.2.2 Other Sentence Alignment Algorithms

Obviously, as translation products of natural languages, many corpora of parallel texts are not literally translation and easy to align as Canadian Parliamentary Debates (Hansards). Because purely length-based alignment fully depends on the length difference of two sentences, it may not lead to a good result if the training parallel texts are not well constructed or include much noise, as pointed out by Simard at al [SFI92] and Chen [Che93]. Therefore, some other supplementary alignment algorithms are proposed.

Algorithm Based on Cognates

All European languages are alphabetic languages and most of them are derived from the same source – ancient Latin language, and there have been many communications and exchange among them. So many words in different languages are literally similar or even the same. These words that share common phonological, orthographic and semantic properties in different languages, are called cognates. In a wider sense, cognates may include numerical expressions, special symbols and punctuation, which are often the same in any language. This property could be utilized to improve alignment algorithm as Simard et al. [SFI92] did.

Usually, cognates in parallel sentences of different languages have similar or the same semantic meaning, and are translation of each other. On the other hand, if two cognates appear in two sentences of parallel texts, it can be reasonably inferred that their sentences match. Simard et al. made a simpler definition of cognates instead of its definition in linguistics: cognates are determined only on the first four letters for their French and English pairs. If a pair of French and English words starts with the same four letters, then these words are cognates [SFI92].

Their experiments showed that the alignment based on cognates alone is worse than that based on length, but better results were obtained by combining cognate clues with the length criterion. Concretely, Simard et al.'s algorithm includes two stages. First, an initial alignment result list is generated by using purely length algorithm, or uses length criterion to filter out unlikely alignment. Second, an improved alignment result is obtained by using cognates to identify the overall best alignments of those candidates that remained in the first stage. This combination experiment can also be executed on the reserve order as [SP98] did. Another cognate-based algorithms include "Char_align" developed by Church [Chu93], when it is difficult to determine paragraphs and sentences in the texts to be aligned as is the case of OCRed texts. This algorithm uses Simard et al. cognate-based algorithm, but aligns texts at character level rather than at paragraph or sentence level.

Word-based Algorithm

Cognates require similar phonological or orthographic features between words in different languages, this is not applicable to some languages such as Chinese. Rather than using cognates which are not shared by many languages, another possible lexical clue is words themselves. Those algorithms use words as alignment anchors, are called word-based alignment algorithm. Among those algorithms, Chen proposed a pure word-based algorithm [Che93], which sets up a statistical word-to-word translation model on the fly during sentence alignment and searches for the algorithm that maximizes the probability of generating the corpus with the model.

The basic rational of Chen's algorithm is that the 1-to-1 alignment takes most proportion and higher probability. They view a bilingual corpus as a sequence of sentence beads, which corresponds to an irreducible group of sentences that align with each other. Furthermore, each sentence bead is considered as a sequence of word beads that includes 1:0, 0:1 and 1:1 three types. In this algorithm, five cases of 1-to-0, 0-to-1, 1-to-1, 2-to-1 and 1-to-2 sentence alignment are considered. First, manually align some sample sentences to create a basic translation model, then expand it to the complete translation model during alignment. Over this model that includes the translation probability of word bead, matching probability of sentence beads are calculated by generating target sentence from source sentence word by word. Similar to length-based algorithm, the results are finally sent into a dynamic programming framework with thresholding to find out the maximum possible alignment result, by converting into a minimum distance problem, between sentences [Che93]. Chen obtained a low error rate of approximately 0.4% over Canadian Hansard with his algorithm. In contrary, Kay and Roscheisen proposed an algorithm that repetitively provides word level and sentence level alignments [KR93].

Theoretically, this algorithm is more robust than pure length-based one when there are many deletions in corpus. Chen has got better result over the same corpus. However, when automatically treat with noisy corpus of distinct languages, its disadvantages are also obvious:

- it requires human intervention of manually aligning sample sentences;
- its word bead model usually applies to those languages that have large amount of cognates, such as English and French. And it assumes sentence structures of two languages are similar.
- it considers the word order and grammatical role that are usually ignored in IR;
- its probability computation is very complicated and not efficient;

Those assumptions and considerations limit its utilization over noisy, poor parallel and non-alphabetic language alignment.

Similarly, Utsuro et al. set up a bilingual text matching framework for Japanese and English, which includes two steps: sentence alignment and structural matching of bilingual sentences [UIYM94]. In the text matching framework, texts are viewed as sequences of sentence beads, which are considered as sequences of word beads. Before sentence alignment, content words are extracted from each sentence (after each sentence is morphologically analyzed if necessary), and word correspondences are found using both existing bilingual dictionaries and statistical information source for word correspondence. Then this correspondence is used to align sentences by calculating the alignment score of a sentence bead, which will also be optimized with dynamic programming.

The main disadvantage of [UIYM94] algorithm is that it depends too much on the existing dictionary and statistical information source, which both cannot cover all information needed for correctly align texts. Furthermore, it requires the parallel texts to be domain specific, and requires extracting content words and morphological analysis first, which are both infeasible for the corpus of parallel Web pages

Algorithms for Non-European Languages

Between those languages that do not have any common property, such as alphabetic (phonetic) European languages and non-alphabetic (pictographic) Asian languages, there is not any inherent lexical property, likes cognates or words that could be directly utilized, as pure lexical algorithm does, to help sentence alignment. On the other hand, pure length-based algorithm also was proven to have poor performance over English and Chinese corpus with a little noise [Wu94]. So some other approaches have to be created to meet those requirements. The algorithm used by Wu over English-Chinese corpus is one of such efforts [Wu94].

Chinese is an Asian language completely different from English. The correspondence between words or characters is very weak between alphabetic English and pictographic Chinese, so pure length-based or lexical algorithm often gets poor performance. Wu's research is a hybrid of length-based and lexical algorithm, it is conducted over the debate recordings of Hong Kong Legislative Council in Chinese and English. Parliamentary debate recordings usually include head and body parts. The body parts of most recordings are fully translated, which can be well aligned by length-based algorithm. On the other hand, the head parts are format fixed and domain specific, but contain a lot of deletions and other mismatching, so the alignment here is often poor and error prone. It is found over the debate corpus that the alignment accuracy is only 86.4% for the entire text using pure length-based algorithm, while 95.2% is obtained when only aligning the body part. It is also found that a few words appear very frequently in head parts and can be looked at as lexical anchors. So they created a small and typically domain-specific lexical cue set as Table 2.2(a)(b) shows to help alignment [(1)]. Besides using length-based Gale-Church algorithm, they imported a lexical parameter basing on the occurrence number of these lexical cues to calculate the alignment probability and to improve alignment. Formally, equation 2.2 becomes as:

$$\begin{aligned} & \arg\max_A \Pr(A|T_1, T_2) \\ & \approx \arg\max_A \Pi_A \Pr(S_1 \Leftrightarrow S_2 \,|\, S_1, S_2) \\ & = \arg\min_A \Sigma_A \text{-}\log \Pr(S_1 \Leftrightarrow S_2 \,|\, S_1, S_2) \\ & \approx \arg\min_A \Sigma_A \text{-}\log \Pr(S_1 \Leftrightarrow S_2 \,|\, l_1, l_2, v_1, w_1, \ldots v_n, w_n) \qquad (2.4) \end{aligned}$$

where v_i = number of occurrence of ith English cue appear in S_1, and w_i = number of occurrence of ith Chinese cue appear in S_2. Again, the dependence is encapsulated in parameters δ_i, which are assumed independent each other and normally distributed, as equation (2.5) shows. Finally, the same dynamic programming optimization of Gale-Church is used.

$$\begin{aligned} & \Pr(S_1 \Leftrightarrow S_2 \,|\, S_1, S_2) \\ & \approx \Pr(S_1 \Leftrightarrow S_2 \,|\, \delta_0 (l_1, l_2), \delta_1 (v_1, w_1), \ldots, \delta_n(v_n, w_n)) \qquad (2.5) \end{aligned}$$

The improvement is obvious as the alignment accuracy over the entire corpus rises to 92.1% from 86.4%.

Although the lexical cues listed in Table 2.2(a)(b) are limited and domain specific: only 28 pairs of title and date words, Wu's algorithm still obtained an encouraged improvement result. The main reasons include the following facts:

[(1)] Colons are the only delimiters to separate cataloges and their contents in the head part.

English	***Chinese***
:(1)	:(2)
Governor	总督

Table 2.2(a) Lexical cues for paragraph alignment [Wu94]

English	***Chinese***	***English***	***Chinese***	***English***	***Chinese***
C.B.E.	C.B.E.	C.M.G.	C.M.G.	I.S.O.	I.S.O.
J.B.E.	J.B.E.	J.P.	J.P.	K.B.E.	K.B.E.
Q.C.	Q.C.	January	一月	February	二月
March	三月	April	四月	May	五月
June	六月	July	七月	August	八月
September	九月	October	十月	November	十一月
December	十二月	Monday	星期一	Tuesday	星期二
Wednesday	星期三	Thursday	星期四	Friday	星期五
Saturday	星期六	Sunday	星期天		

Table 2.2(b) Lexical cues for sentence alignment [Wu94]

- The lexical cues are directly selected from the specific parts (head) of texts of the corpus themselves.
- The mismatching of parallel texts mainly occurs in the specific part.
- The sampling parts of all texts in the corpus are domain specific, while the rest parts are fully translated and well aligned.

For a particular domain specific corpus, it is possible to set up such a small number of lexical cues. Unfortunately this premise is not true to our diverse and noisy corpus. However, in some degree, Wu's algorithm still provides a good example to

(1) colon in one-byte ASCII code

(2) colon in two-byte Chinese Big5 code

parallel text alignment involving two completely different languages. In our project, we will expand and modify this approach by combining with other algorithms to align our noisy English-Chinese corpus.

Fung and McKeown proposed their "DK-vec", an algorithm for producing a small bilingual lexicon from noisy parallel texts according to frequency, position and recency information [FM94]. This algorithm does not consider natural sentence boundaries so as to avoid errors caused by noises in corpus. The created new corpus might be used as anchor points in the following alignment stage, which may use any independent alignment algorithm. It has been tested over English-Chinese and English-Japanese corpora. Its sibling algorithm "K-vec", developed by Fung and Church [FC94], used word distribution to align parallel texts. The involved lexicon was not an external bilingual dictionary but created by investigating and calculating the occurrence distribution of frequently used words in two languages. Fung also showed an algorithm which extracts a bilingual lexicon from noisy parallel corpus without sentence alignment [Fun95]. However, the lexicon extracted only covers a small part of the strongest lexical translation relationships. So this approach is not appropriate to CLIR, which requires a balance between precision and recall.

In the project of ARCADE, Langlais et al. tested several sentence alignment algorithms [LSV98]. They found that, although separating a text first into blocs (e.g. paragraphs) is a good way to reduce the search space, and to increase the efficiency of the alignment algorithm, this step is not mandatory. With an efficient search algorithm, one can directly proceed sentence alignment.

2.3 PTMiner and CLIR Using Translation Model

As we mentioned in section 2.1, the availability of parallel corpus is the bottleneck for constructing statistical translation model. As there are many exchanges and communication among European languages, many documents are published with two or more European languages, and many collections of such parallel documents are available. Most of such documents are manually constructed and elaborately translated. For example, the famous Canadian Hansards, the collection of Canadian parliamentary proceedings are published in both English and French. They are the natural resources of parallel texts. Basing on those well-constructed parallel corpora, many CLIR approaches with statistical translation model over European languages are used in the past years, such as Davis et al. did over English-Spanish parallel corpus [DDO95].

Unfortunately, there are few such corpora for Asian languages. Then, the first work is to create a parallel text corpus for Asian languages such as Chinese. Some small size corpora such as HKUST English-Chinese Parallel Bilingual Corpus have been manually created [WX95]. However, it is not accessible to public. And its texts are too domain specific, which only contains the debate records of Hong Kong Legislative Council.

In recent years, due to the rapid development and popularization of World Wide Web, more and more parallel texts appear on the Internet, even for Asian languages. Internet is a boundless ocean of information, and many Web pages are parallel, i.e. there is a translation of a Web page in another language. This provides us an opportunity of creating large-scale parallel text corpus by collecting texts from the Web. The first version of an automatic miner for parallel texts was described in [NSID99]. This miner was later further developed by Chen into its current form – PTMiner [Che00].

PTMiner, is a multi-tier distributed parallel text miner developed to search for parallel texts from the Web. PTMiner is mainly developed from the Web Crawler package of the Intelligent Miner for Text of IBM [IMT99] [Tka98]. It relies on some traditional Web search engine (as AltaVista) to obtain potential bilingual sites and URLs of documents in these candidate sites, then discovers parallel documents according to common naming patterns. PTMiner's architecture can be briefly depicted as Figure 2.1, Its mining processes include:

- Step 1, Candidate Web sites search – search from the Web search engines for the candidate sites that could contain parallel pages. AltaVista search engine is used for this step. A query such as "anchor: Chinese AND anchor: English" is sent to AltaVista to look for documents in English containing the anchor text "Chinese". The anchor text in such a document is likely to be a link to its Chinese version. All sites containing such a document are selected as a candidate site.
- Step 2, File name fetching – for each candidate site, fetch the URLs of Web pages that are indexed by the search engines. This base on the assumption of parallel texts existing in the same site, and all files' names are collected by PTMiner from the site, then are scanned.
- Step 3, Host crawling – starting from above collected URLs, crawl each candidate site separately for more URLs. This step tries to obtain as many Web pages as possible from the candidate sites. It uses the same crawling algorithm of the web crawler of search engine. Chen found 55,971 file names from one specific domain (hk).
- Step 4, Pair scan – scan for possible parallel pairs among all document identified according to common naming patterns from each obtained URL. The assumption is that two parallel Web pages often have similar names such as "index_e.html" and "index_c.html".
- Step 5, Downloading and verifying – download the possible parallel pages, and determine file size, language identification and character set of each page, and then roughly filter out non-parallel pairs.

In the architecture of Figure 2.1, PTMiner DB serves as the storage of intermediate and final mining results as well as working situation of the servers. PTMonitor is a GUI interface to facilitate the monitoring of the whole mining process. Scanner Sever and Crawler Sever both are CORBA servers, they register in the database and notify PTMonitor. Crawler Server receives invocation from PTMiner Server, then fetch the candidate site and fetch file names. Scanner Server sends message to PTMonitor, takes site name and scans for parallel pairs by naming patterns. PTMiner sever is the central control unit, it synchronizes the real workers and other servers according to information in the database.

