

# **AUTOMATIC BITEXT ALIGNMENT FOR SOUTHEAST ASIAN LANGUAGES**

by

Lwin Moe

A thesis submitted in partial fulfillment of the requirements  
for the degree of Master of Science  
in Computer Science

Examination Committee:      Dr. Paul Janecek (Chairman)  
   Dr. Matthew Dailey (Member)  
   Doug Cooper (External Member)

Nationality:                      Myanmar  
Previous Degree:                Bachelor of Science  
   Indiana-Purdue University  
   Indiana, United States of America

Scholarship Donor:            AIT Fellowship

Asian Institute of Technology  
School of Engineering and Technology  
Bangkok, Thailand  
December 2008

## **Acknowledgement**

First of all, I would like to thank my parents for their love and support.

I wish to express my deep appreciation and sincere gratitude to all my teachers throughout primary, middle, high school and college.

I would like to thank the following people for their help and guidance in my life:

- Neil and Diana Sowards
- Dr. Beomjin Kim, Dr. David Erbach
- Dr. Paul Janecek, Dr. Matthew Dailey

I am also grateful to Rikker Dockum for answering my questions on Thai.

Last but not least, I would like to thank Doug Cooper, for his encouragement, inspiration and invaluable suggestions.

## Abstract

Bitext alignment is the task of aligning words, phrases or sentences in one language with the equivalent translation in another. Aligned bitexts help lay the groundwork for statistical machine translation, are useful for language teaching, provide data for cross-language information retrieval, and have a variety of other applications.

This thesis investigates the problem of bitext alignment for English and Southeast Asian languages. Although bitext alignment in general has been well studied, most algorithms, implementations, and even performance metrics depend on the assumption that both texts have been regularly divided into words and sentences. Bitext alignment of Southeast Asian languages has not benefited from previous work because they are not normally divided this way. There is no completely reliable machine method for dividing such texts into words and sentences.

We will use Thai as our example and test language because experimental data are readily available. However, our goal is to develop insights into the best methods of automatically aligning “low resource” Southeast Asian languages like Burmese, Khmer, and Lao.

This thesis will explore *dictionary-based* alignment methods to improve basic length-based method. We will begin by introducing existing European and Asian bitext corpora, and then discuss current approaches to bitext alignment problems. First, we discuss the basic length-based approach that we use as our baseline method. We then look at the use of lexical features and semantic analysis; for example, using dictionary-based similarity and WordNet relatedness measures, to enhance the baseline methods. Finally, we test different approaches to adapting a Southeast Asian language, Thai, to work with these methods.

Before aligning with dictionary-based methods, we pre-segment the Thai input using various techniques and prepare the English and Thai input using stemming, stopword removal or normalization of derived forms in English.

This thesis will make the following contributions:

1. It will establish the baseline performance of the naïve basic method.
2. It will introduce metrics for evaluating the performance of bitext alignment, taking both sentence boundary detection and alignment of individual Thai segments into account.
3. It will test and measure different approaches to Southeast Asian word segmentation in the input text preparation before determining similarity between Thai sentence segments and English sentences.
4. It will compare the effectiveness of English-to-English comparison (that is, translate the Thai segments to English first) versus Thai-to-Thai comparison (that is, translate the English sentences to Thai first).
5. It will test and measure the effects of using different types of dictionaries for translation and alignment.

6. It will test and measure the effects of stopword removal, stemming, simplification of derived forms on dictionary-based realignment.
7. It will test WordNet relatedness analysis to realign the output of length-based method.
8. It will provide data that will be useful for ongoing research into such problems as detection and correction of misordered or missing alignment pairs.
9. It will make Southeast Asian language-specific recommendations on performance measurement, segmentation algorithm, segmentation dictionary, translation type, translation dictionary and different approaches to improve the segment and sentence similarity test.

## Table of Contents

<b>Chapter</b>	<b>Title</b>	<b>Page</b>
	Title Page	i
	Acknowledgement	ii
	Abstract	iii
	Table of Contents	iv
	List of Figures	vii
	List of Tables	viii
Chapter 1	Introduction	1
	1.1 What is a bitext corpus?	1
	1.2 Applications of bitext corpora	1
	1.3 Bitext alignment problem	1
	1.4 Objectives	2
	1.5 Scope of the thesis	3
	1.6 Outline	3
Chapter 2	Literature Review	5
	2.1 Existing bitext corpora	5
	2.2 Current approaches to bitext alignment	7
	2.3 Problem of misordered or missing sentences in the translation	8
	2.4 Word segmentations and bitext alignment in Southeast Asia	9
	2.5 Sentence boundary detection	12
	2.6 Summary	12
Chapter 3	Methodology	13
	3.1 Linguistic resources	13
	3.2 Segmenting Thai	14
	3.3 Preparing for alignment	15
	3.4 Alignment methods	17
	3.5 Performance Metrics	19
	3.6 Summary	21

Chapter 4	Preliminary Tests	23
4.1	Test cases	23
4.2	Testing similarity of English sentences and Thai segments: 18 methods	23
4.3	Score calculation, normalization and comparison	25
4.4	Discussion	27
4.5	Summary	28
Chapter 5	Results	29
5.1	Results of alignment using different segmentation algorithms, different segmentation and translation dictionaries	29
5.2	Results of alignment based on English-to-Thai rough translation with input variations	31
5.3	Results of alignment based on Thai-to-English rough translation with input variations	32
5.4	WordNet relatedness analysis	34
5.5	Summary	34
Chapter 6	Discussion	36
6.1	Segmentation issues	36
6.2	Preparing input text for dictionary-based alignment	43
6.3	Alignment methods	46
Chapter 7	Conclusions and Recommendations	50
7.1	Metrics	50
7.2	Length-based method as a baseline	50
7.3	Segmentation	51
7.4	Translation to and from English	51
7.5	Finding similarity between Thai segments and English sentences	52
7.6	Conclusion	52
	Bibliography	53
	Appendices	57

## List of Figures

<b>Figure</b>	<b>Title</b>	<b>Page</b>
3.1	WordNet hierarchy for <i>canary</i> , <i>house</i> and <i>gazebo</i>	19
3.2	Alignments between languages L1 and L2. Both methods 1 and 2 have the same number of correct alignment, but method 1 is clearly closer to being correct	20
3.3	Visual comparison of average sentence similarity scores and standard deviations for each method. Method 16 and 17 have higher average similarity scores for true sentence pairs and low average similarity scores for random pairs.	26

## List of Tables

Table	Title	Page
3.1	SEALang library's hand-aligned corpora	14
4.1	Table of standard deviations and average similarity scores for each method	26
5.1	Alignment results with various segmentation dictionaries and algorithms, and various translation dictionaries (previously discussed as methods 1 to 10 in Chapter 4) on the Wanakam corpus.	30
5.2	Alignment results using English-to-Thai translation (method 11 to 13)	32
5.3	Alignment results using Thai-to-English translation (method 15 to 17)	33
5.4	Alignment results comparing English-to-Thai translation to Thai-to-English translation (method 12 uses English-to-Thai translation and method 16 uses Thai-to-English translation)	34
6.1	Segmentation of <b>ทั้งวัน</b> with Wordcut (maximal match) using default and SWATH dictionary	38
6.2	Segmentation of <b>หลังไหล</b> with Wordcut (maximal match) using default and SWATH dictionary	39
6.3	Correct segmentation of <b>หมอยากล่าว</b> using maximal match	40
6.4	Incorrect segmentation of <b>หมอยากล่าว</b> using longest match	40
6.5	Highlighted words are correctly matched. Thai phrases are segmented using maximal match approach	40
6.6	Segmentation of <b>ของดร.</b> using maximal match	41
6.7	Segmentation of <b>ของดร.</b> using shortest minimal match	41
6.8	Sentence with a typo in Thai	41
6.9	Segmentation of Thai phrase with a typo	42
6.10	Sentence with proper noun, Sally	42
6.11	Sentence with proper noun, Aladdin ( <b>อลาดติน</b> )	43
6.12	<b>ต่อ</b> is incorrectly matched with the previous English sentence	44
6.13	Dictionary definitions of <b>ต่อ</b>	45
6.14	Comparing <i>leave</i> and <i>left</i>	46
6.15	Alignment between English and Thai using the naïve length-based method. The numbers in the parenthesis are character counts. [Eng-Thai] is the difference in character count between English and Thai sentences.	47
6.16	Alignment using naïve length-based method with the help of a dictionary-based analysis. The highlighted Thai segment was moved back up to match with the corresponding English sentence as the result of dictionary-based analysis.	48
6.17	WordNet relatedness analysis	49
6.18	Incorrect WordNet relatedness analysis	49



# Chapter 1

## Introduction

This chapter introduces the research problem: bitext alignment for Southeast Asian languages and English. It explains what bitext corpora are and what applications they have. It then discusses what the challenges are for Southeast Asian languages. Finally, it describes the aims and objectives of the thesis.

### 1.1 What is a bitext corpus?

A bitext corpus is a collection of electronic texts in two languages. It comprises a variety of texts in a source language and their parallel translations in a target language. In a bitext corpus, texts are usually aligned in a manner that helps a reader easily compare the source language text with its target translation.

### 1.2 Applications of bitext corpora

Bitext corpora have a very broad range of uses, including supporting human and machine translations, assisting in cross-language information retrieval, and aiding in second language acquisition, especially in the development of reading skills.

Statistical and example-based machine translation systems use bitext corpora to derive the parameters for their statistical models. Example-based machine translation systems also use large bitext corpora to learn example sentences to be able to do translation jobs.

Cross-language information retrieval makes use of bitext corpora. In cross-language information retrieval, a user sends the query in one language and gets results back in another. Bitext corpora are useful because they can be used to find exact phrase equivalences of the query.

Learners of foreign languages can greatly benefit from bitext corpora. The availability of a vast number of sample sentences with their parallel translations can not only improve the student's reading skills, but helps with their ability to produce and translate language as well.

Finally, translation memory systems are designed to assist human translators by seeking out phrase equivalents. They can be extremely useful for technical or formal writing in which long fixed phrases are frequently used.

### 1.3 Bitext alignment problem

Bitext alignment is the task of aligning words, phrases or sentences in one language with the equivalent translation in another. Texts are usually aligned at the phrase or sentence level. Bitext alignment is not a trivial computer task because two languages rarely have

an exact sentence-to-sentence correspondence in translation. A very short sentence might have a very long translation—one sentence may be translated into two or more sentences in the other language or vice versa. Or, a sentence or paragraph may even be left out completely.

Various methods have been proposed to solve bitext alignment problems for European languages, as will be discussed in the following chapter. As a rule, they depend on the fact that both languages are broken into words, sentences, and paragraphs.

On the contrary, there has been little work on Southeast Asian languages. Unlike English and other European languages, most Southeast Asian languages do not normally segment text at the word level, and they do not clearly mark sentence breaks. As a consequence, the standard algorithms and implementations for bitext alignment do not work well for those languages, and there are no well-studied alternative approaches. No previous research has been done to establish baseline performance of the bitext alignment for Southeast Asian languages.

## 1.4 Objectives

Proven and effective methods for bitext alignment for European languages include the basic Gale and Church algorithm [1], as well as more sophisticated approaches seen in implementations like *hunalign* [2]. Both of these are discussed in the next chapter. We will try to adapt a Southeast Asian language, Thai, to work with the existing methods. We will presegment the Thai input using various techniques before doing the dictionary-based translation.

The objectives are:

- To establish the baseline performance of the naïve length-based method.
- To introduce metrics for evaluating the performance of bitext alignment, taking both sentence boundary detection and alignment of individual Thai segments into account.
- To test and measure different approaches to Southeast Asian word segmentation in the input text preparation before determining similarity between Thai sentence segments and English sentences.
- To compare the effectiveness of English-to-English comparison (that is, translate the Thai segments to English first) versus Thai-to-Thai comparison (that is, translate the English sentences to Thai first).
- To test and measure the effects of using different types of dictionaries for translation and alignment.
- To test and measure the effects of stopword removal, stemming, simplification of derived forms on dictionary-based realignment.
- To test WordNet relatedness analysis to realign the naïve method output.
- To provide data that will be useful for ongoing research into such problems as detection and correction of misordered or missing alignment pairs.
- To make Southeast Asian language-specific recommendations on performance measurement, segmentation algorithm, segmentation dictionary, translation type, translation dictionary and different approaches to improve the segment and sentence similarity test.

## 1.5 Scope of the thesis

In this thesis, we will work on the problem of automated bitext alignment for Thai and English texts. We will:

- discuss the background of the problem, describe existing approaches to solving it, and discuss the particular problems posed by Southeast Asian languages.
- test and measure baseline performance for the basic length-based method.
- test and measure the effects on the alignment of various segmentation algorithms (see section 2.4 for the discussion of these algorithms.)
- test and measure the effect of stopword removal and stemming on the alignment algorithm.
- test and measure the effect of simplification of English derived forms (“*leave|leaves|leaving|left*” to *leave*) on the alignment.
- test and measure the use of WordNet relatedness analysis on the alignment algorithm.
- propose alignment quality metrics for the above tests and results.

## 1.6 Outline

This thesis is organized as follows.

In Chapter 2, we will introduce existing European and Asian bitext corpora and discuss current approaches to bitext alignment problems. First, we will discuss the basic length-based approach that will be used as our baseline method. We will then look at the use of lexical features and semantic analysis such as using a dictionary to enhance the baseline methods. The problem of misordered or missing sentences in the translation and text segmentation issues will also be discussed.

In Chapter 3, we will discuss different ways of preparing input text before doing the alignment: segmenting Thai, rough translation, stopword removal, stemming English, simplification of derived forms. We will then discuss three alignment methods: 1) naïve length-based method, 2) dictionary-based method and 3) WordNet relatedness measure-based method.

In Chapter 4, we will look at the preliminary results of different approaches to comparing English sentences with individual Thai segments. Based on the results, we will describe how reliable the similarity scores obtained by different approaches to comparison are and which method or methods perform better than others. The similarity scores obtained by these approaches will be used to enhance our baseline method in aligning different corpora in the following chapter.

In Chapter 5, alignment results for different corpora will be presented. First, we will present results of alignment using different segmentation algorithms, different dictionaries for segmentation and translation. Second, we will present results of alignment based on English-to-Thai rough translation with such input variations as stopword removal, stemming and simplification of derived forms. Third, we will present results of alignment based on Thai-to-English rough translation with similar input variations as above. Finally, WordNet relatedness analysis is used to align a corpus.

In Chapter 6, detailed findings and insights of the alignment methodology will be discussed. Many illustrative examples taken from the test will also be presented.

In Chapter 7, we will discuss how the methodology can be applied to align a Southeast Asian language and give specific recommendations.

## Chapter 2

### Literature Review

This chapter begins by introducing existing European and Asian bitext corpora. Then, current approaches to bitext alignment and problem of misordered or missing sentences in the alignment are discussed. Finally, text segmentation issues—breaking Southeast Asian text into sentences and words—are discussed.

#### 2.1 Existing bitext corpora

The usefulness of bitext corpora has led to several large scale projects for many languages. However, the availability of bitext corpora is still limited, especially for less common languages. Some bitext corpora for European and Asian languages are introduced here.

##### 2.1.1 *European languages*

One of the most widely referenced bitext corpus in computational linguistics research is the Canadian Hansard corpus. The Canadian Hansard is the printed transcripts of the Canadian parliamentary debates. The transcripts are maintained in English and French. Several versions of the Canadian Hansard exist. The University of Southern California version [3] is freely available; it comprises the records of the 36<sup>th</sup> Canadian Parliament from 1997 to 2000. This version has around 2 million words in English and French. Another version maintained by the Linguistic Data Consortium [4] has the records from mid-1979 to 1988. It contains 2.87 million parallel sentence pairs.

