NOTE TO USERS

This reproduction is the best copy available.

PARALLEL TEXT MAPPING OF WEB-BASED BILINGUAL CORPUS MATERIALS

By
Qibo Zhu

A thesis submitted to
the Faculty of Graduate Studies and Research
in partial fulfilment of
the requirements for the degree of
DOCTOR OF PHILOSOPHY

Institute of Cognitive Science
CARLETON UNIVERSITY

Ottawa, Ontario
June, 2009

Your file Votre référence
ISBN: 978-0-494-60127-3
Our file Notre référence
ISBN: 978-0-494-60127-3

Abstract

The chief objective of this thesis is to design and develop a Bitext Mapping Intelligent Agent (BMIA), a computational model that can be used to pair and compare translations texts. There are two main components in BMIA. The first component is the StatCan Daily Translation Extraction System (SDTES) which automatically extracts translations from web-based materials to construct the StatCan Daily Corpus (SDC). At the same time, a translation concordance system (TransConcord) has been developed to provide ready access to SDC and other bilingual corpora. The second component of BMIA is the StatCan Bilingual Text Comparison System (TextComp) that aims at aligning and comparing bilingual texts for translation discrepancy detection and Translation Correspondence Profiling (TCPro). To deal with potentially noisier data sets in the translation checking process, different text mapping algorithms have been designed to parse the input texts, align them, and scan through them to detect translation discrepancies. In order to give a more detailed picture of translation correspondences, TextComp maps translations at a more fine-grained level: the translation constituent level. A TCPro scaling metric is designed to compute the TCPro score for each aligned segment pair so that levels of translation correspondence can be estimated and distinguished. This scale-based view can help in identifying correspondence deviations and objectively assessing the faithfulness of translations. The two component systems in BMIA not only support human translators, but also shed light on machine translation, translation studies, and translation quality assessment.

Acknowledgements

I would like to thank my supervisor Ash Asudeh who, over the past few years, spent a lot of time discussing with me problems and issues relating to bilingual text alignment. He read, commented on and corrected my drafts, and guided me through the thesis project right from setting the thesis topic to modifying the last version. I could not have brought this piece of work to a timely finish without his help and encouragement.

I acknowledge with gratitude the support I received from the other members of the thesis committee and the examination board. Diana Inkpen, David Sankoff, Joel Martin all gave me very valuable input on various parts of the thesis. I feel lucky to have all these people in my thesis committee to help me in related areas such as algorithm design, statistical data analysis, and bilingual text alignment. I am also grateful to Pierre Isabelle and Guillaume Gentil for their insightful comments and detailed suggestions.

The Institute of Cognitive Science, Carleton University, offered me a very friendly and supportive atmosphere for research. I owe my thanks to many professors, staff members, and fellow graduate students of the institute. Andrew Brook, Jim Davies, and Jo-Anne LeFevre discussed with me the thesis topic, and helped me shape some of the ideas in the thesis. John Logan, Ida Toivonen, Mark MacLeod, and Robert West all helped me in one way or another in the process of the preparation of this thesis. The kindness and willingness of Colleen Fulton, Lianne Dubreuil, and May Hyde to assist me have had a very positive influence on my work efficiency at Carleton.

My summer time in 2007 at the FrameNet group at University of California, Berkeley, was extremely valuable to me. Charles J. Fillmore, who guided my first bilingual text

alignment project a few years ago, gave me insightful comments about the StatCan Translation Extraction System and the resultant aligned StatCan Daily Corpus. He also arranged for me to meet, talk with some professors and researchers at ICSI about the StatCan bitext datasets. The feedback helped towards refining the corpus data analysis part of the thesis. To Charles J. Fillmore, Collin Baker, and other researchers at ICSI, I extend my gratitude and thanks.

My employer Statistics Canada funded my Ph.D research at Carleton. I am thankful to management and colleagues at StatCan. Michael Wolfson and Wayne Smith, two Assistant Chief Statisticians of Statistics Canada, met with me at different occasions to discuss my research and the StatCan TextComp system. Other people of the senior management such as Vicky Crompton, Jonathan Massey-Smith, Franscoir Borde, Eric St. John, Martine Grenier, have all been very supportive of my research at Carleton. My managers Grant Niman and Loui Massicotte did a lot of things to make my work schedule flexible but productive. Current management of CLSD played an important role in pushing for TextComp as an important tool of quality assurance at StatCan. Many colleagues from my home division and other divisions, such as the official translations division and the client services division, are involved in talks, discussions and activities of testing the TextComp system, evaluating it, and improving it. Without the full support of these people, particularly people like Loui Massicotte, this thesis would have taken me much longer.

I would like to thank my family and friends near and afar who have never lost faith in me and in this long-term project.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

This thesis is about exploring bilingual text mapping techniques to build translation corpora and to detect potential translation discrepancies and problems. A Bitext Mapping Intelligent Agent (BMIA) model was designed to map and compare web-based bilingual materials. There are two major components in the BMIA model. The first component is the StatCan Daily Translation Extraction System (SDTES) which contains protocols and algorithms for automatically extracting translations from officially published web-based materials. The extracted bitext segments are assembled to construct the StatCan Daily Corpus (SDC). A translation concordance system (TransConcord) with novel features of KWIC search in bilingual information retrieval has been developed to provide ready access to SDC and other bilingual corpora. The StatCan Bilingual Text Comparison System (TextComp) is the second important component in BMIA. It aims at translation discrepancy detection and Translation Correspondence Profiling (TCPro). New algorithms and methods are proposed in TextComp for aligning and comparing bilingual texts that are in the process of being translated, edited or checked prior to publication. At the same time, a TCPro scaling metric is designed to compute the TCPro score for each aligned text segment pair. Together with different display schemes and view modes, the TCPro score can offer a better understanding of the levels of formal correspondences between the two halves of texts in aligned translation segment pairs.

Bilingual texts are the building blocks of bilingual corpora. Bilingual corpus building and bilingual text mapping are areas of research in corpus linguistics and machine translation that have been facing issues and challenges on different participating frontiers of cognitive science. In the 1950s, the advent of Chomsky's revolution (Chomsky 1957; Searle 1972) brought to the fore a very general and basic question for philosophers, psychologists and linguists alike: can corpus materials be a good source for the study of language and language acquisition? Throughout the development period of corpus linguistics, the major challenges for AI researchers in this field have been to seek the best algorithms or strategies for building, analyzing, and exploring corpus data. Recently, with the increasingly widespread availability of computing infrastructure, a new venue has opened for people to consult large collections of bilingual texts on-line. As researchers who are interested in bilingual text mining and corpus linguistics, we ask ourselves an important question: web materials are different from conventional hand-picked corpus materials, but can we tap the rich resources of the web for the building and processing of bilingual corpora? For web-based bilingual materials, can we pair and compare the bilingual text segments in a very efficient way so that the results can be put to direct use in such practical applications as bilingual corpus building, translation analysis and translation checking (Isabelle *et al.* 1993), translation correspondence profiling, machine translation, and translation quality assurance?

It is these questions and others that have motivated the plan for the present investigation. The BMIA model is designed to serve a dual purpose: extracting translation pairs from published web-based materials for parallel corpus building, and measuring the degree of translation correspondence and detecting translation

discrepancies for pre-publication translation quality assurance. The inspiration in designing such a software agent system comes from the philosophy of collaboration of man and machine. In a parallel corpus building setting, computers are supposed to boost human productivity in such a way that translations can be extracted and checked in a batch mode by computers automatically and not piecemeal by human translators manually. In the translation discrepancy detection process, some of the tedious and less complicated cognitive tasks can be delegated to the software agent system for execution because "... there is much about it [translation] that is mechanical and routine and, if this were given over to a machine, the productivity of the translator would not only be magnified but his work would become more rewarding, more exciting, more human" (Kay 1997).

Mapping assembled bilingual texts to build a bilingual corpus is one of the most challenging problems in natural language processing (NLP), particularly in bilingual text mining. The BMIA model in this thesis will make contributions to corpus linguistics, to machine translation (MT) and to machine assisted translation (MAT) in the following ways.

1) The StatCan Daily Translation Extraction System (SDTES) in BMIA can be used to extract translations from texts of current web formats --- HTML pages, XML documents and other texts with markups for styles of presentation or for meta information. This will have implications in web content mining that aims at building much needed domain-specific parallel corpora directly from bilingual websites.

2) One of the intended goals of BMIA is to automatically detect translation discrepancies. The TextComp system of BMIA not only represents a novel, feasible,

independent, and fast bilingual text mapping system that can be readily used for bilingual text alignment and comparison, but also includes innovative and integrated algorithms that can shed light on models of bilingual corpus data processing, corpus-based MT, example-based MT, machine translation evaluation, and translation studies. There have been good text alignment models such as the Gale-Church length-based algorithm, the IBM alignment models etc. However, most of them cannot be directly applied to web-based bilingual text mapping or on-line bilingual text comparison. They either need preprocessing or other lexical knowledge, or are computationally very expensive for real-time web-based applications. The algorithms proposed in this thesis such as AMS, *k*-band, linear regression forecasting, noise filtering in word correspondence mapping, and translation constituent annealing are targeted at robust and fast mapping of bilingual texts. They are mostly unsupervised, and do not rely on external lexical knowledge bases such as bilingual dictionaries or part of speech parsers. They are designed for returning reliable results for practical, web-based implementations that can compete with more sophisticated text alignment models.

3) The Translation Correspondence Profiling (TCPro) component in BMIA is a novel attempt in automatically mapping the associative links in bilingual texts and in objectively measuring the structural and textual correspondences of these links. The profiling of correspondence permits the projection of information from one language to the other and thus makes the translation texts comparable. The model can serve the immediate needs of translation discrepancy detection. In addition, the capacity to establish links of translation correspondence at different levels and to scale the grades of correspondence can open up new fields of research in translation studies, in translation

analysis for quality assessment, in machine translation evaluation and other NLP tasks because "Being able to establish links between two languages allows for transferring resources from one language to another" (Dorr and Monz 2004). For example, TCPro can provide fresh sources of mapped and scaled translation texts for studying the spectrum of different translation styles ranging from literal translation style on one end to free translation style on the other. It can also offer detailed information and new insights about what constitute important parameters towards more objective assessment of human translation and more accurate evaluation of machine translation.

4) A parallel corpus, the StatCan Daily Corpus (SDC), has been compiled using the BMIA model. Some other bilingual corpora have also been built with the same model. Together, these corpora contain tens of thousands of well-aligned, clean translation pairs that can serve as gold standards or good reference data for research in language analysis, translation studies, word sense disambiguation, and terminology building. They can also be used as training data in machine learning, bilingual information retrieval, machine translation and machine assisted translation. At the same time a new bilingual concordance search system (TransConcord) which is based on current search engine technologies has been developed to offer direct and easy access to bilingual corpus materials. Large bilingual corpora equipped with a powerful and convenient translation concordance search system can offer great potential for exploring, mining, and analyzing collections of bitexts.

1.1 Organization of the Thesis

This thesis comprises five chapters, the first chapter being this introduction.

Bitext data is a type of text corpus data, no matter how large or how small the corpora are. Mapping text correspondences of existing translations is closely related to assembling and processing collections of translations to build bilingual corpora. The second chapter discusses corpus development from early word counts to modern bilingual corpus building. It introduces some historical background and theoretical issues in constructing corpora and using corpus data. In this chapter, we will examine characteristics of bilingual corpora, and problems and challenges of bilingual corpus building. We will also describe actual applications of parallel corpora in such fields as foreign language teaching, terminology extraction and word sense disambiguation, human translation and translation studies, machine translation and machine assisted translation.

The third chapter is a review of previous work in bilingual text alignment including some core algorithms and techniques which underlie parallel text mapping. Many methodologies have been developed for parallel language alignment and bilingual text mapping since the early 1990's. This chapter will focus on those statistical, lexical, and hybrid approaches that are influential and are related to our approaches in BMIA. They include the Gale and Church length-based alignment model, the IBM 1 statistical alignment algorithm, the Kay and Röscheisen lexical alignment algorithm, and the K-vec algorithm. Principal methods that are shared in many alignment models and key factors that can affect the performance of the alignment algorithms are also discussed.

Chapter four covers the general framework of BMIA. In this chapter, two major components of BMIA are described. The first component is the StatCan Daily Extracting System (SDTES), a software system that extracts translations from web-based materials,

aligns the translations by text segments, and compiles a bilingual corpus. The end result of SDTES is the bilingual StatCan Daily Corpus (SDC). In this chapter we will describe how the Gale and Church algorithm and the K-vec algorithm are integrated with our own Acceptable Matching Sequence (AMS) method to align text segments, extract cognates, and locate misaligned areas. The second major component is the Bilingual Text Comparison System (TextComp) for translation discrepancy identification and Translation Correspondence Profiling (TCPro). A linear regression forecasting model is built on the basis of the statistical data from the StatCan Daily Corpus. Modified dynamic programming algorithms and the linear regression forecasting model are employed at two levels of translation text alignment: one at the paragraph level, and the other at the text segment level. When text segments are aligned, BMIA compares the translation pairs for structural and lexical clues to see if there are translation discrepancies. In addition, algorithms are designed to establish translation constituent correspondences for translation correspondence profiling. In this chapter, these algorithms and strategies will be explored together with the steps and procedures for bilingual text comparision, for translation discrepancy identification, and for translation correspondence profiling.

The fifth chapter contains the conclusion with proposals for future work. Strong points and limitations of BMIA will be discussed. Potential applications will be examined of the subsystems that are proposed and developed in this thesis. This chapter will also look at some BMIA related work that is open to further investigation and research.

Chapter 2

From Early Word Counts to Modern Corpora

> Not that long ago, all linguistics was corpus linguistics.
>
> (Malouf 2006)

In this chapter, we will introduce some key corpus research work from word frequency counts by Thorndike (1921) to large collections of natural language texts such as the British National Corpus (BNC). We will discuss the theoretical issues that played an essential role in checking the progress of corpus linguistics in the 1950s. The bilingual corpus is a special type of natural language corpus and mapping bilingual texts is closely related to building bilingual corpora. We will explain the principles, problems and challenges of compiling bilingual corpora in this chapter. We will also examine the applications of bilingual corpora from different perspectives such as language teaching, word sense disambiguation, translation studies and machine translation.

Corpus development has experienced its ups and downs in the past. As language corpora can be used for many linguistic and multidisciplinary inquiries and as corpus-based analysis has the potential to yield highly interesting, and often surprising, new insights about language studies and other sciences, corpus-based studies had become a hot testbed for theoretical hypotheses and methods in linguistics, information processing, artificial intelligence and other fields of cognitive science. For about 20 years before 1957, corpus linguistics was a strong and flourishing field of study. Then corpus

development suffered some setbacks because of Chomsky's revolution. In the 1980s, interests in corpus linguistics revived with the advent of advances in computer science. Since then, merits and benefits of corpus-based studies have become increasingly well-recognized, and more and more linguists and researchers in natural language processing have become aware that facts and patterns manifested in corpora "are not minor details of linguistic behavior" (Sankoff 2001). Now, corpus linguistics has become an important part of computational linguistics.

2.1 Historical Background of Corpus-Based Research

Early text collection efforts mostly focused on the study of words and graphemes (Zipf 1935). In the pre-computer period, people collected linguistic data manually on tens of thousands of paper slips. Word frequency lists and collocation lists were extracted from the collected texts for the study of natural language usage, language acquisition, lexicography, and language pedagogy (Thorndike 1921; Palmer 1933; Thorndike and Lorge 1938; Fries and Traver 1940; Thorndike and Lorge 1944). During the 1940s and early 1950s, empiricism was at its peak, dominating cognitive science disciplines ranging from language learning theory to information theory. It was common practice in linguistics to classify words not only on the basis of their meanings but also on the basis of their frequency and their co-occurrence with other words. To some linguists, the corpus or selected collection of texts was seen as the sole source of evidence in the formation of linguistic theory (Leech 1991). For such linguists, the evidence from sufficiently large body of naturally occurring language data was primary and necessary, and intuitive evidence was sometimes rejected altogether. Some of the word counts and

frequency lists produced during this period are still very influential for general linguistics, psycholinguistics, applied linguistics and natural language processing.

Interest in corpus building faded in the late 1950s and early 1960s as the cognitive revolution gained momentum. At that time, Noam Chomsky changed the direction of linguistics away from empiricism towards rationalism (Searle 1972). Although the criticisms of empiricism discredited corpus linguistics, some linguists continued their work in corpus development. The ambitious *Survey of English Usage* was planned in 1960 (Quirk 1960), and work on the flagship American English corpus, the Brown Corpus, began in 1961 (Kucera and Francis 1967). Some other important corpus work including *The American Heritage Word Frequency Book* (Carroll *et al.* 1971) continued during this period and the immediately ensuing years.

Gradually after the Chomskyan revolution, the computer became a chief and central force pushing the development of corpus linguistics. Svartvik computerized the *Survey of English Usage* (Svartvik and Quirk 1980) and as a consequence produced "an unmatched resource for studying spoken English" (Leech 1991). The famous British counterpart of the Brown Corpus, the Lancaster-Oslo-Bergen (LOB) Corpus (Johansson 1980), was completed in 1976.

The 1980s witnessed the reemergence of corpus-based research on a fuller scale (Aarts 1990; Leech 1991). Perhaps the most immediate reason for this empirical renaissance was the advances in computer science which enabled the availability of huge quantities of text data. The creation and storage of corpora was much easier than ever before. This in turn brought about the development of new approaches, algorithms, models and tools such as probabilistic part-of-speech taggers and syntactic parsers. People from all

participating disciplines of cognitive science were joining in to work on the theoretical, methodological and application issues. Since then, corpus linguistics has become a multidisciplinary branch in computational linguistics. Many influential corpora were built during that period and thereafter. The first fully corpus-based dictionary (Sinclair 1987a; 1987b) was based on the COBUILD Main Corpus of about 7 million words in 1982 (Renouf 1987). In 1995, the 100 million British National Corpus (BNC) was built. And now we have the Bank of English, running into more than 500 million words. At the same time, famous text collection projects are in full swing such as the ACL Data Collection Initiative (ACL/DCI), the European Corpus Initiative (ECI), the International Computer Archive of Modern and Medieval English (ICAME), the Linguistic Data Consortium (LDC), and many others.

2.2 Nativist Views and Corpus Development

The dominant nativist paradigm in linguistics was formed in the 1950s under the influence of Chomsky. Under this view, the domain of linguistic inquiry should be linguistic competence. Chomsky argued that our capacity for learning languages is innate, and perhaps through evolution, our brains became somehow programmed for language (Chomsky 1957). Competence is our tacit, internalized knowledge of a language, and performance is the set of actual behaviors produced on particular occasions by the application of competence. Performance can be affected by factors other than our internalized knowledge of a language. For example, whether or not we have been drinking alcohol can alter the way we speak. This brings us to the core of Chomsky's argument against the use of corpora: the information corpora yield is biased more

towards performance than competence and is thus overly descriptive rather than theoretical (Leech 1992). To Chomsky, a corpus is by its very nature only performance data. The collection of externalized utterances is only a poor mirror of competence, and thus cannot be a good guide to modeling linguistic competence.

> Any natural corpus will be skewed. Some sentences won't occur because they are obvious, others because they are false, still others because they are impolite. The corpus, if natural, will be so wildly skewed that the description [of language based on the corpus] would be no more than a mere list. (Chomsky 1957)

In conjunction with this focus on competence, Chomsky advanced the notion of Universal Grammar (UG). Chomsky suggested that UG is genetically determined, structured in the human mind, and common to all human languages. UG has parameters that remain open to be fixed by experience. Limited evidence will suffice for the development of rich and complex systems in the mind, and a small change in parameters may lead to what appears to be a radical change in the resulting system (Chomsky 1980).

The UG theory was advanced to account for the problem of language acquisition in children. Fundamental to theories of UG is the observation that children must learn language from "meager and unspecific evidence" (Chomsky 1986). Given that the evidence presented to language learners is so impoverished, the only possible explanation for universal language acquisition is that some of the knowledge of language must be pre-specified in Universal Grammar. UG constrains and guides the language acquisition

process in such a way that, even given the paucity of evidence available to children, language is reliably acquired. As Chomsky puts it,

> My own suspicion is that a central part of what we call "learning" is actually better understood as the growth of cognitive structures along an internally directed course under the triggering and partially shaping effect of the environment. (Chomsky 1980)

Chomsky holds that human beings are born with a rich structure of cognition already in place. Children can acquire a language mostly because of Universal Grammar, which can be accounted for only by description of its rules - not by enumeration of its sentences. UG can generate an infinite number of sentences but a corpus can cover only a finite number of sentences. This apparently invalidates the corpus as a source of evidence in linguistic inquiry, as John Searle observes:

> This conception of the goal of linguistics then altered the conception of the methods and the subject matter. Chomsky argued that since any language contains an infinite number of sentences, any "corpus," even if it contained as many sentences as there are in all the books of the Library of Congress, would still be trivially small. (Searle 1972)

As a result of the wide acceptance of Chomsky's theories of language, there was little tolerance for the approaches that generative grammarians deemed unacceptable linguistic

practice. The first standard computer corpus, the Brown Corpus, was not warmly accepted by the linguistics community when it was first created in the early 1960s. At one time, it was labeled by a key generative grammarian as "a useless and foolhardy enterprise" because corpus information is not intuitive information, and intuitions are the only legitimate source of grammatical knowledge (Meyer 2002).

The influence of Chomsky's revolution is a very lasting one. In the recent Minimalist Program (Chomsky 1995), a distinction is made between the core elements of a language and the peripheral part of a language. The core is comprised of "pure instantiations of U.G." and the periphery "marked exceptions" that are a consequence of "historical accident, dialect mixture, personal idiosyncracies, and the like" (Chomsky 1995: 19–20). Elements belonging to the periphery of a language are considered not relevant for purposes of theory construction.

While many linguists follow Chomsky from his generative grammar theory to the current Minimalist theory, linguists are now more open to the idea of using linguistic corpora for both descriptive and theoretical studies of language. Corpus linguistics has gradually extended its scope and influence, so that, as far as speech and text processing are concerned, natural language corpus data analysis has become a critical component. The value of the corpus as a source of linguistic analysis, for systematically retrieving data and for text processing software development has become widely recognized. Previously generative grammarians and corpus linguists, each with different goals, were rarely in the mood to communicate with each other, as Fillmore joked:

> These two don't speak to each other very often, but when they do, the corpus linguist says to the armchair linguist, "Why should I think that what you tell me is true?", and the armchair linguist says to the corpus linguist, "Why should I think that what you tell me is interesting?" (Fillmore 1992)

Now more linguists take the view that the analysis of corpora can contribute to the study of language and the construction of linguistic theories, and that robust corpus analysis tools can make many language inquiries possible and easier. Many linguists working in the fields of corpus analysis are actively engaged in issues of language theory, while many generative grammarians have shown an increasing concern for the data upon which their theories are based.

We cannot expect that corpus linguistics will revive the "all linguistics was corpus linguistics" glory. Nor can we expect that corpora will ever be used very widely by generative grammarians, even though some generative discussions of language have been based on the use of corpus data. However, we can reasonably assume that cognitive structures are related in some way to language performance. Texts collected in corpora provide some of the best evidence we have about the nature of cognitive structures because the natural language corpus can be "representative of all the structures and usage of natural speech" (Sankoff and Sankoff 1973). Besides, by analyzing current corpora, we have good reason to say that not all language data sources are messy and riddled with errors. Most of the field-collected samples in the corpora are good and reliable texts, except for the portions that are randomly chosen for the purpose of recording informal and error usage to reflect the authentic nature of language usage.

2.3 Bilingual Corpus Building

During the historical development of corpora, different types of corpora emerged. We have historical corpora and regional corpora, learner corpora and native speaker corpora, spoken language corpora, annotated corpora and plain unannotated corpora. Based on the representativeness criterion of corpus building, we can have general corpora and specialized corpora. General corpora aim at representing a language or variety as a whole. They are usually large in size and contain spoken and written language, formal and informal usage in various social and situational strata. Many large scale monolingual corpora such as the British National Corpus are general corpora. Specialized corpora are usually domain-specific and are assembled for a specific purpose. They can vary in size and composition according to their purpose. Their main advantage is that they can focus on a particular domain so that materials in that domain will be covered on a wider scale with more domain-specific texts and less noise, and the word frequency distribution can be better represented (Zhu 1991). The Guangzhou Petroleum English Corpus is an example of a domain-specific corpus in the monolingual category (Zhu 1989; Leech 1991; Orr 2006). If we group corpora according to the languages involved, we can have monolingual and bilingual corpora. For bilingual corpora, if the source texts and their translations are aligned, they are called parallel corpora; if the texts in two languages are similar in content but are not mutual translations, they are called comparable corpora.

For bilingual corpus building, data sparseness poses a major challenge. For compiling a monolingual corpus, we may have more data sets available for us to choose from. Thus we can divide the data into categories and subcategories, and make sure that the subcategories are balanced properly so that no subcategory exerts an undue influence on

the parent category (Biber 1993a). Given the very nature of bilingual corpus, building a comprehensive and general type of bilingual corpus may prove to be difficult. The insufficient quantity of quality translation texts can sometimes simply stop our corpus compilation process. In some cases, we are able to find the bilingual texts, but they are not eligible to be included in a bilingual corpus. The reason is that some texts are translated only at a more general theme level, or at an outlining "script" level. They are not good candidate documents for a quality bilingual corpus because parallel alignment will be almost impossible. Sometimes, we know that we have the translated texts, but we need to pay a fortune to gain access to the texts. An alternative to this problem is to concentrate on only a specific area of language usage and make it a domain-specific or genre-specific type of corpus. For example, we can build a bilingual corpus using legal documents, parliamentary proceedings or bilingual materials on a website. In this sense, texts for bilingual corpora are more likely to be collected according to the data sources we already have on a specific topic, or data sources that can be made available when chances for collection arise. In most practical applications, this is not a bad thing: people want to focus on translations that are more related to their domain of text instead of searching for translations that will apply universally in all cases.