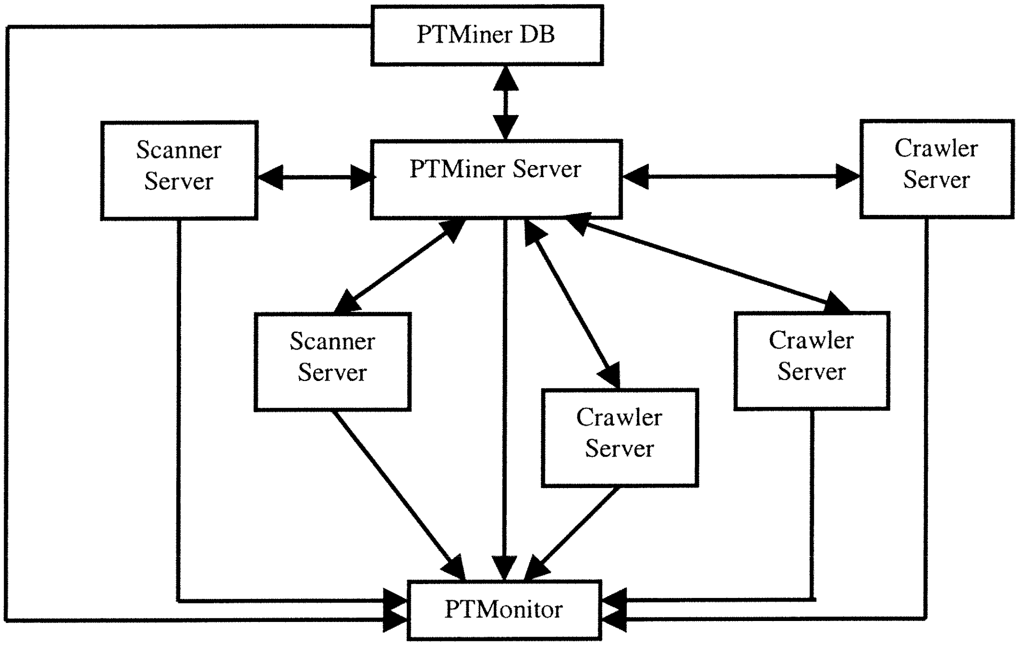

Figure 2.1 Architecture of PTMiner

The result obtained by PTMiner is a parallel corpus, containing 19,835 pairs of parallel Web pages in Chinese and English downloaded from hk (Hong Kong) domain after Step 4 “pair scan”. Among them 14,820 pairs are finally verified in Step 5 as true parallel pairs, including 117.2M Chinese texts and 136.5M English texts. By examining randomly selected samples by human judge, the parallelism of created corpus is estimated to be 82%. This means that a lot of noise exists.

Once the pseudo-parallel text corpus has been created, translation models between English and Chinese can be trained over it. But before training is conducted, the raw parallel files needed to be treated into the form suitable to train. The pre-training process flow of training system is illustrated as Figure 2.2.

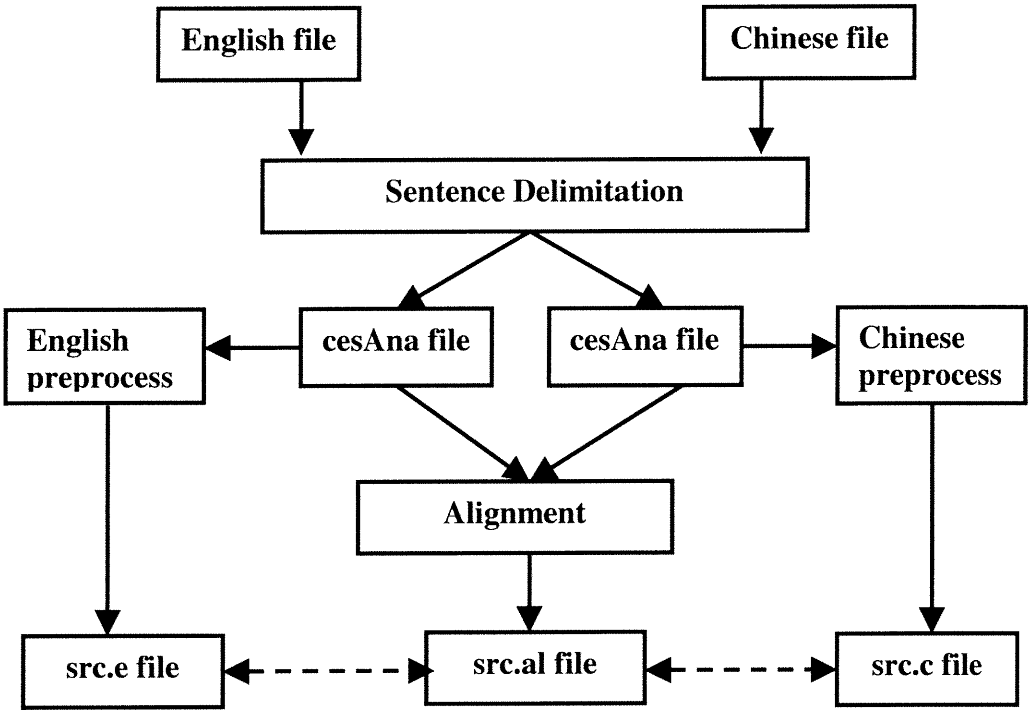

Figure 2.2 Pre-training procedures of Chen

The first pre-training stage is sentence delimitation. Chen used both punctuation and HTML tags to delimit sentences because all files are in HTML format. Then necessary preprocesses include striping off HTML markups, English citation, English expression extraction, Chinese code conversion and Chinese segmentation are carried out on English and Chinese texts respectively. Meanwhile, parallel files are converted into cesAna format, upon which paired texts are aligned under a derivation of SFI alignment algorithm [SFI92]. The results of pre-training processes are two source files: src.e and src.c, and a file containing the alignment src.al (see Figure 2.2). These three files are the input to the training process of statistical models.

Chen used the parallel corpus to train an IBM model I translation model. The accuracy of the resulting models in both directions is listed as Table 2.3. The accuracy was tested with 200 randomly selected words. Only the first translation of

Translation model	English-Chinese	Chinese-English
Translation accuracy	81.5%	77%

Table 2.3 Accuracy of translation model

each word is accessed. Finally, Chen verified these translation models with CLIR experiments. Both English to Chinese and Chinese to English directions are tested, TREC 5, 6, or 7 data are used for the experiments. If only the generated translation models are used, the CLIR performance in average precision is around 40% of that of monolingual IR performance, as shown in Table 2.4. If an extra bilingual dictionary is used together with the translation model, the precision is well improved as shown in Table 2.5.

Chen's work is a useful and successful attempt to create parallel corpus for translation model by searching and collecting online materials over European language (English) and Asian language (Chinese). His results proved that it is a feasible approach to automatically mine parallel Web pages and used them for CLIR. However, the final translation accuracy and CLIR precision of Chen's results still have room for improvement, comparing to those results obtained from manually constructed corpus between European languages that are typically around 80% of the monolingual IR performance.

Besides the larger difference between English and Chinese, another important factor is the translation of the parallelism of the training corpus, which is the basis of statistical translation model-based training. However, Chen's corpus only has 82% truly parallel texts. This naturally leads to an unsatisfactory CLIR performance. The questions we raise in our current study are:

- Is it possible to filter the corpus to improve its parallelism?
- How to effectively carry out the filtering?
- What is the impact of this filtering process on CLIR?

CLIR direction	*CLIR precision*	*Ratio of mono-IR*
English to Chinese	15.91%	40.00%
Chinese to English	16.54%	42.8%

Table 2.4 CLIR precision only using TM

CLIR direction	*Combination ratio (TM:Dict)*	*CLIR precision*	*Ratio of mono-IR*
English to Chinese	2:1	22.32%	56.10%
Chinese to English	1:1	25.83%	66.90%

Table 2.5 CLIR precision using both TM and dictionary

In our project, we will take the corpus downloaded by Chen as our original parallel text corpus. We notice that Chen did not carry out an elaborated verification once Web pages were downloaded. Only the text length was used. In fact the alignment process is also an effective tool to detect if two texts are parallel: usually, non-parallel texts will have difficulty to be aligned into parallel sentences. Therefore, in our study, we will mainly elaborate different strategies of using and modifying the alignment algorithms for the filtering process. This process will be discussed in detail in Chapter 3. In the following subsections, we will continue our presentation of the necessary preprocessing on Chinese and English texts, as well as the translation model training process.

2.4 Preprocessing to Chinese and English Text

The training of a translation model requires aligned parallel text input, in which the sentences or paragraphs are clearly delimited, words are explicitly defined and in proper form. Both texts of a pair should be arranged in the same or comparable form.

Then we can statistically obtain the mapping of individual words between source and target languages, and finally determine the translation relationship between words. However, both original English and Chinese texts are not suitable to import to translation model training. For example, the same words in English texts may appear in various forms, it is better to transform them into a standard form – the citation form; there is no separation between Chinese words, so they are necessary to be segmented.

2.4.1 Chinese Text Properties and Treatments

Chinese is mostly a pictographic language, rather than alphabetic language. It has following major specific characteristics and needs following treatments respectively.

Coded Character Sets

Chinese characters are stored in computer using different encoding schemes. Two commonly used schemes are GB for simplified Chinese and Big5 for traditional. Both GB and Big5 use two bytes for each Chinese character, called full-width character comparing to one-byte alphabet, which is called half-width character.

The texts used in our tests are encoded in GB, whereas the parallel Web pages downloaded from Hong Kong are mostly encoded in Big5. Therefore, we convert all the parallel Web pages into GB format. This conversion will create some errors because there is not always 1-1 correspondence between traditional Chinese characters and simplified characters. However, this small error rate will not affect much the CLIR process. Several conversion tools exist for public uses. In our case, we used the conversion tool NCF (Network Hanzi Filter) [http://www.ifcss.org/].

Punctuation

Punctuation is the natural delimiter of sentences. However, the punctuation system of Chinese is different from that of English. In both GB and Big5 code sets, each Chinese punctuation symbol is encoded in two bytes, while that in English is encoded in one byte. Therefore, we convert all Chinese punctuation of full-width into ASCII format that takes one byte. So that the punctuation marks can be recognizing by an alignment program. The code conversion tool we utilized is bd2punc [Che00], which uses a manually created mapping table between two punctuation systems.

Segmentation

Writing Chinese sentence (also Japanese and Korean) is a continuous string of characters without space between words. No character gives morphological hints to word boundaries and any Chinese character could be a word. Our goal is to train a translation model, or to determine the mapping relationship between words of two languages, So it is important to determine the word boundaries, or to carry out a word segmentation process.

The difficulties of Chinese segmentation mainly come from the vagueness of word definition and the so-called word-chain problem [Liu87] [HB98]. In the past decade, many segmentation approaches have been developed. They can be categorized into dictionary-based approach as [LZ91] [CK92] and statistical approaches as [Ca91] [SS91]. Dictionary-based approaches rely on dictionaries and heuristic rules that corresponding to common word structures. Heuristic segmentation rules include maximum matching, error-driven learning, overlapping ambiguity detection, combination ambiguity detection, etc. [QTS92] [LC94] [HB98]. Statistical approaches first learn statistical information (e.g. a Markov model) from training corpora, and use it to determine words. The coverage and size of the training corpora are crucial to the performance of segmentation. Some hybrid approaches combining the above two approaches are also proposed, such as flexibly incorporating statistical information with dictionaries and heuristic rule of [NJH94] and [NRB95]. In our project, a segmentation tool called "mansegment" developed by Zhibiao Wu, at Linguistic Data Consortium available at [http://www.ldc.upenn.edu/] was used. This program uses the length matching algorithm combined with the frequency of words. This idea is to choose the most frequent and the longest words. However, the dictionary used "Mandarin.fre" is limited in size. It contains 44,405 entries. To extend the word coverage, we enriched this dictionary by several dictionaries found on the Web. They are "Berkeley.Chinese.Dictionary" and "LDC_CE_DICT2.0". The final dictionary contains 310,430 entries, including some overlaps.

2.4.2 English Text Properties and Treatments

Citation Form

English text is composed of different combination of limited alphabets and symbols, and its words change forms depend on subject, tense, mood, etc. Training translation model aims to figure out the mapping relationship between prototype words, so it is necessary to convert morphological transformed words back into their prototypes.

The morphological conversion called citation, implements a search and matching algorithm, to find out and converts all transformation words in the text. It searches all words of the input text in the dictionaries, if any transformation entry is matched, then this word will be replaced with its prototype. For example, words "am", "are", "is", "was", "were", "been" are all transformed into "be". In our project, the citation tool developed by Rali group is used for citation [http://www-rali.iro.umontreal.ca].

Expression Extraction

Idioms or fixed collocated phrases often appear in English. They should be treated as basic units in the translation models. Therefore, idioms and phrases of English have to be replaced by some non-space characters or symbols before training translation model. This procedure is called expression extraction.

In our project, expression extraction is processed by comparing all input word sequences with external dictionary, all expressions matching with any entry of the dictionary are reformatted into a single word by replacing spaces with underscore "_". For instance, phrase "get rid of" is transferred into "get_rid_of". Moreover, except prepositions and conjunctions, other member words of an expression usually have their own semantics meaning. In order to enable the translation of single words, that are members of expression, we keep both the extracted expression as an unit and its member words. For example, for expression "children day", we recognize "children" and "day" as well as "children_day".

2.5 Translation Model Training

Training statistical translation model is to learn the translation probability between words of two languages and setup models of translation relationship between them by statistically comparing the existing translation examples, which are those sentence- aligned parallel texts of our filtered corpus here. Our translation models are IBM model I [BPM93], whose basic rationale is: given aligned translations, if two words often co-appear in both the source and target sentences, there is a higher possibility that they are translation of each other. Specifically, from a large collection of alignments, the model learns the probability $p(t|s)$ of having a word t in the translation of a sentence containing word s. For an input sentence, the model then calculates a sequence of words that are most probable in its translation.

The principle of training can be described as followings. For a single alignment a_k between the source sentence S and the target sentence T, we have two sets of words:

$$S = \{s_1, s_2, s_3, \ldots, s_m\},$$

$$T = \{t_1, t_2, t_3, \ldots, t_n\}.$$

We consider each word $t_{j,} j = 1, 2, 3, .., m$ in T as a possible translation of each word $s_{i,} i = 1, 2, 3, .., n$ in S. All the possibilities are treated as equivalent. We then have

$$p(t_j|s_i, a_k) = C_T/m$$

where C_T is a parameter related to the length of the target sentence. Now, for a set of alignment A, we calculate the overall probability $p(t_j|s_i, A)$ from all $p(t_j|s_i, a_k)$ by

$$p(t_j|s_i, A) = C_A \Sigma_k\, p(t_j|s_i, a_k)$$

where C_A is a normalization factor. With the Expectation Maximization algorithm, the probability $p(t_j|s_i)$ is finally determine from $p(t_j|s_i, A)$.

Given a sentence S, the probability of having word t in its translation is determined by all the words in S. In fact

$$p(t|S) = C_S \Sigma_i\, p(t|s_i)$$

where C_S is another normalization parameter related to the length of S.