The Europarl (European Parliament Proceedings) [5] is a collection of proceedings from the European Parliament. The proceedings are from 1996 through 2006. Eleven languages available in the corpus are French, Italian, Spanish, Portuguese, English, Dutch, German, Danish, Swedish, Greek and Finnish. The corpus consists of around 44 million words per language.

The English-Norwegian Parallel Corpus [6] comprises original texts in both English and Norwegian and their translations. The texts are from both fiction and non-fiction books. The corpus has 100 original texts and their parallel translations. The total number of words is nearly 2.6 million. They were collected over the period of 1994 through 1997.

The English-Swedish Parallel Corpus [7] is very similar to English-Norwegian corpus. This corpus has 64 English text and their translations in Swedish. In addition, it also has 72 Swedish texts and their translations in English. The texts include both fiction and non-fiction materials. The total number of words in the corpus is 2.8 million words. The project was conducted over the period of 1997 to 2001.

The Hunglish corpus [8] [2] consists of Hungarian-English parallel texts collected from literature, religious texts, legal texts, software documentation, movie subtitles, magazines

and news. The corpus comprises about 54.2 million words in 2.07 million sentences. It was built to discuss the methodology to build a bitext corpus for medium density languages such as Hungarian and Romanian.

### 2.1.2 Asian languages

Hong Kong Parallel Text [9], produced by Linguistic Data Consortium, is the combination of three corpora. The three corpora are Hong Kong Hansards, Hong Kong Laws and Hong Kong News. Hong Kong Hansards is the collection of the proceedings of the Legislative Council of Hong Kong. This corpus contains records from October, 1995 to April, 2003. 714 documents in English and Chinese have a total of 36 millions English words and 56 millions Chinese words. Hong Kong Laws contains statute laws established by the Department of Justice of Hong Kong up to the year 2000. It has a total of 8 millions English words and 14 millions Chinese words in 42,255 documents. Hong Kong News contains press releases from Hong Kong government. The press releases are from July 1997 to October 2003. Hong Kong News has a total of 59 millions English words and 98 millions Chinese words in 87,590 documents.

ASAHI Corpus [10] is a collection of articles from the Asahi Shimbun newspaper in Japan. Asahi Shimbun newspaper is one of Japan's oldest newspapers, and is published in both Japanese and English editions. The corpus comprises 472 articles in Japanese and their parallel translations from the years 1989 to 1991.

English-Vietnamese Corpus [11] comprises translations from computer books, Longman lexicon of contemporary English dictionary (Vietnamese version by Tran Tat Thang), English-Vietnamese bilingual dictionaries, translation of SUSANNE corpus, electronic books, children's encyclopedia, and other books. It has a total of 5 million Vietnamese and English words. Sentences were manually aligned if the source text had to be typed, and a Gale and Church aligner was used for automatic alignment if the source was already in electronic format.

Southeast Asian Languages (SEALang) library [12] has Thai-English and Khmer-English bitext corpora. Thai bitexts are collected from Wanakam World Classics in Thai project [13], Thai Fiction in Translation [14] project, and the weekly *Translate It* section of the Bangkok Post [15]. Khmer bitexts are compiled by extracting example sentences from the Headley Cambodian-English Dictionary [16]. Sentences are manually aligned in these corpora.

Asia Online [17] is a private company registered in Bangkok, Thailand. Their business activities involve development of software and delivery of services in the areas of machine translation, internet portals and search. Their machine translation system uses bitexts which are aligned automatically using n-gram models and later manually checked.

In contrast to European corpora, most Asian bitext corpora are aligned manually. Even though research has been done for Chinese, Japanese and Korean [18], there has been very little work for Southeast Asian languages because of limited resources.

## 2.2 Current approaches to bitext alignment

Several approaches have been devised to tackle the bitext alignment problem for European languages. The approaches can be divided into three categories, namely 1) length-based approaches, 2) lexical feature-based and semantic approaches and 3) combination of the above two approaches. They will be discussed in the following sections.

### 2.2.1 Length-based approaches

Length-based approaches rely on the fact that the lengths of text segments are usually directly proportional to their equivalent translations. The earliest sentence-level bitext alignment techniques include Gale and Church [1], who proposed a method using character counting (later implemented as *Vanilla aligner* [19]), and Brown et al. [20] who counted words for alignment purposes. One of the weaknesses of length-based approaches is their inability to detect misordered, missing, or extra sentences in the translation.

The Vanilla aligner has a more subtle problem—for efficiency, it only allows one-to-two sentence matches. This is a problem for Southeast Asian languages, which do not always mark sentence boundaries. As we will see in section 3.2.1, Thai text is broken into pseudo-sentences using pre-existing spaces as sentence boundaries. This produces more short sentence segments than there really are supposed to be. Vanilla aligner fails when one English sentence has to be aligned with several short Thai segments.

But despite the weaknesses of the underlying method, and the Vanilla implementation, this approach is fairly language independent, especially among European languages, and works very well.

### 2.2.2 Lexical feature-based and semantic approaches

Lexical features, such as cognates, collocations, and ‘anchor’ words (described below) can be used in the alignment. Looking more deeply into semantics, with the help of bilingual dictionaries, extends this approach. Some of the approaches using lexical features, semantics, and the combination are discussed below.

Some approaches choose specific words to serve as anchor points in the alignment. The words are chosen manually or by distribution. Kay and Roscheisen [21] uses words with similar distributions in the set of sentences that are potential matches as anchor points in the sentence alignment. Fung [22] uses vectors to take notes of the distributions of words in the arbitrary segments of the text. The distribution information is then used to build a set of anchor words that can be used for the alignment. Nevado et al. [23] also used a set of anchor words, which they manually defined. The set consists, for example, of “*for*”, “*and*”, “*I would like*”, and “*I wish*.”

Simard et al. [24] applied cognates as the main criterion instead of character length in their alignment approach. Cognates are words that have the same origin and are therefore phonologically or orthographically similar. The words ‘*haus*’ in German and ‘*house*’ in English are examples of cognates. Using the orthographic similarity as the basic idea,

Simard et al. assume, in their approach, that the cognates share at least the first four characters of the words, which have to be at least four characters long. Obviously, phonologically similar pairs such as ‘*haus*’ and ‘*house*’ will not be recognized as cognates in their approach. Cognates can only be applied in the alignment of language pairs that share the same origin.

Collocational frequencies can also be applied to build a list of words that can be used in the alignment [25]. Collocations are combinations of words that co-occur more frequently than by chance, and are thus assumed to be meaningful as a set. For example, “*stock market*” and “*make a decision*” are collocations that have specific meanings and usage patterns. Regular correspondences of this kind form the basis of an alignment word list.

Semantic approaches take the meaning of sentences into consideration in the alignment. *Hunalign* [2], for example, uses a dictionary-based rough translation to check the similarity of the sentences in the source and target language text. The Piperidis et al. method [26] looks for the meanings of verbs, nouns, adjectives and adverbs in sentences. They determine the “*semantic load*” of a sentence based on those words. It is then used as a criterion for the alignment.

### 2.2.3 Combination of length-based and lexical approaches

Some approaches apply a combination of length-based, lexical feature-based and semantic approaches. Brown [20] uses the idea of a set of anchor words to divide the text into smaller chunks before aligning sentences with a word-counting method. Simard et al. [24] and Hoftland [27] apply cognates to improve a length-based alignment.

The best-developed combined approach is probably *hunalign* [2]. Varga et al. apply *hunalign* algorithm, which uses both Gale and Church approach and lexical information, to align a Hungarian-English corpus.

First, *hunalign* finds length-based and token-based similarity scores for each sentence. Length-based similarity scores are calculated counting characters in both languages' texts. Token-based similarity scores are calculated using a dictionary if it is available. A simple rough translation from source to target language is done using the dictionary. The translation is then compared with the text from the target language to calculate token-based similarity scores for each sentence.

In the next step, the initial alignment is done using length-based and token-based similarity scores. After the first initial alignment, a dictionary is automatically built. The newly built dictionary is used to improve the existing one. If none exists, the bootstrapped dictionary is used. Next, alignment is done using either the bootstrapped dictionary or the improved dictionary.

## 2.3 Problem of misordered or missing sentences in the translation

One of the problems in bitext alignment is misordered or missing sentences in the translation. In principle, sentence similarity measures can help handle this problem. During or



after alignment, corresponding sentences can be checked for similarity. If two neighbors are more similar than their partners, they can be switched.

Language 1		Language 2
aaaaa		BBBBB
bbbbb		AAAAA

In the above example, sentences aaaaa / bbbbb are in one language, while BBBB / AAAAA are in another. We assume that aaaaa and AAAAA are more similar than aaaaa and BBBB (likewise for bbbbb and AAAAA), so we swap their order.

Ways to assess sentence similarity include:

- Naïve approaches such as character counting and word counting of sentences
- Checking for lexical cues such as cognates, collocations and anchor words in a sentence
- Semantic approaches such as checking meanings of sentences using dictionary-based rough translation, and comparing the meaning of verbs, nouns, adjectives and adverbs in sentences

In a sense the final approach is built into *hunalign*. It compares words from target sentences and rough translations of words from source sentences to produce a token-based similarity score, which can be retrieved by the user.

This approach can be carried further with WordNet [28], an English lexicon that groups words into a concept hierarchy. WordNet can be called an “improved thesaurus”, which encodes associations of words in an innovative way. Pedersen et al. [29] use WordNet to calculate semantic similarity and relatedness between words. They use hierarchy relationships defined in WordNet to find the similarity score between concepts of words. For example, *car* and *bus* are more likely to be similar or related compared to *car* and *dog* because *car* and *bus* are types of *vehicle*. *Dog*, on the other hand, is a type of *mammal*.

Even though using similarity measures to deal with misaligned or missing sentences appears to be attractive and intuitive, the reality is not so easy. Two very different sentences may have close similarity scores purely by coincidence, or two similar sentences may have very different scores because of choices made by the translator.

Before we attempt to use similarity measures to detect and fix sentence mismatch problems, it is necessary to develop a better understanding of how to interpret the similarity measures discussed above. This would be a very interesting follow-up research problem once we have more data on baseline similarity measurements.

## 2.4 Word segmentations and bitext alignment in Southeast Asia

Word segmentation has long been an issue in language processing for languages without obvious word boundaries. Southeast Asian languages such as Thai, Khmer, Lao and

Burmese do not normally segment text at the word level. Instead, words are separated at unmarked phrase or sentence boundaries. Word segmentation is an ongoing research issue in the above-mentioned languages.

Most of the methods for bitext alignment previously described rely on words in sentences. In short, the methods will not give accurate results unless texts are divided into sentences and words. Obviously, it will give the wrong counts in the word-counting approaches. The character-counting approach also needs to know where to break sentences after it finds the proportional character counts in the target language text. If the text is not properly segmented, at least at the phrase level, the sentence breaking would be wrong.

Unless sentences have been divided into words, lexical cues such as anchor words, cognates and collocations cannot be determined correctly. Indeed, the dictionary lookup will be incorrect or the words will not be found in the dictionary as proper dictionary lookup is required to calculate sentence similarity score based on rough dictionary translation, as was described in *hunalign's* approach.

Some of the well-established methods for word-segmentation are discussed below.

#### 2.4.1 Syllable-based approaches

Syllable tables were originally used to suggest line breaking. Most Southeast Asian languages have hand-generated tables, for example [30] for Burmese. Rules for consonants, vowels, diacritical marks are used to decide whether a break between characters is possible. For example, *seg* and *ment* are possible from *segment*.

Therefore, it is possible to break the line between *seg* and *ment*, but not between *se* and *gm*. Although syllable-based approaches were good enough for line breaking, they do not correctly determine word boundaries.

#### 2.4.2 Dictionary-based approaches

Dictionary-based approaches were the next innovation, and were meant to do a better job of finding true word boundaries.

In the longest match approach used by Poowarawan [31], the longest possible match from the dictionary is chosen at each point. For example, 'freezebra' would be segmented as 'freeze' (the longest dictionary match starting at 'f') and 'bra' (the longest match beginning with 'b'), instead of *free* and *zebra*. We can clearly see that the longest match gave us the wrong answer.

Another approach, maximal matching [32], is based on the observation that the segmentation with the fewest words is usually correct. For example, 'autobiography' is likely to be correct, even though it might be segmented further into 'auto' and 'biography'. In this approach, all possible segmentations are found. The segmentation with the fewest words is then chosen as the correct one.

However, equal counts of segmentations are still problematic as in the above example, *freezebra*. The question arises on which to choose: ‘freeze’ and ‘bra’ or ‘free’ and ‘zebra’?

Even the basic assumption that segmentation with fewer words is going to be correct may be flawed in some cases. The choice between ‘*I did not pick up the bill*’ and ‘*I did not pickup the bill*’ will obviously be wrong.

On the other hand, the dictionary-based approach cannot identify missing or misspelled words in the text. No dictionary can hope to include every possible word. Proper nouns, including people's names, are a major problem. Some authors such as Mark Twain intentionally misspelled words: “*Tain't thunder, becuz thunder* –” in *The Adventures of Tom Sawyer*. It is clear that *becuz* is not in the dictionary. Web pages or blogs that include slang can also cause problems for dictionary-based segmentation.

As I have just discussed, weaknesses of dictionary-based approaches include (1) incorrect rule-based segmentation, and (2) words not being in the dictionary. Corpus-based statistical approaches, discussed in the next section, were proposed to take the limitations of dictionary use into account.

### 2.4.3 Statistical approaches

Statistical approaches gather observable data from text. Then, they use these data to help guide decision making in text segmentation. Two easily measured statistics are the co-occurrence of parts-of-speech in successive words, and the co-occurrence of syllables.

Kawtrakul et al. [33] propose a statistical approach involving the use of parts of speech (POS) data and an  $n$ -gram model. A POS tagger was used to find all possible part of speech segments in a text. Then the text was divided into  $n$ -grams, or sub-sequences of length  $n$ . In the Kawtrakul et al. approach, sub-sequences of 3 segments are used. The probabilities for each  $n$ -gram were then used to choose the best segmentation. The adjective + noun ‘*free zebra*’, the example from above, would probably be found more frequently in the corpus than the verb + noun ‘*freeze bra*’.

Aroonmanakun [34] uses a combination of old and new methods. He first segments the text into syllables. He then merges the syllables into all possible combinations of words using a dictionary. Finally, he decides which combination of words is the best based on a statistical measure that weighs the collocational strength between adjacent syllables, and maximizes the sum based on observations from a hand-segmented corpus. His results are almost identical to maximal matching.

Statistical approaches have weaknesses as well. A major problem is the quality or size of the training corpus used to determine probabilities of  $n$ -grams or adjacent syllables. For example, the widely used ORCHID Corpus [35] used in the POS approach was machine-segmented, and is internally inconsistent in its choice of word boundaries. This can reduce the value of the statistical model when it is applied to ambiguous situations.

#### 2.4.4 Feature-based approaches

Features such as immediate collocational information and the presence of context words in the nearest K-word neighborhood are used to disambiguate segmentations [36]. Machine learning algorithms, such as RIPPER and Winnow, use these features to disambiguate preliminary segmentations resulting from maximal matching and Part of Speech tagging.