Recently in building bilingual corpora, just as in building monolingual corpora, the need for building corpora for different subject domains and different text genres has become more and more recognized. A great deal of attention has been paid to relatively smaller but domain-specific corpora rather than only to the large and wide-coverage type of general corpora. With the increase in published bilingual pages on the web, there is a noticeable trend in bilingual corpus building that many more researchers and translators

are going to be interested in translation texts of specific domains and genres. Often people would worry that the size of such a bilingual corpus is smaller because the text domain is limited. In fact, some principles of corpus size and representativeness in building monolingual corpora can be applied to bilingual corpus building. For example, the size of the corpus doesn't have to be extra large. Leech (1991:10) argues that it is naïve to focus merely on corpus size. Murison-Bowie (1993:50) holds that small corpora can be very useful providing they can represent the language of a specific area. Kennedy (1998:68) suggests that, "A huge corpus does not necessarily 'represent' a language or a variety of a language any better than a smaller corpus." For Fillmore, while no corpus is too big in size, small corpora do have their roles to play: "...every corpus that I've had a chance to examine, however small, has taught me facts that I couldn't imagine finding out about in any other way" (Fillmore 1992). Such comments are not only the driving force behind the development of specialized monolingual corpora, but also an impetus to the building of domain-specific or genre-specific bilingual corpora.

The building of bilingual corpora can be traced as far back as translated texts such as those on the Rosetta Stone. The texts on the 196 B.C. stone were believed to convey the honors presented to King Ptolemy V by the temples of Egypt, in the languages of Greek and Egyptian, and in three writing systems (Véronis 2000b). Translation texts such as inscriptions on the tomb, treaties, religious writings, and literature works span over all the important periods of civilization. Many of them are considered to be good examples of parallel texts because the texts are accompanied by their translations.

Attempts at using aligned bilingual texts or bitexts for machine translation began in the late fifties, but the applications were very much limited because of difficulties of

entering, storing and processing large quantities of corpus data at that time. Starting from the 1970s, there were some major developments in building and applying bilingual corpora. One notable development is that two research groups, Bell Communications Research and the IBM T. J. Watson Research Center, developed what was to become one of the most important bilingual corpora in the world --- the Hansard corpus.

The Hansard corpus is composed of transcribed proceedings from the Canadian parliament. As such, it is mostly recordings of spoken language. However, the spoken language is normalized in the process of transcription so that the bilingual corpus is free from pauses, interruptions and false starts which are common in spoken language corpora. Although individual speakers' topics can shift freely and their choices of lexis can be unconstrained, the Canadian Hansard has far more in common with written text corpora than with the usual spoken corpora.

In the past few decades, the Hansard corpus has influenced many research projects in machine translation, machine learning, bilingual corpus data processing, and translation studies. In the meantime, an emerging trend in the use of bilingual texts becomes rather noticeable: the relatively recent use of the web as a bilingual corpus (Resnik 1999; Chen and Nie 2000). Researchers are designing ways to collect, from the web, bilingual texts of such language pairs as English-French, English-German, English-Italian, English-Chinese, English-Arabic and others (Kraaij *et al.* 2003; Resnik and Smith 2003). The application of the web as a bilingual corpus was made possible by the rapid growth in the number of web pages, and the availability of vast quantities of web-based bilingual texts involving many language pairs. Many web-based materials, such as those officially published in Canadian government websites contain clean and exemplary translations.

We believe that mining these web-based bilingual materials for bilingual corpus building can be beneficial to translators, linguists and researchers in many fields.

2.4 Applications of Parallel Corpora

> In fact, it is clear that existing translations contain more solutions to more translation problems than any other available resource. (Isabelle *et al.* 1993)

A rapidly growing body of research on bilingual corpus alignment attests to the importance of aligned bilingual corpora. The applications illustrate significant diversity. Given that the quality of translation is good and the bilingual corpus is well-aligned, the corpus can represent a crucial resource for different natural language processing tasks (Moore 2002; Gey *et al.* 2002), such as machine translation (Hutchins 2005; Deng *et al.* 2006; Simões and Almeida 2006) and cross-language information retrieval (Chen and Gey 2001). Extracted translation pairs can also be a useful reference for translation studies (Neumann and Hansen-Schirra 2005), computer-aided translation (Callison-Burch *et al.* 2005), and computer-assisted revision of translations (Jutras 2000). Texts that are available in two languages can play a pivotal role in language comparison, language analysis, second language learning and teaching, corpus data processing, word sense disambiguation, bilingual dictionary making and terminology extraction.

2.4.1 Foreign Language Teaching

A very common use of aligned texts is for the teaching and learning of foreign languages. Parallel corpora with aligned translation texts have become very useful resources for the development of foreign language reading and writing skills. The applications of bilingual corpora in language teaching and learning are growing in number. For example, concordancers based on bilingual corpora can be used in language teaching or second-language learning (Johns 1986; Mindt 1986; Barlow 2000). In other language learning applications, students of foreign languages use one half of a bitext to practice their reading skills, referring to the other half for translation when they get stuck (Nerbonne *et al.* 1997). There are also tools designed to make use of bilingual corpora and provide the user with interactive facilities for looking up information on lexical units and their translation equivalents.

Some small bilingual corpora are specifically tailored for classroom purposes. Pearson (2003) illustrated how this kind of parallel corpus can allow students' individual activities in the translation class. Students were first provided with a small English-French corpus of popular science articles and then asked to analyze and study how some personal and professional references like titles and names of institutions were translated in the aligned corpus. When they are exposed to natural language corpus data and have read the translation examples from the corpus, they can discover what the translations will be in different contexts. In this way, students can expect to learn more from new resources through the discovery learning opportunities offered by translation corpora.

2.4.2 Terminology Extraction and Word Sense Disambiguation

Over the past few decades, the use of bilingual corpora for the derivation of bilingual dictionaries and terminology databases has gained increasing respectability. Lexical and terminology items can be automatically extracted from bilingual corpora (Smadja *et al.*1996; Brown *et al.* 2000; Melamed 2000) to produce bilingual lexical or semantic resources such as dictionaries or ontologies (Giguet and Luquet 2005), or terminology correspondence banks (Eijk 1993). For example, Dominic *et al.* (2002) used a bilingual vector model for the automatic discovery of German-English term translations. Their model analyzes co-occurrence patterns in the bilingual corpus of medical abstracts, and then applies the cosine similarity measure to arrive at candidate term translations. The correct translations could be added to a multilingual dictionary.

Bilingual corpus data can be explored to tackle monolingual problems as well (Mihalcea and Simard 2005). Some approaches have successfully employed bilingual corpora for monolingual word sense disambiguation (WSD). The assumption is that, from the translations of a word, we can work out the senses of the word (Resnik and Yarowsky 1997). If the word is translated differently in the target language, it may mean that the word has different senses. If one target word is the translation from different source language words, it may indicate that these words are used with similar meanings. Ide (1999) processed a parallel aligned corpus to find out different translations of source words, mapped their translated senses into WordNet senses, and used the information to determine the monolingual sense distinctions. Diab and Resnik (2002) proposed a semi-supervised algorithm to generate an English sense tagged corpus from an aligned parallel

corpus with the aid of an English sense inventory. The sense tagged corpus was then used to train a monolingual WSD algorithm.

2.4.3 Human Translation and Translation Studies

To a certain degree, the availability of parallel corpora has transformed the ways in which the translation professional gathers information. Aligned bilingual corpora can be applied both to the editing process for the recycling of previous translations, and to translation studies for the analysis of translations. In addition, bilingual texts are currently providing the basis for the development of a new generation of software tools to assist human translators and to improve the quality and productivity of their work.

Displaying translations side by side makes bilingual corpora a very useful resource in a translation setting, since it allows translators and researchers to retrieve translation information easily. Bilingual corpora are used for the investigation of translators' styles (Baker 2000; Olohan 2004). For example, linguistic idiosyncrasies in translated texts can be compared with the idiosyncrasies in the original source texts for stylistic studies.

In addition to direct applications in translation studies, new tools have been created based on bilingual corpora to facilitate the human translation process. One initiative is the bilingual concordancing tool that goes with some bilingual corpora. Initially, most of the concordances based on bilingual corpora were designed for language teaching. It is only recently that their potential as translation aids has been recognized (Bowker and Barlow 2004). TotalRecall, developed by Wu *et al.* (2004), is such a tool. TotalRecall is a bilingual concordancer that uses Sinorama Magazine and HKLEGCO corpora as the databases of translation memory. The system accepts a search query in English or

Chinese. It can retrieve and highlight the corresponding translations, and rank the search results according to the translation frequency.

2.4.4 Machine Translation

Parallel texts play a key role in many areas of natural language processing, and machine translation is one of the areas that uses parallel texts most. Many statistical machine translation systems require bilingual texts as the basic ingredients for building statistical alignment models. Parallel corpora are also much used for training automatic systems and machine learning algorithms in machine translation systems (Nagao 1984; Wang and Waibel 1998; Koehn 2005).

The quality of machine translation depends on the quality of the translation knowledge base in the translation system. Many machine translation systems acquire their translation knowledge directly from bilingual corpora. For rule-based methods which mostly depend on linguistic knowledge used in the representation of translation units, we need to infer rules from a good translation knowledge base which can come from a bilingual corpus. For example-based translation systems, the two challenges are the "knowledge access bottleneck" and the "knowledge acquisition bottleneck" which include collecting and selecting similar examples from the paired and syntactically analyzed sentences of bilingual corpora. For pattern-based and corpus-based machine translation systems, bilingual corpora are used for extracting paired lexical or grammatical patterns and structures. These elements can produce the translation dictionaries required for machine translation systems.

For the current research, the main area of application of the aligned StatCan corpus is for language analysis, translation equivalence studies, statistical modeling of alignment algorithms, feature extraction testing, and experimental testing of TextComp methods.

2.5 Summary

This chapter has presented a look back at the history of natural language corpus development and the theoretical issues that brought about important changes in the development of corpus linguistics. We briefly described some major word frequency lists and language corpora that sustained the ups and downs of corpus development. We specifically discussed the dominant nativist paradigm in the 1950s and its negative impact on the progress of corpus development. By comparing the principles of constructing monolingual corpora with those of building bilingual corpora, we examined the problems and challenges facing bilingual corpus building. We proposed that web-based bilingual materials, such as those published in Canadian government websites, could be a good source for building domain-specific bilingual corpora. We also looked at the practical applications of bilingual corpora such as those that can be built from published bilingual texts on the web. To a certain extent, this chapter sets the rationale for building the StatCan Daily Corpus (SDC).

Chapter 3

Previous Work in Bilingual Text Alignment

This chapter presents a literature review on various algorithms and methods for bilingual text alignment. While discussing the statistical alignment method, the lexical alignment method and approaches that employ both statistical and lexical criteria, we will describe some influential algorithms and techniques that are related to the algorithms we use in BMIA, such as the Gale and Church algorithm and the K-vec algorithm. We will also explain some bilingual text mapping methods that are commonly shared in many text alignment models. Close to the end of the chapter, there will be a brief discussion of factors that can potentially affect the performance of bilingual text aligners.

Bilingual text mapping in this thesis means the alignment of best sequences of translation text segments in two languages. A text segment can be a title or subtitle, a phrase in a table cell, or a sentence. In this thesis, a text segment is sometimes referred to as a "bead" of text, as defined in Manning and Schütze (1999:468). Previous related research in parallel text mapping of bilingual corpora has mostly been done in the field of bilingual text alignment, as Isabelle and Church pointed out:

> One important change has been the re-emergence of corpus-based thinking in NLP research, with particular interest in parallel corpora (i.e. collections of human translations). This, in turn, has motivated considerable discussion in techniques such as alignment programs. The availability of these alignment

programs is opening up a whole range of novel possibilities for translation support tools: (semi-)automated terminology extraction, translation memories, translation checkers, etc. (Isabelle and Church 1997)

Source language text segments and target language text segments can be aligned in different alignment patterns such as 1:1 and 1:2 (see Figure 3.1). Bilingual text mapping at the sentence level or text segment level is a useful first step towards phrase alignment or word alignment. Many word alignment models suppose that the corpus is already sentence aligned. Although we have a variety of methods for sentence alignment, there is still room for improvement (Melamed 2000).

[Alignment type: 1:2] Her uncle and aunt were all amazement; and the embarrassment of her manner as she spoke, joined to the circumstance itself, and many of the circumstances of the preceding day, opened to them a new idea on the business. 舅父母听了都非常惊讶。他们看见她说起话来那么窘，再把眼前的事实和昨天种种情景前前后后想一想，便对这件事有了一种新的看法。
[Alignment type: 1:1] Nothing had ever suggested it before, but they now felt that there was no other way of accounting for such attentions from such a quarter than by supposing a partiality for their niece. 他们以前虽然完全蒙在鼓里，没有看出达西先生爱上了他们的外甥女儿，可是他们现在觉得一定是这么回事，否则他这百般殷勤就无法解释了。
[Alignment type: 1:1] While these newly-born notions were passing in their heads, the perturbation of Elizabeth's feelings was every moment increasing. 他们脑子里不断地转着这些新的念头，伊丽莎白本人也不禁越来越心慌意乱。

Figure 3.1. Aligned sentences from block 1373 of the translated novel "Pride and Prejudice" (Zhu 1999)

3.1 Related Bilingual Text Alignment Models

For the majority of sentence alignment models, there seems to be a notable degree of consensus in alignment methodology, such as principles of statistical alignment and alignment based on linguistic clues. Many statistical alignment techniques employ heuristic approaches that are based on empirical results or observations. If the languages involved are closely related and the translations are clean and consistent, statistical or quantitative measures can be used to determine sentence-level translation equivalents with good results. In the early 1990s, two research teams independently discovered that paragraph lengths of texts involving clean translations are highly correlated (see Figure 3.2) and that relative sentence lengths can be used to determine sentence-level translation links (Brown, Lai, and Mercer 1991; Gale and Church 1991). Gale and Church (1991) were using sentence length, measured in characters, to determine the likelihood of alignment between sentences in two languages, while Brown, Lai, and Mercer (1991) formulated the problem as a hidden Markov model (HMM), and produced alignment links by matching similar sequence lengths in terms of words.

At the same time, some researchers used a linguistic approach, arguing alignments generated by lexical methods can be more accurate and reliable. In general, lexical methods depend more on the lexical, grammatical, and morphological features for the pairing of sentences in the bilingual corpus. In 1988, Kay and Röscheisen proposed a method based on the premise that if two sentences are a translated pair, word distributions within the sentences will be similar (Kay and Röscheisen 1993). They used an iterative process to induce a maximum likelihood for sentence alignment and then used the sentence alignment links to refine the word-level alignment estimates. This is the

first approach to iteratively calculate word co-occurrence distributions to identify anchors for sentence alignment.

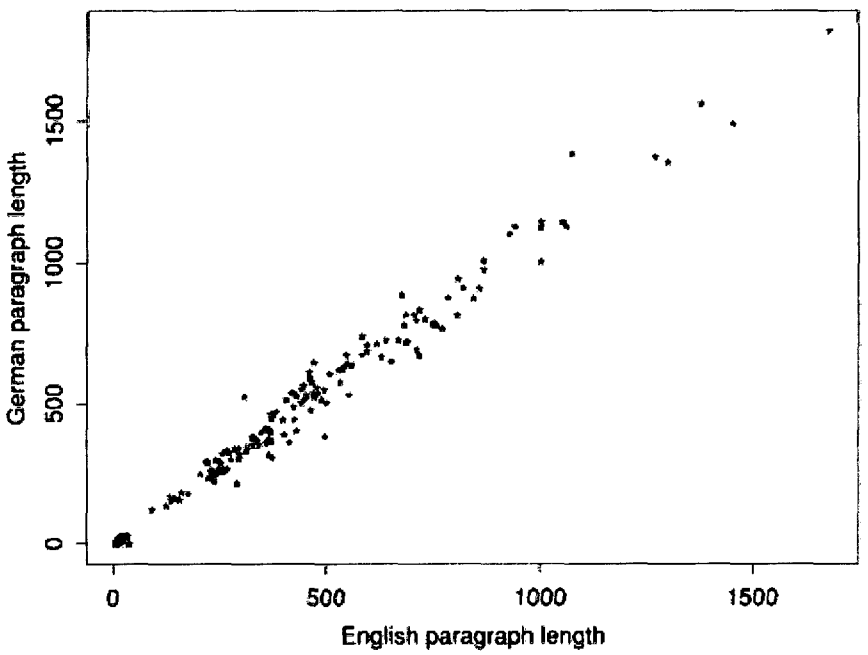


Figure 3.2. Paragraph lengths are highly correlated (Gale and Church 1991:80)

Researchers later found that the performance of alignment was not as good for some pairing tasks and for some types of corpora, if only the statistical or the lexical method is used. Statistical methods run faster, but have the risks of massive misalignment. Lexical methods usually render more accurate results, but the slow speed of the algorithms can make them unsuitable for practical purposes. However, when the two approaches are combined, and a more composite method is employed, alignment performance can be improved. Early integrated models include those developed by Simard *et al.* (1992), Wu (1994), and Chen (1996). Simard, Foster and Isabelle (1992) discovered that cognate words can help improve alignment when combined with other alignment methods. Wu

(1994) proposed to add a domain controlled translation lexicon to the Gale and Church alignment algorithm to enhance alignment for English-Chinese texts. Chen (1996) added a statistical translation algorithm to the statistical alignment model developed by Brown, Lai, and Mercer (1991).

These approaches are good examples indicating that the statistical method and the linguistic method are not mutually exclusive, but can be complementary and can be used together for better alignment results. Many current alignment methods have more or less a hybrid flavour, integrating linguistic features and the statistical models in the alignment algorithms. Lexical data in the composite models mainly include bilingual dictionaries or translation lexicons (Debili and Sammouda 1992; Wu 1994; Haruno and Yamazaki 1996), tag information (Kupiec 1993), or cognates (Simard, Foster and Isabelle 1992; Melamed 1996). For example, Melamed (1996) designed a model called the Smooth Injective Map Recognizer (SIMR) at the University of Pennsylvania. SIMR attempts to find "true points of correspondence" (TPC's) in the bilingual text search space, and the TPC's are identified by using a machine readable bilingual dictionary or cognate-based matching techniques. Many of the composite models (Wu 1994; Haruno and Yamazaki 1996; Fung and Church 1994) are more robust in that they can be used for alignment for a language pair that is not so closely related. Some can handle more noisy data (Fung and McKeown 1997). Some models not only integrate statistical and lexical methods, they also combine existing components in various models into one alignment package. For example, Choueka *et al.* (2000) developed a text alignment system for a rather disparate language pair --- Hebrew-English. In the system, they used a combination of DK-vec

(Fung and McKeown 1997), the *word_align* algorithm (Dagan *et al.* 1993), and an extension of IBM Model 2 (Brown *et al.* 1993).

Now let us focus on some of the influential approaches that are most related to the text mapping approaches in BMIA.

3.1.1 The Gale and Church Algorithm

The basic assumption of the Gale and Church (1991) algorithm is that for historically related language pairs such as English and French, the number of characters in the source language text can be a predictor for the number of characters in target language text. There is a strong likelihood that a long sentence in English corresponds to a long sentence in French; similarly a short sentence in one language corresponds to a short sentence in the other. Roughly speaking, if the average lengths of sentences in French and English are known, it is possible to set up a distribution of alignment possibilities from the sentence length information.

Let us suppose that for parallel texts S and T, S and T can be split into n segments each. Each segment in S (s_i) is the translation of a segment in T (t_i), and they are aligned to form an alignment segment pair a_i. For each i, $1 \leq i \leq n$. A can be defined as the alignment of S and T that consists of a series of aligned segment pairs: $A \equiv \langle a_1,...,a_n \rangle$.

Gale and Church (1993) estimate the alignment parameters through a series of approximations. For the set B of all possible alignments ($A \in B$), the goal is to find the maximum-likelihood alignment:

$$A_{\max} = \arg\max_{A \in B} \Pr(A \mid S,T) \qquad (1)$$

It is assumed that the probability of any aligned segment pair is independent of any other segment pair:

$$A_{\max} = \arg\max_{A \in B} \prod_{i=1}^{|B|} \Pr(a_i \mid s_i, t_i) \qquad (2)$$

Gale and Church introduced a function $d(s_i, t_i)$ as the length difference of the text segments. In their algorithm, they assume this function is the only feature of s_i and t_i that influences the alignment probability:

$$A_{\max} = \arg\max_{A \in B} \prod_{i=1}^{|B|} \Pr(a_i \mid d(s_i, t_i)) \qquad (3)$$

By Bayes' theorem, which states $P(M \mid N) = \frac{P(N \mid M)P(M)}{P(N)}$, equation (3) becomes:

$$A_{\max} = \arg\max_{A \in B} \prod_{i=1}^{|B|} \frac{\Pr(d(s_i, t_i) \mid a_i)\Pr(a_i)}{\Pr(d(s_i, t_i))} \qquad (4)$$

where $\Pr(d(s_i, t_i))$ is the normalizing constant. When they ignore the denominator and use the logarithm, they have,

$$A_{\max} = \arg\max_{A \in B} \sum_{i=1}^{|B|} \log \Pr(d(s_i, t_i) \mid a_i)\Pr(a_i) \qquad (5)$$

Gale and Church used hand-aligned data to arrive at an estimation of the distributions $\Pr(d(s_i, t_i) \mid a_i)$ and $\Pr(a_i)$. Dynamic programming (Sankoff and Kruskal 1983) is employed as an efficient search technique for establishing the optimal alignment to solve Equation (5).

The Gale and Church method also accounts for the fact that there is sometimes a lack of a one-to-one correspondence between sentences. A sentence in one language may

correspond to zero, one, two, or several sentences in the other language. The model considers the *a priori* likelihood of alignment types. There are six types:

- 1:1 substitution; meaning, for example, one English sentence is aligned with one French sentence;
- 1:0 deletion; meaning the English sentence is not translated;
- 0:1 insertion; meaning the French sentence is not translated;
- 2:1 contraction; meaning two English sentences are translated by a single French sentence;
- 1:2 expansion; meaning a single English sentence is translated by two French sentences;
- 2:2 merging; meaning two English sentences are translated by two French sentences, where each of the English sentences does not have an individual translated correspondence in French.

The alignment types can be illustrated by the alignment of the simple corpus in Figure 3.3.

There are three possible alignments of this small corpus. First, we might have a 1:2 alignment, where the single English sentence e_1 is translated by two French sentences f_1 and f_2. Second, we might have a 1:1 alignment which is followed by a 0:1 alignment. The English sentence e_1 is translated by the French sentence f_1 only, while the French sentence f_2 has no English translation. Third, we might have a 0:1 alignment followed by a 1:1 alignment, where the French sentence f_1 has no English translation, and the English sentence e_1 is translated by the French sentence f_2 alone.

e_1. "All beneficiaries" includes all claimants receiving regular benefits (for example, as a result of layoff) or special benefits (for example, as a result of illness) and are representative of data for the Labour Force Survey reference week which is usually the week containing the 15th.of the month.
f_1. L'ensemble des bénéficiaires inclut tous les prestataires recevant des prestations de type ordinaire (par exemple, en raison d'un licenciement) ou des prestations spéciales (par exemple, pour cause de maladie). f_2. Ces bénéficiaires comprennent toutes les personnes qui ont reçu des prestations pour la semaine de référence de l'Enquête sur la population active qui comprend habituellement le 15e jour du mois.

Figure 3.3. A simple corpus for sentence alignment from file d040629c

As can be seen from the example, in the Gale and Church length-based model a given sentence in one language can correspond to one, zero or two sentences in the other language. From the analysis of the hand-aligned data, Gale and Church found that 1:1 aligned sentences were the most common (see Table 3.1). In order to discourage the algorithm from proposing non-1:1 alignments (which are less likely), a penalty formula is used:

penalty = 100 * log([probability of match type] / [probability of 1-1 match])

so that in the program we have (Gale and Church 1991),

```
int penalty21 = 230;        /* -100 * log([prob of 2-1 match] / [prob of 1-1 match]) */
int penalty22 = 440;        /* -100 * log([prob of 2-2 match] / [prob of 1-1 match]) */
int penalty01 = 450;        /* -100 * log([prob of 0-1 match] / [prob of 1-1 match]) */
```

And the resultant penalties are (Table 1):

Type	Probability	Frequency	Penalty
1:1	.89	1167	0
1:0 or 0:1	.0099	13	450
2:1 or 1:2	.089	117	230
2:2	.011	15	440

Table 3.1. Alignment types and penalties in the Gale and Church algorithm

While the Gale and Church algorithm has generated good outputs for language pairs like English-French and English-German, there is still great scope for improvement. For disparate language pairs, such as Chinese and English, the performance is not as good. The algorithm is not robust with respect to non-literal translations and deletions. Also with an algorithm relying only on length of the sentence, it is quite difficult to automatically recover from misalignment triggered by large deletions.

3.1.2 The IBM 1 Alignment Algorithm

IBM Model 1 (Brown *et al.* 1993), though not originally designed for sentence alignment, is widely used in parallel sentence alignment systems. Examples include extracting parallel sentences from comparable corpora (Munteanu *et al.* 2002), bilingual sentence alignment (Moore 2002), aligning syntactic-tree fragments (Ding *et al.* 2003), and estimating phrase translation probabilities (Venugopal *et al.* 2003). IBM Model 1 won the distinction of "truly significant improvement" that could improve a state-of-the-art translation system at the 2003 Johns Hopkins summer workshop on statistical machine translation (Och *et al.* 2004). GIZA++ is a recently developed statistical machine translation package based on IBM Models and an HMM word alignment model (Och and

Ney 2003). With the popularity of GIZA++, IBM Model 1 has become an increasingly common statistical alignment model for alignment learning and parameter estimations in more elaborate models.

IBM Model 1 takes a bag-of-words approach and ignores word order. As far as 1:1 sentence alignment is concerned, IBM Model 1 achieves good results that are very similar to the Gale and Church sentence length model. However, the IBM algorithm uses a different definition of length from that of Gale and Church. Whereas in the sentence length model, Gale and Church measure sentence length by characters, the IBM alignment method is calculating length based on tokens (See Figure 3.4).