In practice, the training of translation model only uses the one to one aligned sentences and ignores others because the 1-to-1 alignments are the most reliable. This is also why we will also examine the 1–to-1 alignments in our later experiments. IBM model I ignores syntactical and positional information of words. It cannot be used to deal with syntactic problems of natural language. However, the goal of our project is to train translation models for CLIR, which do not require syntactical correctness of translation. The most important aspects are the correct selection of translation words and an appropriate weighting. IBM model I is enough for these two aspects.

2.6 Summary

In this chapter, we overviewed the background and related studies of CLIR. We also introduced the general problems of CLIR and compared the similarities and differences between CLIR and traditional IR. We then introduced main approaches of CLIR translation, gave an introduction to methods of query translation, and analyzed their advantages and disadvantages. The rapid development of Web resources provides us with the possibility to construct parallel corpus with Web pages. We introduced the PTMiner system of Chen that has successfully constructed a Chinese-English parallel corpus of Web pages.

Sentence alignment is an important step before translation model training. In this Chapter we described several well-known sentence alignment algorithms, especially the length-based Gale-Church algorithm. The principle of IBM model I is also briefly described.

However, there is much noise in our original parallel corpus. This leads a poor sentence alignment performance and poor translation model. So improving the parallelism quality of the final training corpus is necessary and inevitable. In next chapter, we will describe in detail our corpus filtering approach and improved sentence alignment algorithm.

Chapter 3

Filtering Parallel Text Corpora

Using PTMiner, Chen created a bilingual parallel text corpus. But as we mentioned in last chapter, this corpus is noisy because the parallelism of Web pages varies in an unpredictable range. The poor parallelism of corpus is an important reason to the poor translation accuracy of translation model trained from this corpus and the poor performance of CLIR. In order to improve the final translation accuracy and CLIR effectiveness, we try to develop some approaches to reduce noise and improve the parallelism of the corpus. In this chapter, we will introduce the rationale and our approaches for filtering the raw corpus in order to obtain a higher translation accuracy and better CLIR performance.

3.1 Principles of Filtering

As we know, a pair of parallel text consists of many parallel sentences, so we could judge the truth of parallelism of a text pair by judging the parallelism of its sentences. For a manually constructed corpus, the text parallelism is ideally 100%, paragraph parallelism is 99.1% [GC91], and most sentences are usually matching exactly one to one as shown by the statistical results of Gale-Church listed in Table 2.1.

In our case, parallelism between texts is only 82% [Che00]. It should be pointed out that this evaluation done by Chen tolerates some non-strict parallelism, i.e., even if two texts are not strict translations of each other, if they are about the same topic, these texts are still judged parallel. In fact, as our evaluation reported in Chapter 4, there is a very high percentage of false sentence alignment from the raw corpus. This will raise serious problem to model training.

We manually investigated the matching and parallelism of some randomly selected pairs of our raw corpus. Not like law documents, online parallel pages don't require exact translation and sometimes they take one language as the basic or main language. Usually, the pages in the basic language are more detailed and fully covered in information, but the pages in other language sometimes only give some basic information. Moreover, the layout of parallel pages could be different. Generally, most false parallel text pairs show the following common phenomena:

- There often exist omissions of translation for blocks of paragraphs, the contents of two texts may differ very much. So one text may be much shorter than another, and the length ratio of such two texts could be very different from the average ratio.
- Even if there is not omission of entire blocks, there may be mismatching inside blocks because of sentence deletions and insertions between matching blocks. This means that many sentences match to nothing, i.e. the proportion of n-to-0 or 0-to-n alignment is quite high.
- The structures of two texts are different even though their contents are the same, for various reasons. In some paired texts, the same contents may be arranged in a different order.

These phenomena are the main reasons of noise in our parallel text corpus. As we can judge the parallelism of texts by the parallelism of sentences, we can determine text parallelism through sentence alignment. This is the key idea of our approach.

Usually, organized objects are structurally clearer and easier to be classified and characterized than unstructured ones. In our case, an aligned text pair is easier to judge for parallelism with some reference. Furthermore, some filtering principle requires check the proportion of specific matching model. So we cannot absolutely separate filtering and alignment procedures. But for clarity, we describe filtering principles and alignment algorithm separately in two sections. In the following parts, we propose the filtering principles.

3.1.1 Checking Length Ratio Difference

As we mentioned in Chapter 2, texts or sentences of any language can be viewed as consequence strings of characters. Characters are electronically stored in bytes in computer, no matter what kind of word format of different languages. Then texts can be measured by the quantity of characters, or character length. Moreover, statistical investigations found there is a relatively stable ratio between text lengths of two languages if they are translation of each other [GC91][WX95]. Our observations of the corpus also indicate that most true parallel pairs have a similar text length ratio in character. It means that there is a statistical standard value for length ratio between parallel texts. Thus, in reverse, we can judge the parallelism of two texts then filter the corpus by examining their character length ratio whether satisfies the standard value or not and adjusting the match probability of alignment.

The accuracy of length ratio judgement depends on the parallelism quality of the text pair, as well as the scale of investigated range. Because the average ratio is a statistical value, the range of statistical investigation is crucial. Some sentence pairs may not satisfy the average ratio just because they are too short, even if they're exactly parallel. For example, the result of [WX95] shows that the average length ratio in byte of English/Chinese texts is about 2 (1.98), but the following two examples of Table 3.1 and Table 3.2 show the influence of investigated range:

Language	***English***	***Chinese***
Parallel text	Hello, world!	世界，你好！
Total length	12 bytes[1]	12 bytes

Table 3.1 A short parallel pair example

Language	***English***	***Chinese***
Parallel text	Abstract In order to establish a basis for redesigning initial teacher preparation, the author examines two instances during his career when he was excited in his work as a teacher educator. He also analyzes external barriers to achieving excitement in teacher education, including not only state and national regulation of teacher education but also the low status of teacher education in the United States.	摘要 作者检视他过往热心地训练准教师时的两个事例，作为改革教师教育的根据。 作者分析导致美国的教师了无生气的外在原因，除了州政府和国家颁下的规定外，还有教师社会地位的低微。
Total length	345 bytes	172 bytes

Table 3.2 A long parallel pair example

Both the examples of Table 3.1 and Tale 3.2 are exact translations, but they are distinctly different in the length measured in bytes. The two texts of the short

[1] The average langth ratio 2 of English/Chinese texts is measured in byte.

example of Table 3.1 both are 12 bytes long, or the ratio is 1 that dissatisfies the standard value of about 2. On the other hand, those of the long example of Table 3.2 have 345 bytes and 172 bytes respectively, whose ratio 345/172 = 2.0058 that is very approximate to the standard ratio value. Obviously, if the investigating range is big enough, the ratio gets close to the standard value.

Our corpus is noisy, many nonparallel pairs are assumed parallel. Because there exist the standard length ratio between two parallel texts, we can judge the parallelism by comparing the length ratio at the text level. Moreover, texts can be divided into one or several structural blocks, depending on the contents and structure of the text. Usually two texts of a parallel pair have the same number structural blocks and are parallel at block level, or are translation of each other. In order to ensure the effectiveness of the filtering principle, we judge the parallelism of texts basing on the length ratio both of the entire texts and the structural blocks.

In detail, we try to set an acceptable fluctuating range of length ratio around the standard value of two languages. The parallel pair whose ratio falls in this range will be considered be truly parallel and a positive matching probability adjustment will be added to the matching score of its all sentence members during sentence alignment. Furthermore, if the length ratio of a certain structural block also satisfies this range, then another positive adjustment will also be granted to all sentence members of the block. In other words, if given the standard text length ratio ϕ of two languages and the deviation limit δ, then only those sentence pairs that belong to the text pairs or block pairs whose text length ratio $\mu \in [\phi - \delta, \phi + \delta]$ or block length ratio $\lambda \in [\phi - \delta, \phi + \delta]$ will be granted a positive adjustment φ or ψ respectively. The equation 2.2 then becomes as follows:

$$\begin{aligned} &\arg\max_A \Pr(A|T_1, T_2) \\ &\approx \arg\min_A \Sigma_A -\log(\Pr(S_1 \Leftrightarrow S_2 \,|\, l_1, l_2) + \varphi + \psi) \qquad (3.1) \end{aligned}$$

where $\varphi > 0$ if $\phi - \mu < \delta$ and $\psi > 0$ if $\phi - \lambda < \delta$. Concretely, for English and Chinese language, we set the standard length ratio value as English/Chinese $\phi = 2$, and tried several length ranges with different value of δ in our tests.

3.1.2 Examining Empty Alignment Proportion

A certain proportion of empty alignment is inevitable even for exactly parallel texts. But the proportion is very small for high quality parallel texts as shown in Table 2.1. For a noisy corpus, n–to-0 or 0–to-n alignment will occur more frequently. When a certain portion of a text pair cannot be aligned (or aligned to zero sentence), this would be a strong indication that the pair is not truly parallel, which is what we observed from our corpus. Therefore, we will use a threshold of the proportion of the empty alignment to filter out likely nonparallel texts.

3.1.3 Using Translation Words in Sentence Alignment

If two sentences are really parallel, they usually contain some known translations. These translations are those words that can be found in a bilingual dictionary or lexicon. In reverse, if we can find some known word translation pairs in possible parallel sentences, it is strong evidence that these sentences are parallel. The more we can find known translation in a pair of sentences, the more probable the two sentences are truly parallel. In fact, for someone who is not familiar with both languages, but knows some words, the appearance of these known words and their translations in two sentences is a valid criterion to judge if the sentences could be parallel. This idea was used by Wu to align the noisy head parts of his well translated corpus as introduced in Chapter 2. However, Wu's lexical cue set (Table 2.2) is small and domain specific, and cannot be used for our corpus. In our process, we exploit this same principle of considering known translation as alignment anchors. But we use a large size and full domain collection of lexical cues such as a medium or large bilingual dictionary. In detail, when we observe that there are some known translations in the corresponding sentences, a positive probability adjustment will be assigned to the sentence pair for every existing known translation. Then the correspondence between the two sentences is increased by a certain degree, which is $\sigma \times \rho$, where σ is the known translation weight, ρ is the proportion, which is defined as the proportion of known translation pairs comparing to the total word pairs in the sentence. This degree is then integrated as "lexical cue" into basic degree based on other principles. Then the alignment equation 2.2 changes to equation 3.2:

$$\begin{aligned} &\arg\min_A \Pr(A|T_1, T_2) \\ &\approx \arg\min_A \Sigma_A -\log(\Pr(S_1 \Leftrightarrow S_2 \,|\, l_1, l_2) + \sigma \times \rho) \qquad (3.2) \end{aligned}$$

Usually the weight coefficient σ of known translation is given to 1, which brings good result for high quality parallel corpus. However, this may be too weak to correct the mismatching of the noisy corpus by known translations as lexical anchors. So a larger or smaller known translation weight coefficient may be used, and we will conduct a series of tests later to find out the proper empirical value.

3.1.4 Combinations

The above filtering principles are mutually complementary. There may be different ways to combine them in a filtering process, for example, by a linear combination. Here, we will use a different strategy: we use them one after another, then find out the better one from different combinations. We will conduct a series of experiments in next Chapters to investigate the combinations.

3.2 Integration of Filtering Criteria with Sentence Alignment Algorithm

When we introduced the principles of filtering in last section, we mentioned the utilization of alignment results. Indeed, the filtering criteria will be implemented with the sentence alignment process. In this section, we will describe our integration of the filtering criteria in the alignment algorithm. Sentence alignment is to figure out the translation mapping relationship between sentences. Although English and Chinese are two completely different languages, text length is still the comparable common property of their texts. Naturally, our sentence alignment will take length-based algorithm as the starting point. Since there is not any morphological comparability between Chinese and English words, we take Gale-Church algorithm, which counts the character length, as the backbone of our sentence alignment algorithm.

However, Gale-Church algorithm is proposed over a parallel corpus with good quality of parallelism, which is rather different from ours. It calculates probabilities of matching sentence pairs only based on the sentence length, which is also not very reliable in our corpus. So some inevitable modifications have to be made. Indeed, besides the need to integrate the three filtering criteria into the algorithm, we still have to add some other alignment adjustments to make it adapt to our parallel corpus. Details are described in the following section.

HTML Tags

Gale-Church algorithm was tested on two manually constructed parallel corpora, whose paragraph parallelism even reaches 99.1%. Therefore, it was assumed that there are exactly equal paragraph numbers between two texts to be aligned. Such the algorithm reduced computation by first aligning the texts into parallel paragraphs, then aligning sentences within correspondingly paragraphs. However, the assumption that there is equal number of paragraphs in a pair of parallel texts is no longer true for our corpus where parallelism is poor. On the other hand, to get the best alignment, dynamic programming used in the alignment algorithm is important. Dynamic programming requires time quadratic in the length of the text aligned, therefore, it unpractical to align a pair of texts as a single unit. Such the text to be aligned is usually subdivided into smaller chunks to reduce the computation.

Therefore, we will realize this requirement by relaxing the dividing criteria in our modified Gale-Church algorithm.

Although most corresponding pages in our corpus do not match exactly in paragraph, they usually can be subdivided into the same number of semantic chunks or structural blocks, depending on the page's length. Structural blocks may be a single paragraph or several paragraphs, but they usually match each other well between parallel pages. On the other hand, all of our parallel pages are in HTML format. So the matching structural blocks of parallel pages are often delimited with the same HTML markups, such as <H1></H1>, <H2></H2>, <Table></Table>, etc. Therefore, in our alignment approach, we use predefined HTML tags as "cognates" or "delimiters" to identify the structural blocks, which functionally corresponding to paragraphs of original Gale-Church algorithm. Chen also used this approach in his project, but to help identify paragraphs [Che00].

Character Ratio

Character length ratio is used as a filtering criterion, but it is the statistical length ratio of the two texts or structural blocks aligned. For sentence alignment, Gale-Church algorithm also uses character concept at sentence level: character ratio parameter c, the expected character number generated in target language by per character in source language, which is set to 1 in their experiments over English, French and German. As mentioned previously, the character length ratio between parallel Chinese and English texts is English/Chinese = 1.98 [WX95]. So we set the English/Chinese character parameter $c = 2$, which is used to replace the original one in Gale-Church algorithm.

Known Translations

Using lexical cues or lexical information in sentence alignment is not a new attempt. But former studies either use language dependent literal features as [SFI92] or use a small size and domain specific collection of lexical cues as [Wu94]. As introduced in previous section, we will use known translations as lexical anchors to sentence alignment. Here, we just emphasize the influence of lexical cue collection. The size of the bilingual dictionary may have a significant impact on this method. So we will test the use of several dictionaries of different sizes in order to evaluate this impact in a later chapter.