### 2.5 Sentence boundary detection

Some implementations of bitext alignment algorithms, namely *Vanilla* [19] and *hunalign* [2], need the input text to be segmented at the sentence level. However, not much work on sentence boundary detection for Thai, Lao or Khmer has been found in the literature.

Looking for spaces between sentences is the essence of most Thai sentence breaking techniques. Spaces in Thai text can be between phrases, clauses or sentences. All approaches, so far, use a training corpus, in which words are tagged with appropriate part-of-speech tags and spaces with “sentence-break space” or “non-sentence-break space” tags.

In the first approach, the frequency distributions of three consecutive items, one word before and after a space, are calculated from the training corpus [37]. They were applied to disambiguate sentence-breaking and non-sentence-breaking spaces in the text.

Another approach using the machine-learning algorithm ‘Winnow’ [38] also employs a training corpus to gather collocational information and number of words before and after spaces for disambiguation of sentence-breaking and non-sentence-breaking spaces.

Both methods use the ORCHID corpus for training and testing. The average accuracy of these methods is about 80%, and there are no better alternatives. The quality of the training corpus is also a limitation for both of these methods.

### 2.6 Summary

This chapter discussed existing European and Asian bitext corpora. It then discussed current approaches to bitext alignment: 1) length-based method, 2) lexical feature-based and semantic methods, and 3) a combination of length-based and lexical approaches. It also addressed the issues of misordered or missing sentences in the alignment. Finally, word segmentation and sentence boundary detection issues for Southeast Asian languages were discussed.

## Chapter 3

### Methodology

As we have just seen in the previous chapter, there are a) several well-developed approaches to bitext alignment, all of which rely on word-segmented text, and b) several well-developed approaches to word segmentation, none of which can definitively be called “correct”.

We will align different corpora listed in section 3.1 using some of those methods.

In section 3.2, we discuss how Thai text is broken into segments using pre-existing spaces as sentence boundaries. This step is necessary for naïve length-based method which works on the concept of sentence length.

We also discuss that Thai text needs to be segmented into words and/or compounds to do dictionary-based rough translation. This is necessary to do dictionary lookups to compare the Thai segments with English sentences in realigning the output of naïve length-based method.

Section 3.3 discusses how input texts are prepared using dictionary-based rough translation, stopword removal, and stemming or simplification of derived forms before the dictionary-based realignment.

In section 3.4, we discuss three main methods of bitext alignment: 1) naïve length-based method 2) dictionary-based realignment and 3) WordNet relatedness-based realignment. Note that naïve length-based method only requires the Thai text to be broken into segments to simulate sentence boundaries. It does not require word segmentation or other preparation steps.

Finally, in section 3.5 we discuss performance metrics to account for “close mistakes” in scoring.

#### 3.1 Linguistic resources

The SEALang library [12] has hand-aligned bitext corpora for Thai and English, as mentioned in the previous chapter. The following corpora from SEALang were used for both testing and checking:

- Wanakam: World classics in Thai [13]
- Bangkok Post: Translate It (Sunee Canary) [15]
- Haas: (Mary Haas Thai Reader) [12]
- LangNet: (Language Learning Support System) [39]
- Scribner Messenger, 2007: Translating Newspaper Thai into English [12]

Table 3.1 SEALang library’s hand-aligned corpora

	Number of sentences	Number of paragraphs	Average number of sentences in each paragraph	Average character counts in an English sentence	Average character counts in a Thai sentence	Total number of articles and stories
Wanakam	10700	3445	3	84	215	64
Bangkok Post	2677	1473	2	108	328	223
Haas	712	173	4	64	136	44
LangNet	2104	969	2	97	234	60
Scribner	126	70	2	162	350	14

## 3.2 Segmenting Thai

Two issues were involved in segmenting Thai text: determining sentence boundaries and breaking continuously written text into individual words.

### 3.2.1 Sentence boundary detection

Determining sentence boundaries in Thai involves identifying sentence-breaking spaces as we saw in the previous chapter. As a naïve approach, pre-existing spaces were used as sentence breaks. Even though this produced more sentences than there really are, the alignment algorithm recombined them when it tried to match them with their English equivalents.

Language-specific rules were used to prevent unnecessary breaks. Spaces between some acronyms such as ค. ศ. (Christian Era) and พ. ศ. (Buddhist Era) were not treated as sentence breaks. Spaces were also ignored before the punctuation mark ฯ (mai yamok), which indicates that the phrase it follows is repeated. Spaces after Thai digits (๐๑๒๓๔๕๖๗๘๙) in lists like 1) ..., 2) ... were also ignored.

### 3.2.2 Word segmentation

Different approaches to word segmentation will have different outcomes. For example, maximal matching as explained in section 2.4.2 will likely produce fewer words, but more compounds. A ‘minimal match’ approach will produce fewer compounds but more words in total. Both methods will be affected by the dictionary employed, since bigger dictionaries will have more compounds, but may have more small, specialized words that can lead to incorrect segmentation as well.

The following three segmentation dictionaries were chosen for trials with different segmentation algorithms:

- Headwords from Haas dictionary (small dictionary with just headwords, 8119 words) [12]
- Wordcut dictionary: 19304 words [40]
- SWATH dictionary 23944 words [42]
- Lexitron Thai-English dictionary: 40851 words [41]

We tried several algorithms using the above-mentioned dictionaries:

- Maximal match with SWATH using SWATH dictionary
- Longest match with SWATH using SWATH dictionary
- Maximal match with Wordcut using Lexitron Thai-English dictionary
- Maximal match with Wordcut using Haas dictionary
- Shortest minimal match with SWATH using SWATH dictionary

Another reason for allowing so many possibilities for input is that segmentation dictionaries and dictionaries used for rough translations (see section 3.3.1) are not necessarily the same. We cannot predict whether segmentation that produces many headwords, or fewer compound words, will be better when we attempt dictionary-based semantic approaches to alignment.

### **3.3 Preparing for alignment**

Our dictionary-based alignment is based on comparison of Thai segments and English sentences aligned by naïve length-based method. Since naïve length-based method aligned English sentences with Thai segments broken at the pre-existing spaces as described in 3.2.1, some segments that are not proper sentence boundaries are incorrectly aligned with neighboring sentences.

The basic problem is deciding whether a boundary Thai segment should be aligned with the current sentence or the preceding English sentence (if the segment is the first in the current sentence) or next (if the segment is the last in the current sentence). We assume that the boundary segment will, in general, share some observable surface or lexical features, or semantic content, with the correct English sentence.

In section 3.4.2, we discuss exactly how two segments are compared, and how their similarity is scored. First, we will describe several methods of preparing both Thai and English segments for comparison.

#### *3.3.1 Rough translation*

Beyond the naïve length-based approach (3.4.1, below), we must either translate the Thai segment (so that it may be compared with the English sentence) or vice versa. Dictionaries of various sizes were used for English-to-Thai and Thai-to-English rough translation prior to comparison:

- Lexitron Thai-English dictionary (large dictionary with many headwords and compounds, 40851 words) [41]

- Lexitron English-Thai dictionary (large dictionary with many headwords and compounds, 83206 words) [41]
- Headwords from Haas dictionary with Lexitron definitions (small dictionary with just headwords, 8119 words) [12]

### 3.3.2 Stopword removal

English sentences and Thai segments that we compare often contain words with little semantic content, but which may give the appearance that two segments are similar. Such words are defined as stopwords in information retrieval, and are regularly removed. We tested the effect of removing stopwords such as “a, an, the, of, in” in our sentence comparison tests.

Both English and Thai stopwords lists can be found in Appendix A. English stopwords from the Perl module, `Lingua::EN::StopWords` [43], were used at first. However, we found that results were better after several prepositions, adverbs and pronouns such as “*anybody, anyone, anything, anywhere, before, behind*” were removed from the stopword list. As a result, a smaller stopword list (shown in Appendix A) was used in the end.

Thai stopwords (shown in Appendix A), on the other hand, were chosen from words that were commonly found in the corpus. Common words such as `กัน|ทาง|กร|กรรม`, which are similar or equivalent to *through, -ing, -ment, -ness* in English, were chosen.

### 3.3.3 Stemming English

Stemming, used in information retrieval, is a method of reducing a word to an approximation of its “root” value. There are various stemming algorithms, of which the Porter stemming algorithm [44] is probably best-known. Using the Porter algorithm, *operational, operating, operative* will be stemmed as *oper*.

Even though meaning is sometimes lost in stemming, it may improve alignment performance slightly. For example, in *hunalign*, Varga et al [2] tried stemming both the dictionary and the text before bitext alignment and found out that the alignment improved. We tested the effect of stemming on the alignment by:

- stemming input English
- stemming the Thai glossing dictionary

Although we stemmed English texts and English definitions of Thai words, stemming is not applicable to Thai, which is an isolating language. We achieve the same effect in the prior step of stopword removal.

### 3.3.4 Normalizing derived forms

Stemming operates only on the surface form of a word, and does not analyze its meaning or derivational morphology. Thus, there are many cases where stemming will not help. For example, the Porter algorithm will stem *leave* and *left* to *leav* and *left* respectively, without recognizing that they are different forms of the same word.



To address this problem, derivational analysis is considered, again as in information retrieval; e.g. Jacquemin et al. [45]. For example, a list of the different morphological forms of *leave*, `|leave|leaves|leaving|left|`, can be used to select a common item for comparison. The list helps us reduce both *leave* and *left* to the same root form. In our alignment, we use a list from the SEALang Library [12] that is used in their corpus lookup tool.

### 3.4 Alignment methods

Three basic methods were used for alignment:

1. Naïve length-based approach
2. Alignment with dictionary-based rough translation
3. Alignment based on WordNet relatedness test

The naïve approach, described in 3.4.1, provides a rough alignment that serves the following two purposes:

- it provides a baseline, against which we can measure performance improvement
- it provides a starting condition for our alternative approaches.

As we will discuss below, aligned texts using the naïve approach consist of a single English sentence aligned to several short Thai segments. The first and last Thai segments that are aligned with the English sentence will be defined as *boundary segments*. The naïve approach typically fails in its handling of one or more Thai boundary segments. The first Thai segment should have been aligned with the previous English sentence or the last Thai segment with the next English sentence.

We will describe various approaches we used to try to determine whether the resulting Thai boundary segments were correctly matched with the current sentence. In doing this, we attempt to determine the *similarity* between the leading (or trailing) Thai segment, and the preceding (or following) English sentence, and the sentence it is currently aligned with.

It is expected that the Thai boundary segments belong to the English sentences they are most similar to. However, it is not clear exactly which method can best show this “similarity,” or how to best compare different similarity measures. Before we work on bitext alignment, we will carry out a series of tests that explore the design, implementation and apparent effectiveness of similarity testing. The work provides a useful result on its own, and is discussed in Chapter 4.

#### 3.4.1 Naïve length-based

A naïve length-based method was used to establish baseline performance, because this is the simplest and most fundamental alignment method.

English sentences were aligned with several short Thai segments broken at pre-existing spaces, using character count as the criterion for the alignment. Although Thai text is not divided into words, spaces are used for several purposes, and are invariably found at sentence boundaries. Two or more sentences are never run together. Using this assumption,

Thai text was broken into segments at pre-existing spaces as described in 3.2.1. The resulting segments were aligned with English sentences on the basis of character counts. Alignment performance was tested, both counting and ignoring spaces between English words.

Breaking Thai text into segments at pre-existing spaces will almost certainly cause an error in sentence boundary detection, and subsequently in alignment, because there might be spaces between phrases in the sentence. As a result, some boundary segments will be misaligned by our naïve approach.

To determine if the resulting Thai boundary segments were correctly aligned with the current English sentence or not, we attempt to determine the similarity between the leading (or trailing) Thai segment, the preceding (or following) English sentence, and the sentence it is currently aligned with. We discuss this in the following section.

### *3.4.2 Alignment with dictionary-based rough translation*

As discussed above, Thai boundary segments—the results of naïve length-based alignment—may need to be moved up or down to neighboring English sentences.

Boundary segments were moved based on a similarity score between Thai segments and English sentences. The input texts were prepared in advance using various approaches discussed in sections 3.2 and 3.3 before doing the word comparison in the sentence similarity test. The similarity scores were calculated by counting exact word matches between Thai segments or English sentences and their dictionary-based rough translations.

This combination of segment comparison test and paragraph realignment using the naïve approach continued until the end of a paragraph in aligning a document. The boundary segments were moved to the English sentence to which they were more similar (in other words, the number of exact word matches is higher). After moving the boundary segments from the first sentence pair, we realigned the paragraph from the second sentence onwards using our naïve approach. After realigning the paragraph, dictionary-based analysis was performed again on the second sentence pair. The process of realignment continued until the end of the paragraph.

### *3.4.3 Alignment based on WordNet measure of relatedness*

Two sentences may have the same meaning. Nevertheless, they might use different words. For example, the following two sentences discuss the same concept even though they use different words for it.

- *They had their lunch at the gazebo.*
- *They ate at the summer building in the garden.*

In such cases, our alignment method based on exact word matches will fail because they do not share any exact words. We used an alternative approach that allows us to compare words based on the concept.

The *Lesk* measure of relatedness counts gloss overlaps between concepts in WordNet hierarchy to calculate a relatedness score [29]. In Figure 3.1, the WordNet hierarchy for *canary*, *house* and *gazebo* is shown. If we were to compare these three words, *house* and *gazebo* would be more “related” because the gloss overlaps of the concepts for *house* and *gazebo* would be higher than that of *house* and *canary*.

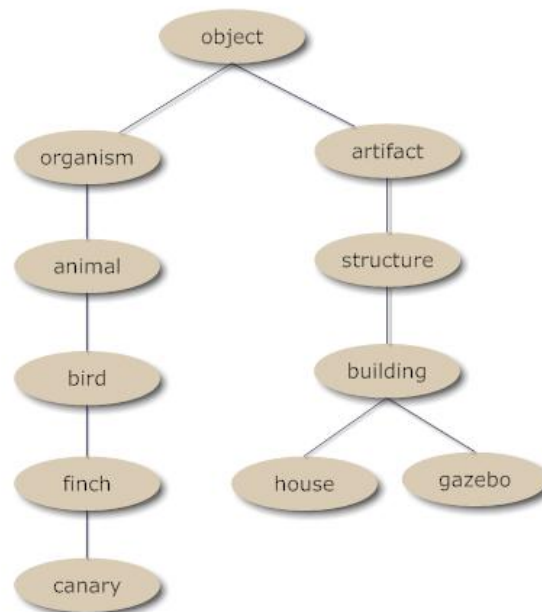


Figure 3.1 WordNet hierarchy for *canary*, *house* and *gazebo*

The *Lesk* measure of relatedness, from the Perl module implementation WordNet::SenseRelate::WordToSet [46], was used in the sentence similarity test to move border segments. Each word from dictionary-based rough translation was compared with sets of words from current English sentence and preceding or following English sentence. The module returned the *Lesk* measure of relatedness of the word to the English sentences. The scores for each word were calculated and the average was found, in order to compare the similarity of the segment with the current sentence and the preceding or following sentence.

The combination of *Lesk* measure of relatedness test and paragraph realignment using naïve approach continued until the end of a paragraph in aligning a document. The segments were moved according to the *Lesk* relatedness score. After moving the segments, the paragraph was realigned using the naïve approach from the next sentence onwards. This process was repeated until the end of the paragraph.

### 3.5 Performance Metrics

In the experiments, hand-aligned text was used as ground truth for evaluating the results of the alignment.