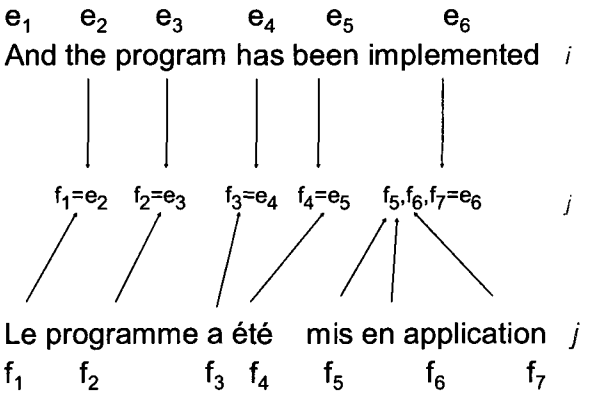


Figure 3.4. Alignment based on token numbers.

The IBM researchers also adopted a different methodological approach. Gale and Church took some linguistic data, aligned it by hand, and then used the figures thus obtained in order to generate the values of the distributions. The IBM researchers devised a method to iteratively refine translation probabilities (Dorr and Monz 2004), and use Expectation-Maximization (EM) for word alignment learning. The method assumes

initial probability estimates and uses them to determine the likelihood of an alignment. From the initial estimates of the probabilities, new probability estimates are derived, and in turn a new and probably more accurate alignment is performed. The process is repeated over and over again.

Let us see how word alignment is learnt through EM in IBM Model 1. The following are the notations and steps.

f = the French sentence

e = the English sentence

m = length of f

l = length of e

e_i = the $_i$th word in e

f_j = the $_j$th word in f

- Step 1: for each French word type and each English word type in the corpus, assign uniform probabilities to $P(f_j|e_i)$.
- Step 2: for the same sentence pair e and f, and for each word position j in f,

  (a) Set sum_p(f_j) to zero

  (b) For each of the l word positions in e, increment counts for sum_p(f_j),

$$\sum_{i'} P(f_j \mid e_{i'}) += \sum_{i=1}^{l} P(f_j \mid e_i)$$

  (c) For each of the l positions in e, increment counts for tc($f_j|e_i$), i.e.

$$tc(f_j \mid e_i) += \frac{P(f_j \mid e_i)}{\sum_{i'} P(f_j \mid e_{i'})}$$

- Step 3: for the same sentence pair *e* and *f*, and for each word position *i* in *e*

  (a) Set sum_ tc(e_i) to zero

  (b) For each of the *m* word positions in *f*, increment counts for sum_tc(e_i),

$$\sum_{j'} tc(f_{j'} \mid e_i) += \sum_{j=1}^{m} tc(f_j \mid e_i)$$

  (c) For each of the *m* word positions in *f*, increment counts for P(f_j | e_i), i.e.

$$P(f_j \mid e_i) = \frac{tc(f_j \mid e_i)}{\sum_{j'} tc(f_{j'} \mid e_i)}$$

- Step 4: Repeat steps 2 and 3 and iterate until values of P(f_j |e_i) have converged to the desired degree.

Here is the pseudo code for the algorithm:

; for each word type in e and f, set P(f_j|e_i) uniform, including P(f_j|null)

; initialize P(f_j|e_i) uniformly

; iteratively refine translation probabilities

for n iterations **do**

  set tc (translation count) to zero

  for each sentence pair (e,f) of lengths (l,m) **do**

for j ← 1 to m **do**

sum_p(f_j) ← 0

for i ← 1 to l **do**

sum_p(f_j) += P(f_j | e_i)

for i ← 1 to l **do**

tc(f_j | e_i) += P(f_j | e_i)/ sum_p(f_j)

for i ← 1 to l **do**

sum_ tc(e_i) ← 0

for j ← 1 to m **do**

sum_ tc(e_i) += tc(f_j | e_i)

for j ← 1 to m **do**

P(f_j | e_i) = tc(f_j | e_i) / sum_tc(e_i)

Based on the algorithms described in Dorr and Monz (2004), we wrote programs for the word alignment learning with Expectation-Maximization in IBM Model 1. Suppose we have this "corpus":

Corpus (1)

the house :: la maison

the blue house :: la maison bleue

the dog :: la chien

Note that in Corpus (1), "la chien" is purposefully mistaken so that later we can compare the results with "le chien" in Corpus (2). First, we give uniform probabilities to all word translations (Table 3.2.), and then we do the iteration.

P(la|house) = 0.2500 P(la|the) = 0.2500 P(la|blue) = 0.2500 P(la|dog) = 0.2500	P(bleue|house) = 0.2500 P(bleue|the) = 0.2500 P(bleue|blue) = 0.2500 P(bleue|dog) = 0.2500	P(chien|house) = 0.2500 P(chien|the) = 0.2500 P(chien|blue) = 0.2500 P(chien|dog) = 0.2500	P(maison|house) = 0.2500 P(maison|the) = 0.2500 P(maison|blue) = 0.2500 P(maison|dog) = 0.2500
P(la|null) = 0.2500	P(bleue|null) = 0.2500	P(chien|null) = 0.2500	P(maison|null) = 0.2500

Table 3.2. Uniform probabilities of word translations.

Iteration 1

Computing tc for NULL the house :: la maison

sum_p(la) = P(la|null)+P(la|the)+P(la|house) = 0.2500+0.2500+0.2500 = 0.7500

tc(la|null) +=P(la|null)/sum_p(la) = 0 += 0.2500/0.7500 = 0.3333

tc(la|the) +=P(la|the)/sum_p(la) = 0+= 0.2500/0.7500 = 0.3333

tc(la|house) +=P(la|house)/sum_p(la) = 0+= 0.2500/0.7500 = 0.3333

sum_p(maison) = P(maison|null)+P(maison|the)+P(maison|house) = 0.2500+0.2500+ 0.2500 = 0.7500

tc(maison|null) +=P(maison|null)/sum_p(maison) = 0 += 0.2500/0.7500 = 0.3333

tc(maison|the) +=P(maison|the)/sum_p(maison) = 0+= 0.2500/0.7500 = 0.3333

tc(maison|house) +=P(maison|house)/sum_p(maison) = 0 += 0.2500/0.7500 = 0.3333

Computing tc for NULL the blue house :: la maison bleue

sum_p(la) = P(la|null)+P(la|the)+P(la|blue)+P(la|house) = 0.25+0.25 +0.2500 +0.25= 1

tc(la|null) +=P(la|null)/sum_p(la) = 0.3333 += 0.2500/1.0000 = 0.5833

tc(la|the) +=P(la|the)/sum_p(la) = 0.3333 += 0.2500/1.0000 = 0.5833

tc(la|blue) +=P(la|blue)/sum_p(la) = 0+= 0.2500/1.0000 = 0.2500

tc(la|house) +=P(la|house)/sum_p(la) = 0.3333 += 0.2500/1.0000 = 0.5833

sum_p(maison) = P(maison|null)+P(maison|the)+P(maison|blue)+P(maison|house) = 0.2500+0.2500+0.2500+0.2500 = 1.0000

tc(maison|null)+=P(maison|null)/sum_p(maison) = 0.3333 += 0.2500/1.0000 = 0.5833

tc(maison|the) +=P(maison|the)/sum_p(maison) = 0.3333 += 0.2500/1.0000 = 0.5833

tc(maison|blue)+=P(maison|blue)/sum_p(maison) =0+= 0.2500/1.0000 = 0.2500

tc(maison|house)+=P(maison|house)/sum_p(maison)=0.3333+=0.25/1= 0.5833

sum_p(bleue)=P(bleue|null)+P(bleue|the)+P(bleue|blue)+P(bleue|house)=0.25+0.25+ 0.25 + 0.25 = 1

tc(bleue|null) +=P(bleue|null)/sum_p(bleue) = 0 += 0.2500/1.0000 = 0.2500

tc(bleue|the) +=P(bleue|the)/sum_p(bleue) = 0 += 0.2500/1.0000 = 0.2500

tc(bleue|blue) +=P(bleue|blue)/sum_p(bleue) = 0+= 0.2500/1.0000 = 0.2500

tc(bleue|house) +=P(bleue|house)/sum_p(bleue) = 0+= 0.2500/1.0000 = 0.2500

Computing tc for NULL the dog :: la chien

sum_p(la) = P(la|null)+P(la|the)+P(la|dog) = 0.2500+0.2500+0.2500 = 0.7500

tc(la|null) +=P(la|null)/sum_p(la) = 0.5833 += 0.2500/0.7500 = 0.9167

tc(la|the) +=P(la|the)/sum_p(la) = 0.5833 += 0.2500/0.7500 = 0.9167

tc(la|dog) +=P(la|dog)/sum_p(la) = 0+= 0.2500/0.7500 = 0.3333

sum_p(chien)=P(chien|null)+P(chien|the)+P(chien|dog)=0.2500+0.2500+0.2500= 0.75

tc(chien|null) +=P(chien|null)/sum_p(chien) = 0+= 0.2500/0.7500 = 0.3333

tc(chien|the) +=P(chien|the)/sum_p(chien) = 0 += 0.2500/0.7500 = 0.3333

tc(chien|dog) +=P(chien|dog)/sum_p(chien) = 0+= 0.2500/0.7500 = 0.3333

Computing P

sum_tc(null) = tc(la|null)+tc(bleue|null)+tc(chien|null)+tc(maison|null) = 0.9167+ 0.2500 + 0.3333 +0.5833 = 2.0833

P(la|null) =tc(la|null)/sum_tc(null) = 0.9167/2.0833 = 0.4400

P(bleue|null) =tc(bleue|null)/sum_tc(null) = 0.2500/2.0833 = 0.1200

P(chien|null) =tc(chien|null)/sum_tc(null) = 0.3333/2.0833 = 0.1600

P(maison|null) =tc(maison|null)/sum_tc(null) = 0.5833/2.0833 = 0.2800

sum_tc(house)=tc(la|house)+tc(bleue|house)+tc(chien|house)+tc(maison|house) = 0.5833 + 0.2500+0.0000+0.5833 = 1.4167

P(la|house) =tc(la|house)/sum_tc(house) = 0.5833/1.4167 = 0.4118

P(bleue|house) =tc(bleue|house)/sum_tc(house) = 0.2500/1.4167 = 0.1765

P(maison|house) =tc(maison|house)/sum_tc(house) = 0.5833/1.4167 = 0.4118

sum_tc(the) = tc(la|the)+tc(bleue|the)+tc(chien|the)+tc(maison|the) = 0.9167+0.2500 +0.3333+0.5833 = 2.0833

P(la|the) =tc(la|the)/sum_tc(the) = 0.9167/2.0833 = 0.4400

P(bleue|the) =tc(bleue|the)/sum_tc(the) = 0.2500/2.0833 = 0.1200

P(chien|the) =tc(chien|the)/sum_tc(the) = 0.3333/2.0833 = 0.1600

P(maison|the) =tc(maison|the)/sum_tc(the) = 0.5833/2.0833 = 0.2800

sum_tc(blue)=tc(la|blue)+tc(bleue|blue)+tc(chien|blue)+tc(maison|blue)=0.2500 +0.2500 +0.0000+0.2500 = 0.7500

P(la|blue) =tc(la|blue)/sum_tc(blue) = 0.2500/0.7500 = 0.3333

P(bleue|blue) =tc(bleue|blue)/sum_tc(blue) = 0.2500/0.7500 = 0.3333

P(maison|blue) =tc(maison|blue)/sum_tc(blue) = 0.2500/0.7500 = 0.3333

sum_tc(dog)=tc(la|dog)+tc(bleue|dog)+tc(chien|dog)+tc(maison|dog)=0.3333+0+0.3333 +0 = 0.6667

P(la|dog) =tc(la|dog)/sum_tc(dog) = 0.3333/0.6667 = 0.5000

P(chien|dog) =tc(chien|dog)/sum_tc(dog) = 0.3333/0.6667 = 0.5000

This is what we can get after 5 iterations (Table 3.3):

P(f|e)	P(f|e)	P(f|e)	P(f|e)
P(chien|dog) = 0.8827 P(bleue|blue) = 0.8125 P(la|null) = 0.7063 P(la|the) = 0.7063 P(maison|house) = 0.6956	P(bleue|dog) = 0.2500 P(chien|blue) = 0.2500 P(chien|house) = 0.2500 P(maison|dog) = 0.2500 P(maison|the) = 0.2400	P(maison|null) = 0.2400 P(la|house) = 0.2327 P(maison|blue) = 0.1337 P(la|dog) = 0.1173 P(bleue|house) = 0.0717	P(la|blue) = 0.0538 P(chien|null) = 0.0289 P(chien|the) = 0.0289 P(bleue|null) = 0.0247 P(bleue|the) = 0.0247

Table 3.3. Probabilities after 5 iterations; Corpus (1).

and after 100 iterations (Table 3.4):

P(f|e)	P(f|e)	P(f|e)	P(f|e)
P(maison|house) = 1.0000 P(chien|dog) = 1.0000 P(bleue|blue) = 1.0000 P(la|null) = 0.9000 P(la|the) = 0.9000	P(bleue|dog) = 0.2500 P(chien|blue) = 0.2500 P(chien|house) = 0.2500 P(maison|dog) = 0.2500 P(maison|the) = 0.1000	P(maison|null) = 0.1000 P(maison|blue) = 0 P(la|house) = 0 P(bleue|house) = 0 P(bleue|null) = 0	P(bleue|the) = 0 P(chien|null) = 0 P(chien|the) = 0 P(la|dog) = 0 P(la|blue) = 0

Table 3.4. Probabilities after 100 iterations; Corpus (1).

The special NULL word here means that a word in French does not correspond to any word in English. When the probabilities are sorted (see Table 3.4), and when the NULL words are ignored, we can obtain the most likely word pairs that can be aligned. In this

case, the best translations are *maison* and *house*, *chien* and *dog*, *bleue* and *blue*, *la* and *the*. The probabilities of these pairs are already much higher than the others after 5 iterations (see Table 3.3).

It is interesting to note in our word alignment experiment with IBM Model 1 that when the *dog* is not female, the results can be radically different despite the number of iterations attempted. Here is the corpus:

Corpus (2)

the house :: la maison

the blue house :: la maison bleue

the dog :: le chien

When compared with the bilingual corpus we used earlier in Corpus (1), the only difference lies in the translation for "the dog" (**le** chien vs. **la** chien). However, it turns out to be much harder for the EM iteration process to find the correct one-to-one word correspondence beyond one pair in the second corpus. Here is what we got after 5 iterations (Table 3.5):

P(f|e)	P(f|e)	P(f|e)	P(f|e)
P(bleue|blue) = 0.7739 P(le|dog) = 0.5000 P(chien|dog) = 0.5000 P(maison|house) = 0.4717 P(la|house) = 0.4717 P(maison|the) = 0.3834 P(maison|null) = 0.3834	P(la|null) = 0.3834 P(la|the) = 0.3834 P(le|blue) = 0.2500 P(la|dog) = 0.2500 P(bleue|dog) = 0.2500 P(chien|blue) = 0.2500	P(chien|house) = 0.2500 P(maison|dog) = 0.2500 P(le|house) = 0.2500 P(la|blue) = 0.1131 P(maison|blue) = 0.1131 P(chien|null) = 0.0936	P(le|null) = 0.0936 P(le|the) = 0.0936 P(chien|the) = 0.0936 P(bleue|house) = 0.0567 P(bleue|null) = 0.0461 P(bleue|the) = 0.0461

Table 3.5. Probabilities after 5 iterations; Corpus (2).

and after 100 iterations (Table 3.6):

P(f|e)	P(f|e)	P(f|e)	P(f|e)
P(bleue|blue) = 1.0000 P(maison|house) = 0.5000 P(la|house) = 0.5000 P(le|dog) = 0.5000 P(chien|dog) = 0.5000 P(maison|the) = 0.4167 P(maison|null) = 0.4167	P(la|null) = 0.4167 P(la|the) = 0.4167 P(le|blue) = 0.2500 P(la|dog) = 0.2500 P(bleue|dog) = 0.2500 P(chien|blue) = 0.2500	P(chien|house) = 0.2500 P(maison|dog) = 0.2500 P(le|house) = 0.2500 P(chien|null) = 0.0833 P(le|null) = 0.0833 P(le|the) = 0.0833	P(chien|the) = 0.0833 P(la|blue) = 0 P(maison|blue) = 0 P(bleue|house) = 0 P(bleue|null) = 0 P(bleue|the) = 0

Table 3.6. Probabilities after 100 iterations; Corpus (2).

The number of iterations did not improve the performance much in identifying the second or third translation pair. In the probability list after 100 iterations, there is only one word that can be safely paired as translations: *bleue* and *blue*. The other pairs cannot reach a probability of more than .5, and are intuitively not necessarily good pairs to be aligned such as *la* and *house*, *le* and *dog*, *maison* and *the*.

IBM Model 1 is relatively less expensive computationally when it is compared with the other IBM alignment models. It can achieve good results for 1:1 correspondences. However, the method is not without its drawbacks. Usually to get good results, it requires large volumes of data and many EM iterations which can be computationally intensive and thus very time-consuming. Another shortcoming is that each word in the target sentence can be generated by at most one word in the source sentence. IBM Model 1 did not consider fertility and distortion. It does not carry out a 2:2 alignment like the Gale and Church model, and thus cannot tackle situations in which a phrase in one language is translated as a single word in another language. These limitations cannot be remedied easily without making the model significantly more complicated.

3.1.3 The Kay and Röscheisen Algorithm

The Kay and Röscheisen algorithm (Kay and Röscheisen 1993) differs in its approach from the Gale and Church model. The Gale and Church model makes no use of lexical information in alignment. However, the Kay and Röscheisen method mostly selects source and target sentences which contain possible lexical correspondences. In this algorithm, words in the two languages are aligned based on a similarity score to calculate how similar their distributions in the sentences are. The algorithm uses this information to decide which words within those sentences are most likely to be translations of each other. An important feature of this algorithm is a relaxation method to iteratively align bilingual texts using the word correspondences acquired during the alignment process.

The algorithm performs two functions simultaneously: sentence alignment and word alignment. It begins by generating an initial Alignable Sentence Table (AST), and from the AST, a Word Alignment Table (WAT) listing pairs of words together with similarities and frequencies in their respective texts. The WAT can be considered a word probability dictionary. The words that are aligned by comparing their distributions in the texts are presumed to be translation equivalents. It is assumed that when certain words have been aligned, the sentences in which these words occur must also be aligned. Thus we can obtain a Sentence Alignment Table (SAT) that records for each pair of sentences how many times the two sentences are set in correspondence by the algorithm. The process is iterated. For each iteration, sentence alignments are recorded in SAT, word alignments are recorded in WAT, and the AST is adjusted. Gradually, partial word alignments are used to induce a maximum likelihood sentence alignment, which is used

again to refine the word level alignment. Typically, after around 5 iterations like this, we can have sentences well-aligned.

Here is a diagram (Figure 3.5) showing how the algorithm works depending on the tables and lists:

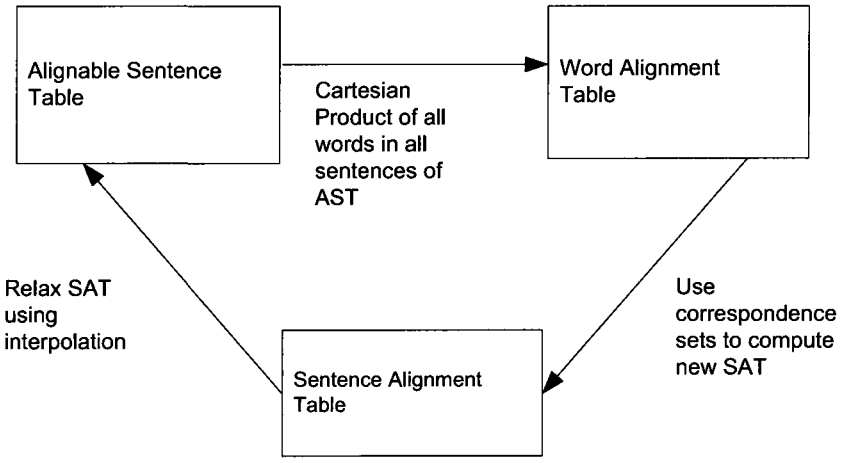


Figure 3.5. Relationship between tables and lists in the Kay and Röscheisen algorithm (Julapalli and Dhond 2003).

The algorithm can be broken down to several steps (Kay and Röscheisen 1993). The steps include:

1) Construct initial AST. The AST indicates which sentences in one language could possibly correspond to sentences in the other language.

2) Build the WAT from the AST. Pairs of words are entered in the WAT if the association between them is so close that it is not likely to be the result of a random event. Along with the closely associated words, the WAT also contains scores indicating

how similar the distributions in their corresponding sentences are. It is obtained by taking a Cartesian product of the words and calculating their similarity.

$$Similarity = \frac{2c}{Na(v) + Nb(w)}$$

where c is the size of the largest correspondence set (i.e. the set of sentences in which both words appear). $Na(v)$ is the frequency of word v in text a, and $Nb(w)$ is the frequency of word w in text b.

3) Sort the WAT. The WAT is portioned into several segments based on frequency. Greater values are given to those words that appear with high frequency and high similarity.

4) Compute a SAT from the WAT. Iterate through the WAT in order. For any words that are paired, pair their corresponding sentences for the SAT as well. Throw away any pairs whose similarity score is below the threshold.

5) Build a new AST. Use the entries from the SAT as fixed points in the AST, and interpolate between these fixed points to produce a new AST.

6) Iterate. Repeat building a new WAT in step 2 and continue to do steps 3 and 4 until the SAT converges. The SAT contains the final aligned sentences, and the WAT includes the aligned words. After computing the SAT, throw out any pairs that do not have a large number of correspondences.

The Kay and Röscheisen algorithm is a more complicated process than the Gale and Church model or IBM Model 1. Kay and Röscheisen (1993) claimed to have achieved good results with four or five iterations. For example, on Scientific American articles, the Kay and Röscheisen algorithm achieved 96% coverage after four passes. The

advantage of the approach is that it does not just align whole sentences: it also aligns partial sentences and words. Therefore it is very robust (Manning and Schütze 1999:478-480). But on the other hand, it is computationally expensive, particularly if one begins with a large text. Another problem with this algorithm is that, for some non-European languages, the method does not work well. For example, with Japanese-English translations, the method does not capture enough word correspondences to permit alignment.

3.1.4 The K-vec Algorithm

Another text alignment approach is the distribution-based K-vec algorithm (Fung and Church 1994). The assumption for this algorithm is that if two words are mutual translations, they are likely to occur in approximately the same regions of the parallel text. In this algorithm, parallel texts are split into *K* equal-sized pieces and *K*-dimensional binary vectors are created to record the distributions of each word. Suppose we have a small bilingual corpus of two texts: an English text containing 1000 words and a Chinese text with 900 words. We divide both texts into 10 equal-size pieces, and thus there are 100 words in each English piece and 90 words in each Chinese piece. The distributions of each word in these pieces are represented in a *1* ... *K* binary vector. For instance, the Chinese word "政府" (zheng4 fu3) and the English word "government" can appear in this bilingual corpus. A vector is created for both of the terms respectively. Suppose the frequency distributions of these two words in the small bilingual corpus is like this: for the Chinese text, "政府" occurs 3 times in the second piece, 5 times in the fourth piece and 7 times in the seventh piece; for the word "government" in the English text, there are

4 occurrences in the second piece, 4 occurrences in the fourth segment and 6 occurrences in the seventh segment. On the basis of the frequency distributions, we can have the values for the vectors for both of the terms:

V_z=<0,1,0,1,0,0,1,0,0,0>

V_g=<0,1,0,1,0,0,1,0,0,0>

Here, the value 1 in the binary vectors indicates that there is at least one occurrence of the word in the piece. Value 0 means that the word does not occur in the piece. The distribution information in these vectors does not consider the total counts of the words in a piece. Next, the binary vectors are compared and the similarity of the two distributions is quantified using two measures, Mutual Information (MI) and a *t*-score.

$$MI(V_g, V_z) = \log_2 \frac{prob(V_g, V_z)}{prob(V_g)prob(V_z)}$$

$$t \approx \frac{prob(V_g, V_z) - prob(V_g)prob(V_z)}{\sqrt{\frac{1}{K} prob(V_g, V_z)}}$$

The *t*-score is computed to filter out insignificant mutual information values. If these two vectors *Vz* and *Vg* are similar, this would suggest the possibility that "government" is the translation of "政府". Suppose *a* is the number of pieces where both the English word and the Chinese word are found, *b* is the number of pieces where only the English word is

found, c is the number of pieces where only the Chinese word is found, and d is the number of pieces where neither word is found, then, the probability of seeing the two words in the same piece is:

$$prob(V_g, V_z) = \frac{a}{a+b+c+d}$$

and the marginal probabilities are:

$$prob(V_g) = \frac{a+b}{a+b+c+d}$$

$$prob(V_z) = \frac{a+c}{a+b+c+d}$$

If the MI value of two words is high but the t-score is low, the strong association suggested by MI is probably the result of pure chance. The threshold values of MI and t-score are set to be 0 and 1.65 respectively. Using these measures, translation candidates such as "government" and "政府"can be ranked. The selection criterion is that only word pairs that have scores higher than the predetermined threshold values and that are in the frequency range 3 to 10 are considered to be potential mutual translations.

Good results were initially reported for this algorithm (Fung and Church 1994), but for later alignment tests with other languages such as English, Japanese and German, the performances of the K-vec method were not as impressive as the initial results (Jones and

Somers 1995). To make the K-vec algorithm more robust and capable of handling noisy texts, Fung and Mckeown developed the DK-vec method (Fung and Mckeown 1994; Fung and Mckeown 1997). The K-vec method assumes that the pattern of distance between successive occurrences of a word is similar to that of its translation. In the DK-vec method, however, the number of characters since the previous occurrence of the same word is defined as *recency*. All the recency values, i.e. the distances between the occurrences of the source language words and the target language words, are recorded in a vector known as "DK-vec" (Fung and McKeown 1994). For example, if the word "government" occurs at positions 200, 250 and 540 in the English text, its recency vector would be <200, 50, 290>. In K-vec, the pieces are of similar sizes; but in DK-vec, vectors can be of variable sizes due to different frequencies. The vectors in the DK-vec method are considered as signals and Fung and McKeown used the Dynamic Time Warping algorithm to find matching signals on the source language side and the target language side. When the recency vectors are compared, the pattern of distance differences in the vectors can indicate the likelihood of words as mutual translations: those with the most similar vectors are considered to be the most likely translation pairs. In this way, the most closely correlated pairs of words can be identified, so that these pairs can be used as reference points to align the parallel text.