Integration with the Alignment Algorithm

The above algorithm adjusting factors are merged into the basic Gale-Church algorithm, as well as the integration of filtering criteria. Here we summarize our sentence alignment algorithm as followings:

```
/* determine thresholds and parallel pairs */
set standard length ratio in byte of two languages $\phi$
set permitted length deviation $\delta$
let texts $T_i$ and $T_j$, $i, j = 1, ...m$ are two parallel texts
  basing on HTML tags, divide two paired texts into same structural blocks
    $B_i$ and $B_j$, $i, j = 1, ...n$

/* check length ratio  and determine adjuster */
scan through paired texts $T_1$ and $T_2$
  let $L_1$ and $L_2$ be the text length in character (byte) respectively
    if ( $L_1/L_2$ - $\phi$ ) $\leq \delta$, then
      set adjuster $\varphi > 0$
    else $\varphi = 0$
  for two paired structural blocks $B_1$ and $B_2$
    let $lb_1$ and $lb_2$ be their text length in character (byte) respectively
      if ( $lb_1/lb_2$ - $\phi$ ) $\leq \delta$, then
        set adjuster $\psi > 0$
      else $\psi = 0$

/* calculate matching probability among sentences */
for supposed parallel sentences $S_{i,\, i=1,\ldots,p}$ and $S_{j,\, j=1,\ldots,q}$ of $B_1$ and $B_2$
  let $S_1$ and $S_2$ are two matching sentences
    let $l_1$, $l_2$ be the text length in character of $S_1$, $S_2$ respectively
    let $v_{i,\, i=0,\ldots,n}$ and $w_{i,\, i=0,\ldots,n}$ be the known translation pairs in $S_1$ and $S_2$
    let $\sigma$ be the known translation weight coefficient
    let $\rho$ be the proportion of known translation
    let $\beta$ be the matching probability function only of $l_1$, $l_2$
    let $A$ be the alignment, then
      the matching probability of $S_1$, $S_2$ is
        $P( S_1 \Leftrightarrow S_2 \,|\, A) = P( S_1 \Leftrightarrow S_2 \,|\, \beta ( l_1, l_2) + \varphi + \psi + \sigma \times \rho)$
set alignment distance function between $S_1$, $S_2$ as
  $d = -\log (P( S_1 \Leftrightarrow S_2 \,|\, A)$

/* find out the minimum alignment distance among possible matches */
let $D(i, j)$ be the minimum distance of $S_i$, $S_j$,
  initialize $D(i, j) = 0$
then, calculate $D(i, j)$ by dynamic programming
  $D(i, j) = \min \{ D(i, j-1) + d(0, t_j, 0, 0); D(i-1, j) + d(s_i, 0, 0, 0);$
                 $D(i-1, j-1) + d(s_i, t_j, 0, 0); D(i-1, j-2) + d(s_i, t_j, 0, t_{j-1});$
                 $D(i-2, j-1) + d(s_i, t_j, s_{i-1}, 0); D(i-2, j-2) + d(s_i, t_j, s_{i-1}, t_{j-1}) \}$

/* conclusion */
then, the alignment $A$ between texts $T_i$ and $T_j$ is
  $P(A|T_i, T_j) \approx \arg\max_A \Pi_A P(S_i \Leftrightarrow S_j \,|\, S_i, S_j ) = \Sigma_A D(i. j)$
```

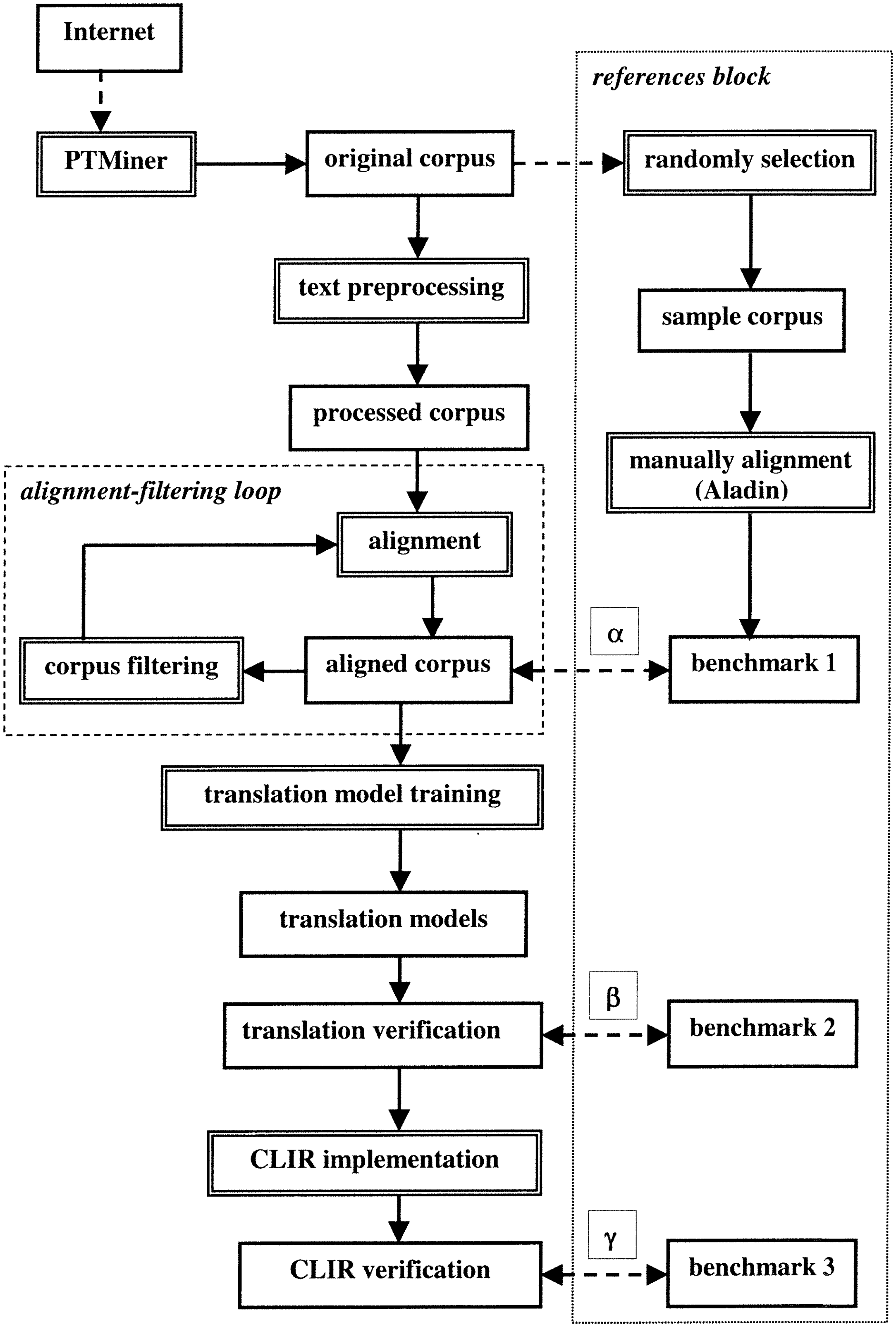

Figure 3.1 Global framework

3.3 Overview of Evaluation

Our study is conducted on the English-Chinese corpus of parallel Web pages created with PTMiner. The final goal is to train a better statistical translation model from a purified parallel text corpus, and to use it to obtain a better CLIR effectiveness. The question now is whether the filtering of likely nonparallel texts has a positive impact on the translation model and on CLIR performance. The answer is not obvious: while we filter out truly nonparallel texts, some parallel texts can also be removed. So the filtering process can result in a less noise but smaller corpus. The smaller size can affect the coverage of the model. Therefore, we will conduct a series of experiments to measure the impact of the filtering process.

Concretely, the detailed global system framework of our project is depicted as Figure 3.1. From the next chapter on, we will describe all our experiments and their results in details according to the framework of figure 3.1. From this figure, we can see that all processes over the parallel text corpus are listed in a sequential string in the left side, and the right side is the reference block. Reference block includes several manually build comparison benchmarks for sentence alignment, translation model accuracy and CLIR precision. The three association relationships α, β, γ shown in this figure representative the evaluation links between our experiment results and references, and their related experiments will be introduced in Chapter 4, 5 and 6 respectively.

Chapter 4

Experimentation I: Sentence Alignment and Corpus Filtering

In the last chapter, we introduced the methodology and rationale of corpus filtering procedure and the system framework, including sentence alignment, corpus filtering. From this chapter on, we will describe in detail the experiments of the approach. This chapter will describe the experiments on sentence alignment with respect to corpus filtering. This maps to the evaluation α of Figure 3.1, in which we are interested in the parallelism between texts, as well as the correctness of sentence alignment.

4.1 Reference Benchmark Setup

To evaluate how a filtering criterion performs, we need to set up a set of text pairs as references. For these text pairs, we manually examine their parallelism and their sentence alignment. We randomly picked up 484 pairs of texts from the original corpus of Web pages to form a sample corpus. Then we manually aligned all the sentences of the corpus. During the alignment, an alignment software with GUI developed by RALI group of University of Montreal is used. It can automatically make the first alignment, in which there may be errors. It also provides a user interface to correct the alignment errors. However, the software does not support Chinese. So we only use its user interface and the sentence alignment is completely manually aligned. To display Chinese character, we employed another software NJStar Communicator [NJS] to display Chinese during alignment.

The final result of this stage is a collection of parallel texts with sentence aligned, which consists of 484 text pairs or 38,798 sentence pairs. In this corpus, 10,875 sentence pairs or 28.03% of the total number are empty alignment pairs, i.e. of types n-to-0, or 0-to-n; 22,338 sentence pairs or 57.58% are 1–to-1 mapping. Basing on manual judgments, there are 396 pairs of texts, or 81.82% are considered parallel, which is approximate to Chen's results of 82% [Che00]. These are depicted in Table 4.1, in which we examine different types of match between sentences and between texts.

Matching element	***Matching type***	***Number of pairs***	***Proportion***
Sentence	***0-to-n or n-to-0 match***	10,875	28.03%
	1-to-1 match	22,338	57.58%
	Total	38,798	100%
Text	***Parallel pairs***	396	81.82%
	Total	484	100%

Table 4.1 Data of reference corpus

Notice that a more detailed analysis in this direction can eventually result in a new set of probabilities of different matches (1-to-1, 1-to-2, 2-to-2, etc.). These probabilities could be used to replace these used in Gale-Church algorithm (Table 2.1). However, this has not been integrated in our current implementation.

During the alignment of sample corpus, we noticed following facts:

- The empty matching proportion is much higher for online parallel pages than other manually constructed parallel corpora.

- There is a much higher proportion of deletion, insertion, reversion for the online parallel pages.
- Some parallel texts are only matching for their URL addresses and file names, but the contents are completely unparallel.
- Some Web parallel pages are in graph or other non-textual format. Although they match each other, they are not useful for our purpose.
- High quality parallel pairs usually are long and literal texts, while poor ones are short and diverse.

These phenomena are common for online parallel texts, and they are the main reason for higher noise of corpus consisted of online parallel texts and reason of low sentence alignment accuracy. And these observations are also the basis of our corpus-filtering principles described in Chapter 3.

4.2 Experiment Measurements

The task of corpus filtering is similar to IR: we want to keep all and only truly parallel text pairs. In practice, our filtering will remove nonparallel text pairs along with some parallel pairs. Therefore, the precision and recall [SM83] measures commonly used in IR are also appropriate for our evaluation. The two measures are defined as follows:

$$\text{precision} = \frac{\text{number of correct elements kept}}{\text{total number of elements kept}} \qquad (4.1)$$

$$\text{recall} = \frac{\text{number of correct elements kept}}{\text{total number of correct elements}} \qquad (4.2)$$

Usually, precision and recall are two contradictory requirements. When precision increases, recall decreases and verse vice. It is often difficult to judge which is better between high precision and high recall. So a combination measure is used to evaluate the global quality, called F-measure [Rij79]:

$$\text{F-measure} = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \tag{4.3}$$

F-measure is a harmonic average of precision and recall. We would expect that the higher F-measure value, the better IR performance (This hypothesis will be further verified in a later chapter). So the algorithm quality will be basically evaluated by F-measure in this chapter.

In our filtering and evaluation procedures, we are interested in several kinds of elements in text and sentence. Because of the corpus are assumed parallel at text level and some is noise, we need to check the parallelism at text level. On the other hand, the input to translation model training requires matched sentences, or 1-to-1 aligned sentences. So we also pay attention to the parallelism at sentence level, include that of 1-to-1 sentences. Moreover, the proportion of empty aligned sentence is a benchmark of text parallelism, so we consider non-empty sentence alignment as well. Therefore, for all experiments in this stage, we calculate the precision, recall and F-measurement at different elements include text alignment, sentence alignment, nonempty sentence alignment and 1-to-1 sentence alignment. All these comparing matrixes will reflect the quality of the filtered corpus and alignment.

4.3 Corpus Filtering and Alignment Experiments

As we described in the last chapter, corpus filtering and sentence alignment are dependent each other. The effect of filtering is evaluated on the result of sentence alignment. In the framework of Figure 3.1, they are combined in a loop, so their experiments and evaluation are conducted together. Because we hope to examine

Unit	Measure	Empty alignment proportion						
		No filter	≤ 50%	≤ 40%	≤ 30%	≤ 20%	≤ 10%	≤ 5%
Text	Precision	65.83 %	67.73 %	69.63 %	71.34 %	75.24 %	79.03 %	83.18 %
	Recall	100.00 %	97.57 %	94.15 %	89.87 %	86.45 %	84.74 %	82.49 %
	F.measure	79.39 %	79.96 %	80.05 %	79.54 %	80.46 %	81.79 %	82.83 %
Sentence	Precision	67.43 %	69.35 %	72.33 %	74.07 %	78.86 %	80.49 %	84.02 %
	Recall	100.00 %	97.63 %	94.51 %	90.31 %	87.05 %	85.21 %	83.04 %
	F.measure	80.55 %	81.09 %	81.95 %	81.39 %	82.75 %	82.78 %	83.53 %
Nonempty	Precision	68.66 %	71.96 %	73.29 %	75.36 %	79.21 %	82.15 %	84.55 %
	Recall	100.00 %	95.65 %	92.64 %	89.52 %	86.49 %	84.05 %	83.34 %
	F.measure	81.42 %	82.13 %	81.84 %	81.83 %	82.69 %	83.09 %	83.94 %
1-to-1	Precision	71.69 %	74.61 %	77.36 %	78.58 %	82.11 %	86.08 %	90.32 %
	Recall	100.00 %	96.01%	93.61 %	90.61 %	87.44 %	85.31 %	83.82 %
	F.measure	83.51 %	83.97 %	84.71 %	84.17 %	84.69 %	85.69 %	86.43 %

Table 4.2 influence of empty alignment proportion

each filtering criterion independently and different combinations of them, we arrange all experiments in two phases, which are described in following two subsections.

4.3.1 Individual Experiments

In the first phase of experiment procedure, we test every filtering criterion individually to observe the influence of each single factor.