Scoring bitext alignment is a difficult and subtle problem. The simplest approach is to calculate scores for the correct alignments only, and ignore all incorrect alignments. However, this is misleading, because it does not distinguish between close mistakes, in which the alignment was only off by a sentence or two, and major mistakes, in which the alignment was off by several sentences. We can see this situation below in Figure 3.2.

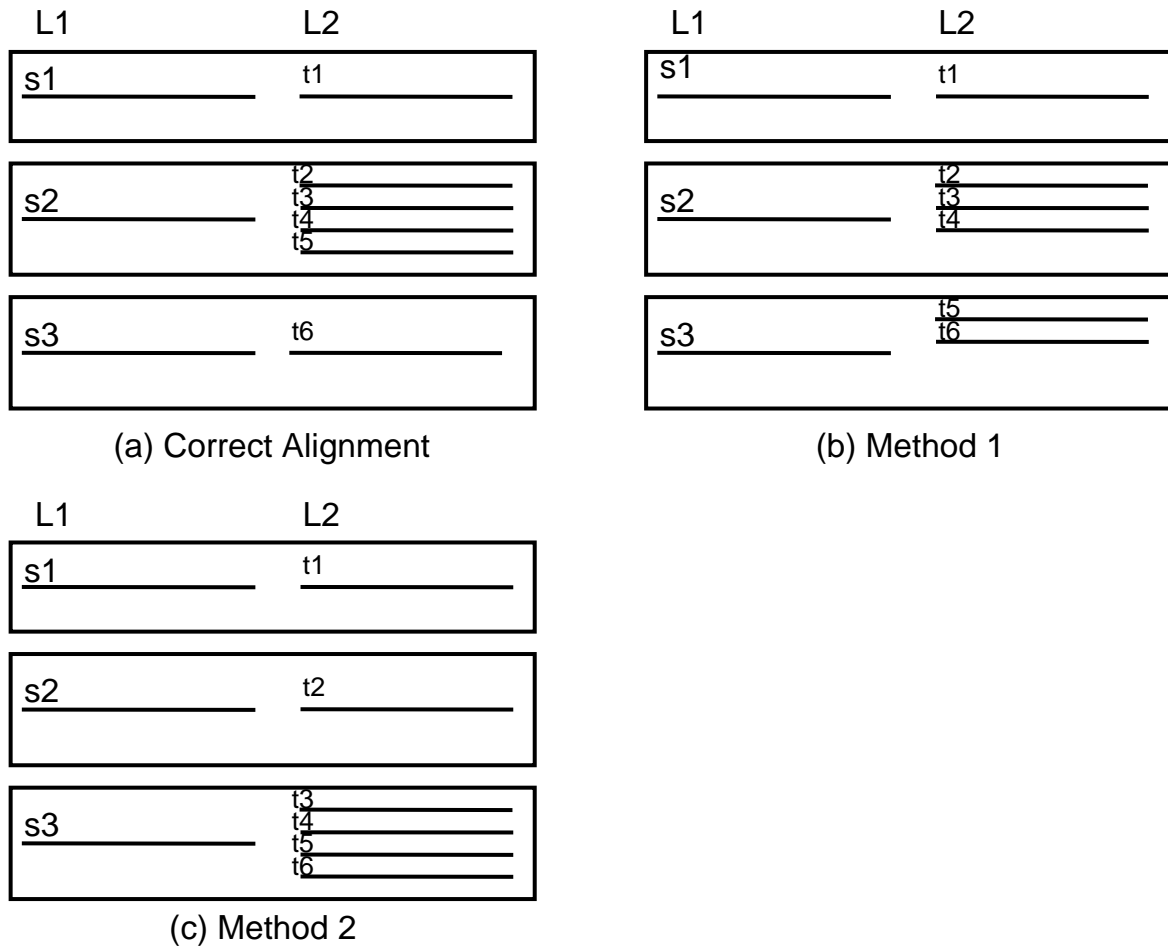


Figure 3.2 Alignments between languages L1 and L2. Both methods 1 and 2 have the same number of correct alignment, but method 1 is clearly closer to being correct

Figure 3.2 (a) is the correct alignment between languages L1 and L2. Figure 3.2 (b) and (c) show alignment results from methods 1 and 2. Both methods 1 and 2 produce incorrect alignments for sentence 2 (s2). However, we can clearly see that method 1 outperforms method 2.

If we had used the simplest approach, each method would have been given the same score. Instead, we used a different approach that is similar to the “ladder” approach used by Varga et al. [2]. Here is how we calculated precision and recall in the case above.

Correct alignment pairs set:

$$\{(\{s1\}, \{t1\}), (\{s2\}, \{t2\}), (\{s2\}, \{t3\}), (\{s2\}, \{t4\}), (\{s2\}, \{t5\}), (\{s3\}, \{t6\})\}$$

Method 1 result set:

$\{(\{s1\}, \{t1\}), (\{s2\}, \{t2\}), (\{s2\}, \{t3\}), (\{s2\}, \{t4\}), (\{s3\}, \{t5\}), (\{s3\}, \{t6\})\}$

Only one element for method 1,  $(\{s3\}, \{t5\})$ , will be marked “incorrect” using this approach.

Method 2 result set:

$\{(\{s1\}, \{t1\}), (\{s2\}, \{t2\}), (\{s3\}, \{t3\}), (\{s3\}, \{t4\}), (\{s3\}, \{t5\}), (\{s3\}, \{t6\})\}$

For method 2,  $(\{s3\}, \{t3\})$ ,  $(\{s3\}, \{t4\})$ ,  $(\{s3\}, \{t5\})$  will be marked “incorrect” using this approach.

The recall score counts the number of correctly aligned segments, **regardless of position**. Therefore, the recall score for method 1, which has 5 correct segments aligned, is  $0.83$  ( $5/6$ ), whereas the recall score for method 2, which has 3 correct segments aligned, is  $0.50$  ( $3/6$ ).

The precision score, on the other hand, is calculated by the number of **correct first segments** in the sentences. This gives us an intuitive sense of ‘proper’ alignment. Both methods, in our example, gave the correct first segments for sentence 1 and sentence 2:  $(\{s1\}, \{t1\})$ ,  $(\{s2\}, \{t2\})$ . The precision score for both methods, therefore, is  $0.33$  ( $2/6$ ).

In general, a method with good precision says that more sentences start off properly aligned, and agrees with our intuitive sense of what alignment means. A method with high recall score means nearly all *segments* are aligned properly as well. This helps give credit to methods that are almost correct, but which misalign the first segment.

### 3.6 Summary

We first broke Thai texts into segments using spaces as sentence boundaries, and aligned them with English sentences using naïve length-based method. Naïve length-based methods produced incorrectly-aligned boundary segments as a result of incorrect sentence boundary detection in breaking Thai texts.

Next, we used various approaches of segmentation schemes and different segmentation dictionaries to segment Thai texts into words and/or compounds.

Next, rough translations in both English-to-Thai and Thai-to-English translations were compared with the actual text to check for similarity, and the segments were realigned accordingly. We translated the segmented Thai words to English using translation dictionaries of different sizes in Thai-to-English translation. The translated English words were then compared with the actual English text. In English-to-Thai translation, however, translated Thai words of an English sentence were compared with the segmented Thai text. In both cases, segments were realigned based on the similarity between the rough translations and the actual text.

We also applied WordNet relatedness measure to test similarity between the rough translated words and the actual text.

We discussed various approaches from information retrieval, such as stopword removal, stemming and normalizing derived forms, to improve the word comparison between rough translations and the actual texts.

Finally, we discussed a metric for evaluating performance of all the above methods, taking into account “close mistakes” in scoring.

In the next chapter, we will see how these methods perform in two preliminary test cases.

## Chapter 4

### Preliminary Tests

We have just introduced, in Chapter 3, different approaches to comparing English sentences with individual Thai segments. In this chapter, we will compare these approaches in our preliminary tests.

First, in section 4.1, we discuss how test cases are designed. English and Thai sentence pairs are randomly chosen for comparison.

Next, in section 4.2, we discuss and list different approaches to comparing English sentences with individual Thai segments, which were chosen in test cases (discussed in 4.1). We determine their *similarity scores* using different approaches.

Finally, in section 4.3, we discuss score calculation, normalization and comparison, and then present our results. Similarity scores for each sentence pair in the two test cases (discussed in 4.1) are calculated and normalized. Average scores and standard deviations are then calculated to help us see how each method performs.

The preliminary results help us establish methodology for, and reliability of, obtaining similarity scores by different methods. This will let us determine which method or methods appear to perform better than the others, and are the best methods to aid in bitext alignment.

#### 4.1 Test cases

Two test samples were selected from the various corpora mentioned in the previous chapter. In the first set, 1,000 English sentences with their corresponding Thai translations were randomly chosen as “correct” pairs. In the second set, 1,000 English and Thai sentences were randomly chosen as “incorrect” pairs. In the second set, the English and Thai did *not* come from corresponding translations, which were intentionally excluded.

In test cases, we expect that Thai segments will show a high degree of similarity with the proper sentences, and a low degree of similarity with the random sentences. We tested this, using different approaches to input variations as explained in the following section.

#### 4.2 Testing similarity of English sentences and Thai segments: 18 methods

A variety of approaches were taken to preparing the English sentences and/or Thai segments for comparison. Methods 1 – 13 translated the English sentences into Thai, and then did Thai-to-Thai comparison with the original Thai segments. Methods 14 – 18 translated the Thai segments into English, and then did English-to-English comparison with the original English sentences.

**English-to-Thai translation:** English sentences translate to Thai.

1. Comparison without using segmentation; Lexitron Thai-English dictionary was used for translation
2. Maximal match using Lexitron Thai-English dictionary; Lexitron Thai-English dictionary was used for translation
3. Maximal match using Lexitron Thai-English dictionary; Haas dictionary headwords with Lexitron definitions were used for translation
4. Maximal match using headwords from Haas dictionary; Haas dictionary headwords with Lexitron definitions were used for translation
5. Maximal match using SWATH with SWATH dictionary, Lexitron Thai-English dictionary was used as a translation dictionary
6. Maximal match using SWATH with SWATH dictionary, Haas dictionary headwords with Lexitron definitions were used as a translation dictionary
7. Shortest minimal match using SWATH dictionary, Lexitron Thai-English dictionary was used as a translation dictionary
8. Shortest minimal match using SWATH dictionary, Haas dictionary headwords with Lexitron definitions were used as a translation dictionary
9. Longest match with SWATH dictionary, Lexitron Thai-English dictionary was used as a translation dictionary
10. Longest match with SWATH dictionary, Haas dictionary headwords with Lexitron definitions were used as a translation dictionary
11. Method 5 + Thai stopwords removal
12. Method 5 + Thai stopwords removal + Stemming
13. Method 5 + Thai stopwords removal + Derivatives

**Thai-to-English translation:** Thai segments translated to English.

14. Maximal match using SWATH dictionary; Lexitron Thai-English dictionary was used for translation
15. Method 14 + English stopwords removal
16. Method 14 + English stopwords removal + stemming
17. Method 14 + English stopwords removal + derivatives
18. Method 14 + English stopwords removal + WordNet

Various word segmentation algorithms were tested in methods 1 to 10 of the experiments. Note that in method 1, no segmentation was used. Instead, the translated Thai words were matched as strings embedded in the non-segmented sentences. In methods 2 to 10, maximal match, longest match and shortest minimal match algorithms were tested.

One of the best segmentation algorithms, maximal matching, was chosen after testing methods 1 to 10 and applying those methods on the Wanakam corpus. We looked at both the score comparison (in Figure 4.1) and the segmented results of the text in choosing the best segmentation algorithm (discussed in 5.1). The chosen algorithm was used in methods 11, 12, 13, 14, 15, 16, 17 and 18.

Different segmentation and translation dictionaries of various sizes (discussed in sections 3.2.2 and 3.3.1) were tested in methods 1 to 10 of the experiments. Several combinations of small and large dictionaries for segmentation and translation were tested. A small set of head words from Haas dictionary (8119 words), medium-sized SWATH dictionary (23944 words) and large lexitron Thai-English dictionary (40851 words) were used.



One of the best combinations of segmentation dictionary and translation dictionary was chosen by looking at the score comparison (in Figure 4.1) and the results of the alignment on the Wanakam corpus (discussed in 5.1). It was then used in methods 11, 12, 13, 14, 15, 16, 17 and 18.

### 4.3 Score calculation, normalization and comparison

In this section, we will discuss our methodology. We will first explain how sentence *similarity scores* are calculated and normalized for each method listed in the previous section. Next, we will discuss how average similarity scores and standard deviations are calculated to compare different methods, and briefly explain why WordNet relatedness scores cannot directly be compared with the other scores. Then, we present our results in Table 4.1 and Figure 4.1.

Our methodology is as follows. *Similarity scores* for each sentence pair using each method (listed in section 4.2) were calculated and normalized. English sentences were compared with the first and last Thai segments from corresponding Thai sentences, and exact word matches were counted as explained in section 3.4.2. The number of word matches was then divided by word counts of the English sentence to normalize the scores.

The average similarity scores and standard deviations for each method were calculated so that they could be compared to establish the reliability of each method. Similarity scores for each sentence pair were first used to calculate means and standard deviations.

Table 4.1, below, compares average similarity scores and standard deviations for different sentence similarity tests. Figure 4.1, in turn, shows a comparison graph of average similarity scores and standard deviations for different methods involving both English-to-Thai and Thai-to-English rough translations.

Note that average similarity scores for WordNet relatedness approach cannot directly be compared with the other average similarity scores. Scores for methods 1 – 17 are based on the number of word matches between Thai segments and the English sentence. In contrast, the WordNet relatedness test weights scores on the basis of match length, i.e. phrase matches are given higher scores than single word matches [46]. We scaled the WordNet relatedness scores to fit in the same range as our other similarity scores in order to reveal their basic problem: they do not adequately distinguish between correct and incorrect matches of translated sentences and/or segments.