One obvious advantage of the K-vec and DK-vec methods is that they can handle many types of noisy texts such as OCR (Optical Character Recognition) input and unparallel corpora because the methods do not assume sentence boundaries. Also, the frequency, position and recency parameters are easier to manipulate than the parameters in EM based algorithms. There are also limitations in the K-vec and DK-vec methods. If,

in a bilingual corpus, there are many recurrent terms or proper names, the co-occurrence ratio of translation pairs will be high. This will give K-vec a better performance. However, if the bilingual corpus contains translations with relatively few identical repetitions of words, or with more morphological variants or synonyms in one language, the performance of the K-vec and the DK-vec algorithms will be degraded.

3.2 Key Methods in Bilingual Text Mapping

There are some key methods that are shared by most of the alignment algorithms proposed, and many other alignment approaches are either derived from these methods or are variants of them. The basic assumption for the majority of previous alignment models is that to align words or sentences we need anchor points to start with. Anchors can be boundary markers, morphological endings, sequences of characters, words, lengths, or distances between word occurrences. If we use words as anchors, as many alignment algorithms proposed, we have to find an efficient way to find corresponding word pairs such as cognates. When we have extracted the anchoring information, we can use a similarity measure such as Dice's coefficient to compute similarity scores, and the scores can be compared to find the best alignment at the sentence level or the word level. Also, we can apply dynamic programming to these anchor points and find the best match amongst the many candidate translation pairs.

3.2.1 Anchor Points

Anchor points are positions in one text which seem to match up with positions in a parallel text. They can be delimiters that indicate "hard and soft boundaries" (Gale and Church 1993:89), or "true points of correspondence" (Melamed 1999:107) in alignment. Through the use of anchor points, regions of text can be identified where further alignments can be sought. Most alignment methods make use of anchor points. One obvious type of anchor point includes the paragraph or sentence boundary markers. Given a bilingual corpus, there is going to be a large number of such markers. Another type of anchor points can be mark-ups that go with the text and that reveal the meta-information or style information about the text. For example some commonly used HTML tags, such as `h1, h2, h3, p, hr, table, i, pre, form, img, a`. For a more detailed list of structural tags, format tags, and content tags that can be used in alignment, see Sanchez-Villamil *et al.* (2006). When using web pages to build a bilingual corpus, we are likely to have a proliferation of such HTML markups regardless of the languages involved. Usually, if a text in language *A* contains a markup for a section in italics, then the corresponding section in language *B* is likely to have the same markup. HTML style tags can be very useful in dividing texts into chunks or segments, or in deciding on sentence or paragraph boundaries (see Table 3.7).

Specific lexical units such as words or phrases can also be anchor points. For example, in the alignment algorithm proposed by Kay and Röscheisen (1993), translation word-pairs are taken as anchor points. They use word alignment to strengthen links in sentence alignment, and use sentence alignment results to reinforce word alignment. The alignment model proposed by Brown, Lai, and Mercer (1991) treated phrases as anchor

points. They used some phrases with a high ratio of co-occurrences as the anchoring information such as "Mr. Speaker" and "M. le Président". DK-vec (Fung and McKeown 1997) also employed distances between occurrences of the same word as anchors to induce word translations.

English	French
 18 December 2006 Prime Minister Stephen Harper fulfilled a longstanding commitment today by announcing the establishment of a sales program for the 11,000 acres of farmland in Mirabel. [More] 	 18 décembre 2006 Le premier ministre Stephen Harper a tenu aujourd'hui l’engagement qu'il avait pris il y a longtemps en annonçant la création d'un programme de vente de 11 000 acres de terres agricoles à Mirabel, au Québec. [Pour en savoir plus]

Table 3.7. HTML markups can be anchor points in parallel corpus alignment.

Similarly, numbers in texts can serve as anchors in alignment. They are good indications of correspondence, because a number in one language is usually interpreted as a number in the other language. Some punctuation marks can also be anchor points. For example, if there is a question mark in English, normally we are expecting a corresponding question mark in its translation text in French.

3.2.2 Cognates

Since Simard, Foster and Isabelle (1992) suggested using cognates to improve the alignment of bitexts, there have been different definitions of cognates, different ways of automatically extracting cognates, and different models of using cognates to aid bilingual text alignment (Church 1993; McEnery and Oakes 1996; Melamed 1999; Danielsson and Muhlenbock 2000; Ribeiro *et al.* 2001; Inkpen *et al.* 2005). In the parallel alignment context, cognates are not necessarily words with etymological ties. They can be identical or graphically similar occurrences in the two languages (Véronis 2000a). Cognates also include borrowings between languages which are not historically related. Simard, Foster and Isabelle (1992) were the first to suggest that the discovery of cognate word pairs could assist in the process of sentence alignment. Their assumption is that there are more cognate pairs in sentences of mutual translations than in random pairs of sentences. They proposed a way to measure the level of "cognateness" (y) in two pieces of text:

$$\gamma = \frac{c}{(n+m)/2}$$

Here, c is the maximal number of pairs of cognates, counted in word types, in the current text unit; n is the number of tokens in the source language and m is the number of tokens in the target language. By analyzing hand-aligned sample portions of texts from the Canadian Hansard, they found that when the sentence pairs are translation pairs, the average cognateness y is 0.21, and when sentence pairs are not translations of each other, y is only 0.06. They proposed that the first four characters of each word could be used for the identification of cognate pairs. For example, these pairs are considered to be cognate pairs:

1. produits, products
2. activités, activities
3. secteurs, sectors
4. groupes, groups
5. qualité, quality
6. cours, court
7. consiste, considers

and these are not :

1. gouvernement, government
2. grands, brands
3. couleur, colour
4. mains, gains
5. mars, March
6. fiscale, scale
7. nombre, number
8. erreur, error

As can be seen in the 2 lists above, there are pairs of words that are false positives or false negatives. However, generally speaking, if the first four characters match for a pair of words in a restricted span of translation text, the likelihood that they are cognates is high. Although some false cognates can be included in the cognate pair lists, and some true cognate pairs cannot be recognized, examples of accidental overlap of forms that

cause misidentifications in alignment take up only a small proportion. In addition, the cognate-based alignment model proposed by Simard, Foster and Isabelle (1992) is easy to implement and fast in speed, and has the advantage of being able to effectuate stemming by truncating variants of different conjugations, derivations and inflections.

Also, we can set rules for frequent morphological or syntactic properties that correspond reliably in specific language pairs although these properties do not bear much orthographic resemblance. For example, in many cases, the English ending 'ic' appears as 'ique' in French. This means that if an English word ends with the sequence 'ic', we may immediately consider the possibility that the sequence can be replaced by 'ique' in French. Thus for the English 'statistic' we can have 'statistique' in French.

3.2.3 Dice Similarity Coefficient

Dice's similarity coefficient was originally developed in the field of biology (Dice 1945) to describe the degree of similarity between two species of plant according to the number of features that they had in common. In natural language processing, this coefficient is widely used to measure similarity between words and sentences of the same language or of two languages.

Adamson and Boreham (1974) adapted the Dice coefficient as a measure of orthographic similarity between two words. Their technique is to compare words using the number of matching bi-grams or groups of two written letters in the string. Dice's similarity coefficient S is calculated as

$$S = \frac{2a}{(b+c)}$$

where a is the number of matching bigrams; b and c are the total number of bigrams in each term.

McEnery and Oakes (1996) used Dice's coefficient to measure the degree of similarity of word pairs in two languages, such as *colour* in English and *couleur* in French. Words are first separated into lists of adjacent characters or bigrams, and then bigrams that match are counted. For example for the candidate pair 'government' and 'gouvernement', the bigrams are:

go ov ve er rn nm me en nt

go ou uv ve er rn ne em me en nt

The number of matching bigrams (go ve er rn me en nt) is 7, while the total number of bigrams in both words is 9 + 11 = 20. Dice's similarity coefficient is twice the number of matches, divided by the total number of observable features – in this case bigrams – which is (2*7)/ 20 = 0.7. This provides a measure of the average match, that is, the number of real matches over the number of possible matches. If the result of applying the formula is 1, there is an exact match; if the result is 0, then there is no match at all. For the word pair 'government' and 'gouvernement', Dice's coefficient is 0.7, which is a fairly high figure. For the two words in this particular case there is a very good chance that they are a cognate pair.

Empirical results for English-French (McEnery and Oakes 1996; Smadja 1996) and English-Polish (Lewandowska-Tomaszczyk *et al.* 1999) showed that the greater the similarity coefficient between two words in two languages, the more likely it is that these

two words are mutual translations. Dice's coefficient is also used in word translation co-occurrence evaluations. It is assumed that if two words are translations, they tend to occur with roughly the same frequency in approximately the same regions of texts, and Dice's coefficient can be employed to measure the co-occurrence ratio. Smadja *et al.* (1996) took this approach and used Dice's coefficient in the *Champollion* system to identify individual word pairs that are highly correlated.

Another way of using Dice's coefficient is to establish correspondent alignment links among candidate sentence pairs. In this case the similarity coefficient scores at the sentence level are computed and compared. Suppose we are trying to find a possible match from one sentence in English and two sentences in French, we have the two candidate alignments:

English sentence 1 = e1, **e2**, e3, e4, e5, **e6**, e7, e8, e9, e10, e11.

French sentence 1 = f1, f2, **f3**, f4, f5, f6, f7, f8, f9, f10, **f11**, f12.

In this sentence pair, e2 and f3, e6 and f11 are two cognate pairs.

English sentence 1 = e1, e2, **e3**, e4, e5, **e6**, e7, e8, e9, **e10**, **e11**.

French sentence 2 = f1, **f2**, f3, f4, f5, f6, f7, **f8**, f9, f10, f11, f12, f13, **f14**, **f15**.

In this sentence pair, e3 and f2, e6 and f8, e10 and f14, e11 and f15 are cognate pairs. If we consider the number of matching cognate pairs and the total number of words in each sentence, we can use Dice's coefficient to compare the scores of similarity, and we can

arrive at a value that indicates which of the two French sentences in the possible alignments can form a likely translation pair with the English sentence.

3.2.4 Dynamic Programming

Dynamic programming (DP) was the brainchild of an American mathematician, Richard Bellman. It has been used to compare sequences, time-warping functions, and continuous functions (Sankoff and Kruskal 1983). In molecular biology, DP can be used to compare and align RNA sequences and DNA strands. In computer science, the dynamic programming approach refers to a collection of algorithms that can be used to compute optimal substructures. In natural language processing, DP can be used for decision making when two elements partially match and it must be decided what has to be done in order to make the elements match completely. DP is used in many corpus processing tools and analysis algorithms. For example, it is the mathematical equivalent to the Viterbi algorithm used for resolving ambiguous tags in the CLAWS part of speech tagger (Garside 1987). In both K-vec and DK-vec (Fung and Church 1994; Fung and McKeown 1997), dynamic programming is used to compute and compare the similarity or dissimilarity scores between vectors. In the alignment model proposed by Brown *et al* (1991), dynamic programming is used in two passes: first to determine the alignment of anchors, and then to establish alignment of the text between the anchors. Gale and Church (1993) introduced operations in addition to the basic dynamic programming algorithm of insertions, deletions and substitutions. In their length-based alignment model, Gale and Church also considered 2:1 or 1:2 correspondences, denoting that two sentences of one language correspond with just one of the other, as well as 2:2 "merges".

They also allowed for the fact that some alignment types are more commonly encountered in real data than others, and assigned higher penalties to less frequently alignment types such as 2:1 or 1:2. Here is how DP works in the Gale-Church model:

Let D(i, j) be the lowest cost alignment between sentences s_1, …, s_i, …, s_I and t_1, …, t_j, …, t_m. The base is D$_{(0,0)}$ = 0.

$$D(i,j) = \min \begin{cases} D(i, j-1) + \text{cost}(0:1 \text{ align } \phi, t_j) & \text{Inserting} \\ D(i-1, j) + \text{cost}(1:0 \text{ align } s_i, \phi) & \text{Deleting} \\ D(i-1, j-1) + \text{cost}(1:1 \text{ align } s_i, t_j) & \text{Substituting} \\ D(i-1, j-2) + \text{cost}(1:2 \text{ align } s_i, t_{j-1}, t_j) & \text{Expanding} \\ D(i-2, j-1) + \text{cost}(2:1 \text{ align } s_{i-1}, s_i, t_j) & \text{Contracting} \\ D(i-2, j-2) + \text{cost}(2:2 \text{ align } s_{i-1}, s_i, t_{j-1}, t_j) & \text{Merging} \end{cases}$$

In addition, dynamic programming can be used for cognate detection and word alignment. First, we ask how much we have to change one word in order to make it match the other, and what the cost is of achieving the match. For instance, we might suspect that the French word 'prix' is the match for the English word 'price'. The objective is then to modify 'prix' with minimal cost to make it match 'price'. There are possible operations like insertion, deletion and substitution for modifying or rewriting a word, each of which is associated with a certain cost. In this case, we would have to substitute the 'c' for 'x' and then delete 'e' or delete 'c' before we replace 'e' for 'x'. The cost of the modification operations can be adjusted in light of the language pairs involved such that the cost of substituting English 'er' by French 're' as in *october-octobre* and *september-septembre* should be much lower than, say, for changing the 'er' in English to

an 'ir' or an 'or' in French. Normally when the lengths of two words are equal, the lower the total cost of making the match is, the more likely it is that the two words are cognates.

3.3 Factors Affecting the Performance of Algorithms

The performance of a bilingual text alignment algorithm depends on some identifiable factors. In many cases, an alignment algorithm can achieve good results for a language pair such as English and French, but when the language pair is English and Chinese, the performance falls short of our expectation. Sometimes, we can even make predictions about whether the performance will increase or decrease on the basis of the factors. In designing algorithms for text alignment of a bilingual corpus, we have to keep these factors in mind, and tune some parameters accordingly so that our aligners can obtain the best performance.

3.3.1 Language Difference

Some aspects of human language are universal or near-universal, others diverge greatly. Languages like Japanese and English may differ substantially from each other in their alphabet, morphology and syntax. As a consequence, various assumptions that were used in some alignment programs do not hold for such disparate languages, due to radical differences in total text length, partitioning to words and sentences, part-of-speech usage, and word order. The high complexity of morphology for some languages may require complex monolingual processing prior to alignment. For example, Haruno and Yamazaki (1996) showed that for structurally very different languages, function words impede

alignment. For this reason, they eliminate all function words in the corpus using a POS Tagger. They also suggested the use of a language specific dictionary for matching word pairs to avoid sparse data problems in aligning short texts with the Kay and Röscheisen method (Manning and Schütze 1999:483).

For some languages, the first challenge we face is to segment the texts. Writing systems such as that of Chinese and Japanese do not have word boundaries marked. And languages like modern standard Arabic and Chinese tend to have sentences that are quite long. In many cases a sentence can be as long as a paragraph in English. In Chinese, the majority of words are just one or two characters long, although collocations up to four characters are also common. At the same time, there are several thousand characters in daily use, unlike the 26 alphabetical letters in English. Such lexical differences make it even less obvious whether pure sentence-length criteria are adequately discriminating for statistical alignment. Even when we know that we have to segment the Chinese sentences for alignment, we have different standards to segment them. Bing Zhao *et al.* (2003) described three approaches of segmenting Chinese sentence strings.

- Both English and Chinese sentence are measured in bytes
- Both English and Chinese sentence are measured in words
- English sentence is measured in words and Chinese sentence is measured in bytes

In addition, word order arrangements in non-Indo-European languages can depart greatly from Indo-European languages. Languages such as English, French and Chinese have SVO (Subject-Verb-Object) structures, but languages like Japanese and Hindi have

SOV order. In the following examples, the Japanese sentences have their verbs located around the end of a sentence, while the English translations are SVO structures.

j_1. 私は音楽特にロックが大好きだ。

e_1. I love music, especially rock.

j_2. 私も音楽が大好きです。

e_2. I love music, too.

For some languages, although they have the same SVO structure, there are part of speech divergences:

English: She likes/VERB to sing

German: Sie singt gerne/ADV

English: I'm hungry/ADJ

Spanish: tengo hambre/NOUN

For alignment models based on sentence length and for IBM Model 1, this doesn't matter much since they don't take the word order into account. However, for those alignment algorithms that depend on lexical context information, dissimilarities in linguistic structures can adversely affect the performance of the algorithm. Usually the models will fail to efficiently capture the correspondence properties between text sections, if the structures of two languages are drastically different. We may need to explore ways to do the reordering, which might result in a higher search error rate.

Moreover, some languages may share more cognates than others, and this can be another factor that can affect performance in parallel sentence alignment. Take the Gale-Church model as an example. Although it has been suggested that length-based methods are language-independent (Gale and Church 1991), they may in fact rely, to some extent, on length correlations arising from the etymological relationships of the languages involved. If the two languages in the bilingual corpus share many cognates, chances are good that the length of the sentence and the length of its translations are correlated.

3.3.2 Difference in Translation Style

In bilingual corpora, the level of correspondence between texts varies as a result of the style of translation. Translation of a text can be fairly literal or it can be of a free style. It can even be a re-creation or anything between the two extremes. The degree of comparability of translated texts will affect what we can do with the alignment of text segments and the mapping of translation correspondences.

The translation correspondence variance can be different from genre to genre: it is less predictable in narrative fiction, for example, than in technical language. In narrative fiction translations, we are more likely to have absence of one-to-one correspondences. For the more paraphrasing style of translation, text alignment will be more difficult.

The more literal and consistent the translation is, the easier it is to apply alignment algorithms, particularly statistical alignment algorithms. Of course, the easiest would be texts with word-to-word translations, data collections of which are relatively few and far between in the real world. In a free style of translation, paragraphs and sentences can be dropped or added; sentences can also be merged or split. Sometimes, sections of texts are

so noisy that we can only call them partially-paralleled sections. For those sections of texts, we cannot reliably do a 1:1 sentence alignment. Most of the alignment will be many:many, many:0, 0:many, or n:m (n and m are integers). To some extent, this type of corpora may fall into the category of comparable corpora, but they can still be used for parallel word or phrase extraction. Corpora like the Hong Kong News Corpus and the Xinhua News Corpus are in fact of this nature. The texts are only rough translations of each other, focusing on the same thematic topics, with some insertions and deletions of paragraphs. Here is an example for a potential bilingual corpus. It is collected from two sections of online news.xinhuanet.com:

> WASHINGTON, Dec. 19 (Xinhua) -- U.S. President George W. Bush said Tuesday that he plans to expand the size of the U.S. military to meet the challenges of "a long-term global war against terrorists."
>
> In an interview with The Washington Post at the White House, Bush said it was a response to warnings that sustained deployments in Iraq and Afghanistan have stretched the armed forces to near the breaking point.
>
> He said he has instructed newly sworn-in Defense Secretary Robert Gates to report back to him with a plan to increase ground forces.
>
> The president gave no estimates about how many troops may be added but indicated that he agreed with suggestions in the Pentagon and on Capitol Hill that the current military is stretched too thin to cope with the demands placed on it.

The decision comes at a time when he is rethinking his strategy in Iraq and considering, among other options, a short-term surge in troop levels to try to secure violence-torn Baghdad.

In describing his decision, Bush tied it to the broader struggle against extremists around the world rather than Iraq specifically.

新华网华盛顿１２月１９日电（记者潘云召 杨晴川）

美国总统布什１９日说，他计划扩大美军规模，以应对全球反恐战争的挑战。

布什在接受《华盛顿邮报》记者采访时说，他已要求新近上任的国防部长盖茨与国防部官员讨论之后提出具体的增兵计划。目前美军兵力过于分散，军队难以达到对其提出的要求。但布什没有说明美军具体增加多少人。

布什表示，增加美军人数不仅是考虑到伊拉克战争，而且考虑到了在全球打击恐怖分子的需要。

As we can see, the source language and the target language texts differ in size because of the style of translation. Here the size of the English texts is bigger. Although mostly the two texts are talking about the same thing, the sentence alignment would be hard on this one because the passage is translated with a very loose and free style. In addition to the fact that the paragraphs do not match, some of the phrases in English do not have translations in Chinese. All these factors affect the performance of an algorithm in terms of, say, precision and recall values.

3.4 Summary

This chapter has provided an investigation of previous work in bilingual text alignment. In this chapter, we introduced two approaches of bilingual text alignment, one is the statistical method such as the Gale and Church algorithm, and the other is the lexical method represented by the Kay and Röscheisen algorithm. In the meantime, we looked at some hybrid text alignment models which combined the two approaches. It is found that statistical methods can post gains in speed while lexical methods can render more accurate alignment results. There are four alignment models that are of particular interest to us: the Gale-Church algorithm, IBM Model 1, The Kay and Röscheisen algorithm, and the K-vec algorithm. We gave a detailed description of each of them. In addition, we identified some basic textual elements and key methods that are shared amongst most of the text alignment algorithms. These elements and methods include anchor points, cognates, Dice similarity coefficient and dynamic programming. We argued that the same alignment algorithm could yield different results for different types of texts or for texts of different languages. Therefore, in designing parallel text mapping algorithms, we should take into consideration factors such as language difference and translation style that are likely to affect the performance of bilingual text aligners.

Chapter 4

Framework of the Bitext Mapping Intelligent Agent

The objective of the Bitext Mapping Intelligent Agent (BMIA) model is to carry out tasks of aligning text segments for building bilingual corpora, mapping translation correspondences and detecting potential translation problems. To achieve this objective, we designed two major systems for BMIA: the StatCan Daily Translation Extraction System (SDTES) and the StatCan Bilingual Text Comparison System (TextComp). This chapter contains two main parts that describe these two systems respectively.

In the first part, we will concentrate on the protocols and algorithms we adopted for SDTES. We will explain how BMIA aligns the StatCan *Daily* news release texts, and how the agent detects and filters regions of texts that are misaligned. Evaluation results are presented of the performance of the algorithms for text alignment and for text misalignment detection. We will also introduce the StatCan Daily Corpus (SDC) which includes all the aligned translation pairs that are processed in SDTES. At the same time, the main features of the translation concordance search system (TransConcord) that we designed for BMIA will be described in this part of the chapter.

The second part of this chapter describes the StatCan Bilingual Text Comparison System (TextComp) which is designed to align and compare more noisy bilingual texts. In this part, we will first present our algorithms for bilingual text alignment at the paragraph level and at the text segment level. Then we will describe how BMIA parses the aligned text segments and compares the bitexts to identify potential translation

problems, and how association links are established in Translation Correspondence Profiling (TCPro). The TCPro scoring metric that measures the levels of correspondence between translation constituents will be explained. In addition, there will be a discussion section focusing on challenges and problems that TextComp encounters in its effort to accurately align the texts and to efficiently compare the texts for the identification of translation discrepancies.

For BMIA, the chief purpose of bilingual text mapping is to extract translations from web-based materials to build a bilingual corpus and to establish translation correspondence links, so that translation discrepancies can be detected, translation correspondences can be profiled, and the degree of association can be scaled. The BMIA approach for parallel text mapping is a composite one with integrated statistical and lexical methods. The assumption underlying the main algorithms in this chapter is that translation equivalence is compositional: translated source text has its textual features reflected, either implicitly or explicitly, in the target text through choice of words, morphological features, styles, structures etc. Bitext mapping analysis can thus be translation representation analysis. The global representational correspondences between source and target texts are analyzable into sets of finer anchoring points. These individual points of features can help us decide if a candidate pair of text segments is of a good matching relation. If in the text mapping process, some key textual properties do not match or are missing in either the source text or the target text, then chances are good that there is an error in alignment or translation.

4.1 BMIA in Translation Extraction for the StatCan Daily Corpus

Texts used in the building of bilingual corpora can come from a variety of resources. Some of these resources are unannotated transcriptions of recorded meeting minutes, scanned copies of literary works, and data collections of text-only flat files. For the building of the StatCan Daily Corpus (SDC), the bilingual texts are from the previous translations published on the official StatCan website. The need for a domain-specific or text-genre specific bilingual corpus such as SDC is obvious for StatCan: it can be used in translation memory systems, in bilingual information retrieval systems, and in the updating of the terminology bank, so that when people are writing releases for *the Daily*, their translations can be more consistent and standard. The corpus data can be exported to other translation memory systems and information retrieval systems to benefit other departments and institutions in Canada. The aligned translation segments can be the gold standard in the prepublication quality check process. SDC can also be a valuable source for translation studies and statistical machine translation systems. In addition, BMIA is designed to mine widely available web resources to bridge the gap in the scarcity of bilingual data resources, and this will have implications in how to explore online resources for the building of parallel corpora. Automatic translation extraction has the potential of enriching currently available electronic translation resources with limited manpower and lexicographical expertise (Tufiş *et al.* 2004), and the BMIA model brings this advantage into full play.

There are two major parts in the translation extraction system: a text alignment part and a misalignment detection part. The bilingual text aligner includes a two-round

procedure that adopts the Gale-Church statistical model for the alignment of web contents. Without this two-round procedure, using the Gale-Church length model to align *the Daily* HTML texts would have either become impossible or would have generated many spurious correspondences. The misalignment detection part integrates different alignment techniques, and takes advantage of some important textual and structural information in HTML texts. It can help distinguish correctly aligned pairs that are true translations from those pairs that are misaligned and unusable. There is a two-step cognate extraction method to generate cognate lists that could be used to assist the text alignment process for the purpose of detecting misaligned translation pairs. Results show that BMIA can generate rather clean translation pairs – translation pairs that are of good quality and almost error free as far as alignment is concerned – because the few problematic translation pairs can be automatically identified and eliminated. The precision and recall of the alignment mechanism and of the misalignment detection mechanism are very high. The algorithms and methods are straightforward, robust and can be easily applied to most of the government web materials in Canada, where federal government websites have to present content in both English and French.

4.1.1 Data Preparation

The StatCan Daily Corpus is a bilingual corpus consisting of English and French texts of *the Daily* news releases of Statistics Canada. Each release text is an HTML document published officially at www.statcan.ca. The file name of the document bears information about the language in which it is published and the date of publication. Release texts can contain expository texts, numerical tables, notes, graphs and note-to-reader text chunks.