Experiment 1: Using Empty Alignments Proportion

To examine the filtering effect of using only the criterion of empty alignment proportion, we directly use Gale-Church algorithm with the character generating parameter *c* (length ratio) set 2 to align the corpus. Then we examine the alignment results and remove those text pairs with a proportion of empty alignment higher than a predefined threshold. Generally, we considered the two texts are nonparallel if more than a half of their sentences were aligned to null in either side, so the lowest filtering threshold was naturally set as 50%. Furthermore, in order to find out the impact of the proportion, we also compared the alignment results at some stricter situations when the threshold was set at 40%, 30%, 20%, 10% and even 5%. The final alignment results are listed in Table 4.2. In this table, we show the alignment ratio between different units: texts and sentences. In addition, in the case of sentence alignment, we consider two particular cases: non-empty alignments and 1-to-1 alignments, that will have the most important impact on translation model training later. The numbers correspond to the values of these measures with different thresholds on empty alignment proportion. For example, the upper-left number (65.83%) is the precision of text alignment when no filtering on empty alignment proportion is applied.

By comparing the numbers in the same row or column, or between them, we can observe some conclusions from Table 4.2:

- The unfiltered precision of parallel text is only 65.83%, which is lower than Chen's 82% [Che00]. A possible reason is the criteria used are different. In our evaluation, the criterion seems to be stricter.
- For any evaluation in Table 4.2, when the empty alignment proportion threshold becomes stricter, the precision increases while the recall drops. It means the filtered corpus with fewer empty alignment pairs will lead to better alignment consequence at both text and sentence level.
- Stricter empty alignment proportion leads to a higher F-measure, or a better comprehensive performance. This may mean that the speed of precision increase is higher than that of recall drop as the threshold gets stricter.
- There is a similar increase/decrease for text, sentence, nonempty matching sentence and 1-to-1matching sentence.

Experiment 2: Using of Length Ratio

Measurement		***Adjustment value: $\varphi$and ψ***		
Unit	***Measure***	***0.8***	***1.0***	***1.5***
Text	***Precision***	71.46%	71.83%	72.19%
	Recall	92.78%	92.55%	92.03%
	F-measure	80.70%	80.88%	80.91%
Sentence	***Precision***	71.68%	72.52%	73.11%
	Recall	91.89%	91.23%	90.43%
	F-measure	80.54%	80.81%	80.85%

Table 4.3 adjustment value comparison

In the experiments using length ratio criterion to filter corpus, we use a modified Gale-Church algorithm to align the corpus. As we introduced in Chapter 3, we check the length ratio between the entire texts and matching structural blocks. And all sentence pairs within the text or block will be granted a positive probability adjustment φ or ψ if the length ratio deviation of the text or block is smaller than a predefined tolerated threshold δ. In order to determine the value of the adjustments, we conducted some comparison experiments with three values under the same algorithm while the tolerated deviation is set as $\delta = 0.5$. For simplification, we take the same value for φ and ψ. The results are listed in Table 4.3.

We can observe that the experiment results of different adjustment values are similar in Table 4.3, so we will take 1 as the adjustment value in the following experiments. In order to check the influences of length ratio under different tolerated deviation thresholds, we predefined several length difference deviations from the loosest 0.5 up to the strictest 0.05. The alignment results of different elements are listed in Table 4.4, from where we can compare the precision, recall and F-measure and conclude the followings:

- For all measuring units, stricter length constraint (smaller δ) brings better parallelism quality (higher precision) and narrow coverage (lower recall), but F-measure also gets better. These results suggest that a stricter length constraint leads to a better comprehensive result.

- The strictest δ value here 0.05 is an extreme threshold, but still brings a better result that exceeds our expectation. We believe the reason is that the real parallel Web pages are usually relatively longer, and then have a length ratio that is closer to the standard value. But it's seemly the smallest value we can accept, because the recall drops very low now.

Experiment 3: Using Known Translation as Alignment Cues

Before we investigate the influence on alignment of known translation with different weight, we have to decide how many known translations we should use, or what kind of known translation collection we should use. Is the small size lexicon of Table 2.2

Unit	*Measure*	*Length ratio difference deviation δ*						
		∞	*0.5*	*0.4*	*0.3*	*0.2*	*0.1*	*0.05*
Text	*Precision*	65.83%	71.83 %	75.47 %	79.36 %	82.09 %	83.47 %	85.70 %
	Recall	100.00 %	92.55 %	88.57 %	85.15 %	84.02 %	82.89 %	81.66 %
	F.measure	79.39 %	80.88 %	81.50 %	82.15 %	83.04 %	83.18 %	83.63 %
Sentence	*Precision*	67.43 %	72.52 %	76.35 %	80.16 %	83.79 %	85.18 %	86.50 %
	Recall	100.00 %	91.23 %	86.63 %	85.71 %	84.96 %	83.21 %	82.91 %
	F.measure	80.55 %	80.81 %	81.17 %	82.84 %	84.37 %	84.18 %	83.19 %
Nonempty	*Precision*	68.66 %	75.69 %	78.95 %	82.19 %	84.87 %	86.00 %	87.64 %
	Recall	100.00 %	93.83 %	90.56 %	87.48 %	86.11 %	85.35 %	84.54 %
	F.measure	81.42 %	83.79 %	84.36 %	84.75 %	85.49 %	85.67 %	86.06 %
1-to-1	*Precision*	71.69 %	78.86 %	82.12 %	85.54 %	88.34 %	90.23 %	91.02 %
	Recall	100.00 %	93.47 %	89.78 %	87.44 %	86.93 %	85.34 %	84.48 %
	F.measure	83.51 %	85.55 %	85.78 %	86.48 %	87.64 %	87.72 %	87.63 %

Table 4.4 influence of permitted length ratio range

[Wu94] or medium size lexicon used by Wu [WX94] effective enough? What is the influence of involving a larger lexicon on the alignment accuracy? In order to figure out the impact of lexicon size, we took a large size bilingual dictionary as the base dictionary. The base dictionary "Mandarin.fre" has 44,405 entries, which are sorted on usage frequency. Then we created several Chinese-English dictionaries with different size from the base dictionary. All these specific dictionaries are created from the base dictionary by selecting a certain portion of the most frequent words. For example, the dictionary with 500 words has the most frequently used 500 words of the dictionary with 1,000 words and so on. Then we conduct a series of alignment experiments individually with the specific dictionaries as the known translation collection under the same algorithm. In this series of experiments, the weight of known translation adjustment is set to 1. We then compare the results with the reference benchmark. The alignment accuracy is defined as the proportion of correctly aligned sentence pairs during the corpus alignment:

$$\text{Alignment accuracy} = \frac{\text{number of correct sentence alignment}}{\text{total number of sentence alignment}}$$

The comparison result is listed in Table 4.5:

Dictionary size	***Alignment accuracy***	***Improvement***
Without dictionary	72.31%	
500 words	72.95%.	0.89%
1,000 words	74.36%	2.84%
3,000 words	77.89%	7.72%
5,000 words	80.82%	11.77%
8,000 words	81.78%	13.10%
10,000 words	82.16%	13.62%

Table 4.5 influence of dictionary size

Unit	*Measure*	***Known translation weight coefficient***						
		1.0	***1.2***	***1.3***	***1.4***	***1.5***	***1.8***	***2.0***
Text	***Precision***	68.76 %	72.60 %	76.09 %	80.11 %	84.52 %	75.43 %	70.25 %
	Recall	94.89 %	90.70 %	88.81 %	86.66 %	84.06 %	89.21 %	94.38 %
	F.measure	79.74 %	80.65 %	81.96 %	83.26 %	84.29 %	81.74 %	80.55 %
Sentence	***Precision***	70.42 %	73.72 %	76.77 %	81.15 %	85.98 %	75.52 %	70.14 %
	Recall	95.31 %	91.04 %	88.49 %	86.84 %	84.84 %	90.16 %	96.40 %
	F.measure	81.00 %	81.47 %	82.21 %	83.90 %	85.68 %	82.19 %	81.20 %
Nonempty	***Precision***	72.05 %	74.69 %	78.62 %	82.59 %	86.64 %	77.03 %	72.07 %
	Recall	94.12 %	92.07 %	90.54 %	88.78 %	86.26 %	91.26 %	95.35 %
	F.measure	81.62 %	82.47 %	84.16 %	85.57 %	86.45 %	83.54 %	82.09 %
1-to-1	***Precision***	75.81 %	79.19 %	80.56 %	84.12 %	89.99 %	79.33 %	76.17 %
	Recall	95.09 %	91.62 %	90.63 %	88.28 %	87.04 %	91.35 %	94.27 %
	F.measure	84.36 %	84.95 %	85.30 %	86.15 %	88.49 %	84.91 %	84.26 %

Table 4.6 influence of weight coefficient of known translation

Form Table 4.5, we notice that the use of dictionary with suitable size brings higher improvement of alignment accuracy, and a larger dictionary brings better result. We also notice that a too small dictionary (500 words here) cannot make significant impact on alignment, but a medium size dictionary (here 3,000~5,000 words) brings significant improvements. Further improvements can be obtained by even larger dictionary, however, the improvement rate is decreased, as we can see in the case of 8,000 words and of 10,000 words here in the Table 4.5. Nevertheless, it seems clear that the larger the bilingual dictionary, the better the alignment result. In order to increase the accuracy of Chinese segmentation and alignment as most as possible, we used a combined dictionary in our following experiments and Chinese text segmentation. This dictionary is created by concatenating three dictionaries together, forming a dictionary with 310,430 entries.

When we used known translations as cognates, the default weight coefficient of known translation is often 1.0. But we are not sure this is the proper value that will bring encouraging result. Therefore, we try to increase the weight coefficient of known words, to see their influences over alignment. We tried several experiments with some values from 1 to 2. The experiment results are listed in Table 4.6 where is also arranged according to text, sentence, nonempty sentence, 1-to-1 sentence and nonempty 1-to-1 sentence alignment categories.

From Table 4.6, following facts can be observed:

- Higher weight coefficients for known word anchors yield better alignment performance than default weight coefficient.
- It does not always bring better result with continuously increasing coefficient. There is a peak value for final alignment result, at about 1.5. After this, a bigger coefficient leads to worse results.

4.3.2 Combination Experiments

In the last phase, we tested the effects of every individual filtering criterion and compared their alignment results. In this phase, we try to combine different filtering criteria together, two by two first and then all three together, to test their influences on each other and alignment results. The experiment results of last phase show that the impact of all individual criteria tends to increase or decrease at one direction as the criterion value increasing or decreasing. In order to simplify the experiment procedure, every combination experiment will only combine some typical points of individual experiment. Then we determine our final filtering algorithm as the combination with the best experiment result.

Experiment 4: Empty Alignment Proportion and Length Ratio

In this experiment, we test the combination of empty alignment proportion criterion and length ratio difference criterion.

From the results of experiment 1 and experiment 2, we know that for single criterion of empty alignment proportion, threshold of 50% provides the best recall while 5% brings the best precision; for single length ratio criterion, deviation $\delta = 0.5$ has best recall while $\delta = 0.05$ leads to best precision. On the other hand, the F-measure value increases while the precision improves for both criteria. So we only select the two extreme thresholds for each criterion to form four combinations, with which we test their influences on alignment. The experiment results are shown in Table 4.7.

In Table 4.7, we use the same measurements of precision, recall and F-measure as that of single criterion experiment. We can find that all results in Table 4.7 are better than that of its individual element. For example, the text F–measure of combination 50% + 0.5 reaches 81.99%, is better than 79.96% of 50% empty alignment alone and 80.88% of set derivation 0.5 alone, and so do other F-measures. This means

Unit	*Measure*	***Combination*** ***(empty alignment proportion + length ratio deviation δ)***			
		50% + 0.5	***50% + 0.05***	***5% + 0.5***	***5% + 0.05***
Text	***Precision***	73.52 %	85.25 %	82.63 %	88.34 %
	Recall	92.67 %	84.57 %	85.15 %	79.87 %
	F.measure	81.99 %	84.91 %	83.87 %	83.89 %
Sentence	***Precision***	73.68 %	86.29 %	83.33 %	88.07 %
	Recall	92.87 %	85.73 %	85.51 %	80.31 %
	F.measure	82.17 %	86.01 %	84.41 %	84.01 %
Nonempty	***Precision***	75.42 %	87.43 %	84.29 %	89.36 %
	Recall	94.73%	86.65 %	86.64 %	81.52 %
	F.measure	83.98 %	84.35 %	85.45 %	85.26 %
1-to-1	***Precision***	78.69 %	90.61 %	86.56 %	92.58 %
	Recall	81.61 %	75.29 %	75.31 %	71.30 %
	F.measure	80.12 %	82.24 %	80.54 %	80.59 %

Table 4.7 combination of empty proportion and length ratio

Unit	*Measure*	***Combination*** ***(empty alignment proportion + known translation weight)***			
		50% + 1.0	***50% + 1.5***	***5% + 1.0***	***5% + 1.5***
Text	***Precision***	70.17 %	73.73 %	85.01 %	88.34 %
	Recall	95.87 %	92.57 %	83.15 %	82.87 %
	F.measure	81.03 %	82.08 %	84.07 %	85.52 %
Sentence	***Precision***	71.43 %	73.35 %	85.37 %	89.07 %
	Recall	95.02 %	91.63 %	84.21 %	83.31 %
	F.measure	81.55 %	81.48 %	84.79 %	86.09 %
Nonempty	***Precision***	72.86 %	75.46 %	85.92 %	90.36 %
	Recall	95.34 %	90.65 %	84.64 %	83.52 %
	F.measure	82.60 %	82.36 %	85.28 %	86.81 %
1-to-1	***Precision***	76.69 %	79.61 %	90.36 %	93.58 %
	Recall	83.61 %	80.29 %	75.31 %	73.43 %
	F.measure	80.00 %	79.95 %	82.15 %	82.29 %

Table 4.8 combination of empty proportion and known word weight

combination of criteria will improve filtering and alignment. Moreover, we notice that combination of stricter criteria (small empty alignment proportion threshold and length ratio deviation) lead to a higher precision, lower recall as well as a better F-measure. For example, the precision of 1-to-1 alignment reaches 92.58% while the recall falls to 80.61%, but the F-measure is still higher at 96.87%. So we can conclude that stricter combination of these two criteria is the better choice for corpus filtering and alignment.

Experiment 5: Empty Alignment Proportion and Known Translation Weight

In this experiment, we combine empty alignment proportion with different known translation weight coefficient. From experiment 3, we know that there is a best value for known translation weight coefficient, and it is about 1.5. On the other hand, for most cases, the weight value of 1.0 is used. Therefore, we only combine and test these two weight values with two specific empty proportion thresholds 50% and 5% in this experiment. The experiment results are listed in Table 4.8.

Table 4.8 tells us that: first, combination of these two criteria brings better filtering and alignment results than its individual counterparts respectively; second, the combination of stricter empty alignment proportion (5%) and the best known translation weight value (1.5) will brings better precision and F-measure, even though the recall is worse.

Experiment 6: Length Ratio and Known Translation Weight

The final experiment of combination of two criteria was taken between length ratio difference criterion and known translation weight coefficient. In the experiment, we combine two sets of criterion values, $\delta = 0.5$ and $\delta = 0.05$ for length ratio difference, coefficient 1.0 and 1.5 for known translation weight. The experiment results are illustrated in Table 4.9.