Table 4.1 Table of standard deviations and average similarity scores for each method

	Test 1 (true pairs)		Test 2 (random pairs)	
	Avg.	$\sigma$	Avg.	$\sigma$
Method 1	0.179187745	0.013456600	0.046450850	0.000017230
Method 2	0.132014726	0.010523400	0.027355625	0.000093450
Method 3	0.100719330	0.010164300	0.027508806	0.000028720
Method 4	0.076128191	0.010713901	0.009826139	0.000065324
Method 5	0.137214050	0.010873400	0.030694395	0.000099320
Method 6	0.114747883	0.010354600	0.027474300	0.000069810
Method 7	0.133464344	0.010123400	0.032723396	0.000013890
Method 8	0.116331676	0.010337800	0.028296341	0.000049810
Method 9	0.135917603	0.016413400	0.030995672	0.000016870
Method 10	0.111970161	0.010743200	0.031752052	0.000096520
Method 11	0.105690524	0.015989200	0.008109223	0.000019870
Method 12	0.103264817	0.014791200	0.009673071	0.000016980
Method 13	0.156677553	0.016459800	0.009673071	0.000019230
Method 14	0.130261839	0.013489100	0.032128441	0.000019240
Method 15	0.107509742	0.017198300	0.008925367	0.000012340
Method 16	0.176935220	0.016658200	0.008775753	0.000015640
Method 17	0.167815304	0.010225582	0.008145698	0.000037533
Method 18	0.201884000	0.103002100	0.091613000	0.043486800

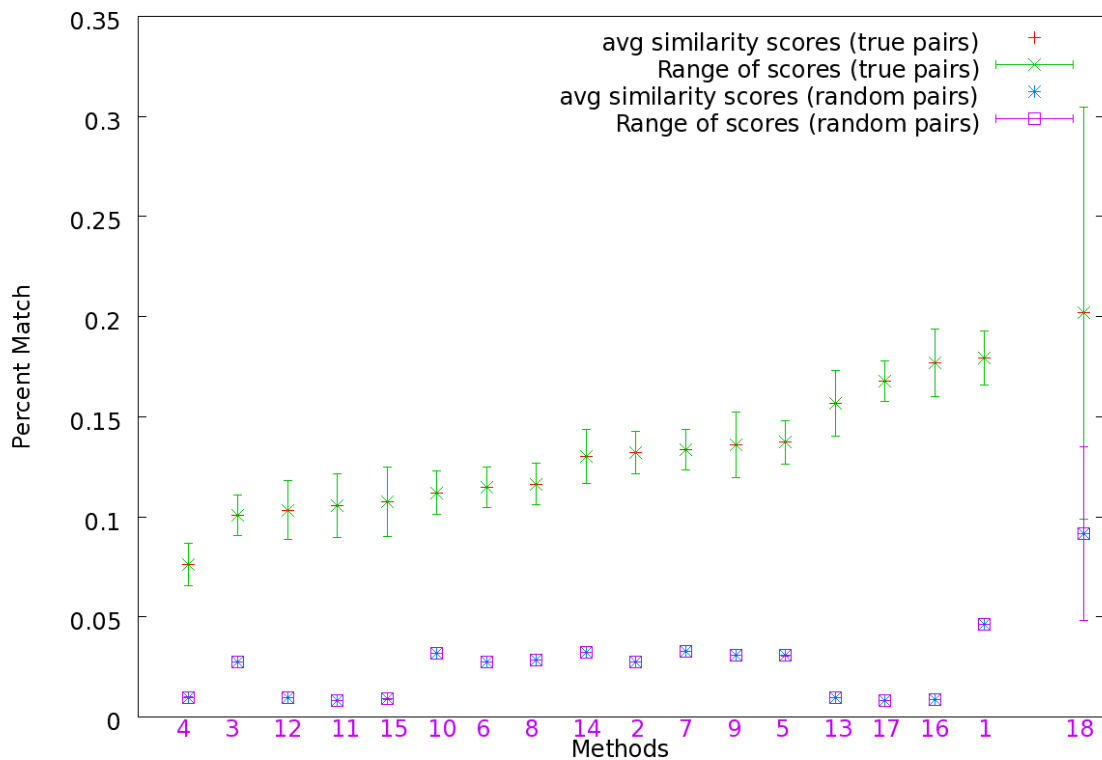


Figure 4.1 Visual comparison of average sentence similarity scores and standard deviations for each method. Method 16 and 17 have higher average similarity scores for true sentence pairs and low average similarity scores for random pairs. (Note that method 18 scores are scaled to fit in this range. It cannot be directly compared with the average scores of other method.)

## 4.4 Discussion

Table 4.1 and Figure 4.1 in the previous section compare various approaches to establishing similarity (and dissimilarity). These include different approaches to word segmentations, stemming, and the simplification of derived forms and the use of segmentation and translation dictionaries of various sizes.

First, we will discuss which methods perform better than others. Next, we will discuss if stopword removal is effective or not. We will also discuss the effect of large translation dictionary in the comparison tests. We will then discuss how shortest minimal match segmentation helps to alleviate the limited size of the translation dictionary. We will then discuss WordNet relatedness scores. Last, we will discuss the limitation of the test cases.

The results show that methods **13**, **16** and **17**, which use stopword removal, stemming and expansion of derived forms, perform better than others because of their high average similarity scores for “correct” sentence pairs and low average similarity scores for “incorrect” sentence pairs.

We also see that stopword removal helped in identifying false matches, as methods 11 and 15 had low average similarity scores for “incorrect” pairs.

Using a large translation dictionary with many headwords, as in methods 5, 7 and 9, is also clearly better than using small translation dictionary, as in methods 6, 8 and 10.

However, using shortest minimal match segmentation algorithm helps to alleviate the limited size of the translation dictionary. Shortest minimal match segmentation produces more headwords than compounds, which works well with the fact that the translation dictionary actually has only headwords and no compounds.

This is seen in comparing methods 3, 6, 8 and 10. All of those methods use a small translation dictionary with only headwords and no compounds. We can see that the average similarity scores for the “incorrect” pairs for those methods are not much different from each other. However, method 8, which uses shortest minimal match algorithm with the SWATH segmentation dictionary, has the best average similarity scores for “correct” pairs.

We cannot compare the WordNet relatedness approach’s average similarity scores directly with the other average scores, which are based on the number of word matches between Thai segments and the English sentence. However, we can see from the graph in Figure 4.1 that the standard deviations of the scores for both “correct” pairs and “incorrect” pairs are much higher compared to the other methods. We can also see that the variations from the average score for the “correct” pairs and the variations from the average score for the “incorrect” pairs actually overlap. We will further discuss this in 5.4 and 6.3.3.

Note that, in all of the test cases, we are testing an idealized case: one set of tests involves all correct pairs while the other has all incorrect pairs. However, in real-world texts, there might be cases where the segments are more similar to neighboring sentences even though the segment should actually belong to the current sentence.

This occurs because sentences in a paragraph are discussing the same topic. For example, consider *John walked five miles on his way back from the library. He wasn't happy that it was so far from his house.* In this example, the segment containing “*library*” from the first sentence might also be similar to its neighboring sentence in cases where the translation uses “*library*” instead of the pronoun, “*it*”. We will discuss this again in section 6.3.2.

We can reasonably expect this to occur when we attempt to align real texts. We will, therefore, apply these methods to real-world bitext corpora in the following chapter and report the results.

## 4.5 Summary

In this chapter, we designed test cases to compare *similarity scores* by 17 different approaches to comparing English sentences with individual Thai segments.

We found that:

- Maximal match algorithm with a small segmentation dictionary scores best.
- Shortest minimal match algorithm helps to alleviate the limited size of the translation dictionary.
- Using a large translation dictionary with many headwords is better in general than using a small one.
- Stopword removal, stemming and simplification of derived forms steadily improve the similarity scores.

In the next chapter, we will apply these methods to real-world bitext corpora and report the results.

## Chapter 5

### Results

In this chapter, the precision and recall figures for different alignment methods will be reported. We have already noted in section 3.5 that in our results, precision is the number of first segments that are correctly aligned and recall is the number of segments that are correctly aligned overall.

First, section 5.1 discusses results of aligning the Wanakam corpus. In the experiments in this section, the translation direction (from English to Thai) was held constant. However, different segmentation algorithms, and different segmentation and translation dictionaries were used. The best performance here will guide our subsequent tests.

Second, we will report, in section 5.2, results of alignment of all five corpora, using the best-observed combination of segmentation dictionary and algorithm, and translation dictionary (as discussed in section 5.1). In this section, we use English-to-Thai rough translation for all tests, but vary the way we prepare the input text, using stopword removal, stemming and simplification of derived forms.

Third, in section 5.3 we reverse the translation direction, and report the results of alignment of five corpora. In this section, we use Thai-to-English rough translation for all tests, but again vary the way we prepare the input text, using stopword removal, stemming and simplification of derived forms.

Fourth, we will briefly discuss alignment results obtained using WordNet relatedness analysis in section 5.4.

Finally, we summarize the results in section 5.5.

#### **5.1 Results of alignment using different segmentation algorithms, different segmentation and translation dictionaries**

In the alignment methods using English-to-Thai rough translation, the segmentation algorithms discussed in section 3.2.2 were all tested. We already know that the maximal matching algorithm and a small dictionary work better in isolated tests of single English sentences and Thai segments (as seen in Figure 4.1), but we want to find out if the segmentation algorithm and/or the segmentation dictionary will have a significant effect on alignment of complete bitexts.

Both small (headwords from Mary Haas dictionary with Lexitron definitions) and large (Lexitron Thai-English) translation dictionaries were tested. As above, we already know from the test cases in Figure 4.1 that the large translation dictionary (Lexitron Thai-English) works better in isolated tests of single English sentences and Thai segments, but again, we want to test if the changes in translation dictionary will have an effect on the alignment of complete bitexts.

The alignment method, using similarity scores based on English-to-Thai translation, was held constant, and used throughout all the experiments.

The following table shows the precision and recall on the alignment of the Wanakam corpus.

Table 5.1 Alignment results with various segmentation dictionaries and algorithms, and various translation dictionaries (previously discussed as methods 1 to 10 in Chapter 4) on the Wanakam corpus. The results show that using a large translation dictionary is clearly better than using a small translation dictionary. (Note: telex is the Lexitron Thai-English dictionary.)

<b>Methods</b>	<b>Precision</b>	<b>Recall</b>
Naïve length-based approach	0.79	0.88
(4) Maximal + Haas, small Haas dictionary	0.81	0.89
(3) Maximal + telex, small Haas dictionary	0.81	0.89
(6) Maximal + Swath dictionary, small Haas dictionary	0.82	0.89
(8) Shortest minimal + Swath dictionary, small Haas dictionary	0.82	0.89
(10) Longest match + Swath dictionary, small Haas dictionary	0.82	0.89
(1) Without segmentation	0.83	0.89
(2) Maximal + telex, large telex dictionary	0.85	0.91
(5) Maximal + Swath dictionary, large telex dictionary	0.85	0.91
(7) Shortest minimal + Swath dictionary, large telex dictionary	0.85	0.91
(9) Longest match + Swath dictionary, large telex dictionary	0.85	0.91

In each case reported in Table 5.1, we used the same basic method for alignment: similarity scores based on English-to-Thai translation. However, we vary the segmentation algorithm, the segmentation dictionary, and the translation dictionary. The table shows that of these three variations, the most significant improvement comes from using a large translation dictionary instead of a small one.

It may be unexpected that different segmentation algorithms and dictionaries did not have much effect on the alignment. This is not consistent with our results from Chapter 4, which compared sentences and segments in isolation. Although different segmentation

algorithms and dictionaries change similarity scores in isolation, they are not significant enough to have much effect on the actual alignment of complete texts. Here is the reason why this happens:

Language 1	Language 2
AAA BBB CCC DDD	aaabbbcccd

Different segmentations for Language 2:

- **aaa** bbbccc **ddd** (shortest minimal)
- aaabbbccc **ddd** (maximal)
- aaabbbccc **ddd** (longest)

In the above example, the similarity score for the shortest minimal match algorithm is 2 because segmented words “aaa” and “ddd” match with “AAA” and “DDD.” On the other hand, the score using the other two segmentation algorithms is 1 because only “ddd” matches with “DDD.” The border segment in question “aaabbbcccd” will still match with the corresponding sentence “AAABBBCCCD” even though the similarity scores for the second two segmentations are lower than the first one. (We will discuss different segmentation issues in Chapter 6.)

We already know from Chapter 4 that maximal matching algorithm with a small segmentation dictionary give the best result for simple calculation of similarity scores in isolation.

The results reported above in Table 5.1 show that the large translation dictionary is most effective for real-world alignment tests.

Below, in Sections 5.2 and 5.3, we will use these methods—small segmentation dictionary, maximal matching algorithm, and large translation dictionary—as we try to improve alignment performance further. We will test the effect of using techniques from information retrieval, such as stopword removal, stemming and simplification of derived forms, before the similarity comparison. Section 5.2 uses rough translation from English to Thai, and Section 5.3 uses rough translation from Thai to English.

## 5.2 Results of alignment based on English-to-Thai rough translation with input variations

In alignment methods 11, 12 and 13 using English-to-Thai rough translation, only maximal match word segmentation algorithm using the SWATH dictionary was tested. This is because we know from the previous sets of experiments (case studies in Chapter 4) that maximal match segmentation with the small segmentation dictionary performs better than other segmentation methods in alignment (as discussed in 6.1.)

Note that we were using a synthetic English-Thai dictionary created by reversing the Lexitron Thai-English dictionary (40851 words) for English-to-Thai translation. We discussed in 6.2.1 the reasons why the Lexitron English-Thai dictionary was not suitable for our purpose.

In these experiments, we did not change the segmentation method, segmentation dictionary, translation dictionary and translation direction (English to Thai). We *do* change the preliminary steps we take before calculating the similarity between English sentences and Thai segments (we use this information to decide if segments need to be moved for a better alignment).

We will test three different variations, all taken from information retrieval, and first discussed in Chapter 3. They are 1) stopword removal, 2) stemming, and 3) simplification of derived forms.

In Table 5.2 below, we contrast the performance of methods 11, 12 and 13 and the naïve length-based approach. These methods use Thai stopword removal, English stemming and simplification of English derived forms. The details of the methods are explained in section 4.2.

Table 5.2 Alignment results using English-to-Thai translation (method 11 to 13)

	Naïve length-based		Stopword removal + segmentation (11)		Stopword removal + Stemming + segmentation (12)		Stopword removal + Derivatives + segmentation (13)	
	Pre	Rec	Pre	Rec	Pre	Rec	Pre	Rec
Wanakam	0.79	0.88	0.85	0.91	0.84	0.90	0.86	0.91
Bangkok Post	0.89	0.96	0.89	0.96	0.89	0.96	0.90	0.96
Haas	0.82	0.91	0.87	0.92	0.86	0.92	0.89	0.94
LangNet	0.82	0.93	0.83	0.93	0.84	0.93	0.84	0.93
Scribner	0.88	0.94	0.88	0.94	0.90	0.95	0.90	0.95

We see from the above table that the use of stopword removal, stemming, and simplification of derived forms steadily improves the alignment performance compared to the naïve approach. This is further discussed in Chapter 6.

In our next experiment, we will change our starting condition by reversing the translation direction. We will use Thai-to-English rough translation to do the alignment.

### 5.3 Results of alignment based on Thai-to-English rough translation with input variations

In the following experiments using alignment methods 15, 16 and 17, the preliminary segmentation step was held constant, as in section 5.2: maximal match word segmentation algorithm using the SWATH dictionary. This is because we know from the previous sets of experiments (case studies in Chapter 4) that maximal match segmentation with the



small segmentation dictionary performs better than other segmentation methods in alignment (as discussed in 6.1.)

The large Lexitron Thai-English dictionary with many headwords (40851 words) was used for rough English-to-Thai translation because it also showed better results (as reported in figure 4.1).

In these experiments, we did not change the segmentation method, segmentation dictionary, translation dictionary and translation direction (Thai to English). We *do* change the preliminary steps we take before calculating the similarity between English sentences and Thai segments (we use this information to decide if segments need to be moved for a better alignment).

We will test three different variations, all taken from information retrieval, and first discussed in Chapter 3. They are 1) stopword removal, 2) stemming, and 3) simplification of derived forms.

In Table 5.3 below, we contrast the performance of methods 15, 16, and 17 to the naïve length-based approach. These methods use English stopword removal, English stemming and simplification of English derived forms. The details of the methods are explained in section 4.2.

Table 5.3 Alignment results using Thai-to-English translation (method 15 to 17)

	Naïve length-based		Stopwords removed (15)		Stopword removal + stemming (16)		Stopword removal + derivatives (17)	
	Pre	Rec	Pre	Rec	Pre	Rec	Pre	Rec
Wanakam	0.79	0.88	0.85	0.91	0.86	0.92	0.87	0.92
Bangkok Post	0.89	0.96	0.89	0.95	0.90	0.96	0.90	0.96
Haas	0.82	0.91	0.86	0.92	0.90	0.93	0.88	0.93
LangNet	0.82	0.93	0.84	0.93	0.85	0.93	0.84	0.94
Scribner	0.88	0.94	0.91	0.95	0.92	0.96	0.92	0.96

We can see from the above table that the use of stopword removal, stemming, and derived forms steadily improves the alignment performance compared to the naïve approach. This is further discussed in Chapter 6.