The Daily is the flagship publication of Statistics Canada, and arguably the most important document of the agency. It is published every working day, in two official languages. Usually it publishes an average of 4 to 5 text releases in a day. Some important major releases are long, running for pages, and some regular minor releases are short, containing only a few lines of texts. It is Agency policy that every new product or data set at Statistics Canada must be announced to the public through *the Daily* in one form or another. Therefore it is Statistics Canada's main vehicle for informing the public about much needed information items such as key indicators of economy, consumer price index, labor force survey, population census and many others.

Producing *the Daily* is a cooperative effort. There are people providing draft contents, people assembling and processing the texts and tables, people generating charts and graphs, people editing and polishing the texts, people checking the contents, links, and styles, and people disseminating the finished documents. As there are so many people working on *the Daily* on a daily basis, there are guidelines for the submission of release articles, templates for the recurring releases, rules for the use of HTML markups, and procedures for quality checking. There are dedicated professional editors on the job to make sure that *the Daily* release texts are consistent in style of writing, and that the same content meaning is accurately conveyed in both languages. As a result, the finished products of *the Daily* can be expected to be good quality translations, free from variations like deletions and insertions. As the first line of communication between Statistics Canada and the public, *the Daily* texts are domain-specific and are standard written (in contrast with spoken) language in nature. Thus, we can expect to find many

correspondences in structural features like the use of cognates and HTML markups (see Table 4.1) in English and French.

English	French
<IMG SRC="/english/dai-quo/SSI/dailylogo.gif" border=0 ALT="The Daily"> <B> <A HREF="td061102.htm">Thursday, November 2, 2006</A> </B> <H2>Study: Neighbourhood characteristics and the distribution of crime in Regina</H2> 2001 <P>This study, the third of its kind by Statistics Canada, investigated neighbourhood-level crime patterns in Regina by examining how police-reported crimes are distributed across city neighbourhoods, and whether the crime rate in a given neighbourhood is associated with factors specific to that neighbourhood, such as the incidence of low income, the education level of residents, the housing conditions and land-use characteristics.</P>	<IMG SRC="/francais/dai-quo/SSI/dailylogo_f.gif" border=0 ALT="Le Quotidien"> <B> <A HREF="tq061102.htm">Le jeudi 2 novembre 2006</A> </B> <H2>Étude : Caractéristiques des quartiers et répartition de la criminalité à Regina</H2> 2001 <P>Cette étude, qui est la troisième en son genre réalisée par Statistique Canada, a servi à étudier les modèles de la criminalité à l'échelon des quartiers à Regina en examinant la répartition des actes criminels déclarés par la police entre les quartiers de la ville et le lien possible entre le taux de criminalité d'un quartier et des facteurs qui lui sont propres, comme la fréquence du faible revenu, le niveau de scolarité des résidents ainsi que les caractéristiques liées aux conditions de logement et à l'utilisation du territoire.</P>

Table 4.1. Similarities between HTML structures of two languages

However, because of the writing conventions of different languages, there are sometimes structural differences in expressing the same content. For example, in some texts we use a superscript in the French text, while we do not need to do so in English. In

addition, because *the Daily* is also published in a PDF version, to avoid graphs and tables crossing over pages, we may have floating charts and tables. This means that charts, graphs, and tables can end up in different positions in texts of different languages. This can be demonstrated in Figure 4.1 where the floating positioning of the "Note to readers" text block disrupts the order of paragraphs. Meanwhile, in the published editions of *the Daily* we still have texts that are quite noisy in structural markups, and texts that have errors in translations.

The first batch of texts collected for the StatCan Daily corpus are the release texts of *the Daily* published from 2004 to 2006. Since every file is officially published on the Statistics Canada website, the release texts can be obtained by crawling the archive section of the http://www.statcan.ca/. Once all the data files are assembled, the next thing is to preprocess the HTML documents to pave the way for the application of the statistical alignment models. The preprocessing involves the removal of the header part and the footer part of the HTML documents. Special French HTML coding is converted so that most of the accent characters in French can be properly and consistently handled. Another important task in the preprocessing stage is to re-organize some structures of the documents. As can be seen in Figure 4.1, *the Daily* release texts can contain frequent floating tables and graphs. For reasons of PDF pagination, some charts and text blocks can chop off different numbers of paragraphs in between. If texts are aligned as they are, the length based alignment model can easily generate confusing alignment pairs. To correct this problem, all the tables, graphs, charts and blocks of texts that can be in floating positions are moved to the end of each document.

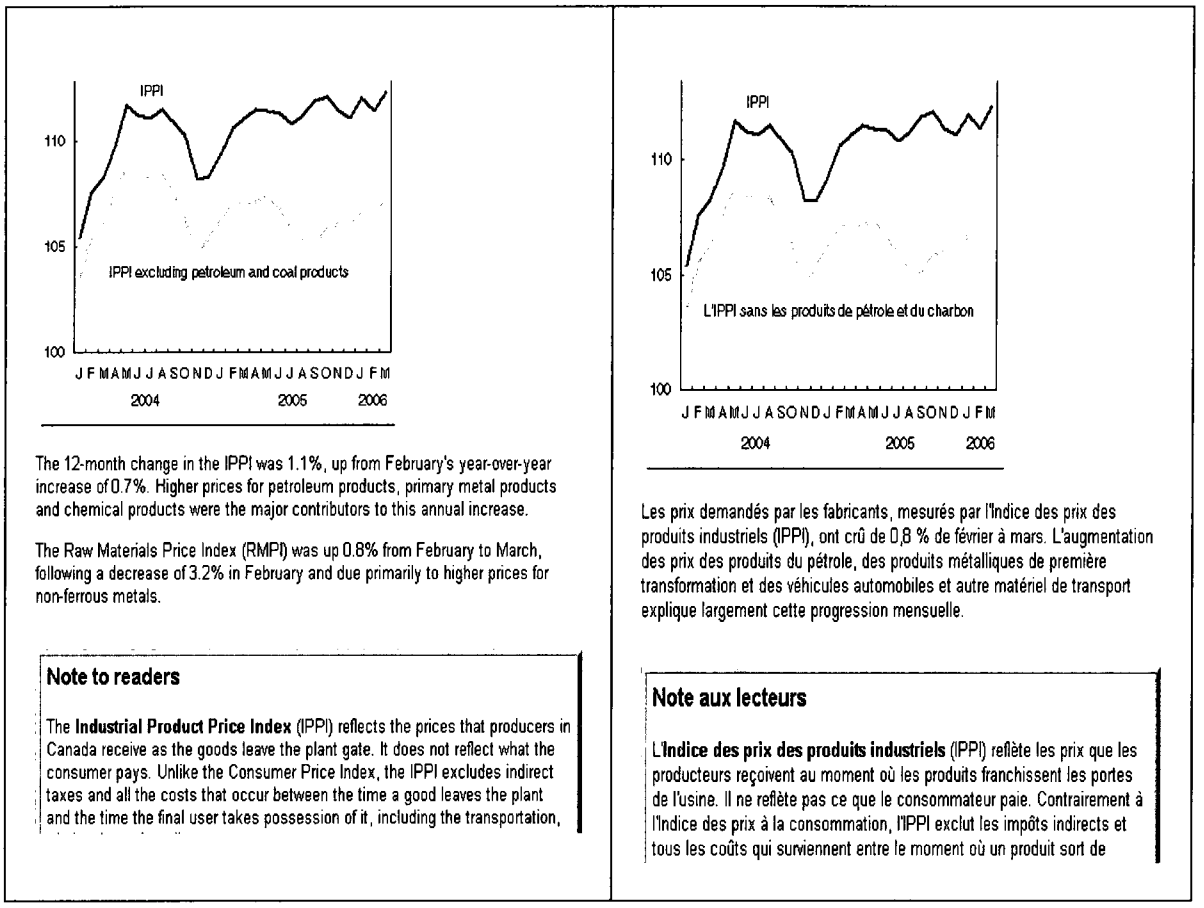


Figure 4.1. Floating images and tables disrupting the order of paragraphs

4.1.2 Text Mapping Using the Gale-Church Statistical Model

BMIA uses the Gale and Church statistical alignment model (Gale and Church 1991) for the text alignment of the StatCan Daily web pages. Here, the task is to extract translations from web-based materials. Generally speaking the published *Daily* pages are consistent translations. If there are deletions in translation for *the Daily* materials, they are very few in number. For this type of bilingual material, Gale and Church method of alignment can be a good choice. A detailed formal description of the Gale and Church algorithm can be found in Section 3.1.1.

The Gale-Church algorithm accepts input from two text files at a time: one text file for each of the languages. The preparation of the text files involves (1) breaking the text files into lines of words, one word per line and (2) adding two types of place-holding markers: one for the end of the paragraph (`.EOP`) and one for the end of the sentence (`.EOS`). For the application of the Gale-Church model, the actual sentence-ending or paragraph-ending markers such as `.EOS` and `.EOP` can take arbitrary names, but the number of paragraph-ending markers (`.EOP`) in the English file should be equivalent to the number of paragraph-ending markers in French. If the paragraph numbers are different, the program will not proceed. The variance in the number of sentences (or `.EOS`) will not matter as much because in aligning sentences, we can have different matching types such as 1:2, 2:1, 1:0 and 0:1.

It is not uncommon to find a discrepancy in the number of paragraphs in *the Daily* document pairing process. Usually the gap is not great. Nevertheless, as noted above, a slight difference of one or two can bring the alignment process to a halt. It would be rather time-consuming to manually identify the exact places where the end-of-paragraph markings are missing, or are to be inserted. To tackle this problem, we defined the two types of boundary marking symbols in the first round of text alignment as follows:

`.EOA` is a paragraph pseudo-boundary marker that indicates a block of text containing, in most cases, a series of paragraphs separated by a main HTML element. In the first round of alignment, an `.EOA` marked text block is treated as a "large" paragraph.

.EOP is a sentence pseudo-boundary marker that actually indicates the end of a paragraph. An .EOP marked paragraph, though it usually contains more than one sentence, is considered a "large" sentence in the first round of text alignment.

In SDTES, a limited number of HTML tags that can be used to mark the beginning of a text block are categorized as the main HTML elements. They include tags such as `title`, `table`, `h1`, `h2`, `h3`. In the first round of text alignment, SDTES counts these few main HTML elements to see if the pair of text files has the same number of main HTML features in them. If the numbers are the same, the system splits the texts into blocks separated by these feature HTML tags. After this, the system marks these text blocks using the paragraph pseudo-boundary symbol `.EOA`, and marks the original paragraph ending places using the sentence pseudo-boundary marker `.EOP`. If the numbers of major HTML elements are different, the system treats the whole text document as a single pseudo paragraph and the original paragraphs as pseudo sentences. So the parameters for the first implementation of the Gale-Church algorithm are:

```
align -D '.EOA' -d '.EOP' filename_en filename_fr
```

Recent research has shown that some types of HTML mark-up codes can help guide the initial splitting of text in bilingual text alignment (Sanchez-Villamil *et al.* 2006). In the first round of text alignment for SDTES, we focused only on a few selected HTML tags that are almost certain to appear in the texts of both languages. An obvious advantage of doing the macro level alignment based on these main HTML elements is that by introducing the HTML structural information into the statistical method, we can avoid misalignment across text blocks. Even if there are misalignments, they will be kept

within a minimum text region. At the same time, by using only a few key HTML tags, we can avoid some problems that are likely to arise when different HTML tags are used for the same representational effects in the texts of two languages or when some structural tags such as `<tbody>` and `<ul>` are missing in one part of the translation units.

*** Link: 1 - 1 ***
<H3>Mineral wool including fibrous glass insulation <hpara=e6:f7> .EOS
<H3>Livraisons de laine minérale y compris les isolants en fibre de verre <hpara=e6:f7> .EOS

*** Link: 1 - 1 ***
</H3> .EOS
</H3> .EOS

*** Link: 1 - 2 ***
February 2000 <p>Manufacturers shipped 1 879 513 square metres of R12 factor (RSI 2.1) mineral wool batts in February, down 26.8% from 2 566 724 square metres in February 1999 and down 14.8% from 2 206 111 square metres in January. .EOS Year-to-date shipments to the end of February totalled 3 560 137 square metres, a 36.0% drop from the same period in 1999.</P> .EOS
Février 2000 <p>En février dernier, les fabricants ont livré 1 879 513 mètres carrés de laine minérale de facteurs R12 (RSI 2.1) en nattes, en baisse de 26,8 % par rapport aux 2 566 724 mètres carrés livrés en février 1999 et en baisse de 14,8 % comparativement aux 2 206 111 mètres carrés livrés le mois précédent.</P> .EOS <p>Les livraisons cumulatives pour 2000 à la fin de février se situaient à 3 560 137 mètres carrés, en baisse de 36,0 % comparativement à la même période en 1999.</P> .EOS

*** Link: 1 - 1 ***
<p><B>Available on CANSIM: matrices <a alink>40</A> and <a alink>122</A> (series 32 and 33).</B></P> .EOS
<p><B>Données stockées dans CANSIM: matrices <a alink>40</A> et <a alink>122</A> (séries 32 et 33).</B></P> .EOS

*** Link: 1 - 1 ***
<p>The February issue of <EM>Mineral wool including fibrous glass insulation</EM> (<a alink>44-004-XIB</A>, $5/$47) is now available. .EOS See <I>How to order publications</I>.</P> .EOS
<p>Le numéro de février 2000 de <EM>Laine minérale y compris les isolants en fibre de verre</EM> (<a alink>44-004-XIB</A>, 5 $ / 47 $) est maintenant en vente. .EOS Voir <I>Pour commander les publications.</I></P> .EOS

Figure 4.2. Results after the first round of alignment using the Gale-Church algorithm

To do the second round of alignment, the software agent automatically reconstructs the English document and the French document from the aligned paragraphs of the output file. When the first round of text alignment has been completed using the length-based

model, the alignment results are put in a file ending in .al with translation text blocks containing paragraphs clearly indicated (see Figure 4.2).

However, to realign the texts, the aligned pairs have to be separated into two files: one for the English texts, and the other for the French texts. So, the texts in different languages were reassembled and new definitions were given to the two types of boundary markers:

.EOP marks the end of each aligned text block in the output file that is generated as a result of the first round of alignment. In most cases it contains one paragraph, but in some cases it can contain two paragraphs.

.EOS is the text segment ending symbol in the second round of text alignment. It marks the end of a text segment which in most cases includes one sentence.

By reorganizing the text structures on the basis of the paragraph pairing results in the first around of text alignment, SDTES was able to reset the English and French documents to the original text format prior to the initial alignment. The two input files are processed with the newly assigned boundary symbols `.EOP` and `.EOS`:

```
align -D '.EOP' -d '.EOS' filename_en filename_fr
```

Figure 4.3 contains some examples that are produced in the second round of text alignment in SDTES. They are results of text alignment at a more fine-grained text segment level. We also call the text segment unit that is aligned with its translation counterpart in each of the translation pairs "a bead of text".

*** Link: 1 - 2 ***
In 2005, employment grew by 1.2% with gains in a number of industries, most notably educational services (+13.3%), business, building and other support services (+8.8%) and construction (+6.4%) while manufacturing continued to shed jobs (-4.1%).</P>
En 2005, le nombre d'emplois s'est accru de 1,2 %, des hausses ayant été relevées dans plusieurs branches d'activité, notamment dans les services d'enseignement (+13,3 %), dans les services aux entreprises, les services relatifs aux bâtiments et les autres services de soutien (+8,8%), ainsi que dans la construction (+6,4 %). Le secteur de la fabrication, en revanche, a continué de perdre des emplois (-4,1 %).</P>

*** Link: 1 - 1 ***
<P>Despite little change in December, employment in Alberta rose by 1.8% in 2005.
<P>Bien qu'il ait peu varié en décembre, le nombre d'emplois en Alberta a augmenté de 1,8 % en 2005.

*** Link: 1 - 1 ***
Employment jumped in professional, scientific and technical services (+21.5%) in 2005.
Les services professionnels, scientifiques et techniques ont connu une forte poussée de l'emploi (+21,5 %) en 2005.

*** Link: 2 - 1 ***
Employment also increased in educational services (+11.2%) over the same period. Natural resources continued to strengthen (+5.4%), the result of intense oil and gas activities.
L'emploi a également progressé dans les services d'enseignement (+11,2 %) au cours de la même période, et le secteur des ressources naturelles a continué de s'affermir (+5,4 %), grâce au dynamisme des industries pétrolière et gazière.

*** Link: 1 - 1 ***
The unemployment rate closed the year at 4.1%, unchanged from November.</P>
Le taux de chômage est resté inchangé par rapport à novembre et s'est établi à 4,1 % à la fin de l'année.</P>

Figure 4.3. Paired text segments after the second round of alignment

4.1.3 Cognates Extraction Using K-vec and AMS

In BMIA, cognates are defined as words that share a good portion, either interrupted or uninterrupted, of the character string in a reasonably computable search space. This means that cognates should be "more or less like each other in form" (McArthur 1992) and that for two words to be considered a cognate pair, they should occur in approximately the same text region. BMIA's goal of finding cognates is to use them to assist parallel text alignment in the process of misalignment detection, so it is required

that two cognate words should appear in a reasonable range of segment numbers. If a source word in French appears in paragraph 3, but the target English word is in paragraph 10, even if they are a cognate pair, they are not recruited in our cognate list. This is because the two words in this case are not within a text region of reasonable range and we do not want two widely separated sentences to be aligned as a translation pair.

The first step in the SDTES cognate extracting algorithm is to produce candidate cognate lists using the K-vec algorithm. The main objective is to find cognate candidates within an acceptable text region range and limit the number of words to be considered as cognate pairs. The K-vec method (see also Section 3.1.4) was developed by Fung and Church as a means of generating "a quick-and-dirty estimate of a bilingual lexicon" that "could be used as a starting point for a more detailed alignment algorithm ..." (Fung and Church 1994). The assumption is that if two words are translations of each other, they are likely to occur almost an equal number of times in approximately the same region in the parallel texts. In trying to find word-to-word translations, K-vec does not require any prior knowledge of sentence boundaries. The algorithm divides the two texts into K pieces and looks for word correspondences in corresponding pieces. It can be preferable to use a technique that doesn't rely on sentence boundary information to verify the alignment results of an algorithm that is heavily dependent on markers of sentence boundaries, because at times we can be uncertain about the correct correspondence of sentence boundaries in two texts. Problems usually surface if cognate pairs occur in text segments with corresponding sentence boundary information, but not in the same K piece text area. In going across sentence boundaries, the K-vec technique can capture these problems, and help detect misalignment, particularly massively misaligned text chunks. It

can also confirm the correctness of correspondence of sentence boundaries, when text segments are properly aligned. Although the *K* value can be chosen and adjusted, care should be taken not to make it too large or too small. If *K* is very large, the total number of words in each piece would be small and we may have the risk of missing translations. If *K* is very small, the number of words in each piece would be large and we lose the advantage of dividing the text into pieces for the purpose of locating a word and its translation in corresponding pieces. Fung and Church (1994) suggested that *K* be equal to the square root of the total number of word tokens in the text.

The `K-vec++` package (Pedersen and Varma 2002) is used for the implementation of the K-vec algorithm. This package was designed for applying the K-vec algorithm to finding word correspondences in parallel texts. It is called the `K-vec++` package because the package extends the K-vec algorithm in a number of ways. It is obtainable from `http://www.d.umn.edu/~tpederse/parallel.html`. Using the Perl programs in the `K-vec++` package, SDTES generates a very rough list that might contain cognates for each pair of documents. Figure 4.4 contains some sample lines of a resultant list. For each line in the list such as *industrielle<>industrial<>1 2 2*, the three numbers (*1 2 2*) indicate the corresponding *K* pieces where both the French word and the English word can be found, the *K* pieces containing the French word and the *K* pieces in which the English word occur. These results are further processed in SDTES to extract cognate pairs such as *résidentielle<>residential<>1 1 1* and *industrielle<> industrial<>1 2 2*.

production<>strong<>1 7 3	ces<>experienced<>1 1 1
alors<>rebound<>1 4 1	octobre<>several<>1 10 2
enregistrées<>growth<>1 2 2	industrielle<>industrial<>1 2 2
effaçant<>increase<>1 1 7	diffusion<>of<>1 1 19
production<>as<>1 7 4	rénovation<>by<>1 1 6
à<>sales<>1 11 3	détail<>pharmaceuticals<>1 3 2
ou<>slowed<>1 2 1	hausse<>apartment<>1 7 1
obtenir<>for<>1 1 6	croissance<>were<>1 3 3
chiffres<>stable<>1 1 1	autres<>unchanged<>1 2 3
d'appartements<>of<>1 1 19	cansim<>gross<>1 1 3
travaux<>was<>1 1 6	au<>september<>1 4 4
nombre<>air<>1 1 1	secteur<>automotive<>1 11 2
résidentielle<>residential<>1 1 1	rebond<>sector<>1 1 9
d'octobre<>gross<>1 1 3	cependant<>machinery<>1 1 2
les<>site<>1 15 1	pour<>sector<>2 4 9
connexes<>steel<>1 1 1	comptes<>our<>1 1 2
la<>an<>3 22 5	marchandises<>marked<>1 1 2

Figure 4.4. Sample candidate pairs generated by kvec.pl in the K-vec++ package

In using the K-vec++ package, special consideration is given to the *lcutoff* threshold. The *lcutoff* parameter is a threshold frequency value for a word pair to be taken as a valid corresponding pair and to be present in the output list of K-vec (Pedersen and Varma 2002). SDTES is more interested in finding cognate pairs that have close to 1:1 or 2:2 correspondence than pairs that have a frequency ratio of, say 100:200. The high ratio matching pairs, when used as lexical clues in text alignment, can be computationally expensive, and indiscriminative as an anchoring feature. For example, if the candidate cognate pair information is *industrielle<>industrial<>1 20 2*, it means that *industrielle* appears in twenty pieces of the French text, *industrial* appears in two pieces of the English text, and in only one corresponding piece, they co-occur. This would indicate that the chance of these two words being in the same translation unit is very small, and thus this candidate cognate pair does not have much value in helping detect misalignments in the context. On the other hand, if the *lcutoff* value is low, the French word could have a better chance of pairing with the English word in the corresponding *K* piece. For

example, when the word correspondence information is *industrielle<>industrial<>1 2 2*, it would mean that the chance of *industrielle* and *industrial* being in the same translation unit is high. This is also one of the reasons why, in the candidate cognate list in Figure 4.4, we see so many noisy pairs that are not cognates. As soon as the rough candidate cognate list is produced, there is a filtering device in SDTES to keep only those pairs that have the last three numbers in the output line (such as *industrielle <>industrial<>1 2 2*) very close to each other. If the last three numbers for each candidate cognate pair in the list are represented by h, i and j respectively, the system considers only those word pairs with $\max(h, i, j)<10$ and $\min(h, i, j)>0$. Further constraints are: if $\max(h, i, j)>3$, then $\min(h, i, j)\geq 0.5\max(h, i, j)$. Otherwise, $\min(h, i, j)\leq 3$. These constraints are determined based on our testing results with sampled k-vec outcome lists of *Daily* release texts. Using these thresholds, the system can reject those spurious correspondences like *rebond<>sector<>1 1 9* or *pour<>sector<>2 4 9* before candidate pairs of cognates are further processed.

The second step in the extraction of cognate pairs is to apply our pattern matching algorithm, the Acceptable Matching Sequence (AMS) search. An AMS has two non-overlapping substrings that can be matched in the same order in both of the words in a cognate pair. The algorithm extracts two substrings (θ^a and β^b) from a source word, say a French word (W_1), with a length threshold (T) for the two substrings combined. Then it searches for the string sequence that contains the two substrings in the same order in the target English word (W_2). Skipping some characters is acceptable before, after or between the two substrings; we call these "Don't Care Characters" (DCCs). The initial value a in θ^a is set to 0, and b in β^b to T. If a match does not occur, one substring θ^a is increased in

length ($a=a+1$) while the other substring β^b gets decreased ($b=b-1$). The search continues till a two-substring match is found or $a>T$ or $b<0$. An AMS can be defined as in the following regular expression:

$$W_1 = (.)^*\theta^a (.)^{\{0,c\}}\beta^b(.)^*$$

Where:

$(.)^*$ is a substring of any character combinations. This substring can also be an empty substring.

$(.)^{\{0,c\}}$ is a substring of 0 to c characters. $0<c<4$.

θ^a is the first substring to be matched in W_2. The length of this substring is a.

β^b is the second substring to be matched in W_2. The length of this substring is b.

Let x and y be the lower bound and upper bound for the lengths of W_1 and W_2, and z is the length difference threshold. $x \leq length(W_1) \leq y$ or $x \leq length(W_2) \leq y$. $|length(W_1) - length(W_2)| \leq z$; if $y>10$ then $0 \leq z \leq 4$, otherwise $0 \leq z \leq 3$.

Let T be the combined length of θ^a and β^b. $0 \leq a \leq T$, $0 \leq b \leq T$, $a+b=T$. SDTES sets the T parameter with reference to the upper bound y for the lengths of W_1 and W_2. The system discards word pairs with $y<4$. $T=8$ if $y>10$. For all the rest, SDTES uses the simple linear regression model $T=0.5y+1.8$ to compute the threshold value. The linear regression model is derived from the regression analysis of a hand-picked collection of cognate pairs from *the Daily* release texts.

The AMS search model can be demonstrated as in Figure 4.5. W1 becomes an AMS only when the two substrings (θ^a and β^b) are both matched in the correct order in W2. DCC1, DCC2, DCC3 are "Don't Care Characters". They don't need to be matched in W1 and W2. For example, if the length of the longer string of two words (W1 and W2) is 9 and the length difference between the two words is 3, the system sets $c=2$ and $T=6$.

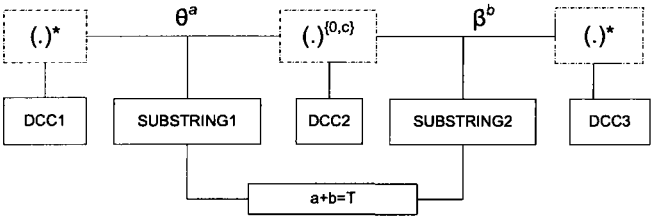


Figure 4.5. AMS search model for $x \leq length(W_1) \leq y$.