Similar to former combinations, we can obtain these conclusions from Table 4.9: first, the combination of criteria brings better results to filtering and alignment than individual criterion; second, the combination of stricter or better element value leads to a better result.

Experiment 7: Combination of All Criteria

From the combination experiments between two criteria, we notice these common conclusions: first, combinations bring better results than individual criterion; second, combination of stricter criteria usually leads to better F-measure as well as higher precision. Therefore, we can generally conclude that criterion combination is preferable for corpus filtering and alignment.

As the second step of combination experiment phase, after having evaluated combinations of every two criteria, we try to combine all three criteria together. The results of experiment 4, 5 and 6 show that the change trend of those combination experiments strictly increases. Therefore, for the same simplification reason, we only experiment some combinations points of specific values of the first step: the highest precision and highest recall points: empty alignment proportion 50% and 5%, length ratio difference 0.5 and 0.05, known translation weight 1.0 and 1.5. The results are listed in Table 4.10.

From the results listed in Table 4.10, we can notice that:

- To filtering with combination of three criteria is better than that of combination between any two criteria.
- The combinations of stricter filtering criteria are better than loose ones if we evaluate by F-measure.

Unit	*Measure*	*Combination* *(length ratio deviation δ + known translation weight)*			
		0.5 + 1.0	***0.05 + 1.0***	***0.5 + 1.5***	***0.05 + 1.5***
Text	***Precision***	72.64 %	84.73 %	76.63 %	87.34 %
	Recall	91.86 %	84.67 %	87.15 %	83.25 %
	F.measure	81.13 %	84.70 %	81.55 %	85.25 %
Sentence	***Precision***	73.43 %	85.35 %	77.13 %	88.07 %
	Recall	90.54 %	84.63 %	87.51 %	83.31 %
	F.measure	81.09 %	84.99 %	81.99 %	85.62 %
Nonempty	***Precision***	75.96 %	86.57 %	78.92 %	88.76 %
	Recall	92.65 %	85.65 %	87.64 %	84.52 %
	F.measure	83.48 %	86.11 %	83.05 %	86.59 %
1-to-1	***Precision***	79.25 %	90.61 %	84.36 %	93.58 %
	Recall	81.61 %	75.19 %	78.39 %	74.30 %
	F.measure	80.41 %	82.18 %	81.27 %	82.83 %

Table 4.9 combination of length ratio and known word weight

Unit	*Measure*	***Combination*** ***(empty alignment proportion + length ratio deviation δ + known translation weight)***							
		50% + 0.5 + 1.0	***50% + 0.5 + 1.5***	***50% + 0.05 + 1.0***	***50% + 0.05 + 1.5***	***5% + 0.5 + 1.0***	***5% + 0.5 +1.5***	***5% + 0.05 + 1.0***	***5% + 0.05 + 1.5***
Text	***Precision***	74.36 %	77.83 %	86.24 %	87.36 %	80.09 %	87.47 %	86.70 %	92.01 %
	Recall	94.57 %	90.55 %	84.57 %	83.15 %	88.02 %	84.89 %	83.66 %	81.34 %
	F.measure	83.26 %	83.71 %	85.40 %	85.20 %	83.87 %	86.16 %	85.15 %	86.35 %
Sentence	***Precision***	74.83 %	78.52 %	86.35 %	88.16 %	80.79 %	87.18 %	87.92 %	92.56 %
	Recall	94.63 %	90.23 %	84.63 %	83.71%	88.96 %	85.21 %	83.91 %	82.23 %
	F.measure	83.57 %	83.97 %	85.48 %	85.88 %	84.68 %	86.18 %	85.87 %	87.09 %
Nonempty	***Precision***	75.96 %	78.69 %	87.95 %	89.19%	82.57 %	88.93 %	88.64 %	93.45 %
	Recall	94.65 %	90.83 %	85.56 %	85.21 %	89.31 %	86.35 %	84.53 %	83.57 %
	F.measure	84.28 %	84.33 %	86.74 %	87.15 %	85.81 %	87.62 %	86.54 %	88.23 %
1-to-1	***Precision***	79.61 %	83.86 %	92.12 %	94.54 %	85.34 %	93.23 %	93.02 %	97.19 %
	Recall	82.29 %	78.92 %	75.42 %	74.27 %	79.85 %	75.50 %	73.51 %	72.48 %
	F.measure	80.93 %	81.32 %	82.94 %	83.19 %	82.50 %	83.43 %	82.12 %	83.04 %

Table 4.10 Combination of three criteria

From above analyses, we can conclude that: first, filtered corpus using single criteria leads to a better alignment result; second, using combination approaches brings a better result than single one; third, the combination of all three criteria will be better than the combinations of two. Moreover, the combination of stricter individual criterion is preferable for our corpus of parallel online pages.

4.4 Summary

In this chapter, we described the experiments of filtering and alignment with different filtering criteria and combinations. The results proved our assumptions about the filtering principles and their influence on alignment result. Every filtering principle brings improvement to alignment precision. We get a higher alignment precision as well as higher F-measure, when we use stricter criteria on empty alignment proportion and length ratio. For the known translation factor, the best value seems to be 1.5. When it is too small or too large, the value of F-measure generally decreases. In general, the alignment results of combination are better than that of individual experiments.

The improvement of alignment performance is not our final goal. What is the impact over the quality of trained translation model and CLIR? In next chapter, we will examine this question.

Chapter 5

Experimentation II: Translation Model Training and Evaluation

In last chapter, we described and compared the corpus filtering and sentence alignment experiments, individually and with combined criteria. We found that all filtering criteria lead to a better alignment result, and combination of stricter criteria brought better results measured with F-measure. On the other hand, the precision and recall move in contrary directions, so we are still not sure that the combination of stricter criteria, which has high precision but low recall, will lead to a better translation model. Therefore, we conducted a series of experiments to compare the translation accuracy of translation models trained with different filtering criterion combinations. This chapter will give out some detailed description to these experiments, and the analyses on their results.

5.1 Translation Model Training and Evaluation

After having filtered and aligned the corpus, we got a collection of aligned sentences. Among them, some are exact one to one matched, which are the basic materials we will use to train our translation models. As mentioned in Chapter 2, we will train our translation models according to IBM model I. This means we will statistically calculate the probability $p(t|s)$ of having the word t in the translation of source sentence containing word *s*, while all words in the target sentence are given equal possibility weight of translation of the word in the source sentence. The Rali group [http://www-rali.iro.umontreal.ca/] has developed tools to train translation model, which we will use do our training. The result of training process is two translation models in two directions, English-Chinese and Chinese-English.

To evaluate translation accuracy of trained translation models, we first set up an evaluation platform. The simple and direct method of check translation accuracy is to check the translation of specific words. Chen created two small sets of 200 English words and 200 Chinese words, randomly selected from the corpora, and used them to test his two translation models [Che00]. For the convenience of comparing with Chen's results, we take the same sets of evaluation words in our evaluation.

For each source word, the translation model gives a list of most probable translations as well as their probabilities. In order to simplify the comparison, we only evaluate the accuracy of the most probable translation of each word as Chen did. So a translation is considered correct if:

- the most probable translation (the first word) is correct if the translation could be one word, or
- the first translations together form a correct translation if the source word should be translated into a group of words.

We use word group rather than phrase or idiom for the second condition means we ignore the order of word in the group. The reason is first that for query translation of CLIR we do not pay much attention to the order but the elements themselves, because CLIR only cares whether there exist those words in both query and documents or not. On the other hand, sometimes it is difficult for some words of source language to be translated into a fixed phrase or idiom in target language.

According to Chen's experiment results, the utilization of stop-list of target language is much helpful for translation model training. A stop-list is a set of the most frequently grammatical words, which are helpless for information retrieval, so we should remove them from the training source. Those words are not of interest for IR since they are context meaningless. Moreover, these words exist in most alignment pairs, they are easily taken as translation of many words since their high appearance frequency, then make the statistical model conclude in wrong translation. Therefore, we also used stop-list of target language in our training procedures.

Usually, human judges conducted the evaluation of translation accuracy. In different project, there may exist some tiny subjective differences of judgement since the human judge may be different, but it's tolerable. For example, in Chen's project, the accuracy of Chinese-English translation model is evaluated at 77%, and for English-Chinese is 81.5%, while in our own evaluation of the same translation models, they are 77% and 80.5% respectively. We can see that the difference due to subjective judgment is quite small.

5.2 Experiments and Result Comparison

The alignment results in the last chapter prove that the precision at individual and combination experiments both increase when the empty alignment proportion and length ratio difference criteria get stricter, and it also increase before the known translation weight criterion reaches the best value. And the results of combination experiments are better than that of individual experiment. Therefore, for the experiments of translation model training, we just tested several different combinations with the loosest and strictest or best thresholds. The goal of these tests is to see whether it is better to have a high precision low recall corpus, or the reverse, and whether the F-measure also reflects the quality of the resulting translation models. The combination filtering factors are shown as following:

	Filtering criteria (empty proportion + length ratio difference + known translation weight)								
	No filtering	***50% + 0.5 + 1.0***	***50% + 0.5 + 1.5***	***50% + 0.05 + 1.0***	***50% + 0.05 + 1.5***	***5% + 0.5 + 1.0***	***5% + 0.5 + 1.5***	***5% + 0.05 + 1.0***	***5% + 0.05 + 1.5***
Correct	161	172	175	176	180	173	178	177	183
Correct rate	80.50%	86.00%	87.50%	88.00%	90.00%	86.50%	89.00%	88.50%	91.50%
Improvement		6.83%	8.70%	9.32%	11.80%	7.45%	10.56%	9.94%	13.05%
No. of tokens in English	76970	65543	63156	56325	54151	61132	58736	52025	49512

Table 5.1 Accuracy of English-Chinese translation

	Filtering criteria (empty proportion + length ratio difference + known translation weight)								
	No filtering	***50% + 0.5 + 1.0***	***50% + 0.5 + 1.5***	***50% + 0.05 + 1.0***	***50% + 0.05 + 1.5***	***5% + 0.5 + 1.0***	***5% + 0.5 +1.5***	***5% + 0.05 + 1.0***	***5% + 0.05 + 1.5***
Correct	154	163	165	167	171	165	169	168	173
Correct rate	77.00%	81.50%	82.50%	83.50%	85.50%	82.50%	84.50%	84.00%	86.50%
Improvement		5.84%	7.14%	8.44%	11.04%	7.14%	9.74%	9.09%	12.33%
No. of tokens in Chinese	53528	46530	44875	41083	39612	42131	41802	37731	35486

Table 5.2 Accuracy of Chinese-English translation

- Empty alignment proportion: no limit, 50%, 5%;
- Text length ratio difference in characters: no limit, 0.5, 0.05;
- Known translations weight coefficient: 0, 1, 1.5.

Therefore, nine different combinations were created, nine experiments were conducted, and nine couples of translation models were trained for both English-Chinese and Chinese-English. The results of English-Chinese translation accuracy upon the selected words is compared in Table 5.1, and that of Chinese-English translation over the specific evaluation platform is compared in Table 5.2. In both tables, row "no. of tokens" means the number of words in source language included by trained translation model.

Closely observing Table 5.1 and 5.2, we can see the following facts:

- The translation accuracy of original model of non-filtered corpus, is 80.50% for English-Chinese and 77.00% for Chinese-English, is approximately the same to Chen's evaluation [Che00]. It means the results of two projects are comparable.
- All translation accuracy results of filtered corpus have certain improvements over that of using original corpus. The improvement range for English-Chinese translation varies from 5.83% to 13.05%, for Chinese-English translation from 5.84% to 12.33%. This means a more correct translation model can be trained upon a filtered corpus than the raw one.
- For both translation models, the strictest combination (5% empty proportion, 0.05 length ratio and 1.5 known translation weight) brings the best accuracy.
- Translation accuracy improves while the filtering criteria get stricter, or it is consistent with the F-measure of sentence alignment. This means that a higher precision is preferable than a higher recall when filter the original corpus. We believe that the reason is due to the reasonably large size of our original corpus. Even a number of parallel texts are eliminated, the remaining texts pairs are still enough for training a better model. And we also believe that this is a common

property of all corpora consisted of downloaded online parallel texts whose resources are theoretically unlimited on the Web.

- When the filtering criteria become stricter, the number of tokens decreases, meaning the coverage of the trained translation model becomes smaller. However, 200 test words are covered by all models. In the next chapter, we will see a stricter model will encounter a few more unknown words in query translation. However, the global impact on CLIR effectiveness is still positive.

As an intuitive example, we compare the first translations and their probabilities of 25 randomly selected words in Tables 5.3 and 5.4. These tables only show the following three groups of translation results:

- Original: non-filtering.
- Filtering I: combination of the loosest filtering criteria (50% empty alignment + 0.5 length ratio difference + 1.0 known word weight).
- Filtering II: combination of the strictest filtering criteria (5% empty alignment + 0.05 length ratio difference + 1.5 known word weight).

In Table 5.3 and Table 5.4, we used T and F to represent correct and wrong translation respectively. Together with each translation word, its probability is also given. We note the following two facts from Table 5.3 and Table 5.4:

- First, the translation results of Filter I and Filter II are usually more correct than that of without filtering, this is consistent with the general conclusion derived from Table 5.1 and Table 5.2.
- Second, even if the evaluation is "T", or the translation is the same in different models, the probability of translation of filtered models is higher than that of without filtering. And that of Filter II is even better than that of Filter I. This means that we can get better translation model by filtering the corpus.