We also saw from Table 5.2 and Table 5.3 that the performance of the alignment based on Thai-to-English translation is slightly better overall than English-to-Thai translation. This is summarized in Table 5.4, below, which contrasts the naïve approach to methods 12 and 16.

Table 5.4 Alignment results comparing English-to-Thai translation to Thai-to-English translation (method 12 uses English-to-Thai translation and method 16 uses Thai-to-English translation)

	Naïve length-based		Stopword removal + Stemming + segmentation (12)		Stopword removal + stemming (16)	
	Pre	Rec	Pre	Rec	Pre	Rec
Wanakam	0.79	0.88	0.84	0.90	0.86	0.92
Bangkok Post	0.89	0.96	0.89	0.96	0.90	0.96
Haas	0.82	0.91	0.86	0.92	0.90	0.93
LangNet	0.82	0.93	0.84	0.93	0.85	0.93
Scribner	0.88	0.94	0.90	0.95	0.92	0.96

#### 5.4 WordNet relatedness analysis

Finally, we report on the use of WordNet relatedness analysis to align *Haas* corpus. The alignment performance with WordNet relatedness analysis was disappointing. The precision and recall were **0.62** and **0.72** respectively.

This is not unexpected. We can see from Figure 4.1 that the variations of the comparison scores are much higher than those of other methods, and the variations from the average score for the true pairs and the average score for the random pairs are overlapping. As a result, the relatedness score was not a very good guide to proper alignment of English sentences and Thai segments.

The basic problem of using WordNet is that we have too much information for English, and not enough for Thai. We end up comparing so many word senses that the final result is not very accurate. Although it might be possible to narrow down the English by part of speech tagging and/or word sense disambiguation, we do not have any such ability for Thai or other non-segmented Southeast Asian languages at present.

#### 5.5 Summary

In conclusion, we reported the effect on alignment of variations in segmentation algorithms, segmentation dictionaries, and translation dictionaries. We tested translating Thai to English and English to Thai before comparing sentences and segments. We contrasted the precision and recall figures for naïve length-based alignment method to dictionary-based alignment methods that took advantage of methods from information retrieval, such as stopword removal, stemming, and reduction of derived forms. We also saw alignment results using WordNet relatedness analysis.

We found that:

- Different segmentation algorithms and segmentation dictionaries do not have much effect on the alignment.
- A large translation dictionary performs better in the alignment.
- Thai-to-English translation performs slightly better than English-to-Thai translation.
- Stopword removal, stemming and simplification of derived forms steadily increase the performance of the alignment.
- WordNet relatedness analysis was not useful for alignment at present, but might be if better tools were available for semantic analysis prior to comparison.

In the following chapter, we will discuss our findings and insights from the experiments.

## Chapter 6

### Discussion

We now discuss our results, focusing on three main issues: segmentation, translation, and alignment. We will show many illustrative examples taken from the test data.

The discussion is outlined as follows:

Segmentation issues unique to Southeast Asian languages are discussed in section 6.1.

- First, we discuss sentence and segment comparison without using segmentation in 6.1.1.
- Second, we discuss the effect on alignment of using segmentation dictionaries of different sizes in 6.1.2.
- Third, we compare the maximal match algorithm against longest “greedy” match, and shortest minimal match algorithms in 6.1.3.
- Fourth, we discuss problems presented by typographical errors in 6.1.4.
- Finally, we discuss the challenges of proper nouns in segmentation in 6.1.5.

Translation and preparation of input text for dictionary-based alignment is discussed in section 6.2.

- First, the use of different dictionaries for rough translation is discussed in 6.2.1.
- Second, the effect of stopword removal on alignment is discussed in 6.2.2.
- Third, the effect of stemming on the alignment is discussed in 6.2.3.
- Finally, the effect of simplifying derived forms is discussed in 6.2.4.

Finally, different approaches to alignment are discussed in section 6.3.

- First, we discuss the baseline performance of the naïve length-based method in 6.3.1.
- Second, we derive insights into dictionary-based realignment of naïve output in 6.3.2.
- Finally, WordNet relatedness-based alignment is mentioned briefly in 6.3.3.

#### **6.1 Segmentation issues**

We will discuss segmentation issues unique to non-segmented Southeast Asian languages in this section.

First, we discuss sentence and segment comparison without using segmentation in section 6.1.1.

Next, in section 6.1.2, we discuss the use of segmentation dictionaries of different sizes (the actual segmentation algorithms are discussed in 6.1.3).

In section 6.1.3, we discuss how different segmentation algorithms affected the segment comparison in the dictionary-based alignment. We will compare maximal match algorithm against longest and shortest minimal match algorithms.

Finally, in sections 6.1.4 and 6.1.5, we discuss problems presented by typographical errors and proper nouns in the segmentation.

### 6.1.1 Sentence and segment comparison without segmentation

We first compared rough Thai translations of English sentence to Thai segments without using segmentation. In other words, words from English sentence were translated to Thai and the translated words were compared to non-segmented Thai segments.

We reported the performance of this method in Figure 4.1. We compared the dictionary lookup word, “AAA” for example, with the segment “BBBAAACCC.” We looked for the word “AAA” in the segment “BBBAAACCC.” If it was found (as in this case) the score was counted as one for “AAA.”

Similarity test results were often inaccurate using this approach. That is because short translated words, like อา ('aa, uncle), will accidentally match longer strings in the non-segmented Thai text.

In the following example, rough dictionary translation of an English sentence and a Thai segment are compared without segmenting the Thai text. Even though they have nothing in common, อา ('aa, uncle) accidentally matches with เอา ('aw, take) because segmentation was not used in the comparison.

Go to your mother and tell her I am coming." Aladdin ran home and told his mother of his newly found <b>uncle</b> .
---

แต่ที่ข้าเพิ่งจะถูกรถม้าชนถึงน้ำคั้นเดียวกันกับที่ชนเกลาซ่า ชนเอาเสียจนขาหักแบบนี้ (= and here have I been run down by that self-same water-cart, and my leg is broken.)
--

Therefore, in the test cases in Figure 4.1, we found that both the average similarity scores for true pairs and random pairs are high using this method. This approach does not help us tell the difference between true matches and random matches.

### 6.1.2 Different segmentation dictionaries

Instead of looking for the word “AAA” in the segment “BBBAAACCC” without segmentation as in the previous section, we applied various segmentation algorithms to “BBBAAACCC” to produce, for example, “BBB AAA CCC”. We compared exact word matches between “AAA” and the words we segmented from “BBB AAA CCC.”

We will discuss different segmentation algorithms in 6.1.3. First, we will discuss how the choice of segmentation dictionaries affects word comparison in this section.

Segmentation using a large dictionary with many compounds produced fewer words but more compounds. On the other hand, segmentation using a small dictionary with fewer compounds produced more words but fewer compounds.

Our results show that segmentation with a smaller dictionary, which contains fewer compounds, was generally better.

Thus, as seen in the comparison for method 2 and 3 in Figure 4.1, the bigger segmentation dictionary (Lexitron Thai-English) did not do well in the average similarity score comparison. When the smaller SWATH dictionary was used for segmentation as in method 5 and 7, average similarity score is higher.

We will now look at an example. In the example below, the first Thai segment was compared with the English sentence. Note that, the English word “day” was translated as ทั้งวัน (t<sup>h</sup>ǎŋ 'wan, for daylong).

<p>She had come all the way in a <b>day</b> coach; her linen duster had become black with soot and her black bonnet grey with dust during the journey.</p>	<p>ท่านนั่งรถไฟชั้นประหยัดมา<b>ทั้งวัน</b>จนผ้าเช็ดหน้ากลายเป็นสีดำจากเขม่า   สวมหมวกสีดำกลายเป็นสีเทาเพราะฝุ่นระหว่างการเดินทาง</p>
--	--

We will compare the results of looking for “day” when the Thai text is segmented using small versus large dictionaries, and then translated.

In Table 6.1, segmentations using both large and small dictionaries are shown. With a small dictionary, two headwords, ทั้ง (t<sup>h</sup>ǎŋ, whole) and วัน ('wan, day) are produced by the segmentation. One compound word ทั้งวัน (t<sup>h</sup>ǎŋ 'wan, for daylong) is produced by the segmentation using the bigger dictionary.

Table 6.1 Segmentation of ทั้งวัน with Wordcut (maximal match) using default and SWATH dictionary

Segmentation dictionary	Segmented words	Definitions
Smaller Wordcut dictionary	ทั้ง วัน	<p>ทั้ง t<sup>h</sup>ǎŋ whole</p> <p>วัน 'wan day</p>
Bigger SWATH dictionary	ทั้งวัน	ทั้งวัน t <sup>h</sup> ǎŋ 'wan for daylong

Since segmenting with the smaller dictionary gave ทั้ง (t<sup>h</sup>ǎŋ, whole) and วัน ('wan, day), we were able to match วัน ('wan, day) with the word “day” from the sentence. In effect, segmenting into smaller headwords gives us more chances for a correct match.

Similarly, in Table 6.2 segmentations for หลั่งไหล (làng lăy) using both large and small dictionaries are shown. When a smaller default dictionary from Wordcut (described in 3.2.2) is used, หลั่งไหล (làng lăy) was segmented as หลั่ง ('làng, pour out) and ไหล ('lăy, flow). With SWATH dictionary, which is bigger and has more compound words, หลั่งไหล (làng lăy) was recognized as a word that means “flow in”, without dividing into smaller head words.

Table 6.2 Segmentation of หลั่งไหล with Wordcut (maximal match) using default and SWATH dictionary

Segmentation dictionary	Segmented words	Definitions
Smaller Wordcut dictionary	หลั่ง ไหล	หลั่ง 'làng pour out ไหล 'lăy flow
Bigger SWATH dictionary	หลั่งไหล	หลั่งไหล làng lăy flow in

In the second case, smaller head word ไหล ('lăy, flow) was matched with the word “flows” from *ebbs and flows*. It was noted in this case that the compound word “flow in” was also matched when stopword removal was used as in 6.2.2.

As we have just seen, segmenting into smaller headwords helped the word comparison and produced better alignment than longer compounds, because the smaller headwords had a better chance to match correctly. It works even though the meanings are not always as accurate as the compounds produced by a bigger dictionary.

### 6.1.3 Maximal vs. longest and shortest minimal match

Different segmentation schemes provided slightly different average similarity scores in the comparison, as seen in Figure 4.1. We will see how each segmentation algorithm performed compared to each other in the alignment. We will compare the maximal match (fewest words) with longest (greedy) and shortest minimal match.

Maximal match is the standard approach [36]. It mostly produced correct segmentations (in terms of semantics) and also did slightly better realignment, as seen in Figure 4.1. Table 6.3 and Table 6.4 show segmentation results for หมอยากล่าว using maximal match and longest match. It was clear from the tables that the maximal match approach segmented the phrase correctly as หมอ ('măw, doctor), ยา ('yaa, medicine) and กล่าว ('klàaw, say) to match with the English sentence “said the physician” as shown in Table 6.5.

Table 6.3 Correct segmentation of หมอยากกล่าว using maximal match

หมอ ('มั่วว)	Doctor.
ยา ('yaa)	medicine, drug.
กล่าว ('klàaw)	to say, declare, mention.

Table 6.4 Incorrect segmentation of หมอยากกล่าว using longest match

หมอ ('มั่วว)	Doctor.
ยาก ('yâak)	1. to be hard, difficult. 2. to be wanting.
ล่า ('lâa)	to withdraw, retreat.
ว (พวว)	low consonant, pronounced w initially and finally.

Table 6.5 Highlighted words are correctly matched. Thai phrases are segmented using maximal match approach

"That is none of my business," <b>said</b> the physician;
" เรื่อง อย่าง นี้ ข้า ไม่ ถนัด ดอก "   หมอ ยา <b>กล่าว</b>

In the analysis for the following English and Thai sentences, the phrase ของดร. was segmented using both maximal match and shortest, minimal match.

It was now so nearly sunset that the chamber had grown duskier than ever; but a mild and moonlike splendor gleamed from within the vase, and rested alike on the four guests and on the doctor's venerable figure.	ยามนี้พระอาทิตย์ใกล้ตก ตัวห้องเริ่มมืดลงกว่าเดิม หากแสงนวลรำไรคล้ายแสงจันทร์ส่องเป็นประกายจากภายในแจกัน สะท้อนไปที่แขกทั้งสี่และร่างอันดูน่าเลื่อมใสของดร.ไฮเดกเกอร์
--	--

Maximal match produced correct words “doctor” and “belonging to” as shown in Table 6.6. The shortest minimal match produced the incorrect segmentation as shown in Table 6.7



Table 6.6 Segmentation of ของดร. using maximal match

ของ ('k <sup>h</sup> ɔ̌w)	owned by, belonging to.
ดร. (dɔ̌k'tɔ̌ɔ)	doctor, Dr.

Table 6.7 Segmentation of ของดร. using shortest minimal match

ขอ ('k <sup>h</sup> ɔ̌w)	to ask for, beg, request.
งด ('ŋɔ̌t)	to stop, halt, cancel.
ร (rɔ̌w)	low consonant, pronounced r initially and n finally.

#### 6.1.4 Shortest minimal match and typos

Even though maximal match is generally better as we saw in the previous section, shortest minimal match is better when there are typographical errors and/or proper nouns in the text (proper nouns will be discussed in the next section.)

The shortest minimal match still gave us the partial correct words that can be matched in the segment comparison even though the meaning was lost in some cases.

In the following example in Table 6.8, กระสับกระส่าย (kra'sàp kra'sàay, nervous) was mis-typed as กระสับกระท่าย. As a result of the typo, while segmenting the phrase กระสับกระท่ายม อง, the shortest match gave us the correct action verb “look” (มอง 'mɔ̌w) whereas the longest match gave us the incorrect combination of ยม ('yom) and อง ('ɔ̌ŋ). [See Table 6.9]

Table 6.8 Sentence with a typo in Thai

The child looked uneasily first at his mother, then at his father, who leant on his gun, looking at him with an expression of concentrated anger.	เด็กชายได้แต่กระสับกระท่ายมองแม่ก่อน แล้วหันไปมองพ่อที่ยืนเท้ากระบอกปืนจ้องเขาด้วยสีหน้าที่อึดแน่นไปด้วยความโกรธเกรี้ยว
---	---

Table 6.9 Segmentation of Thai phrase with a typo

Segmentation method	Segmented Thai phrase with a typo	Definitions of the words in question
Shortest minimal match	เด็กชาย ได้แต่ กระ สับ กระ ทำ ยม มอง แม่ ก่อน	ย 'yow low consonant  มอง 'mooŋ look
Longest match	เด็กชาย ได้แต่ กระ สับ กระ ทำ ยม อง แม่ ก่อน	ยม 'yom god of the under-world  อง 'oŋ title of royalty, equivalent to "prince."

#### 6.1.5 Shortest minimal match and proper nouns

Proper nouns are still challenges in segmentation. In the following example in Table 6.10, แซลลีถาม (Sally 't<sup>h</sup>ǎam) was segmented as แซล (Sal), ลี (ly), ถาม ('t<sup>h</sup>ǎam, ask) by shortest minimal match with syllable segmentation. Longest match, on the other hand, just gave us แซลลีถาม (Sally ask) as one word. Even though both of the segmentations cannot be said to be correct, the first approach at least correctly identified “ถาม” ('t<sup>h</sup>ǎam, ask) to be able to match with the sentence.