Figure 4.6 represents one of the "worst-case" AMS search problems in SDTES. Here, we want to find cognates from two candidate words of which the longer one has a length of more than 10 characters (y=13). Here θ^a = "r", β^b ="sidenti", a=1 and b=7. According to the criterion for the minimum combined matching length requirement, T is 8 when y >10. The two substrings can be separated by 0 to 3 characters ($c=3$). The length difference between the two words is less than or equal to 4 ($|\ length(W_1) - length(W_2)| \leq z;\ z=4$). In the solution to the AMS search problem, we are interested in finding if the properties of the two substructures are shared in both of the words W_1 and W_2. If the two substrings θ^a and β^b are exactly the same in the two words, and they occur in W_1 and W_2 in the same order, we say W_1 and W_2 are cognates. Figure 4.6 shows the AMS search process.

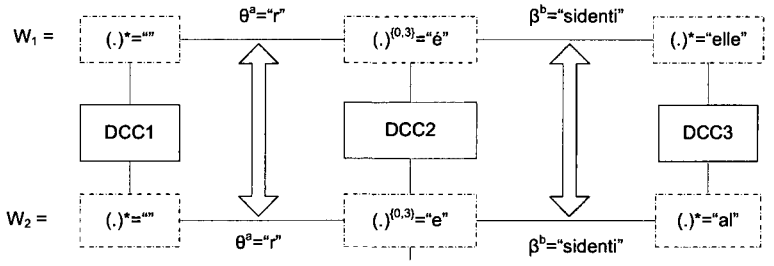


Figure 4.6. AMS search process for the candidate cognate pair W_1 = "résidentielle", W_2 = "residential".

novembre<>november<>2 2 2	industries<>industries<>2 3 2
cansim<>cansim<>1 1 1	durables<>durable<>1 1 1
centres<>centres<>1 1 1	publications<>publications<>1 1 1
définitions<>definitions<>1 1 1	pharmaceutiques<>pharmaceuticals<>1 2 2
observée<>observed<>2 2 2	résidentiels<>non-residential<>1 1 2
visiteurs<>visitors<>1 1 1	industrie<>industry<>3 3 6
secteurs<>sectors<>4 5 6	totale<>total<>1 1 1
situation<>situation<>2 2 2	manufacturière<>manufacturers<>1 2 2
canada<>canada<>1 2 2	marchandises<>merchandise<>1 1 1
résidentielle<>non-residential<>1 1 2	résidentielle<>residential<>1 1 1
consécutif<>consecutive<>1 1 2	module<>module<>1 1 1
site<>site<>1 2 1	activités<>activities<>1 1 1
produits<>products<>1 3 3	persistante<>persistent<>1 1 1
source<>sources<>1 1 1	qualité<>quality<>1 1 1
l'exploration<>exploration<>1 1 1	machines<>machinery<>1 2 2
base<>base<>1 1 1	tourisme<>tourism<>1 1 1
groupes<>groups<>1 1 1	excluant<>excluding<>1 1 1
mines<>mines<>2 3 2	labrador<>labrador<>1 1 1

Figure 4.7. Output cognate list of the AMS algorithm

We wrote Perl programs to implement the AMS search algorithm, taking full advantage of Perl's feature of efficient regular expression matching. By applying the AMS search model to the rough candidate list that was generated by the K-vec algorithm, SDTES was able to produce a cognate matching list (see Figure 4.7) for each of the HTML document pairs.

AMS has the straightforwardness of the naïve matching algorithm of Simard *et al.* (1992). Since in AMS, string matching is conducted at the level of substrings, a substring is treated as if it were a single character unit in the search process. For BMIA, the goal is to find only the acceptable matching sequence, not necessarily the longest common sequence. Once a two-substring match is found, the search stops. There is no need to do calculations of insertion or deletion to get the minimum edit distance. This can reduce the complexity of computational operations. When compared with the string matching algorithm of Simard *et al.* (1992), AMS adds accuracy (two substrings should match rather than one, avoiding matching problems caused by common prefixes), and flexibility (not necessarily the first four letters should match). We executed test runs to compare AMS and the Simard method. The results show that AMS helped in excluding false cognates such as conséquemment/consistent, consultées/constant, consulter/construction, construits/considered, constituées/consistent, constaté/consumer, consommateurs/consult, concurrence/concentrated, comptes/compared, compris/competing, comprenant/ computers, commerce/community. The results also indicate that AMS has the capacity of identifying cognate word pairs that could have been missed in the Sigmard method, such as numéro/number, d'avril/april, échanges/exchanged, remplacé/replaced, pourcentage/percentage, mouvements/ movements, études/studies, établir/establish, marchandises/merchandise, lignes/lines, japon/japan, inchangées/unchanged, enregistrés/registered, dépenses/expenses, d'assurances/insurance, fonds/funds.

At the same time, AMS inherits the strength of no-crossing-links constraint in the Longest Common Subsequence Ratio (LCSR) algorithm by Melamed (1999). However, in specifying that no more than two matching substrings are allowed, AMS overcomes

the inherent weakness of LCSR in positing non-intuitive links because of lack of context sensitivity as noted in recent research by Kondrak and Dorr (2004). This can help reduce the number of false positives for SDTES such as **voitures/sources, ventes/metres, parution/starting, mensuels/results,** and **courtiers/computers.**

In addition, by focusing on only those pairs that can help with the text alignment, the algorithm is able to filter out many candidate pairs that are either not genuine translation correspondences or true cognates that could not help the text alignment prior to the actual substring matching process. Therefore, the AMS search model is easy to use and efficient in achieving our purpose.

4.1.4 Detection of Potential Misalignment

As stated earlier, translations in the published *Daily* releases are mostly clean and consistent translations in agreement with the Canadian guidelines of publications for government websites. Translation errors such as deletion and insertion are rare. But it is possible that the correct translation pairs are misaligned. In the current system, an algorithm is developed for BMIA to detect the regions of possible alignment errors. Figure 4.8 is the alignment detection operation diagram. It shows the major steps involved in arriving at one of the two outcomes: *pass* and *problem*. The detecting process starts from two prior filtering mechanisms. One of them is the length ratio criterion. If a text segment in one language is more than 3 times longer than the corresponding text segment in the other language, the pair is marked as a problem pair. We tested some SDC aligned translation pairs and found that, in more than 98% cases, if the text of one language was 3 times long as the text of the other language, they were either rather

exceptional translation pairs, or spurious translations. The second criterion is the matching type. Because the extracted translations are independent translation pairs that will be used for translation memory systems and cross-language information retrieval systems, matching types like 1:0 and 0:1 have to be discarded. When these two criteria have been checked, BMIA compares the structural and lexical clues of the HTML text segments for further detection. These clues include selected cognates, punctuations, numbers and HTML tags.

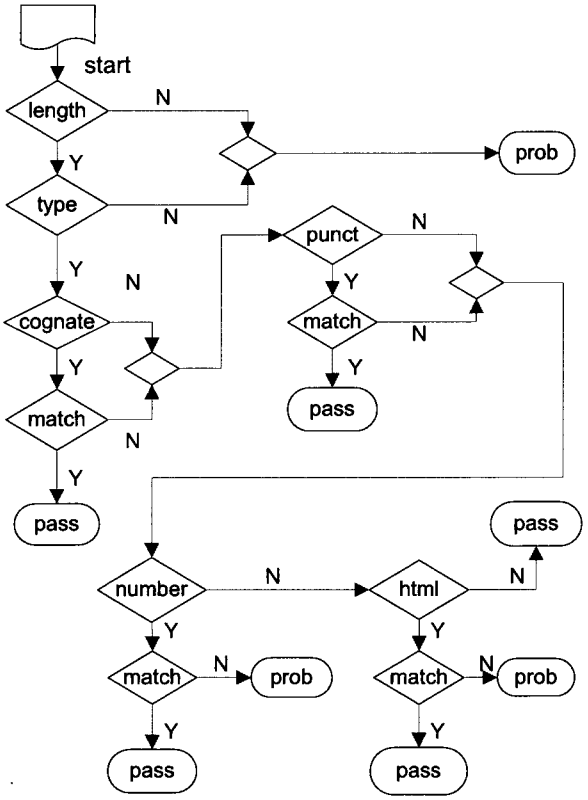


Figure 4.8. Framework for misalignment detection

For the cognates, BMIA uses the list of words that are automatically extracted by the algorithms we described in 4.1.3. When the detecting agent accepts an aligned pair as

input in the misalignment detection operation, it scans through the input text segments to see if there are cognates present. If there are cognates, then the system compares them to see if they match in the texts of two languages. If a match is found, the two text segments are a good translation pair. Otherwise, the candidate pair of text units is passed to the next step of detection. For punctuation, BMIA mainly maps the correspondences using comma, plus sign and minus sign, parenthesis, colon and semi-colon, etc. If the system identifies the equivalence in the use of punctuations (frequency of occurrence $n>0$), it marks the pair as *pass*. It is assumed in SDTES that, in statistical news release texts, numbers are literally translated. If we have numbers in the English text, normally we should have numbers in the corresponding French text segment. If most of the numbers in the candidate pair do not match, the decision is that there might be a problem in alignment or in translation. When the system does not find any numbers in the aligned text segments, it continues the detection by looking for markups of HTML coding. The distinguishing HTML markups BMIA uses include `<h2>` `<h3>` `<i>` `<b>` `<a ...>` `<table>` and others. We did some HTML style unification formatting so that some parts of the HTML codes are highlighted, while some are ignored. For example, the code `<a ...>` becomes `<a alink>` after the unification formatting. Once the system gathers the key HTML structural features in the aligned texts, a comparison is done to see if the HTML tags are the same. If they are, the segments are a good aligned pair of translations; if they are not, the system marks it as a problematic alignment region. Finally, we have the zero-match tolerance principle: if there are no structural and textual clues present, and if the two prior filtering criteria (length and matching type) are checked, we mark the segment as *pass*. Here is a sample of the misalignment detection results list (Figure 4.9)

where 'pass' means the alignment is identified as correct, and 'problem' means the pair is a misalignment.

```
18 (060628c, 1 - 1) --- pass numbers: full match (e= 20 20 f= 20 20)
19 (060628c, 1 - 1) --- pass cognates: match (e=nationally f=nationale)
20 (060628c, 1 - 1) --- problem numbers: not_match (e= 0 2004 3 5 8 f= 5 8)
21 (060628c, 1 - 1) --- pass numbers: full match (e= 2005 f= 2005)
22 (060628c, 1 - 1) --- pass numbers: full match (e= 20 20 f= 20 20)
23 (060628c, 1 - 1) --- pass numbers: no_numbers,html_tag: no_html_tags, no_other_clues: match (e= f= )
24 (060628c, 1 - 1) --- pass cognates: match (e=industries f=l'industrie)
25 (060628c, 1 - 1) --- pass numbers: full match (e= 20 20 f= 20 20)
26 (060628c, 1 - 1) --- pass cognates: match (e=all-important f=importante)
27 (060628c, 1 - 1) --- pass numbers: full match (e= 20 20 f= 20 20)
28 (060628c, 1 - 1) --- pass cognates: match (e=alberta f=l'alberta)
```

Figure 4.9. Sample result lines in misalignment detection

4.1.5 Filtering and Formatting for SDC

We added a filter on the list of stored translations to eliminate aligned pairs such as:

1. Pairs that contain pure meta information coding or codes that are derived from the HTML coding unification process.

```
*** Link: 1 - 1 ***
<hr   size=1>   </zcorpus   text=10>   <zcorpus   table=10>
<hpara=e35:f35>
<hr   size=1>   </zcorpus   text=10>   <zcorpus   table=10>
<hpara=e35:f35>
```

(040107a)

2. Alignment segments that include only the numerical information:

```
*** Link: 1 - 1 ***
2001/02 <hpara=e35:f35>
2001-2002 <hpara=e35:f35>
```

(040107a)

3. Duplicate sentences or similar constructions that have been seen more than once in the collection of texts. Sometimes this involves unifying or discarding some information such as numbers and tags in the text.

```
*** Link: 1 - 1 ***
Retail trade</TITLE>
Commerce de détail</TITLE>
(060821a)
*** Link: 1 - 1 ***
<p><B>Definitions, data sources and methods: survey numbers,
including related surveys, <a alink> 2406 </A> and <a
alink>2408</A> .</B></P>
<p><B>Définitions, source de données et méthodes : numéros
d'enquête, y compris ceux des enquêtes connexes, <a
alink>2406</A> et <a alink>2408 </A>. </B></P>
(060821a)
```

The aligned text segments were cleaned and organized in an XML format for easy exportability into the translation memory system, information retrieval database systems, and other application systems. Meta information items about each of the aligned pairs were recorded such as the string length information (before the HTML codes are stripped), the source of the matched strings, the matching patterns (1:1, 1:2, 2:1, 2:2), and misalignment detection result (*pas* for *pass* and *pro* for *problem*). The current output format is intended to serve as an intermediary form from which different user-friendly formats for different applications can be derived. Figure 4.10 includes some examples of final aligned segments with the English part beginning with 'Wholesale trade activity'.

In the original SDTES format, only the tag </bead> takes line breaks, and for the sake of presentation clarity here, line breaks are added to different levels of XML elements.

```
<bead>
<en>Wholesale trade activity declined 1.4% in July, dragged down by reduced sales of computers and other electronic equipment, lumber and millwork, personal and household products and oil products.</en>
<fr>Le commerce de gros a reculé de 1,4 % en juillet, freiné par la contraction de la demande d'ordinateurs et d'autre matériel électronique, de bois d'oeuvre et de menuiseries, de produits personnels et ménagers ainsi que de produits pétroliers.</fr>
<pa>1:1</pa>
<id>050930a:36</id>
<re>Pas</re>
<le>194=243</le>
</bead>
<bead>
<en>Wholesale trade activity increased 0.9% in June, helped by the demand for computers and other electronic equipment. </en>
<fr>Le commerce de gros a augmenté de 0,9 % en juin, profitant de la demande d'ordinateurs et d'autres équipements électroniques.</fr>
<pa>1:1</pa>
<id> 050831a:64 </id>
<re>pas</re>
<le>116=126</le>
</bead>
<bead>
<en>Wholesale trade activity increased 1.0% in October. </en>
<fr>Le commerce de gros a affiché une croissance de 1,0 % en octobre. </fr>
<pa>1:1</pa>
<id>051223a:28</id>
<re>pas</re>
<le>52=66</le>
</bead>
```

Figure 4.10. Modified XML format for aligned segments generated by SDTES

The interim format is good for use with UNIX system tools and Perl short programs for quick finding of translations, as well as for various sorting, analyzing and statistical tasks. It can be easily converted into different feeding formats in different systems such as *Daily* translation recycling templates system (text block based format), translation memory systems (format with texts of different languages assembled in different files), bilingual information retrieval (format required by the search engine), and bilingual text

navigation interface (format required by the field specifications of the SQL database). The conversion from the basic BMIA format to the required input formats of different systems can be done with a few lines of coding or, in some cases, in a text editor environment.

4.1.6 Evaluation and Results

To evaluate the performance of the alignment, BMIA used a reference collection of manually aligned text segments. The aligned pairs are supposed to be correct in this reference collection. A comparison is made between the reference collection and those pairs aligned by SDTES to see how the machine aligned pairs differ from the manually aligned text segments.

Let M be the set of segments in the manually aligned reference collection, A be the set of machine aligned segments before the misalignment detection device is applied. Precision (P) and recall (R) can be defined as follows:

$$P = \frac{|A \cap M|}{|A|} \qquad R = \frac{|A \cap M|}{|M|}$$

Files in the evaluation are randomly chosen: one aligned file for every two months. The same file is manually aligned for the building of the reference collection. Then the machine aligned pairs were checked to arrive at the number of correctly aligned segments. For the 18 aligned files used for evaluation, the average alignment precision for aligned translation pairs before filtering is 0.98; and the recall is also 0.97 (see Table

4.2). This compares favorably with the precision and recall values of most of the other alignment classifiers.

File	Machine proposed matching pairs before filtering (A)	Manually identified matching pairs (M)	Intersection of A and M	Precision (P)	Recall (R)
040213b	214	214	212	0.99	0.99
040415a	109	109	103	0.94	0.94
040611b	92	94	83	0.90	0.88
040715e	17	16	15	0.88	0.94
041015c	11	11	11	1.00	1.00
041223a	100	100	100	1.00	1.00
050107a	119	119	119	1.00	1.00
050414a	105	105	105	1.00	1.00
050629b	72	72	72	1.00	1.00
050721a	121	120	119	0.98	0.99
050922a	116	116	116	1.00	1.00
051109a	185	190	185	1.00	0.97
060111a	86	87	84	0.98	0.97
060308b	48	48	48	1.00	1.00
060608d	31	34	29	0.94	0.85
060822b	57	58	48	0.84	0.83
061005b	117	117	117	1.00	1.00
061213e	17	16	15	0.88	0.94
Overall	1617	1626	1581	0.98	0.97

Table 4.2. Evaluation parameters for aligned text segments before filtering

The main function of the misalignment detection algorithm is to filter out misaligned translation pairs. The task is to traverse every pair of aligned segments to see if they are indeed a correctly aligned pair. It can happen that the members of the aligned pair are perfect translation of each other, but the detection algorithm wrongly labels the pair as a misaligned one, or a misaligned pair can be detected as a pair of perfect translations and the algorithm fails to capture the misalignment. In evaluating the adequacy of this

filtering component, we used the same files that were randomly chosen for the evaluation of the aligning algorithm. The system gets the proposed automatic alignments (A') by excluding those translation pairs that the filtering device identifies as misaligned segments. M is the reference set, i.e. the number of aligned translation units that the human evaluator thinks the system should have reported. The recall (R) represents the proportion of algorithm-proposed translation units (A') that are right with respect to the reference (M), and the precision (P) is the proportion of correctly proposed alignment segments with respect to the total of those proposed (A').

$$P = \frac{|A' \cap M|}{|A'|} \qquad R = \frac{|A' \cap M|}{|M|}$$

From the results shown in Table 4.3, we can see that the system is accurate in identifying correctly aligned translation pairs (P=.99 and R=.96). A comparison of the precision and recall values for the data sets before and after the filtering shows the effect of the filtering performed by the misalignment detection algorithm. Precision improved from .98 to .99 with a slight loss of recall from .97 to .96. In the context of automatic extraction of translations, where misaligned or doubtful pairs of translations should be omitted, this trade-off for the purpose of maximizing precision can be a preferred option.

It would have been good if we could set the original Gale and Church algorithm (without the initial alignment using the main HTML tags) as the baseline, and compare it with the alignment method we adopted in SDTES to evaluate the effects of the two-round alignment. However, this attempt was deterred by problems in identifying the

paragraph boundaries without the help of HTML tags in the source files. Carriage returns which normally mark the end of a text segment or a paragraph can be anywhere in an HTML source file, and this makes the process of paragraph identification difficult. If we run the algorithm without paragraph detection, the results would be very poor.

File	Machine proposed matching pairs after filtering (A')	Manually identified matching pairs (M)	Intersection of A' and M	Precision (P)	Recall (R)
040213b	212	214	211	1.00	0.99
040415a	105	109	102	0.97	0.94
040611b	78	94	77	0.99	0.82
040715e	15	16	15	1.00	0.94
041015c	11	11	11	1.00	1.00
041223a	100	100	100	1.00	1.00
050107a	118	119	118	1.00	0.99
050414a	105	105	105	1.00	1.00
050629b	72	72	72	1.00	1.00
050721a	121	120	119	0.98	0.99
050922a	116	116	116	1.00	1.00
051109a	183	190	180	0.98	0.95
060111a	84	87	84	1.00	0.97
060308b	47	48	47	1.00	0.98
060608d	27	34	27	1.00	0.79
060822b	49	58	46	0.94	0.79
061005b	116	117	116	1.00	0.99
061213e	15	16	15	1.00	0.94
Overall	1574	1626	1561	0.99	0.96

Table 4.3. Evaluation parameters for aligned text segments after filtering.

SDTES was built on the basis of *the Daily* releases published between 2004 and 2006. 3,874 documents for each language (English and French) were processed and 70,555 translation segments were produced after misalignment detection and filtering. Then the SDTES model was tested with web-based bilingual materials of 5 other government

websites, and more than 200,000 pairs of translations were generated. The same algorithms of SDTES were applied to an expanded data collection of *Daily* releases to build the StatCan Daily Corpus (SDC). All in all, BMIA assembled 32,276 *Daily* release files (16,138 for each language). Each release file represents a *major* release, an *other* release, or a *feature* release in *the Daily*. The collected web documents were published over a period of 3,075 days, from June 13th 1995 to December 24th of 2007. The average release texts per day are around 5 (the actual number is 5.25). Numerals in table cells are deleted and thus not included in the final bilingual corpus. SDC contains more than 14 million words (6,513,895 running words for English and 8,310,827 running words for French). The word translation ratio of French to English is 1.3:1, which means that for every ten English words, we use 13 French words in the StatCan *Daily* texts. After filtering and formatting, BMIA generated 488,646 aligned text segments. Each text segment contains around 13 (rounded from 13.33) words for English and 17 (rounded from 17.01) words for French. They are ready to be used in many applications and systems such as translation memory systems, information retrieval templates, and machine translation systems.

All the text segments in SDC including the HTML tags and meta information lines were aligned with one of the 6 alignment patterns: 0:1, 1:0, 1:1, 1:2, 2:1, 2:2. The most common pattern is 1:1 ($p>.93$) indicating that more than 93% of the text segments are sentence to sentence translations. There are examples of paraphrasing (patterns 2:1, 1:2 and 2:2), but deletions and insertions (patterns 1:0 and 0:1) in translation are rare (see Table 4.4).

Alignment pattern	1:1	1:2	2:1	0:1	2:2	1:0
Percentage	0.93	0.042	0.02	0.003	0.002	0.003

Table 4.4. Alignment types for the StatCan Daily Corpus.

As we can see from the result of the evaluation, the number of misaligned pairs that escaped detection in SDTES was minimal. Although there were examples of false misalignment, 99% of the identified correct alignment pairs are truly reliable translation pairs. While the SDTES methodology has achieved good output for officially published government bilingual text data, there are also limitations. When we examine the aligned pairs before the filtering device is applied, we still find some examples, although the number is very small, of chains of misaligned sentences that are set off by the swapping of positions in the translation texts. For example, if in a translation document pair, the 35^{th} text segment in English is the translation of the 40^{th} text segment in French, although the translation is right, it can amass misaligned segment pairs because the translation spans beyond more than 3 text segments, and the Gale and Church algorithm cannot handle it.

There are some short sentences on only one side of the alignment pairs, and they do not carry much discriminative feature information. These short sentences can be a source of misalignment (see Figure 4.11). Judging by the length-based alignment criterion, appending the short sentence to the previous sentence or combining the short sentence with the following sentence would not make much difference. This increases difficulty in alignment. To solve this problem, we need to add less costly lexical clues to the existing

length-based model to ensure more accurate alignment without putting too heavy a burden on the computer time and speed.

There have been many occasions when three parties, the Conservatives, the Liberals and the Bloc, stood together on an issue and supported a position that was wrong.	À de nombreuses reprises, les conservateurs, les libéraux et les bloquistes ont fait front commun sur une question et défendu une mauvaise position. Nous nous trouvons actuellement dans une situation semblable.
In this case, we are dealing with a similar situation, where in haste we are proceeding with a bill that is flawed and we are not thinking about the long term ramifications.	Nous sommes en train d'étudier à toute vapeur un projet de loi comportant des failles et nous ne pensons pas aux répercussions à long terme.

Figure 4.11. Alignment error analysis: short sentences causing alignment problems.

When examining the SDTES-generated cognate lists word by word, we found some misclassified pairs (see Table 4.5). The false cognates are mostly false positives caused by accidental similarity in the orthographic form. They are likely to be on the lists of other cognate classifiers that depend on measures of orthographic similarity.

French	English	French	English
consiste	considers	sport	report
cours	court	estivaux	festival
abordable	affordable	fiscale	scale
aller	smaller	finlande	mainland
exercer	exerted	grands	brands
variable	available	mains	gains
lever	every		

Table 4.5. Examples of false cognates in SDTES cognate matching.

Although these false cognates can potentially hurt the misalignment detection algorithm in SDTES, their impact on the actual identification of misaligned pairs is not so significant. We extracted all the lines which match "--- pass cognates:" in the misalignment detection result file, and discovered that not many false cognates are

actually used for the cognate matching criterion in the misalignment detection process. Generally speaking, if two cognate words are not true friends, and if the text segments where they occur are not true translations of each other, chances are good that the misalignment would have been detected by the length criterion and the matching type criterion prior to the application of the cognates matching process (see Figure 4.8). In some cases, there are other cognate pairs that are checked before the false cognate pair. Once a cognate match is found in one of these candidate cognate pairs, the aligned text segments are marked as *pass*. We also examined a few instances where the false cognates are actually used as a decisive factor in the misalignment detection process and we found all of the aligned text segments to be true translations. It is often the case that one word of the false cognate pair is the translation of another word in the text segment of the other language.

4.1.7 TransConcord

Any efficient bilingual corpus analysis is dependent on both the creation of the bilingual corpus and software tools to help process and analyze the corpus data. Researchers such as Arthern (1978), Melby (1981) and Isabelle (1992) have observed that a collection of past translations together with computer software to retrieve them can prove to be useful for a variety of purposes. Parallel concordance is one form of technology that can access aligned bilingual corpora and can serve to uncover the complex correspondence relationship in translation texts. Usually when a corpus has been compiled, the researcher either develops or selects a concordancing tool to use with the corpus so that when retrieving information from the corpus, keywords can be displayed in contexts and words

can be studied in collocations and clusters. Since the early nineties, there have been noticeable developments in parallel concordance such as the Church-Gale concordance system (Church and Gale 1991), TransSearch (Simard *et al.* 1993; Macklovitch *et al.* 2000; Macklovitch *et al.* 2008), ParaConc (Barlow 2002), Multiconcord (Romary *et al.* 1995), and TotalRecall (Wu *et al.* 2003). These concordancing systems are aimed at different audiences, such as language learners and teachers, translators, lexicographers, and translation study researchers. Many of these systems offer varied search facilities: word search, phrase search, regular expression search and tag search.