Words	***Without filtering***			***Filter I (50% + 1.5~2.5 + 1.0)***			***Filter II (5% + 1.95~2.05 + 1.5)***		
办事处	T	office	0.395586	T	office	0.416884	T	office	0.435771
保护	T	protection	0.353903	T	protection	0.383481	T	protection	0.421553
报告	T	report	0.395991	T	report	0.409475	T	report	0.424695
备	T	prepare	0.240706	T	prepare	0.264254	T	prepare	0.284341
本地	T	local	0.447433	T	local	0.472830	T	local	0.501273
标准	T	standard	0.463190	T	standard	0.484363	T	standard	0.497283
补校	F	adult	0.050530	F	supplementary	0.052451	F	supplement	0.050794
不足	T	inadequate	0.116634	T	insufficient	0.120776	T	insufficient	0.151587
部分	T	part	0.380860	T	part	0.424638	T	part	0.438723
财经	T	financial	0.181867	T	financial	0.201541	T	financial	0.234781
参观	T	visit	0.355927	T	visit	0.373153	T	visit	0.394701
草案	T	bill	0.475395	T	bill	0.486137	T	bill	0.507388
车辆	T	vehicle	0.472942	T	vehicle	0.503154	T	vehicle	0.521438
储蓄	T	saving	0.184575	T	saving	0.204513	T	saving	0.234621
处理	T	handle	0.135370	T	deal	0.136146	T	deal_with	0.136405
传真	T	fax	0.416355	T	fax	0.434512	T	fax	0.456712
次序	T	order	0.174050	T	order	0.187341	T	order	0.205431
措施	T	measure	0.448414	T	measure	0.468354	T	measure	0.473158
达到	T	achieve	0.235742	T	achieve	0.264283	T	achieve	0.284313
当局	T	administration	0.492832	T	administration	0.521476	T	administration	0.547961
登记	T	registration	0.292372	T	registration	0.304512	T	registration	0.326473
电子	T	electronic	0.354305	T	electronic	0.373485	T	electronic	0.403943
调	T	adjust	0.089118	T	adjust	0.112450	T	adjust	0.154326
定	T	determine	0.201319	T	determine	0.221541	T	determine	0.237814
动态	T	dynamic	0.105815	T	dynamic	0.132022	T	dynamic	0.151746

Table5.3 Chinese - English translation result

Words	***Without filtering***			***Filter I (50% + 1.5~2.5 + 1.0)***			***Filter II (5% + 1.95~2.05 + 1.5)***		
a.m.	T	上午	0.204787	T	上午	0.212679	T	上午	0.223468
access	F	公开	0.095658	F	查阅	0.108218	T	通道	0.105677
adaptation	T	适应	0.185545	T	适应	0.188986	T	适应	0.201543
add	T	补充	0.362725	T	补充	0.383652	T	补充	0.412579
adopt	T	采用	0.271118	T	采用	0.301254	T	采用	0.354878
agent	T	代理人	0.209154	T	代理人	0.215465	T	代理人	0.245357
agree	T	同意	0.403120	T	同意	0.424763	T	同意	0.434561
airline	T	航空公司	0.372547	T	航空公司	0.402674	T	航空公司	0.436489
amendment	T	修订	0.384250	T	修订	0.394768	T	修订	0.414582
appliance	T	用具	0.159941	T	用具	0.173185	T	用具	0.189789
apply	T	适用	0.243746	T	申请	0.201262	T	申请	0.236710
attendance	T	列席	0.178672	T	出席	0.164873	T	出席	0.189734
auditor	F	审核	0.180631	F	核	0.176895	F	审计	0.193061
average	T	平均	0.486271	T	平均	0.497862	T	平均	0.510476
base_on	F	计算	0.130322	F	得出	0.140083	T	基於	0.107234
block	F	大厦	0.178680	F	大楼	0.159845	F	楼	0.160259
bottom	T	最低	0.127479	T	最低	0.156713	T	最低	0.174284
break_law	F	冒险	0.048044	T	犯法	0.065743	T	犯法	0.091032
breath	T	呼气	0.371555	T	呼气	0.401273	T	呼气	0.342671
briefing	T	简报	0.255423	T	简报	0.289753	T	简报	0.315727
building	T	建筑物	0.105820	T	楼宇	0.101225	T	建筑物	0.123705
business	T	业务	0.203324	T	业务	0.209964	T	商业	0.190386
carry	F	工程	0.113120	F	通过	0.106180	T	执行	0.975835
category	T	类别	0.115763	T	类别	0.132587	T	类别	0.151731
census	F	统计	0.173830	T	普查	0.140945	T	人口普查	0.153034

Table 5.4 English - Chinese translation result

5.3 Summary

In this chapter, we described the experiments of translation model training and analyzed their results. The accuracy of translation models under corpus filtering also shows changes consistent with the F-measure of the sentence alignment. It turns out that a stricter filtering criterion on the proportion of empty sentence alignment and on the length ratio leads to a better translation model. The best translation factor is still at 1.5. Again, the combinations of good filtering criteria lead to better translation models. The next question is how will be its impact over CLIR? Next chapter will give details on this question.

Chapter 6

Experimentation III: CLIR

In this chapter, as the final but necessary verification stage, we will test our translation models trained from filtered parallel corpus, in CLIR. The experiments will be carried out on both English-Chinese and Chinese-English directions, all experiment will be tested on TREC (Text REtrieval Conference) collections using the SMART information retrieval system [Buc85]. The results will be compared to those of former researches.

6.1 Chinese-English CLIR

In this section, we describe the Chinese-English CLIR experiments where queries are in Chinese and Documents are in English.

6.1.1 The Collection and Retrieval Tool

For Chinese-English CLIR experiments, Chen used the English (AP) collection of TREC 6 as the test platform [Che00]. We use the same collection in our experiments. The statistical information of topics, or queries in natural language statements including titles and descriptions, and documents is shown in Table 6.1. The original queries have been manually translated from English to into Chinese. We consider these Chinese queries as our original queries, and translate them to English with our translation models. The SMART information retrieval system developed by Cornell University is an efficient and classic IR tool. We used SMART as our experiment tool, as many researchers did. Here are some notes about the experiments:

- Only 21 topics in TREC 6 have relevant documents. So the results depended on the 21 topics;
- All English words in both queries and documents are modified into their citation form;
- All documents are indexed with *ltn* weighting scheme, which is a *tf*idf* weighing scheme often used in IR, of SMART system.
- The original queries are submitted to translation model, which returns a set of translated words and their probabilities. These words and probabilities are then sent to IR tool as the translated queries.
- The effectiveness of a retrieval method is measured in terms of "average precision" [SM83]. The average precision is the average of precision at the following 11 points of recall: 0.0, 0.1, …, 1.0. In addition, we also indicate the percentage of the CLIR effectiveness with respect to that of the monolingual retrieval. The CLIR effectiveness is obtained using the translated queries, whereas the monolingual effectiveness is obtained with the manually translated queries that are provided in the test corpora.

Topic		***Document***		
Number	***Average word number/topic***	***Number***	***Average word number /document***	***Total size***
25	88.4	79,919	468.7	237 MB

Table 6.1 collection of C-E IR

6.1.2 CLIR Experiment with Translation Model

There are two approaches to use translation model to translate the queries: translate word by word or translate by query. In the first approach, we translate the query word by word; in the second approach, we translate the query as a whole. In order to investigate the effect and difference, we compared the CLIR results of two approaches with the translation model trained over a fixed filtering combination of 5% empty alignment proportion, 0.05 length ratio difference and 1.5 known translation weight. For translating by word, we took the first three translations of each word into the translated query. On the other hand, to translating by query, the translation is a series of words that are the most probable translations of the source sentence. Thus it is necessary for us to set the number we take from the output translation words of translation model. Here we import a concept called *length factor* C_{leng}, which is the ratio of the lengths of target query and source query. This means, for a query with N words, we take $C_{leng}N$ words for its translation. According to Chen's results [Che00], for Chinese-English CLIR, we take the best $C_{leng} = 3$. Furthermore, also based on Chen's results, we utilized the word weight given by translation model to each word during translation.

For the 21 English queries of TREC 6, the average precision of monolingual IR is 38.61%. The experiment results are shown in Table 6.2.

Approach	***Ave. Precision***	***Comparison to mono-IR***
Translate by word	20.49%	53.07%
Translate by query	20.63%	53.43%

Table 6.2 Comparison of translation approaches

From table 6.2, we noticed that the effects of both approaches are very similar, so we adopted the query translation approach in the rest experiments. In the followings, we examined the influence of different filtering criterion combination on CLIR precision. Because the alignment precision and translation accuracy of translation model both had positive reaction to the increasing strictness of filtering criteria, we just selected some specific test points, which had approximately equal interval of translation model accuracy as shown in Table 6.3, to conduct the experiments.

The results of Table 6.3 show us that the average IR precision strictly increases when the translation accuracy increase. Or the translation model with highest accuracy (here 91.5%) leads to the highest IR precision (here 20.63%). Comparing to Chen's result obtained from non-filtered corpus [Che00], which has 16.54% average IR precision or is 42.80% of mono-IR, the CLIR results with tools from filtered corpus in table 6.3 have made significant progress (highest 53.43% of mono-IR).

Filtering combination (empty + length deviation + word weight)	***TM accuracy***	***Average Precision***	***Comparison to mono-IR***	***No. of unknown words***
No filtering ([Che00])	77%	16.54%	42.84%	0
50% + 0.5 + 1.0	86%	18.98%	49.16%	0
50% + 0.05 + 1.0	88%	19.52%	50.56%	0
5% + 0.05 + 1.0	90%	20.09%	52.03%	0
5% + 0.05 + 1.5	91.5%	**20.63%**	**53.43%**	0

Table 6.3 CLIR results with translation model

From Table 6.3, we also notice there is no unknown word encountered in the translation of the queries. This is because all the words used in the queries are common words. Only one proper noun is included: Waldheim. This proper noun is covered by the parallel corpus, and is translated correctly. So the deduction in the number of words covered by the model after the filtering does not have any negative impact on these queries.

6.1.3 CLIR Experiments with Dictionary Combined

As Chen found that the enforcement of translation tools by combining translation model with dictionary would improve the CLIR precision. What will happen when we combine our translation model with some dictionary? We tried it in this way: for each query, we took its translation of both translation model and an extra dictionary; discarded the probabilities given by translation model. Then, we gave different weight coefficients to the translations produced by translation model and dictionary. In this experiment, we used a large size Chinese-English dictionary, which has 128,366 words.

Combination ratio	*Ave. precision*	*Comparison to monoIR*
TM:dictionary = 2:1	27.45%	71.10%
TM:dictionary = 1:1	28.11%	**72.81%**
TM:dictionary = 1:2	27.89%	72.24%

Table 6.4 CLIR with different combination ratio

First, we need to determine the combination ratio between translation model and dictionary. We had several experiments of using different combination ratio between the translation model and dictionary as Table 6.4. The experiments are conducted with the same translation model, which is trained with the same filtering criterion combination: 5% empty alignment proportion, 0.05 length ratio difference and 1.5 known translation weight. Based on the results of table 6.4, we set the combination ratio of translation model and dictionary at the best ratio of 1:1. Then we investigated the effects of different filtering criteria. Similarly to the previous experiments, we only test for some specific points, which have equal translation accuracy interval of translation model. The CLIR experiment results are shown in Table 6.5, in which the best precision reaches 28.11%, or 72.81% of mono-IR, and the number of unknown words does mot change.

Filtering combination (empty + length deviation+ word)	*TM accuracy*	*Average Precision*	*Comparison to mono-IR*	*No. of unknown words*
No filtering ([Che00])	77%	25.83%	66.90%	0
50% + 0.5 + 1.0	86%	26.89%	69.64%	0
50% + 0.05 + 1.0	88%	27.36%	70.86%	0
5% + 0.05 + 1.0	90%	27.65%	71.61%	0
5% + 0.05 + 1.5	91.5%	**28.11%**	**72.81%**	0

Table 6.5 CLIR results combined translation model and dictionary

In comparing and analyzing the results from Table 6.2 to Table 6.5 together, we can notice the following points:

- For both query translation approaches, i.e. only using translation model or using extra dictionary and translation model together, the average CLIR precision of those translation models trained from filtered corpus is improved with respect to that of the translation model trained from non-filtered corpus. This means that filtering training corpus will finally benefit CLIR.
- The average precision will further improve when the translation model is trained from a more strictly filtered corpus. The best results are obtained under the strictest filtering combination for both approaches. This means that the changing tendency of CLIR effectiveness is consistent with that of alignment quality (F-measure) and translation model accuracy.
- The expansion of translation tool by combining extra dictionary brings significant improvements to CLIR precision.

6.2 English-Chinese CLIR

In this section, we describe English-Chinese CLIR experiments.

6.2.1 The Collection and Retrieval Tool

For English-Chinese direction, we used the Chinese collection of TREC 5 and TREC 6, which come from two major news media of mainland China: People's Daily and Xinhua News Agency. There are total 54 topics given in both English and Chinese, 28 in TREC5 and 26 in TREC 6 as shown in Table 6.6. We also used SMART system as the IR tool. All documents are indexed in *ltc* weighting scheme, all translated queries with the probabilities given by the translation model are indexed by *mtc*. For the 54 queries, the average precision of monolingual IR is 39.76%.

Topic		***Document***		
Number	***Average word number/topic***	***Number***	***Average word number /document***	***Total size***
54	103.27	164,811	549	170 MB

Table 6.6 collection of E-C IR

6.2.2 CLIR Experiments with Translation Model

First, all words in stoplist are removed and all words are converted into their citation forms, Then we used the translation model to translate 54 queries into Chinese. As in the Chinese-English experiments, we tested the two translation approaches as illustrated in Table 6.7, while the filtering criterion combination is fixed at 5% empty alignment proportion, 0.05 length ratio difference and 1.5 known translation weight. As what we did for Chinese-English direction, we used length factor, and word weight, But this time we set the length factor $C_{leng} = 2$, the value that Chen proved to be the best [Che00]. Table 6.7 shows that two approaches have similar results, so the remaining experiments were conducted with translating by query.

Approach	***Ave. precision***	***Comparison to mono-IR***
Translate by word	19.96%	50.20%
Translate by query	20.13%	50.63%

Table 6.7 Translation approach comparison

Filtering combination (empty +length difference + word)	***TM Accuracy***	***Average precision***	***Comparison to mono-IR***	***No. of unknown words***
No filtering ([Che00])	81.50% [(1)]	15.91%	40.02%	7
50% + 0.5 + 1.0	81.50%	18.43%	47.11%	7
50% + 0.05 + 1.0	83.50%	19.21%	48.31%	9
5% + 0.05 + 1.0	85.50%	19.76%	49.70%	10
5% + 0.05 + 1.5	87.50%	**20.13%**	**50.63%**	10

Table 6.8 CLIR results with translation model

The probabilities given by the translation model were kept. The experiments were conducted over some selected different filtering combination points for the same reason, which have similar translation accuracy interval of translation models. The results are listed in Table 6.8, where the monolingual IR benchmark is 39.76%.

From Table 6.8, we can observe that the average IR precision strictly increases when the translation accuracy increase. Or the translation model with highest accuracy 87.5% leads to the highest IR precision 20.13%. Comparing to Chen's result obtained from non-filtered corpus [Che00], which has 15.91% average IR precision or is 40% of mono-IR, the CLIR results with tools from filtered corpus in Table 6.8 have made significant progress that the highest reaches 50.63% of mono-IR.

[(1)] Translation accuracy is examined by different human judge in [Che00].