Table 6.10 Sentence with proper noun, Sally

Sally asked, picking up the portrait of the man with the umbrella.	แซลลีถาม พลองหยิบภาพถ่ายชายกางร่มขึ้นดู
--	---

Aladdin (อลาดดิน), on the other hand, was not so lucky. The segmentation error for Aladdin triggers false matches in the alignment. As seen in Table 6.11, Aladdin was segmented as อลาด (Alad) and ดิน ('din). ดิน ('din) happens to be a Thai word meaning earth or soil. The Aladdin story also involves many occurrence of “earth” as in the sentence, “Immediately an enormous and frightful genie rose out of the earth, saying: “What wouldst thou with me?””

Table 6.11 Sentence with proper noun, Aladdin (อลาดติน)

อลาด	no definitions
ติน ('din)	earth, soil

## 6.2 Preparing input text for dictionary-based alignment

As we will see in section 6.3, border segments broken at pre-existing spaces from the naïve length-based approach may need to be moved up or down to a neighboring sentence. We assume that segments will be more similar to the sentences they should be aligned with than to the sentences they should not be aligned with.

We used dictionary-based rough translation to compare the similarity between the boundary segments and the current sentence or preceding sentence (if the segment is the first) and/or the following sentence (if the segment is the last), as discussed in section 3.4.2.

Before doing any translation or comparison, the Thai input text was prepared using various segmentation algorithms and segmentation dictionaries, as discussed in the previous section. This is a preliminary step to making dictionary-based rough translations for sentence and segment comparison.

This section will discuss several methods of preparing both Thai and English segments for comparison.

We will first discuss the issue of using different translation dictionaries in doing rough translation in section 6.2.1. We will then discuss the use of techniques from information retrieval such as stopword removal in 6.2.2, stemming in 6.2.3 and simplification of derived forms in 6.2.4 in preparing the text for the segment comparison.

### 6.2.1 Different dictionaries for rough translation

Both English-Thai and Thai-English translation dictionaries were used for rough English-to-Thai and Thai-to-English translation.

We will first discuss the size of the dictionary with regards to the segmentation methods used. Then, we will discuss an important issue regarding the style of definitions in the dictionary.

The choice of translation dictionary depended on the method of segmentation used. A small dictionary with fewer compounds (headwords from the Haas dictionary) performed well with the shortest minimal match segmentation algorithm when compared to other methods of segmentation (Method 8 vs. 3, 4, 6 and 10 in Figure 4.1). However, a large dictionary with more compound words (Lexitron Thai-English dictionary) was necessary for longest match segmentation (Method 9 vs. 10 in Figure 4.1). If more correct and accu-

rate segmentation such as maximal match was used, a large dictionary with more compounds produced better results (Method 5 vs. 6 in Figure 4.1).

How words are defined in the dictionary is also important for semantic analysis. When using the Lexitron Thai-English and English-Thai dictionaries, it was found that the definitions in the Lexitron Thai-English dictionary were better because they were shorter.

The definitions for “high”, for example, in the Lexitron Thai-English dictionary are สูง|อาหาร|โด่ง|เกิน|ตา|เกิน|สูง|สูง. On the other hand, the definitions for the same word in the Lexitron English-Thai dictionary are much more detailed: (ภูเขา, ตึก) สูง|(สระ) เสียงสูง|ก้าวหน้า|เกียรติสูง|ซึ่งอยู่สูง|ดี|ที่สูง|แพง|เมา (ยา, สุรา) (คำแสดง)|รำเริง|รุนแรง (ลม).

We can clearly see that Lexitron Thai-English definitions were better for our purpose. The definitions from Lexitron Thai-English are short, and are exact Thai words for the equivalent English word, compared to the definitions from Lexitron English-Thai, which provide additional context and explanation. The short definitions are more like simple glosses. They are less useful for students and translators trying to understand the deeper meaning and use of words, but they are better for alignment.

As we will point out in Chapter 7, this is an important result: simple glossaries are easier to find or create than complex dictionaries.

### 6.2.2 Stopword removal

Stopword removal helped in reducing the number of false matches. Common prepositions such as *in* and *of* match too randomly and did not help with the alignment. In Table 6.12, ต่อ (‘ต่อ’) is incorrectly matched with the previous sentence because one of the definitions of ต่อ is *with*, as seen in Table 6.13.

Table 6.12 ต่อ is incorrectly matched with the previous English sentence

The trouble <b>with</b> him was that he was without imagination.	ปัญหา คือ การ ที่ ตัว เขา นั้น ปราศจาก จินตนาการ
He was quick and alert in the things of life, but only in the things, and not in the significances. (99)	เขา ตื่นตัว และ ฉับไว ต่อ สิ่ง ต่างๆ   ใน ชีวิต   แต่ แค่ สรรพ สิ่ง เท่านั้น   ไม่ ใช่ การ ตีความ

Table 6.13 Dictionary definitions of ต่อ

ต่อ ('tǔw)	build base wasp to next teach transfer decoy renew against with continue concede renew bargain
------------	--

Just as removing common English stopwords corrected some false matches, removing Thai stopwords also helped the word comparison for realignment. In the following example, when กัน ('kan) was removed before doing the word comparison, the false match of the Thai segment with the neighboring English sentence was corrected.

She hadn't the remotest idea that a book could be made of these adventures, which she had so often heard related that to her they seemed the most commonplace things in the world.	เอนิกไม่ถึงด้วยซ้ำว่าบรรดาตำนานที่เธอได้ ยินได้ฟังบ่อยครั้งจนเธอรู้สึกเหมือนมันเป็นสิ่งสามัญที่ เกิดขึ้นในโลก   จะก่อกำเนิดขึ้นเป็นหนังสือเล่มหนึ่งได้
When she tried to write, she chose material from her books, and with fresh courage she strung together stories of the Sultans in "Thousand and One Nights," Walter Scott's heroes, and Snorre Sturleson's "Kings of Romance."	เมื่อเด็กน้อยเริ่มลงมือเขียน   เธอเลือกเอาสิ่งที่พบเจอ ในหนังสือมาเขียน   เธอร้อยเรียงเรื่องราวของสุลต่าน ใน   นิทานพันหนึ่งราตรี   เหล่าพระเอกของวอลเตอร์ สก็อตต์   และเรื่องใน   คิงส์   ออฟ   โรแมนซ์   ของสนอร์   สเตอร์ลเลสสันเข้าด้วยกัน   ด้วยความกล้า อันบริสุทธิ์ของนักประพันธ์หน้าใหม่

### 6.2.3 Stemming

Stemming, usually found in information retrieval, was useful in the segment comparison. Stemming helped the following English and Thai segment to match because both *ranked* (from the English sentence) and *rank* (the dictionary lookup word for อันดับ an'dàp) stemmed to *rank*.

The cost of living in Taipei, the capital city of Taiwan, was ranked 48th in the world.
สำหรับในส่วนของค่าครองชีพของไทเปนครหลวงของไต้หวันนะครับ ก็ถูกจัดอยู่ในอันดับที่

Stemming helped the similarity test in the following example, too. The dictionary lookup for กล่าว ('klaaw) from the Thai segment is *say*. The English sentence, *said the physician*, is in the past tense. Porter's algorithm stems both *said* and *say* to *sai*.

"That is none of my business," said the physician;
--

" เรื่อง อย่าง นี้ ข้า ไม่ ถนัด ดอก "   หมอ ยา กล่าว
--

However, there are cases where Porter's stemmer could not help. For example, *leave* and *left* are stemmed to *leav* and *left*. We can see this in the example in Table 6.14. The dictionary lookup word for ทิ้ง ('t'ing) is *leave*. The English sentence uses the past tense form, *left*. There are also other cases such as *highest* and *high* where Porter's stemmer did not help. We address these in the next section.

Table 6.14 Comparing *leave* and *left*

"Here we left it," *she said.
-------------------------------

"เรา ทิ้ง มัน ไว้ ที่นี่ "   เธอ กล่าว
--

#### 6.2.4 Simplifying derived forms

As we have just seen in the previous section, there are cases such as *leave* and *left* or *highest* and *high* where stemming did not help.

Simplifying derived forms, however, helped to match some of those cases. The example from the previous section in Table 6.14 was matched when the derived form *left* was simplified to *leave*. In the following example, derived forms, *was* and *buried*, were simplified to *be* (คือ 'k'ueu) and *bury* (ฝัง 'fǎng) to match with their respective dictionary definitions.

Oh, *was that the buried treasure?
------------------------------------

โอ้ * นี้ หรือ คือ สมบัติ ที่ ถูก ฝัง ไว้
---

The derived forms list, unfortunately, is neither complete nor exhaustive. For example, it does not have either *seriously* and *serious* or *high* and *highest*. As a result, *seriously* and *serious*, *high* and *highest* were not matched either with the help of stemming or simplified derived forms.

### 6.3 Alignment methods

Three basic methods were used for alignment:

- Naïve length-based approach (6.3.1)
- Alignment with dictionary-based rough translation (6.3.2)
- Alignment based on WordNet relatedness test (6.3.3)

The results from naïve length-based approach were used as input for the dictionary-based and WordNet-based realignment.

### 6.3.1 Naïve length-based method

The naïve length-based method was used as a baseline to measure the performance of other approaches. Overall, we were able to correctly align 84% of sentence beginnings, and 92% of all segments for all five corpora.

Spaces between English did not have any effect on the alignment, because they tend to be fairly uniformly distributed in English text. Thai segments were aligned proportionately to the character counts of English sentences, and the alignment performance was the same whether we counted spaces in English or not.

Errors occur using the naïve approach when border segments are matched with the wrong neighboring sentences, as seen in Table 6.15. These cases were tackled using the dictionary-based approaches as in the following section.

Table 6.15 Alignment between English and Thai using the naïve length-based method. The numbers in the parenthesis are character counts. [Eng-Thai] is the difference in character count between English and Thai sentences.

<p>The magician cried out in a great hurry: "Make haste and give me the lamp." This Aladdin refused to do until he was out of the cave. (132)</p>	<p>เด๋มาจอมขม้งเวทก็ตะโกนออกมาอย่างรีบเร่ง (114)   "เอาตะเกียงมาให้ล้งเร็วเข้า" (80)   อลาดดินปฏิเสธ (39) (Total: 233) [Eng-Thai:-101]</p>
<p>The magician flew into a terrible passion, and throwing some more powder on to the fire, he said something, and the stone rolled back into its place. (149)</p>	<p>ไม่ทำตามจนกว่าเขาจะได้ออกจากถ้ำเสียก่อน (117)   เด๋มาจอมขม้งเวทบังเกิดโทสะ (75)   จึงโปรยผงลงในกองไฟอีก (63)   แล้วท่องบ่นอะไรบางอย่าง (69)   แผ่นหินจึงกลับเคลื่อนเข้าที่เดิม (96) (Total: 420) [Eng-Thai:-271]</p>

### 6.3.2 Dictionary-based alignment

The dictionary-based realignment was able to move the border segment (like the one in the example from Table 6.15) to the neighboring sentence if the segment belongs to that sentence (as seen in Table 6.16.) The decision to move was based on calculating a similarity measure, as discussed in 3.4.2.

Table 6.16 Alignment using naïve length-based method with the help of a dictionary-based analysis. The highlighted Thai segment was moved back up to match with the corresponding English sentence as the result of dictionary-based analysis.

<p>The magician cried out in a great hurry: "Make haste and give me the lamp." This Aladdin refused to do until he was out of the cave. (132)</p>	<p>เดมาจอมขม้งเวทก็ตะโกนออกมาอย่างรีบเร่ง   "เอาตะเกียงมาให้ลุงเร็วเข้า"   อลาaddinปฏิเสธ   <b>ไม่ทำตามจนกว่าเขาจะได้ออกจากถ้ำเสียก่อน</b></p>
<p>The magician flew into a terrible passion, and throwing some more powder on to the fire, he said something, and the stone rolled back into its place. (149)</p>	<p>เดมาจอมขม้งเวทบังเกิดโทสะ   จึงโปรยผงลงในกองไฟอีก   แล้วท่องมนอะไรบางอย่าง   แผ่นหินจึงกลับเคลื่อนเข้าที่เดิม</p>

There are cases in which dictionary-based alignment could not help move the border segments to proper sentences. In the following example, the second and the third sentences are both talking about “fire.” Even though the two highlighted Thai segments should go down to match with the third sentence, the segments are equally similar to the current sentence because the word “fire” appears in both sentences.

<p>But the brute had its instinct.</p>	<p><b>แต่สัตว์เดรัจฉานมีสัญชาตญาณ</b> มันสัมผัสถึงความรู้สึกคุกคามอันรางเลือนซึ่งคุกคามมันจนต้องเดินแอบติดเท้าเจ้าของ</p>
<p>It experienced a vague but menacing apprehension that subdued it and made it slink along at the man's heels, and that made it question eagerly every unwonted movement of the man as if expecting him to go into camp or to seek shelter somewhere and build a fire.</p>	<p><b>และคอยสงสัยทุกการเคลื่อนไหวที่ผิดแปลกของเขา</b>   ราวกับคาดหวังให้เขาเข้าไปอยู่ในแคมป์หรือ <b>หาที่พักพิงสักแห่งและก่อไฟสักกอง</b>   <b>เจ้าสุนัขรู้จักไฟแล้ว</b>   <b>และมันอยากได้ไฟด้วย</b></p>
<p>The dog had learned fire, and it wanted fire, or else to burrow under the snow and cuddle its warmth away from the air.</p>	<p>หรือไม่กี่ขอบุดโพรงอยู่ในหิมะและขุดตัวกลมให้อบอุ่นจากอากาศหนาวก็ยิ่งดี</p>

### 6.3.3 WordNet relatedness-based alignment

Where simple dictionary-based matches cannot help, it might be useful to do more sophisticated analysis for similarity testing.

In the following example in Table 6.17, WordNet relatedness test was able to analyze correctly that the Thai segment “และช่วยชีวิตตัวเองได้” and the corresponding English sentence were similar. The dictionary lookup words for the Thai segment were highly “related” to the words from the English sentence even though there were no exact word matches; for example, “oneself” from dictionary lookup and “himself” from the sentence.




Table 6.17 WordNet relatedness analysis

Well, here he was; he had had the accident; he was alone; and he had saved himself.	ตอนนี้เขาอยู่ตรงนี้แล้วไงล่ะ เขาเจออุบัติเหตุ ต้องอยู่คนเดียว และช่วยชีวิตตัวเองได้
---	---

However, even though WordNet can be helpful, it is nearly as likely to hurt. In Table 6.18, we can see that WordNet relatedness analysis incorrectly identified the highlighted Thai phrase to be more similar to the following English sentence than to the current one even though that was not the case.

Table 6.18 Incorrect WordNet relatedness analysis

There was a reed that grew alongside a stream.	
And not far above the stream there was a banyan tree, too.	ยังมีต้นอ้อขึ้นอยู่ข้างลำธารแห่งหนึ่ง   และเหนือลำธารขึ้นไปไม่มาก มีต้นไทรขึ้นอยู่ด้วย



In effect, WordNet provides too much information. If we could narrow it down—if we knew more about the proper part of speech and sense of the Thai and English words—the score would be much more reliable. However, while it might be possible to get some of this analysis for English, it is generally unavailable for Thai and other non-segmented Southeast Asian languages at this point.