Although the extracted translation units of SDC can be imported to translation memory systems to search for a translation pair, we also designed a translation concordance search system for ready access of translation pairs in SDC and other bilingual corpora. The StatCan translation concordance search system (TransConcord) is basically a search engine that achieves real-time concordancing of existing translation pairs in a bilingual corpus. This concordance search system is based on the key word in context (KWIC) search technique, and current search engine technologies for indexing, lemmatizing, and query processing. TransConcord can be extremely useful for lexicographers, human translators, learners and teachers in language teaching and training, and researchers who are interested in translation studies, machine translation, word sense disambiguation, and other natural language engineering tasks. TransConcord is in a sense better than a conventional dictionary in that when we need to examine the actual examples in which a query word is used, particularly when the word is polysemous, the concordance search system can help cluster similar uses of translations together.

Some of the unique features in the StatCan translation concordance search system include: 1. unsupervised identification of potential word correspondences and related string sequences; 2. applying the monolingual concordance KWIC technique to help locate query contexts and their translations quickly in the results returned; 3. because of the optimized indexing algorithm used for the search engine, the search system can handle large volumes of data and has the capacity of searching through tens of thousands of translation pairs in the bilingual corpus in a quick and efficient manner. Currently the StatCan translation concordance search system contains 5 collections of officially published bilingual materials from several government departments and agencies in Canada. The collections will grow and expand in the future. In addition, it also includes a large collection of officially released Canada Hansard materials from the 35th Parliament in 1994 to the 40th Parliament at the end of 2008. All the text segments in the Hansard collection are aligned and filtered using the algorithms that are extended from those described in our previous work (Zhu *et al.* 2007). When we search for translations of frequently used words (stop words not included), returned hits of aligned translation pairs can easily exceed the 10,000 mark. For example, when searching the word 'government', the system indicates that it found 143,812 translation pairs that contain the query word in less than one second. Without doubt, the large numbers of rich contexts provided in these pairs can be a good source of information in differentiating the usage of alternate translations and synonymous expressions for translation studies and many other purposes.

have resulted in a modest	**increase** in investments and job cuts	900
Ontario posted a more modest	**increase**.	901
with modest	**increases** in deposits	902
fuel oil showed the biggest	**increase**, a gain of 244 400 cubic	903
second largest	**increase** in sales was observed in the	904
the largest	**increase** since	905
the largest	**increase** was recorded in the Atlantic	906
the largest	**increases** were witnessed by the	907
the largest	**increase**.	908
recorded the largest	**increase** in	909
posted the largest	**increase**, adding 400 megawatts of	910
experiencing the largest	**increases** in tonnage from September	911
the largest	**increase** (in dollars) occurred in	912
largest	**increase** (+26.6%) occurred in the	913
largest	**increase** (in dollars) occurred in	914
largest	**increase** by volume occurred in diesel	915
largest	**increases** were to Vermont	916
the highest	**increase** in the Atlantic	917
the highest	**increase** since the fourth quarter of	918
experienced the highest	**increase** among all the	919
registered their highest	**increase** of the year	920
fastest	**increase** (+2.7%) occurred in	921

Figure 4.12. Monolingual KWIC indexing in TransConcord sorted by the left contexts in the selected text window c_i, c_{i-1}, c_{i-2}, ... c_{i-n}

To facilitate the fine-tuning process in finding the right query expression to search in the bilingual corpus, TransConcord generates a monolingual KWIC view on the initial results returned. In SDC, a corpus of more than 400,000 words, some query words are likely to occur in higher numbers than can be dealt with within a web page. In many cases, we may have hundreds of returned translation pairs. It is often hard to say that one translation pair is more important or more relevant than the other. To help find the translations quickly, TransConcord first returns a monolingual concordance in KWIC

format, displaying only a small amount of co-text on both sides of the query word. There is a sorting function that can sort and reorganize the co-occurring texts (some can be zero) in the selected window by the left hand side c_i, c_{i-1}, c_{i-2}, ... c_{i-n} (Figure 4.12) or by the right hand side c_i, c_{i+1}, c_{i+2}, ... c_{i+n} (Figure 4.13).

visit friends and relatives **increased**	a more modest	574
experienced the highest **increase**	among all the	575
production posted a slight **increase**	And crude oil production	576
spurred significant volume **increases**	And price declines for both	577
manufacturing sector **increased**	at a faster pace than it did	578
economic output **increased**	at a faster pace than the	579
their investments in Canada **increased**	at a faster rate than the	580
in both provinces **increased**	at a slower pace than in	581
these donations **increased**	at an annual average rate of	582
it **increased**	at an annual rate of 2.8% in	583
the core non-profit sector **increased**	at an average annual rate of	584
non-profit organizations **increased**	at an average annual rate of	585
sectors of both nations **increased**	at an identical annual	586
consumer prices **increased**	at rates faster than the	587
with that of investment **increased**	at slower rate in	588
Canada and the United States **increased**	at the same average pace	589
underwent a large **increase**	between 2001 and 2006, not	590
performance is expected to **increase**	by $40 million	591
in the third quarter after **increasing**	by 0.2% in the second	592
total manufacturing sales **increased**	by 0.9% in	593

Figure 4.13. Monolingual KWIC indexing in TransConcord sorted by the right contexts in the selected text window c_i, c_{i+1}, c_{i+2}, ... c_{i+n}.

The selection of the co-text in the window can be interrupted by boundary marks for sentences, clauses, or words. When occurrences of the query word are sorted alphabetically in either direction, they are much more easily uncovered and located.

Several commonly encountered cases can serve as the cognitive basis for setting this display format as the default in TransConcord:

- We try to find the translation of a phrase associated with a word in the source language, but we don't have the information about the context that is needed to complete the search.
- We cannot remember an expression which the query word is part of, and cannot make a decision unless we are provided with alternative candidates.
- We know what the expression is, but we do not know if it is available in the corpus at all, or if the expression has varied forms, for syntactic, morphological or grammatical reasons.

We can thus quickly discard translations that are not so relevant and focus on only those translation clusters that are of interest to us. Narrowing down the query expression range through information about the contexts in the vicinity of the query word makes it much easier to make the decision as to what expression or phrase to select in translation search.

One noted difficulty in past attempts of bilingual concordance search has been with the automatic identification of the "bilingual context" of the translation word in the target language (Simard *et al.* 1993). Particularly with web-based applications, the algorithm has to be fast and efficient so that the user will be able to see quick returns of appropriate search results. Some approaches like IBM Model 1 with the EM algorithm for finding word correspondences can yield good results, but they are time-consuming in the iteration process. Other approaches use a dictionary to match the words in the translation

pair with the query word; however in many cases accuracy will suffer. Many translation words in the dictionary cannot be found in the individual translation pair, and many potential translations are not contained in the look-up lexicon. This is why in most conventional bilingual concordance systems, only the search term is highlighted by the concordancer, but no relative region is marked to identify possible translations. The user needs to read the translation segments to identify where the translated words are and how they are translated.

If a person is called by a **charity** and asks to be placed on the do not call list held by that charity, the charity is forced to comply and is not allowed to call that individual for three years, which is the current time limit.	Si, recevant un appel d'une organisation caritative, un particulier demande à celle-ci de l'inscrire sur sa liste d'exclusion, l'organisation doit se plier à cette demande et elle n'a pas le droit de rappeler cette personne avant trois ans, le délai actuel.	506
If a person is called by a **charity** and asks to be placed on the do not call list held by that charity, the charity is forced to comply and is not allowed to call that individual for three years.	Lorsqu'un organisme de bienfaisance appelle une personne qui demande à être inscrite sur une liste d'exclusion, cet organisme doit y consentir et ne pas rappeler la personne avant trois ans.	507
Thus, the bill changes the rules concerning health, **charities** and business arrangements.	Ainsi, le projet de loi modifie les règles touchant la santé, les organismes de bienfaisance et les arrangements commerciaux.	508
I asked earlier about my concern regarding **charities** and charitable organizations.	J'ai soulevé ma préoccupation relativement aux organismes de bienfaisance.	509
There is one particular very comprehensive legal assessment of this legislation by about three dozen legal experts and academic advisers whose whole careers are invested in academic interests related to human rights, religious rights, and **charity** and constitutional law.	Il existe une évaluation juridique très complète de ce projet de loi; elle a été réalisée par quelque trois douzaines d'experts juristes et de conseillers universitaires dont toute la carrière s'est investie dans des centres d'intérêt universitaires liés aux droits de la personne, aux libertés religieuses, ainsi qu'au droit régissant les organismes caritatifs et au droit constitutionnel.	510

Figure 4.14. Conventional view of translation concordance search which highlights only the query word in the source language

In Figure 4.14, the word 'charity' is searched in the Hansard bitext collection of the StatCan translation concordance search system. Only a few of the 397 matched pairs are shown here, in a way that many other conventional concordancers would have done. The

query word (*charity* or *charities*) in the source language is highlighted for each of the translation pairs, but not the potential translation words in the target language segments (*caritative, bienfaisance, caritatifs*). Without providing an approximate region to find the translations, it would be harder for the reader to go through every part of the translation segments and fish for the potential translation correspondences, particularly when the returned hits are many and the translated segments are long or the query word has polysemous meanings.

TransConcord allows the concordance user to search for a word in one language, retrieve all the instances of the query with the immediate contexts at both sides, and at the same time return the potential translation correspondence in the matching section of the translation text in real time. This is one of the features that surfaces most often from the suggestions of users and that many other conventional concordance systems attempted to achieve (Simard *et al.* 1993). This feature is important in that word correspondences so identified can be of great use for machine translation, machine learning, and translation data mining. It can also bring into play the potential of concordances in many practical tasks in semantic analyses such as distinguishing between different meanings of a word, differentiating literal and metaphorical meanings, and uncovering hidden meanings of words or phrases (Sinclair 2003). In TransConcord, the computation for the likelihood of translation correspondence depends on some convenient, time-saving algorithms we designed and can be processed on the fly in a very short time span. For elaboration of some of the word correspondence identification algorithms, please see Section 4.2.4.1. The translation probability analysis is integrated with some filtering mechanisms for finding word correspondences. This approach does not need pre-search processing of the

data sets or training data for machine learning. Neither does it depend on prior knowledge of word translation co-occurrence probabilities or dictionary matching lists. The system repeats the process of finding the surest word correspondences and excludes the text segments in which these best-fit translations occur as it goes. For the residual pool which cannot be easily handled by the surest-match approach, the software agent employs the position offset information of the query word to guess the approximate position of the potential translation word. This strategy is effective in gradually reducing the search range and eliminating noises when identifying word correspondences in the remaining regions of the search space. The algorithm has the advantage of performing fast, and in the majority of cases, is capable of finding the most relevant translations in the contexts.

However, identifying the correct word translation is not an easy task, and there are still some translation pairs where TransConcord finds it hard to ascertain which word is the most likely corresponding translation word. For example, the algorithm will encounter problems when the search returns very few translation pairs, or when translation equivalents of the query words are evenly distributed among a few words or phrases in the target text segments.

Here are some procedures for identifying translations of the query word, and for displaying them in the bilingual KWIC format:

- Step 1: the system accepts a query word in one language.
- Step 2: the search agent returns the first batch of paired translations in which the query word occurs.
- Step 3: the system builds a word translation probability vector.

- Step 4: the system establishes translation correspondence links using both the lexical clues and the statistical results.

foreign direct investors in Canada	**increased** $0.8 billion to a high of $9.8	502
de dollars en raison d'une	**progression** de 0,8 milliard de dollars.	
liabilities to non-residents	**increased** $12.2 billion in the third quarter	503
net envers les non-résidents a	**augmenté** de 12,2 milliards de dollars au	
direct investment holdings	**increased** $19.1 billion during the third	504
directs étrangers au Canada ont	**augmenté** de 19,1 milliards de dollars au	
of goods resumed an upward trend,	**increasing** $2.0 billion to a record $104.9	505
de biens se sont redressées, ayant	**progressé** de 2,0 milliards de dollars pour	
Columbia also saw a sharp	**increase** (+11.2%) for a total of $3.8	508
a également connu une forte	**progression** (+11,2 %) pour s'établir à 3,8	
fastest	**increase** (+2.7%) occurred in regular	509
sans plomb a enregistré la	**hausse** la plus marquée (+2,7 %).	
the largest	**increase** (in dollars) occurred in British	513
l'échelon provincial, l'	**augmentation** la plus prononcée (en dollars) a été	
largest	**increase** (in dollars) occurred in Alberta	514
plus importante	**croissance** en dollars s'est produite en Alberta	
the rise, as is its rate of natural	**increase** (the excess of births over deaths).	515
au Québec est toujours en	**hausse**, tout comme le taux d'accroissement	
oil production	**increases**	516
	Augmentation de la production de pétrole brut	
of mines, utilities and factories)	**increased** 0.1% in October.	517
publics et des fabricants) s'est	**accrue** de 0,1 % en octobre.	
force, Ontario's unemployment rate	**increased** 0.2 of a percentage point to 6.2% in	518
de chômage dans la province s'est	**accru** de 0,2 point en novembre pour se	
was up 0.1% in September, after	**increasing** 0.2% in August and 0.1% in July.	519
de 0,1 % en septembre, après avoir	**progressé** de 0,2 % en août et de 0,1 % en	
population	**increased** 0.37% to an estimated 12,850,600,	525
population de l'Ontario s'est	**accrue** de 0,37 % pour atteindre un nombre	
of goods and services	**increased** 0.6% in the third quarter, after	529
biens et de services ont connu une	**croissance** de 0,6 % au troisième trimestre,	

Figure 4.15. Bilingual KWIC display of search results in TransConcord (query word: *increase*)

- Step 5: the system prunes texts outside the proximity window, and centers the key word and its translation word.
- Step 6: the system sorts and ranks the results, and displays the results in the KWIC format.

Figure 4.15 shows the results of such a process. The source language line is displayed in KWIC format, with the query word in the centre. After each source language text line, the target language line is shown in KWIC format with the corresponding translation word automatically identified and centered as well. By reading the query word and the corresponding translation word, it is immediately possible to see if the translation is the one the user wants to find. It is also a very convenient form to view the variations of translations for particular patterns that are automatically grouped.

The third format in which search results are shown in TransConcord is the full-text search display format. This is where full translation text segments in both languages are shown. This option is particularly helpful when the user has located the translation pair through the bilingual KWIC display, and wants to explore the translation correspondences further. In many cases wider co-texts are necessary for deciphering the meaning of the keywords and their translations. It is only by looking further than the immediate co-text that we can understand the translation correspondence better. In full-text search display, the query word and the corresponding translation in the other language are highlighted. This is a convenient means by which the user can closely examine and understand how specific elements in one language are translated into another. When investigating translation patterns and collocations extracted from large

amounts of texts in the bilingual text collections, the user can detect and determine which translations are appropriate in which particular contexts, or if the selection of different words in translation makes no difference in meaning at all. Figure 4.16 is a sample of full-text search results that is activated by searching the English query word 'increase' in the StatCan Daily Corpus.

Economic activity was up 0.1% in September, after **increasing** 0.2% in August and 0.1% in July.	L'activité économique a augmenté de 0,1 % en septembre, après avoir **progressé** de 0,2 % en août et de 0,1 % en juillet.	519
October 2007 Previous release Economic activity **increased** 0.2% in October, after growing 0.1% in September.	Octobre 2007 Communiqué précédent L'activité économique a progressé de 0,2 % en octobre, après avoir **augmenté** de 0,1 % en septembre.	521
Ontario's population **increased** 0.37% to an estimated 12,850,600, which represented about 39% of Canada's population.	La population de l'Ontario s'est **accrue** de 0,37 % pour atteindre un nombre estimatif de 12 850 600, ce qui représente environ 39 % de la population du Canada.	525
Natural gas production **increased** 0.4% in 2006 from 2005.	La production de gaz naturel a **augmenté** de 0,4 % en 2006 par rapport à 2005.	526
The CMSPI **increased** 0.5% to 125.8 (2003=100) in November.	En novembre, l'IPSMSM s'est établi à 125,8 (2003=100), en **hausse** de 0,5 %.	527
Activities in the finance and insurance sector **increased** 0.5%.	Les activités du secteur de la finance et des assurances ont **augmenté** de 0,5 %.	528
Exports of goods and services **increased** 0.6% in the third quarter, after growing 0.8% in the second.	Les exportations de biens et de services ont connu une **croissance** de 0,6 % au troisième trimestre, comparativement à 0,8 % au deuxième trimestre.	529
In British Columbia, labour productivity **increased** 0.7% in 2006, slightly below the national average.	La **croissance** de la productivité en Colombie-Britannique s'est chiffrée à 0,7 % en 2006, soit un peu au-dessous de la moyenne nationale.	530
Growth in household net worth slowed significantly during the third quarter, **increasing** 1.2%, about half the pace of the second quarter and the slowest in five quarters.	La croissance de la valeur nette des ménages a ralenti considérablement au cours du troisième trimestre, ayant **progressé** de 1,2 %, ce qui représente environ la moitié du rythme atteint au deuxième trimestre et le rythme le plus lent en cinq trimestres.	536

Figure 4.16. Full-text search in TransConcord.

In addition, the user can enter an English query word and a designated French translation word, and search the pair at the same time. For example, we know that we have different translations for the word 'increase' in English. But if we want to find the word 'increase' translated only as 'hausse' in French, we can enter 'increase' in the English query box, and the word 'hausse' in the French query box. TransConcord can quickly scan all the aligned sentence pairs in the selected bilingual corpus and return only the instances in which the pair of words occurs in the paired segments. This way of bilingual concordance search can be readily utilized for the study of meaning differentiation and word sense disambiguation. Translation segments in search result lists like Figure 4.17 can be compared in a view to verifying if a French word is a dominating choice in the translation of an English word among a diversified selection of alternative words. From such lists, we can also see whether a translation pair is triggered by certain patterns in the context in the source language. The findings can provide valuable insights for many natural language processing applications.

In consideration of large quantities of data that can be returned in the search process, TransConcord places a limit on the number of search results that can be processed in a display page. The number is set to be 500 currently and is much larger than the conventional search engines (10, 20, 25, 30, 50, or 100 for Google, Yahoo, and others). However, the number-limit setting can be adjusted to reflect the preference of the user based on the tradeoff preference between browsing convenience, speed and available computer memory resources.

12.9%, a second consecutive monthly	**increase** after four months of significant	538
%, ce qui représente une deuxième	**hausse** mensuelle consécutive après quatre	
sales made by wholesalers, July's	**increase** also coincided with a turnaround in	539
réalisées par les grossistes, la	**hausse** notée en juillet coïncide aussi avec	
	increased among all major commodity groups.	541
biens et services ont affiché une	**hausse**.	
gas production posted a slight	**increase** and crude oil production slipped.	542
de gaz naturel a connu une légère	**hausse** et la production de pétrole brut a	
that the ozone exposure indicator	**increased** by an average of 0.8% a year between	548
d'exposition à l'ozone a connu une	**hausse** moyenne de 0,8 % par année de 1990 à	
in the province	**increased** by an estimated 30,000, the first	550
a connu une	**hausse** estimative de 30 000 dans la	
from September to October, a slight	**increase** compared with the marked declines of	555
ce qui représente une légère	**hausse** par rapport aux baisses marquées des	
about 40% of the disability rate	**increase** could be explained by population	556
cette période, environ 40 % de la	**hausse** du taux d'incapacité s'expliquerait	
and hogs into the United States and	**increased** dairy and chicken prices.	557
à 13,8 milliards de dollars. Cette	**hausse** est en majeure partie attribuable à	
strong corporate profits, and	**increasing** demand for health and nursing	558
bénéfices des sociétés et la	**hausse** de la demande d'établissements de	
personal income and, in turn,	**increased** demand for new homes and other goods	559
de travailleurs a eu un effet à la	**hausse** sur le revenu personnel et a	
market, productivity gains become	**increasingly** dependent on the continuous	560
sur le marché du travail, les	**hausses** de productivité sont de plus en plus	
male university enrolment would	**increase** dramatically to 2030/2031.	562
de sexe masculin connaîtraient une	**hausse** spectaculaire d'ici 2030-2031.	
province except Alberta, where the	**increase** eased off slightly from the 12-month	564
provinces sauf en Alberta, où la	**hausse** a légèrement ralenti comparativement	
combined impact over the years of	**increased** exports and lower imports expanded	569
combiné au fil des ans de	**hausses** dans les exportations et de chutes	
slowed appreciably, reflecting	**increases** for capital goods.	573
façon appréciable, traduisant une	**hausse** des biens d'équipement.	
price	**increases** for motor vehicles and electrical	576
Des faibles	**hausses** de prix pour les véhicules	
was the first	**increase** for non-durable goods in the past	577
s'agit de la première	**hausse** de biens non durables observée au	

Figure 4.17. Translation correspondence search in TransConcord (words searched: *increase* in English and *hausse* in French)

Strikingly, these algorithms that can be accomplished in real time in a web-based application and that are mostly based on common sense and realistic assumptions about translations can generate results comparable with sophisticated text mapping models. For example, when testing IBM Model 1 to estimate translation word correspondences in target language texts, we discovered that we had to obtain all the word correspondences in the returned translation pairs before we could pick specific translations for the query word. This is obviously a waste of time and the iteration process in EM for doing this makes the process too slow. We also tested with external lexical resources such as a bilingual dictionary that could be used to aid translation identification. We found that there were a few major issues that we had to address. For instance, there is a trade-off between the size of the dictionary and the time needed to retrieve translations in real time. Second, we need to rank the priority of translations in the dictionary so that true translation words can be selected before other options. This is difficult when the priority list has to vary with specific contexts in different translation texts. Third, there are translation equivalents that occur in the translation text but cannot be found in the dictionary, and there are translation equivalents of the query word that are listed in the dictionary but do not match any of the words in the translated text.

Although the translation word association identification approaches we designed in TransConcord cannot guarantee that every single translation pair identified is correct, they are robust, fast, practical and powerful. In the overwhelming majority of cases, the translations identified are reliable. We randomly checked some results to see if most identified word translation pairs are correct. In one instance, we searched the English

word "water" in SDC. The search returned 283 matching pairs. In 3 translation pairs, the system indicated that it could not find a proper word in French as the translation equivalent. For 5 returned pairs, the system wrongly identified words like "ménages" as the correspondent translation of "water". All the other pairs are correct; the correct ratio is 97 percent. Similarly, we checked the translation of words such as "gain" (10,737 pairs), "sales" (23,071 pairs), "school" (1,891 pairs), and found that they all returned a correct ratio of more than 95 percent in translation word identification. Many results returned by TransConcord reveal surprising findings about interesting non-fixed or hidden translation relations that are unseen in conventional dictionaries.

4.2 BMIA for Translation Discrepancy Detection and Translation Correspondence Profiling

Since Isabelle *et al.* (1993) proposed a "translation checker" for translators, there have been key developments in research and in designing such tools to aid human translation. TransCheck (Jutras 2000) and TransType (Foster *et al.* 2002) are two of the translation support systems that pioneered work towards this goal. An important purpose of BMIA is to build the StatCan Bilingual Text Comparison System (TextComp), a computational system to analyze translated texts so that detailed translation correspondences can be profiled and regions of potential translation discrepancies can be detected. Although there are many differences between TransCheck and TextComp in the types of errors to detect, the algorithms for identifying the errors, and the lexical knowledge such as part of speech

information to employ in the process of error detection, both systems share the common goal of being a "translation analyzer" (Isabelle *et al.* 1993) for translation error detection.

It has been observed in Lexical-Functional Grammar that a rich set of structures and correspondences can be posited as constituting the linguistic form-meaning relation (Kaplan 1987; Kaplan 1989; Asudeh and Toivonen 2009). We apply this observation about language to translation studies and assume that if, in a sentence pair, there is a rich set of form correspondences (such as cognates and symbols), there is likely a meaning association (mutual translation relationship) between them. On the basis of this assumption, we propose a Translation Correspondence Profiling (TCPro) component in TextComp for the purpose of initial translation quality assessment. We consider this a further step in exploring Isabelle's "translation checker" concept and in introducing objectivity in judging translation quality that many researchers in translation studies have argued for and have ventured to do (House 1976; Wilss 1982; Baker 1992; Horton 1998). We believe that such an analyzer can have a significant impact on the way we do translation quality assessment, particularly the way we check finished translations prior to official publication on government websites.

4.2.1 General Framework

The principal goal of translation is to communicate the same content message in both the source text and the target text. Translation products are usually evaluated and judged by two important criteria: faithfulness and transparency. The faithfulness criterion measures the fidelity of the translation and examines the extent to which the translated text accurately preserves the meaning of the source text. The aim of the transparency criterion

is to ensure that the translation reads well in the target language, and reflects the morphological, grammatical, syntactic and idiomatic conventions of the target language. Here is a typical scenario of how translation products are checked against the two criteria in the translation quality assurance process prior to publication.

In preparing *the Daily* news release texts for official publication at the Statistics Canada website, subject matter people from different divisions of Statistics Canada first draft their release texts, translate them and submit the texts to *the Daily* editing group for assembling, formatting and modification. The submitted texts are edited individually by two professional editors who are also native speakers of the language involved. When the editing is completed, each native speaker reads the text of his or her language and checks if the text conforms to the linguistic standards and cultural norms. In this way, the goal of translation transparency checking is achieved fairly easily, such that by the end of this process, it is difficult to tell which text is the source text and which one is the target, translated text. However, when it comes to translation faithfulness checking, texts have to be compared with their translation counterparts to see if they convey the same message or if there is any discrepancy in data presentation. Although this is a key step in the quality control process, it is a very challenging process if it is done by humans. Just imagine if we have to 'manually' map many pages of the bitexts chunk by chunk to identify potential deletion and insertion problems and to compare tables of numerical data that are formatted in different numbering systems.