We also note that when the filtering criteria become stricter the number of unknown words to the trained translation model increases a little bit. This means some texts that have these words are filtered out from the corpus. The unknown words appear in the following cases:

- Some unknown words are not covered by the parallel corpus. They are unknown word even before the filtering. Most of these words are proper nouns. For example, the corpus contains no sentence with “Mount Pinatubo” and its Chinese translation “皮拉图博”. Therefore, the Chinese proper is unknown to the model trained with the corpus.
- These new unknown words are not the key words of the queries, so they do not effect the CLIR effectiveness.
- The three new unknown words added after the filtering are: 扶贫 (poverty assistance), 京九(Beijing-Kpwloon) and 希望工程(Hope Project). In the unfiltered model, these words are not translated correctly. 扶贫 is translated as “poverty”, 京九 as “capital” and 希望工程 as “hope engineering”. These translation do not have a positive impact on the CLIR effectiveness

6.2.3 CLIR Experiments with Dictionary Combined

As in the Chinese-English CLIR, we also combined translation model with bilingual dictionary for extensive English-Chinese CLIR experiments. The dictionary we used has 110,834 entries, and three different combination ratios were tested from 2:1 to 1:2. The experiments of Table 6.9 were conducted with the translation model trained with a filtering combination of 5% empty alignment proportion, 0.05 length ratio difference and 1.5 known translation weight. The experiments on influence of different filtering combination over final CLIR performance are conducted with a fixed 1:1 combination rate between translation model and extra dictionary. As for

Combination ratio	*Ave. precision*	*Comparison to mono-IR*
TM:dictionary = 2:1	25.87%	65.07%
TM:dictionary = 1:1	**26.01%**	**65.42%**
TM:dictionary = 1:2	25.46%	64.03%

Table 6.9 CLIR with different combination ratio

other verifying experiments, we only tested several specific filtering combinations. The results are shown in Table 6.10. As for English-Chinese CLIR, the strictest filtering combination also leads to the highest CLIR performance and extra dictionary can cover some more words. We also note that all unknown words are recognized by the dictionary, which has a large amount of entries.

Filtering combination (empty + length difference + word)	*TM accuracy*	*Average precision*	*Comparison to mono-IR*	*No. of unknown words*
No filtering ([Che00])	81.50%	22.32%	56.14%	0
50% + 0.5 + 1.0	81.50%	24.63%	61.95%	0
50% + 0.05 + 1.0	83.50%	24.98%	62.83%	0
5% + 0.05 + 1.0	85.50%	25.37%	63.81%	0
5% + 0.05 + 1.5	87.50%	**26.01%**	**65.42%**	0

Table 6.10 CLIR results with combined translation model and dictionary

From Table 6.7 to Table 6.10, we can conclude that:

- Filtering parallel corpus for training statistical translation model can bring better CLIR precision.
- The CLIR precision tendency is consistent with the strictness of filtering criteria, as well as alignment precision and accuracy of translation model.

- Combining of query translation by translation models with a dictionary will significantly improve the CLIR precision.

6.3 Summary

Up to this chapter, we have finished to describe all experiments of our project. The final CLIR experiment results of this chapter show strong support to our proposal of filtering noisy parallel text corpus will bring better translation model and significant improvement of CLIR performance.

Comparing the best results with former researches as Chen's study [Che00], our translation model trained from the filtered corpus leads to better translation accuracy as well as better CLIR precision as shown in Table 6.11 and Table 6.12.

Comparing item	***Translation accuracy***	***CLIR precision comparing to mono-IR***	
		Translation model	***TM + dictionary***
Non-filtered [Che00]	77.00%	42.84%	66.90%
Filtered	86.50%	53.43%	72.81%
Improvement	**14.84%**	**24.72%**	**8.83%**

Table 6.11 Improvement of C-E direction

Comparing item	***Translation accuracy***	***CLIR precision comparing to mono-IR***	
		Translation model	***TM + dictionary***
Non-filtered [Che00]	81.50%	40.02%	56.14%
Filtered	91.50%	50.63%	65.42%
Improvement	**12.27%**	**26.51%**	**16.53%**

Table 6.12 Improvement of E-C direction

From Table 6.11 and 6.12, we observe that while the translation accuracy increases about 13~15%, the CLIR effectiveness improves about 25~27% (only TM) or 9~17% (TM and dictionary). The best CLIR effectiveness reaches 65.42% or 72.81% of monolingual IR.

The best CLIR effectiveness has been obtained when strict filtering criteria are applies, and the known translation factor is attributed a value of 1.5. The CLIR effectiveness is very consistent with the evaluation on sentence alignment and translation quality of the translation model. We observe that a good text filtering method leads to a good sentence alignment result, which leads to a better translation model. This latter in turn results in a better CLIR effectiveness. This series of experiments show clearly the benefits to have a better parallel corpus. Even if during the filtering, some parallel texts are also removed (leading to a lower recall), globally, the quality of the corpus is improved. During the whole experiment, there is no word that is not covered by the model even the filtering criteria get stricter. This proves that as the number of the remaining texts is still large, a strict filtering will not compromise the coverage of the translation model.

These experiments invalidate somehow common belief that " more data is better data". This is not the case at least for CLIR.

Chapter 7

Conclusions

Our description in this thesis involved four different topics: online parallel pages mining, corpus purification or filtering, translation model training, and cross-language information retrieval. The motivation and final goal of our research aim to finding a high accuracy, low-cost and effective query translation approach for CLIR, by filtering out noise from raw corpora, improving text alignment and translation model.

Rapidly increasing online parallel pages provide us with a theoretically unlimited resource of parallel texts, especially for those languages that have few parallel corpus available, such as between Orient and European languages. The Web significantly extends the possibility and feasibility of training statistical translation model from parallel corpus. But the problem is how to efficiently find out parallel Web pages. PTMiner, developed in Chen's project, is an online parallel page finder without reading the contents of pages, but by checking the URL and file names. Chen's result of using PTMiner is a large English-Chinese parallel corpus.

The poor parallelism of the corpus created by PTMiner, comparing to manually constructed corpus, directly lead to unsatisfied translation accuracy and CLIR precision. We analyzed the noise in the original corpus, compared the original alignment results with manually built sample results. We found that most noise has some common properties:

- large empty alignment proportion
- great length difference
- inexact translation

We then proposed a set of criteria to filter out noise from the raw corpus then improve the text alignment algorithm. In our corpus purification and alignment system, three filtering principles are employed: empty alignment proportion, text length ratio and known translation. We found stricter empty alignment proportion and length ratio difference brought better alignment accuracy, and higher weight for known translation also leads to better results, but the weight should not be too high. The experiment results proved that combinations of above constraints have positive effect on alignment accuracy. Furthermore, the translation accuracy of translation model trained from the filtered corpus and the CLIR performance using this translation model are consistent with the alignment correctness.

We adopted IBM model I and its implementation in the RALI lab when we trained our translation model. However, because the properties of Chinese language and its difference from English, we made some preprocessing on both texts so that they can be aligned correctly in sentence level. Among these preprocesses, we imported Chinese code conversion, Chinese punctuation conversion, Chinese word segmentation, English word transformation into citation form and expression extraction. The sentence alignment algorithm we used was based on Gale-Church's length-based algorithm, but some modifications were made by integrating filtering criteria and other adjustments. The hybrid algorithm incorporated sentence length, HTML tags and known translation and they significantly improved the alignment accuracy.

Evaluated upon randomly selected sample words, we noticed that the translation accuracy of translation model trained from filtered corpus is significantly improved over the new corpus. We obtain relative improvements of 12.27% for English-

English and 14.94% for Chinese-English. On the other hand, it also showed that stricter corpus parallelism, or higher precision rather than higher recall, is preferable for our situation. This is possibly due to the fact that we have sufficient parallel pairs in the corpus for filtering.

Finally, we used the translation models in CLIR query translation. Significant improvements are obtained in both English-Chinese and Chinese-English CLIR. For English-Chinese CLIR, when using only our translation model, the precision reaches 50.63% of monolingual IR. This means an improvement of 26.51% over the model trained directly from the original corpus (when using translation model with extra dictionary, they reach 65.42% and 15.53% respectively). For Chinese-Chinese CLIR, we obtained a precision of 42.84% or improvement of 24.72% for translation model alone, and 72.81% and 8.83% when combining translation models with dictionaries.

This study clearly showed that a filtering process of noisy corpus can greatly help improve CLIR. However, we had not examined every aspect of the filtering process in this study. We believe there are still some potential improvements possible on many aspects along the processing chain. The possible improvements could include the following ones:

- The translation of statistical translation model is influenced and limited by the domain and topics of collected parallel texts. In order to cover most possible translation of multi-meaning words then improve query translation and CLIR precision, expanding the diversity of mining websites is helpful and useful.
- New markup languages like XML are getting more popular on the Internet. This provides other possibilities to further refine the mining and filtering process in the future.
- Sometimes, the CLIR needs are limited in some specific domains. Then rather than mining full domain parallel Web pages, only collecting those domain specific pages to form the corpus and training translation tools from it may lead

to better translation models and better CLIR performance over the specific domain.

- In this study, our alignment algorithm is based on that of Gale-Church with some modifications in order to integrate the additional filtering factors. As we indicated in Section 4.1, it is possible to further modify the coefficients used in this algorithm to adapt it further to our parallel corpus. It is also possible to use a different sentence alignment algorithm such as the one proposed in [LSV98].

References

[BLM91] P. F. Brown, J. C. Lai, and R. L. Mercer. Aligning sentences in parallel corpora. In *Proceeding of 29th Annual Meeting of the Association for Computational Linguistics*, pages 89-94, Berkeley, California, 1991.

[BPM93] P. F. Brown, S. A. Della Pietra, V. J. Della Pietra and R. L. Mercer. The mathematics of machine translation: Parameter estimation. *Computational Linguistics*, 19: 263-311, 1993.

[Che93] S, F, Chen. Aligning sentences in bilingual corpora using lexical information. In *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics*, pages 9-16, Columbus, Ohio, 1993.

[Che00] Jiang Chen. Parallel Text Mining for Cross-Language Information Retrieval using a Statistical Translation Model, *M. Sc. degree thesis*, University of Montreal, 2000.

[Chu93] K. W. Church. Char_align: a program for aligning parallel texts at the character level. In *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics*, pages 1-8, Columbus, Ohio, 1993.

[DDO95] M. W. Davis, T. E. Dunning and W. C. Ogden. Text alignment in the real world: Improving alignments of noisy translation using common lexical features, string matching strategies and n-gram comparisons. In *Proceedings of the Conference of the European Chapter of the Association of Computational Linguistics*. University of Dublin, March 1995, EACL.

[DL96] S.T.Dumais, T. K. Landauer and M. L. Littman. Automatic cross-linguistic information retrieval using latent semantic indexing. In *SIGIR'96 workshop on Cross-Linguistic Information Retrieval*, 1996.

[FC94] Pascale Fung and Kenneth Ward Church. K-vec: A New Approach for Aligning Parallel Texts. In *Proceedings from the 15th International Conference on Computational Linguistics (Coling-94)*, pages 1096-1102. Kyoto, 1994

[FM94] Pascale Fung and Kathleen McKeown. Aligning noisy parallel corpora across language groups: Word pair feature matching by dynamic time warping. In *AMTA 94: Partnerships in Translation Technology*, pages 81-88, Columbia, Maryland, October, 1994.

[Fun95] Pascale Fung. A pattern matching method for finding noun and proper noun translation from noisy parallel corpora. In *Proceedings of the 33rd Annual Conference of the Association for Computational Linguistics*, Boston, Massachusetts, 1995.

[GC91] William A. Gale and Kenneth W. Church. A Program for Aligning Sentence in Bilingual Corpora. In *Proceeding of the 29th Annual Meeting of the Association for Computational Linguistics*, pages 177-184, Berkeley, California, 1991

[Gre98] Gregory Grefenstette. The Problem of Cross-Language Information Retrieval. In *Cross-language Information Retrieval.* Kluwer Academic Publishers. pages 1-9, 1998

[HB98] Julia Hockenmaier and Chris Brew. Error-Driven Learning of Chinese Word Segmentation. In Communications of COLIPS, Volume 8(1), pages 69-84, Singapore, 1998

[IMT99] Intelligent Miner for Text. Available at: http://www.software.ibm.com/data/iminer/fortext/about.html 1999.

[KR93] M. Kay and M. Roscheisen. Text-translation alignment. *Computational Linguistics*, 19:121-142, 1993.

[Kwo99] K.L. Kwok. English-Chinese cross-language information retrieval based on a translation package. In *Workshop of Machine Translation for Cross Language Information Retrieval, Machine Translation Summit VII*, Singapore, 1999.

[Liu87] Y. Q. Liu. Difficulties in Chinese language processing and method to their solution. In *Proc. Of 1987 International Conference on Chinese Information Processing*, Volume 2, pages 125-126, 1987.

[LSV98] Philippe Langlais, Michel Simard, Jean Véonis. Methods and Practical Issues in Evaluating Alignment Techniques. In *Proceedings of 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistic*, Montreal, Canada, August 10-14, 1998.

[NJH94] Jian-Yun Nie, Wanying Jin and M.L. Hannan. A hybrid approach to unknown word detection and segmentation of Chinese. In *International Conference of Chinese Computing*, pages 326-335, Singapore, 1994.

[NJS] NJStar Software Co. NJStar Communicator, available: http://www.njstar.com

[NRB95] Jian-Yun Nie, Xiaobo Ren, And Martin Brisebois. A unifying approach to segmentation of Chinese and its application to text retrieval. In *ROCLING 8*, pages 172-190, August, 1995.

[NSID99] Jiangyun Nie, Michel Simard, Pierre Isabelle and Richard Durand. Cross-language information retrieval based on parallel texts and automatic mining parallel texts from the Web. In *ACM SIGIR '99*, pages 74-81, August 1999.

[Rij79] C. J. Van Rijsbergen. Information Retrieval. 2nd edition. London, Butterworths, 1979.

[SFI92] Michel Simard, George F. Foster and Pierre Isabelle. Using cognates to align sentences in bilingual corpora. In *Proceedings of TMI-92*, Montreal, Quebec, 1992.

[SM83] Gerard Salton and Michael J. McGill. Introduction to Modern Infromation Retrieval. McGraw-Hill Book Company, 1983

[SP98] Michel Simard and Pierre Plamondon. Bilingual sentence alignment: Balancing robustness and accuracy. *Machine Translation*, 13:59-80, 1998.

[SS91] R. Sporat and C. Shih. A statistical method for finding word boundaries in Chinese text. *Computer Processing of Chinese and Oriental Languages*, 4(4):336-351, 1991.

[Tka98] Daniel Tkach. Text mining technology – turning information into knowledge – a white paper from IBM. technical report, *IBM Software Solutions*, February 1998.

[UIYM94] Takehito Utsuro, Hiroshi Ikeda, Masaya Yamane, Yuji Matsumoto and Makoto Nagao. Bilingual Text Matching using Bilingual Dictionary and Statistics. In *15th COLING*, pages 1076-1082, 1994

[Wu94] Dekai Wu. Aligning a parallel English-Chinese corpus statistically with lexical criteria. In *ACL-94: 32nd Annual Meeting of the Association for Computational Linguistics*, pages 80-87, Las Cruces, NM, June 1994.

[WX95] Dekai Wu and Xuanyin Xia. Large-Scale Automatic Extraction of an English-Chinese Translation Lexicon. In *Machine Translation*, 9:3-4, pages 285-313, Kluwer Academic Publishers, Boston, MA, 1995.

Acknowledgements

Here I'd like to express my acknowledgement to my supervisor Dr. Jian-Yun Nie. It's his generous guide that led me finish my research and this thesis. And I also thank Philippe Langlais, Michel Simard, Graham Russell, Guy Lapalme and Elliott Macklovich, who gave me so much help during my research.