## Chapter 7

### Conclusions and Recommendations

The goal of this thesis was to investigate problems of aligning non-segmented Southeast Asian texts with English texts. We used Thai for our experiments because at present it has the most available resources. Our intention, however, was to help establish methodology and baseline performance for approaches to bitext alignment that might reasonably be used with Burmese, Khmer, Lao, and similar low-resource languages.

The following recommendations and insights that are relevant to other Southeast Asian languages were achieved from the experiments. They include:

- An alignment metric that takes into account the sentence boundary detection problem applicable to Southeast Asian languages (section 7.1),
- Naïve length-based method as a baseline (section 7.2),
- Breaking a Southeast Asian language text into words and/or compounds (section 7.3),
- Translating a Southeast Asian language to and from English (section 7.4),
- Finding the similarity between Southeast Asian text segments and English sentences in order to decide whether the segments needed to be moved during re-alignment (section 7.5).

#### 7.1 Metrics

Simple precision and recall based on full sentence matches may be appropriate for Western languages, in which sentence detection is relatively easy. But it is misleading for alignment of Southeast Asian languages (with English) because it does not reflect the difference between completely incorrect pairs and near misses.

We devised an alternative method. The number of correctly aligned *first* segments is used as a *precision* score to give an intuitively sensible idea of how many sentences are correctly aligned. The number of correctly aligned segments *overall*, in turn, is the basis of our *recall* score. It gives us a good sense of just how far off the misaligned sentences really are.

This is easy to calculate, and produces a clear and intuitive result. It also addresses the sentence boundary detection problem applicable to many Southeast Asian languages such as Thai, Khmer and Lao.

#### 7.2 Length-based method as a baseline

Length-based alignment method has not been used to align non-segmented Southeast Asian languages with English. However, it has widely been used with European languages.

Paragraphs in bitext must be segmented into sentences (or segments if sentence boundaries are not clear) to use this method. For languages like Thai, Khmer and Lao where sentence boundaries are not well-defined, it is necessary to detect sentence boundaries in advance (in our case, by using pre-existing spaces).

The method is perfectly suitable for languages like Burmese where sentence boundary is well-defined (as in English), and will perform reasonably well.

Length-based method is the simplest and most fundamental. It is language independent and can be easily implemented. Since it produces a reasonably good result, it is recommended to use as a baseline for the alignment of Southeast Asian language bitexts.

### **7.3 Segmentation**

Southeast Asian language text needs to be segmented into words and/or compounds to use a dictionary-based alignment method. Segmenting Southeast Asian text correctly into words and/or compounds is not an easy and trivial task.

Even though none of the existing algorithms can be said to be correct, maximal match segmentation algorithm with a dictionary consisting mainly of headwords works best for this purpose. It gave mostly correct segmentations except for proper nouns and typographic errors.

The maximal match algorithm is well understood and easy to implement. Once implemented for a Southeast Asian language, it could be used for another by switching the dictionary.

A segmentation dictionary is also easy to compile or extract. There are many monolingual or bilingual Southeast Asian language dictionaries available for language learners. Headwords from such a dictionary can be extracted to use as a segmentation dictionary.

### **7.4 Translation to and from English**

A Southeast Asian language needs to be translated into English or vice versa to compare the rough translation to the actual text for the alignment.

We found that the best dictionary for this purpose had two main characteristics:

- many headwords
- short definitions

Having many headwords clearly will increase the chance to find the word we want to translate. Having short definitions also increases the chance to match with the words from the actual text; for example, “*saunter*” vs. “*walk in a slow relaxed way.*”

Dictionaries of Southeast Asian languages, which are readily available off the shelf, could be used for this purpose. They usually have many headwords with short definitions.

## 7.5 Finding similarity between Thai segments and English sentences

After doing rough translation, the translated words are compared with the actual text to decide if the boundary segments are more similar to the current sentence or a neighboring sentence. Exact word matches are performed to check the similarity.

We found that stopword removal, stemming and simplification of derived forms helped the word comparison.

Stopword removal helped reduce false positives. Compiling such a stopword list for a Southeast Asian language can be done by analyzing a large corpus to look for frequent words and a basic knowledge of the language.

Stemming helped to improve the chance for word matches by reducing words to an approximation of their ‘root’ forms. There are publicly available tools for stemming English (for example, Porter’s stemmer).

Simplification of derived forms also helped to improve the word comparison by normalizing the words to their ‘root’ forms. There are readily available lists of derived forms for English. An example of a Southeast Asian language for which a list of derived forms can be compiled is Burmese. Burmese has a list of prefixes and suffixes to form derivations [47]. For example, adding a prefix အ (a) to verbs and adjectives forms nouns or adverbs. ကြည်ညို (ဖျိ ဂဲ, revere) becomes a noun, အကြည်ညို (အ ဖျိ ဂဲ, reverence), after the prefix is added. There are other ways to make derived words in Burmese; for example, adding a rhymed syllable to a word.

It is recommended to apply these techniques (in both English and Southeast Asian languages if they are available) to enhance dictionary-based alignment.

## 7.6 Conclusion

The methodology in this thesis can be used to align bitext corpora of languages with few resources such as Burmese, Khmer or Lao. Since little advanced research on computational linguistics has been done for these languages, this methodology provides a mechanism to align bitext corpora using the resources that are available off the shelf, with some Southeast Asian language-specific modifications of input text such as word segmentations and using approaches from information retrieval such as stopword removal, stemming and simplification of derived forms.

## References

- [1] W. A. Gale and K. W. Church, “A program for aligning sentences in bilingual corpora,” in *Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics (ACL)*. Morristown, NJ, USA: Association for Computational Linguistics, 1991, pp. 177–184.
- [2] D. Varga, L. Németh, P. Halácsy, A. Kornai, V. Trón, and V. Nagy, “Parallel corpora for medium density languages,” in *Proceedings of the Recent Advances in Natural Language Processing*, 2005, pp. 590–596.
- [3] “Aligned Hansards of the 36th Parliament of Canada,” September 2007. [Online]. Available: <http://www.isi.edu/natural-language/download/hansard/>
- [4] S. Roukos, D. Graff, and D. Melamed, *Hansard French/English*. Philadelphia: Linguistic Data Consortium, 1995.
- [5] P. Koehn, “Europarl: A parallel corpus for statistical machine translation,” in *MT Summit*, 2005. [Online]. Available: <http://www.statmt.org/europarl/>
- [6] “The English-Norwegian parallel corpus,” September 2007. [Online]. Available: <http://www.hf.uio.no/ilos/forskning/forskningsprosjekter/enpc/>
- [7] “The English-Swedish parallel corpus,” September 2007. [Online]. Available: <http://www.englund.lu.se/corpus/corpus/espc.html>
- [8] “Hunglish corpus,” August 2007. [Online]. Available: <http://mokk.bme.hu/resources/hunglishcorpus/indexhtml>
- [9] Xiaoyi Ma, *Hong Kong Parallel Text*. Philadelphia: Linguistic Data Consortium, 2004.
- [10] N. Collier and K. Takahashi, “Sentence alignment in parallel corpora: The Asahi corpus of newspaper editorials,” Centre for Computational Linguistics, UMIST, Manchester, Tech. Rep. 95/11, October 1995.
- [11] Dinh Dien and Hoang Kiem, “POS-Tagger for English-Vietnamese bilingual corpus,” in *Proceedings of the HLT-NAACL 2003 Workshop on Building and using parallel texts*. Morristown, NJ, USA: Association for Computational Linguistics, 2003, pp. 88–95.
- [12] SEALang. (2007, October) Southeast Asian languages library. [Online]. Available: <http://library.sealang.net>
- [13] Wanakam. (2007, October) Wanakam world classics in Thai. [Online]. Available: <http://www.wanakam.com>
- [14] M. Barang. (2007, October) Thai fiction in translation. [Online]. Available: <http://www.thaifiction.com>

- [15] “Bangkok Post, Learning Post,” September 2007. [Online]. Available: <http://www.bangkokpost.net/education/index.htm>
- [16] R. K. Headley, *Cambodian-English dictionary*, 1st ed. Washington D.C: Catholic University of American Press, 1977.
- [17] “Asia Online,” October 2008. [Online]. Available: <http://asiaonline.net>
- [18] I. D. Melamed, “Bitext maps and alignment via pattern recognition,” *Computational Linguistics*, vol. 25, no. 1, pp. 107–130, 1999.
- [19] P. Danielsson and D. Ridings, “Practical presentation of a “Vanilla” aligner,” August 2007. [Online]. Available: <http://nl.ijs.si/telri/Vanilla/doc/ljubljana/>
- [20] P. F. Brown, J. C. Lai, and R. L. Mercer, “Aligning sentences in parallel corpora,” in *Proceedings of the 29th annual meeting on Association for Computational Linguistics*. Morristown, NJ, USA: Association for Computational Linguistics, 1991, pp. 169–176.
- [21] M. Kay and M. Röscheisen, “Text-translation alignment,” *Computational Linguistics*, vol. 19, no. 1, pp. 121–142, 1993.
- [22] P. Fung and K. W. Church, “K-vec: a new approach for aligning parallel texts,” in *Proceedings of the 15th conference on Computational linguistics*. Morristown, NJ, USA: Association for Computational Linguistics, 1994, pp. 1096–1102.
- [23] F. Nevado, F. Casacuberta, and E. Vidal, “Parallel corpora segmentation by using anchor words,” in *Proceedings of EACL 2003 workshop on EAMT, 11th Conference of the European Chapter of the Association for Computational Linguistics*, April 2003.
- [24] M. Simard, G. F. Foster, and P. Isabelle, “Using cognates to align sentences in bilingual corpora,” in *CASCON '93: Proceedings of the 1993 conference of the Centre for Advanced Studies on Collaborative research*. IBM Press, 1993, pp. 1071–1082.
- [25] T. Utsuro, H. Ikeda, M. Yamane, Y. Matsumoto, and M. Nagao, “Bilingual text, matching using bilingual dictionary and statistics,” in *Proceedings of the 15th conference on Computational linguistics*. Morristown, NJ, USA: Association for Computational Linguistics, 1994, pp. 1076–1082.
- [26] S. Piperidis, H. Papageorgiou, and S. Boutsis, “From sentences to words and clauses,” in *Parallel Text Processing: Alignment and use of translation corpora*, J. Véronis, Ed. Paris: Kluwer Academic, 2000, ch. 6.
- [27] K. Hofland, “A program for aligning English and Norwegian sentences,” in *Proceedings of the ACH/ALLC Conference*, July 1995.

- [28] G. A. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K. J. Miller, "Introduction to WordNet: an on-line lexical database," *International Journal of Lexicography*, vol. 3, no. 4, pp. 235–244, 1990.
- [29] T. Pedersen, S. Patwardhan, and J. Michelizzi, "WordNet::Similarity - Measuring the relatedness of concepts," in *Proceedings of Fifth Annual Meeting of the North American Chapter of the Association for Computational Linguistics*, May 2004, pp. 38–41.
- [30] K. Stribley, "Syllable based Dual Weight algorithm for line breaking in Myanmar unicode," October 2007. [Online]. Available: <http://thanlwinsoft.org/ThanLwinSoft/MyanmarUnicode/Parsing/>
- [31] Yuen Poowarawan, "Dictionary-based Thai syllable separation," in *Proceedings of the ninth Electronics Engineering Conference*, 1986.
- [32] V. Sornlertlamvanich, "Word segmentation for Thai in a machine translation system," National Electronics and Computer Technology Center, Bangkok, Tech. Rep., 1993.
- [33] A. Krawtrakul, C. Thumkanon, Y. Poovorawan, and M. Suktarachan, "Automatic Thai unknown word recognition," in *Proceedings of the Natural Language Processing Pacific Rim Symposium 1997 (NLPRS'97)*, 1997.
- [34] W. Aroonmanakun, "Collocation and Thai word segmentation," in *Joint International Conference of SNLP-Oriental COCOSDA*, 2002.
- [35] V. Sornlertlamvanich, T. Charoenporn, and H. Isahara, "ORCHID: Thai part-of-speech tagged corpus," National Electronics and Computer Technology Center, Bangkok, Tech. Rep. TR-NECTEC-1997-001, 1997. [Online]. Available: [www.links.nectec.or.th/orchid/](http://www.links.nectec.or.th/orchid/)
- [36] S. Meknavin, P. Charoenpornasawat, and B. Kijirikul, "Feature-based Thai word segmentation," in *Proceedings of the Natural Language Processing Pacific Rim Symposium 1997 (NLPRS'97)*, 1997, pp. 41–46.
- [37] P. Mittrapiyanuruk and V. Sornlertlamvanich, "The automatic Thai sentence extraction," in *Proceedings of the fourth symposium on Natural Language Processing*, 2000, pp. 23–28.
- [38] P. Charoenpornasawat and V. Sornlertlamvanich, "Automatic sentence break disambiguation for Thai," in *The Nineteenth International Conference on Computer Processing of Oriental Languages (ICCPOL)*, May 2001, pp. 231–235.
- [39] LangNet. (2008, August) LangNet: multilingual advanced learning online. [Online]. Available: <http://www.langnet.org>
- [40] Vee Satayamas and Warat Yingsalee. (2008, August) Thaiwordseg, word segmentation for Thai language. [Online]. Available: <http://thaiwordseg.sourceforge.net>

- [41] NECTEC. (2008, August) Lexitron: Thai-English electronic dictionary. [Online]. Available: <http://lexitron.nectec.or.th/index1.php>
- [42] Paisarn Charoenpornasawat. (2008, August) Software: SWATH - Thai word segmentation. [Online]. Available: <http://www.cs.cmu.edu/paisarn/software.html>
- [43] D. James. (2008, September) Lingua::EN::Stopwords. [Online]. Available: <http://search.cpan.org/splice/Lingua-EN-Segmenter-0.1/lib/Lingua/EN/StopWords.pm>
- [44] M. F. Porter, “An algorithm for suffix stripping,” *Program*, vol. 14, no. 3, pp. 130–137, 1980.
- [45] C. Jacquemin, J. L. Klavans, and E. Tzoukermann, “Expansion of multi-word terms for indexing and retrieval using morphology and syntax,” in *ACL-35: Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics*. Morristown, NJ, USA: Association for Computational Linguistics, 1997, pp. 24–31.
- [46] T. Pedersen. (2008, October) Wordnet::SenseRelate. [Online]. Available: <http://senserelate.sourceforge.net>
- [47] J. Okell, *A reference grammar of colloquial Burmese*. London: Oxford University Press, 1969.



## Appendix A

### Stopword Lists

#### Stopword list from Lingua::EN::StopWords

a about above across adj after again against all almost alone along also  
although always am among an and another any anybody anyone anything anywhere  
apart are around as aside at away be because been before behind being below  
besides between beyond both but by can cannot could deep did do does doing done  
down downwards during each either else enough etc even ever every everybody  
everyone except far few for forth from get gets got had hardly has have having  
her here herself him himself his how however i if in indeed instead into inward  
is it its itself just kept many maybe might mine more most mostly much must  
myself near neither next no nobody none nor not nothing nowhere of off often on  
only onto or other others ought our ours out outside over own p per please plus  
pp quite rather really said seem self selves several shall she should since so  
some somebody somewhat still such than that the their theirs them themselves  
then there therefore these they this thorough thoroughly those through thus to  
together too toward towards under until up upon v very was well were what  
whatever when whenever where whether which while who whom whose will with  
within without would yet young your yourself

#### Reduced English stopword list

a an the of by but for that this here there other another and or in on up at to down he she  
with not it

#### Thai stopword list

กัน|ทาง|การ|ความ|ของ|ที่|กับ|และ|โดย|ด้วย|หรือ|คน