This is the typical problem in translation quality assurance that we want to tackle in the BMIA model. Generally speaking, it is less painful to check the transparency of the translation. A native speaker of the target language reads the translated text, and modifies

the text within the message framework as if it were originally written in the target language. The more difficult job is to compare the translation texts bit by bit to find correspondence mapping problems that may indicate potential translation errors. We have numerous similar translation discrepancy checking tasks every day in government departments and agencies, in universities and private businesses. Currently, the process of this type of translation check is done mostly by hand. The translator has to read the source text and the target text line by line and compare them carefully sentence by sentence and sometimes word by word. When large amounts of bilingual texts are involved, particularly when many tables and numerical data are in the texts, this process of pairing and comparing the bilingual texts becomes a very daunting and stressful task in translation quality assurance.

What we aim to achieve in TextComp is to design algorithms that will help towards automatic translation discrepancy detection and translation correspondence profiling. To implement this, bilingual texts have to be first mapped and aligned, and then compared and checked. Basically, the solution BMIA adopts for text alignment is an approach that employs both statistical and lexical methods. BMIA aligns the paragraphs using the longest common subsequence method. Then it applies a linear regression forecasting model to determine the optimal alignment positions for text segments. In the process for translation discrepancy detection, text segments are checked against the anchoring lexical and structural features extracted from the texts. If the candidate pair can pass these criteria, the alignment link is established. At the same time, any detected regions of alignment problems and translation discrepancies are highlighted and marked. In translation correspondence profiling, the software agent scans through translations,

assembles evidence of correspondent constituents in translation, and gains ideas about breakdown structures of the translated segments. The system then does a profile of what it judges to be mutual translations and reports scores that may reflect the sureness of translations of the texts analyzed.

The BMIA program codes are tested in a Windows-Cygwin environment. Mostly, Perl scripts were written to process the data sets. Figure 4.18 shows the main framework of the TextComp system and demonstrates how it operates in practice. Overall, the system contains three processes. Process 1 is for data inputting (input En, input Fr). Process 2 is for translation text alignment, in which Align 1 is for paragraph alignment and Align 2 for text segment alignment. Process 3 is the process for bitext comparison. Comparison 1 in this process is for the general translation discrepancy detection mode and Comparison 2 for the translation correspondence profiling mode. The striped areas in the graph are the steps that happen in the background and are invisible to the user.

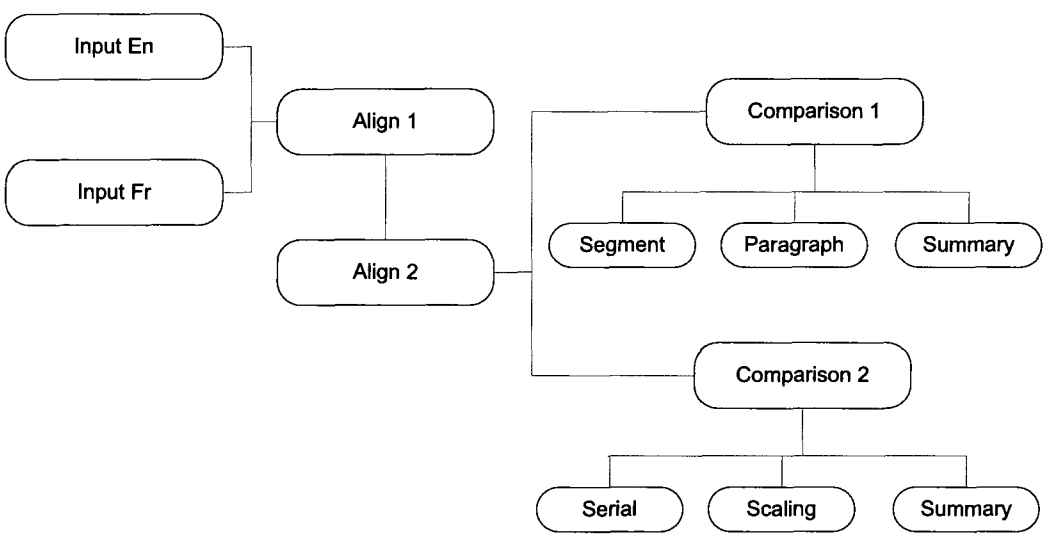


Figure 4.18. General framework of TextComp

4.2.2 Bilingual Text Alignment

Input bilingual texts have to be paired and aligned before they are compared. To align texts for automatic translation discrepancy detection, we designed algorithms that are based on dynamic programming and linear regression interval forecasting, rather than using the same algorithms as in SDTES. There are good reasons for changing the algorithms. First, SDTES is intended to align published, clean and consistent translations. For translation discrepancy detection, the bilingual texts to be compared are usually not finished translation products. They require more robust algorithms that can efficiently handle noisier and more uncertain data sets in the web-based application. The advantage of using algorithms based on dynamic programming and linear regression interval forecasting is that the alignment process relies more on the lexical anchoring information. For noisy data it can render more accurate alignment results with fewer massive misalignments than in the Gale-Church algorithm. Secondly, when texts are aligned in SDTES, tables, graphs and note-to-readers blocks have to be moved or deleted to set the stage for the length-based statistical model. We cannot do the same when our goal is to detect and identify translation errors. We need our text segments presented in the original context so that errors and discrepancies in translation can be checked and verified. This pre-alignment formatting requirement almost rules out the possibility that we can use the same SDTES algorithms for translation discrepancy detection. Thirdly, as we mentioned before, the Gale-Church method cannot deal with texts of different numbers of paragraphs. In SDTES, we depend on the main HTML tags in the source codes to help identify the paragraph boundaries before we apply the Gale and Church algorithm. However, when bilingual texts are cut and pasted into the text input areas in the web

interface for translation discrepancy detection, most of them are without HTML elements in the texts. If we align them using the SDTES algorithms without the help of information about HTML tags, the system would generate a lot of spurious translation correspondences. As a result, we need algorithms that can align noisy data sets efficiently and accurately within the allowable time frame in the web-based application.

TextComp automatically aligns the translations at two levels consecutively: the paragraph level (see Align 1 in Figure 4.18) and the text segment level (see Align 2 in Figure 4.18). The following are the main procedures that the software agent has to go through, like humans, when performing the task of pairing translations.

- Break the texts into chunks, i.e. take sentences within a certain length range from texts of both languages.
- Extract textual and structural properties of texts. This includes finding invariable elements in translation such as numbers and symbols.
- Compare the features extracted to see if they can serve as anchor points for correspondence connections in translation.
- Decide if there are enough related textual and stylistic features to determine that they are bilingually parallel in content and meaning.

4.2.2.1 Aligning Paragraphs

The paragraph level alignment is the initial alignment where bilingual texts are paired and compared. An important step in automatic alignment of paragraphs is to provide the agent with the lexical and structural features for distinguishing a bilingual connection

region from a non-connection region. Textual features that can be used to aid the alignment process include key structural elements in texts such as numbers, symbols, punctuation marks and cognates. Some of the features can have dual roles in rectifying a misalignment, and in checking if the translation is a problematic one. We observe from the training corpus of aligned translation segments that the distinction between mutual translations and unrelated texts becomes prominent when we compare the text segments and examine the degree of correspondences in these features. Overall, if the extracted features can be matched across the two languages, they can provide useful information for effective paragraph alignment. However, when reducing lines of texts to projected lists of features, we have to be careful: if the features are too fine-grained and we have too many features to be matched, the chances of exact matches for each line will be greatly reduced. This is close to directly comparing the two sections of texts in two different languages, and most of the paragraphs will not be aligned. It is very time-consuming too. On the other hand, feature selection cannot be too unrestrained either. Although coarse-grained selection of features can potentially save comparison time, and reduce the number of symbols in the finite alphabet to be used in our aligning algorithm, it can cause massive misalignments. For example, instead of one-to-one hard matching of punctuation marks in parallel texts, we allow no matching or one-to-several matching of punctuation marks.

When the features are extracted from bilingual texts, the system builds a parallel feature vector for each of the paragraphs. The LCS (longest common subsequence) model is applied with the extracted skeleton features in the vectors to find the best alignment paths.

For the past 30 years or so, the LCS model has been applied in various areas of computer science and molecular biology (Sellers 1980; Sankoff and Kruskal 1983). Many LCS problems are biologically motivated such as reconstructing long sequences of DNA from overlapping sequence fragments, storing and retrieving DNA sequences in databases and so on. LCS can be used in sequence comparison for similarities and frequently occurring patterns which can yield information about distances, alignments, traces or listings. For example, for two molecules or other sequences generated by an evolutionary process, if the distance is not small, it may mean that they are not so close, or their common ancestry was long ago (Sankoff and Kruskal 1983). In relation to natural language processing, LCS can find its applications in file revision detection, spelling correction, plagiarism detection, and speech recognition.

Given a sequence $A = (a_1, a_2, ..., a_i)$, another sequence $P = (p_1, p_2, ..., p_k)$ is a subsequence of A if there is an increasing sequence $(i_1, i_2, ..., i_k)$ of indices of A such that for all $j = 1, 2, ..., k$ we have $A_{ij} = P_j$. For example, *dust* is a subsequence of the sequence *industry*. For two sequences A and B, if P is a subsequence of both A and B, P is called a common subsequence. To solve the longest common subsequence (LCS) problem is to find a subsequence like P that has the maximum length.

Let *c[i,j]* be the length of an LCS of $A = (a_1, a_2, ..., a_i)$ and $B = (b_1, b_2, ..., b_j)$. We can compute *c[i,j]* row by row or column by column till *c[m,n]* is obtained:

$$c[i,j] = \begin{cases} 0 & \text{if } i = 0 \text{ or } j = 0 \\ c[i-1, j-1] + 1 & \text{if } i, j > 0 \text{ and } a_i = b_j \\ \max\{c[i, j-1], c[i-1, j]\} & \text{if } i, j > 0 \text{ and } a_i \neq b_j \end{cases}$$

If the alphabet of the two sequences is composed of letters including space, computation of the longest common subsequence usually involves three kinds of local modifications between sequences: (1) insertion of a letter in the sequence, (2) deletion of a base letter from the sequence and (3) substituting a base letter with another. There is a reciprocal relationship between insertion and deletion. Deleting one letter in one sequence may mean inserting one letter in the other. Because of this relationship, insertion and deletion are also called *indel* for short (Sankoff and Kruskal 1983:11). Given two sequences of strings, a subsequence is a sequence that appears in both of the sequences in the same relative order, but not necessarily contiguous. For example, in the string *abcdefg*, "abc", "abg", "bdf", "aeg" are all subsequences.

Recursive Algorithm

A brute force algorithm for finding the LCS of two sequences *A* and *B* involves generating each subsequence in *A* and checking if it is a subsequence of *B*. A string of length *n* has $O(2^n)$ different subsequences,

<a> -- a ($1 = 2^1-1=1$)

<a, b> -- a, ab, b ($2*1+1= 2^2 -1=3$)

<a, b, c> -- a, ab, b, ac, abc, bc, c ($3*2+1=2^3 -1=7$)

<a, b, c, d> -- a, ab, b, ac, abc, bc, c, ad, abd, bd, acd, abcd, bcd, cd, d ($7*2+1=2^4-1=15$)

We can take the shorter string, and test each of its subsequences for presence in the other string greedily. Here is the recursive formulation.

```
LCS_Recursive (a, b, i, j)
; a = English paragraph feature list
; b = French paragraph feature list
; m = size of a
; n = size of b
if i = 0 or j = 0
        then return 0
if a_i  = b_j then
        return LCS_Recursive (i-1, j-1) + 1
else
        if LCS_Recursive (i-1, j) >=  LCS_Recursive (i, j-1)  then
                return LCS_Recursive (i-1, j)
        else
                return LCS–Recursive (i, j-1)
```

This clearly would take a lot of time mapping every subsequence in *A*. The best case is *O(n)* --- when sequences are the same and the stack height is *n*. The worst case is $O(2^n)$ --- when sequences have no common elements, stack height = lg *n*. Finding LCS through recursion is a correct solution but it's very time consuming. This naïve approach of matching up every subsequence in *A* with *B* can yield a huge number of recursive calls. It is an exponential time algorithm because the recursive calls will re-compute c[*i, j*] that might have already been computed. For example, if we want to have the value of c[3, 4], we need to call the function to have the value of c[2, 4], c[3, 3], and for each of these

two, we have to repeatedly call c[2,3] and the nodes under it (see Figure 4.19). This process duplicates the computation by calling the function again and again, and thus increases the complexity.

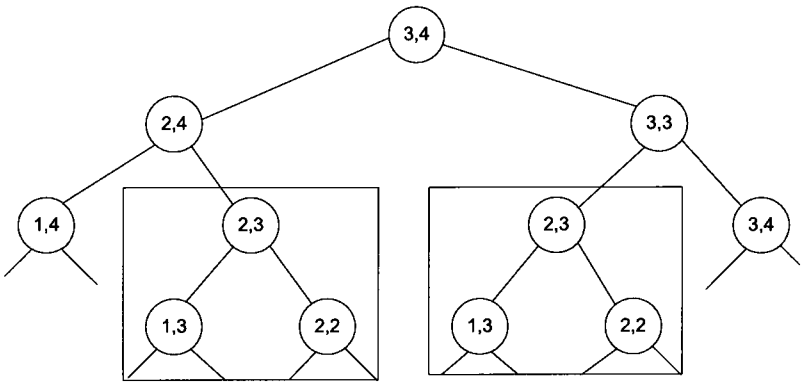


Figure 4.19. Recursion tree with m=3, n=4, height=m+n, potentially 2^{m+n} exponential. The boxed areas indicate sub-problems that have been repeatedly resolved.

Dynamic Programming Approach

Because the recursion in finding LCS can easily take exponential time, we must seek for a more efficient solution for comparing subsequences. We can use dynamic programming (DP) because there are only m*n distinct sub-problems, and we can compute the values in a bottom up fashion. In other words, we can set up a table pre-loaded with the trivial solution (0 for example). Each entry can be calculated depending only on the neighbours on its top, left, and top-left, as shown in the recurrence relation. For example, cell (2,3) depends on cell (1,3), cell (2,2) and cell (1,2). If we fill in the table, we can get the length of the longest common substring in c[*m, n*].

Let s and t be two sequences, and $s=s[1]s[2]...s[m]$ and $t=t[1]t[2]...t[n]$. Let $\sum$ be the finite alphabet for s and t. $\sum$ is augmented with a "space" which is denoted by "-". "-" is not in $\sum$. The alignment alphabet which includes "-" is $\sum'$. Let $\sigma(x,y)$ be the score of aligning x and y, for any $x,y \in \sum'$. *DP(i,j)* is the optimal alignment score for *s[1...i]* and *t[1...j]*. Suppose *DP(i,j)* has k columns, then

$DP(i,0)$ is the alignment score between *s[1...i]* and an empty sequence and $DP(i,0) = \sum_{k=0}^{i} \sigma(s[k],-)$.

$DP(0,j)$ is the alignment score between an empty sequence and *t[1...j]* and $DP(0,j) = \sum_{k=0}^{j} \sigma(-,t[k])$.

For each *DP(i,j)*, $1 \le i \le n$, and $1 \le j \le m$, we choose the one that yields the highest score:

$$DP(i,j) = \max \begin{cases} \mathrm{DP(i-1, j-1)} + \sigma(\mathrm{s[i], t[j]}) \\ \mathrm{DP(i-1, j)} + \sigma(\mathrm{s[i], -}) \\ \mathrm{DP(i, j-1)} + \sigma(\mathrm{-, t[j]}) \end{cases}$$

In BMIA, LCS is used to align paragraphs. We apply tabular computation of optimal alignment to compute *DP(i,j)* for all possible values of i and j. These values are stored in a table of size $(n+1)$ $(m+1)$. When computing the value of a specific cell $DP(i,j)$, only

cells *DP(i-1, j-1)*, *DP(i, j-1)* and *DP(i-1,j)* are examined and compared, along with two elements *s[i]* and *t[j]*. There are *(n+1)* *(m+1)* cells in the table, and the time complexity is thus O(*nm*).

```
LCS1(a, b, m, n)
; a = English paragraph feature list
; b = French paragraph feature list
; m = size of a
; n = size of b
; initialization
for i ← 1 to m do
        pos(i,0) ← 0
for j ← 1 to n do
        pos(0,j) ← 0
; outer loop
for i ← 1 to m do
        ; inner loop
        for j ← 1 to n do
                ; compute the LCS and add trace-marks for the backtracking matrix.
                if a_i = b_j then
                        pos(i, j) ← pos(i-1, j-1) +1
                        trace(i, j) ← UP_AND_LEFT
                else
                        if pos(i-1, j) >= pos(i, j-1) then
```

pos(i, j) ← pos(i-1, j) + 0

trace(i, j) ← UP

else

pos(i, j) ← pos(i, j-1) + 0

trace(i, j) ← LEFT

return pos and trace

Compared with the recursive solution, this approach entirely eliminates recursive calls. This change makes the difference between exponential time and polynomial time. As we fill in the table, we may want to mark which table entry was used so that we can recover the optimal solution (the actual LCS), and not just its length. This can be done by storing additional backtrack information (also referred to as "backpointers" in some textbooks) while computing the optimal values. After finding the length of the LCS, we can use these backpointers to trace backwards (see Table 4.6) from DP(*m,n*) and print out the aligned paragraphs.

		a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9	a_{10}
	0	0	0	0	0	0	0	0	0	0	0
b_1	0	↖ 1	←1	←1	←1	←1	←1	←1	←1	←1	←1
b_2	0	↑1	↖ 2	←2	←2	←2	←2	←2	←2	←2	←2
b_3	0	↑1	↑2	↑2	↑2	↑2	↑2	↖ 3	↑3	↑3	↖3
b_4	0	↑1	↑2	↑2	↑2	↑2	↑2	↑3	↑3	↑3	↑3
b_5	0	↑1	↑2	↑2	↑2	↖ 3	←3	↑3	↑3	↑3	↑3
b_6	0	↑1	↑2	↑2	↑2	↑3	↖4	←4	←4	←4	←4
b_7	0	↑1	↑2	↑2	↑2	↑3	↑4	↖ 5	←5	←5	↖ 5
b_8	0	↑1	↑2	↑2	↑2	↑3	↑4	↑5	↖ 6	←6	←6
b_9	0	↑1	↑2	↑2	↑2	↑3	↑4	↑5	↑6	↑6	↑6

Table 4.6. Tabular computation of the longest common subsequence.

		a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9	a_{10}
	0	0	0	0	0	0	0	0	0	0	0
b_1	0	↖ 1	← 1	← 1	← 1	← 1	← 1	← 1	← 1	← 1	← 1
b_2	0	↑ 1	↖ 2	← 2	← 2	← 2	← 2	← 2	← 2	← 2	← 2
b_3	0	↑ 1	↑ 2	↑ 2	↑ 2	↑ 2	↑ 2	↖ 3	↑ 3	↑ 3	↖ 3
b_4	0	↑ 1	↑ 2	↑ 2	↑ 2	↑ 2	↑ 2	↑ 3	↑ 3	↑ 3	↑ 3
b_5	0	↑ 1	↑ 2	↑ 2	↑ 2	↖ 3	← 3	↑ 3	↑ 3	↑ 3	↑ 3
b_6	0	↑ 1	↑ 2	↑ 2	↑ 2	↑ 3	↖ 4	← 4	← 4	← 4	← 4
b_7	0	↑ 1	↑ 2	↑ 2	↑ 2	↑ 3	↑ 4	↖ 5	← 5	← 5	↖ 5
b_8	0	↑ 1	↑ 2	↑ 2	↑ 2	↑ 3	↑ 4	↑ 5	↖ 6	← 6	← 6
b_9	0	↑ 1	↑ 2	↑ 2	↑ 2	↑ 3	↑ 4	↑ 5	↑ 6	↑ 6	↑ 6

Table 4.7. Trace of backtracking for the longest common subsequence.

Using the established pointers in the cells, we can easily trace back the alignments (see Table 4.7). The direction of the pointer in cell (i,j) indicates the value of which cell was used when DP(i,j) was computed. In the recovery of the optimal solution, the worst case is that there is no common subsequence for the two sequences, then the backtrack marker for the cell is either the up arrow or the left arrow. In this case, the running time of the program is the longest and is $O(mn)$.

Despite its improved time-complexity, the dynamic programming algorithm still makes quite a number of computations. To compute the longest common subsequence of two given sequences A and B of the same length n, the above approach, based on dynamic programming for solving the problem, is to fill a two dimensional dynamic programming table where each entry represents the length of the longest common subsequence between the corresponding prefix of A and the corresponding prefix of B. There are n^2 entries to be filled in the two-dimensional table if the two strings are of the same length. The matrix in the dynamic programming algorithm grows quadratically with

the lengths of the sequences. Two 100 item sequences would require a 10,000-item matrix and 10,000 comparisons would need to be done. Generally speaking, scanning through and searching of every cell of the table makes it computationally costly, particularly when the number of paragraphs to be compared is large. When many comparisons and computations have to be handled on the fly for the text alignment interface, the standard DP approach becomes too slow to be considered for the time critical web-based applications. Optimizations can be made to the algorithm above to speed it up. Since most of the time is spent performing comparisons between items in the sequences, we should be able to gain some reduction in computational complexity by narrowing the search space to save some comparisons.

When analyzing the alignment patterns in SDC, we notice that if text *E* is the translation of text *F*, beginning paragraphs in *E* and ending paragraphs in *F* rarely align as a translation pair. Paragraphs that are far away in position are not mapped as translations. For example, it is very uncommon for the first paragraph in one language to be aligned with the 30th paragraph or the 40th paragraph in another language. The overwhelming majority of paragraphs in *E* align only with paragraphs in *F* that are in the neighbourhood or with only a very short distance gap (Table 4.8). It can be thus observed that there is a band size we can use to limit the range of paragraphs to be considered for alignment. In comparing specific paragraphs, most paragraphs that are not in the band range can be ruled out as possible translation candidates. In most real world alignment cases, for the 55th paragraph in one language, the optimal alignment paragraph in another language is somewhere within the 45th to 65th paragraph range if for both languages the number of paragraphs is approximately the same (see Figure 4.20).

Aligned items	Aligned block	English paragraph number	French paragraph number	Alignment type
1	45	44	46	1:1
2	46	45	47	1:1
3	47	46	48	1:1
4	48	NULL	49	0:1
5	49	47	50	1:1
6	50	48	51	1:1
7	51	49	52	1:2
8	51	49	53	
9	52	50	54	1:1
10	53	51	55	1:1
11	54	52	56	1:1
12	55	53	57	1:1
13	56	54	58	1:1
14	57	55	59	1:1
15	58	NULL	60	0:1
16	59	56	61	1:1
17	60	57	62	1:1
18	61	58	63	1:1
19	62	59	NULL	1:0
20	63	60	64	1:1

Table 4.8. SDC blocks showing that aligned paragraph numbers are within a certain band range.

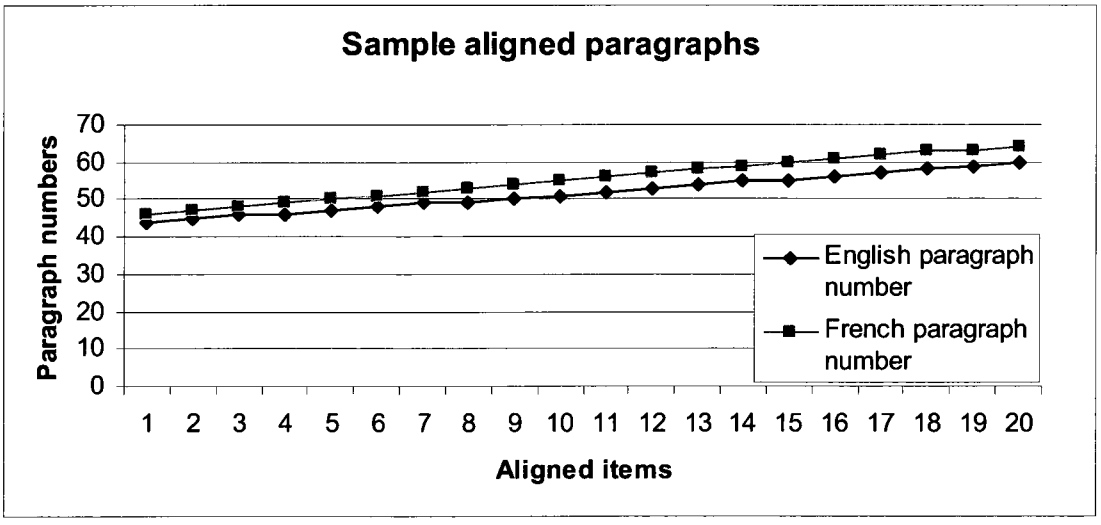


Figure 4.20. SDC aligned paragraph numbers are very close to each other.

K-band Algorithm

Based on the characteristics of bilingual text mapping in TextComp, we modified the standard DP model and added a band-size factor to limit the space in the search for optimal alignment. In TextComp, we consider a diagonal band of entries starting from the middle diagonal and ignore entries outside the chosen band. Usually for bilingual text mapping, there can be a hidden range of distance within which the elements in one text sequence can be aligned with the elements in another text sequence. This means that we can set a band range term k around the best scoring diagonal (Figure 4.21), so that alignment candidates can be mapped more efficiently. We can compute, instead of the entire LCS, only those candidates within the band range. This can be done in $O(km)$, not $O(n^2)$ space. In this way, the performance of the algorithm for the LCS problem is partly dependent on variables other than the sizes (m, n) of the two input text sequences.

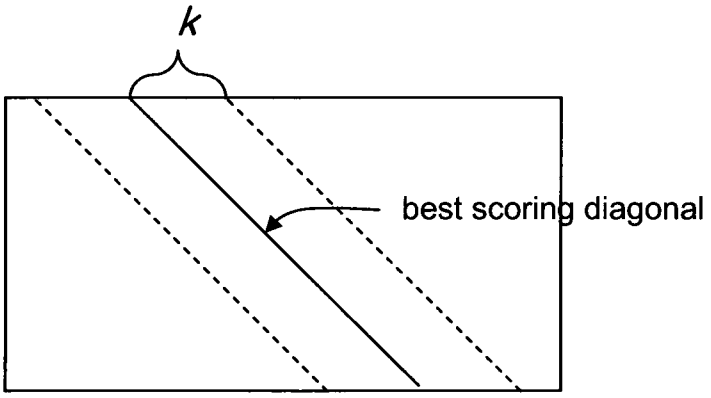


Figure 4.21. The k-band search space for TextComp.

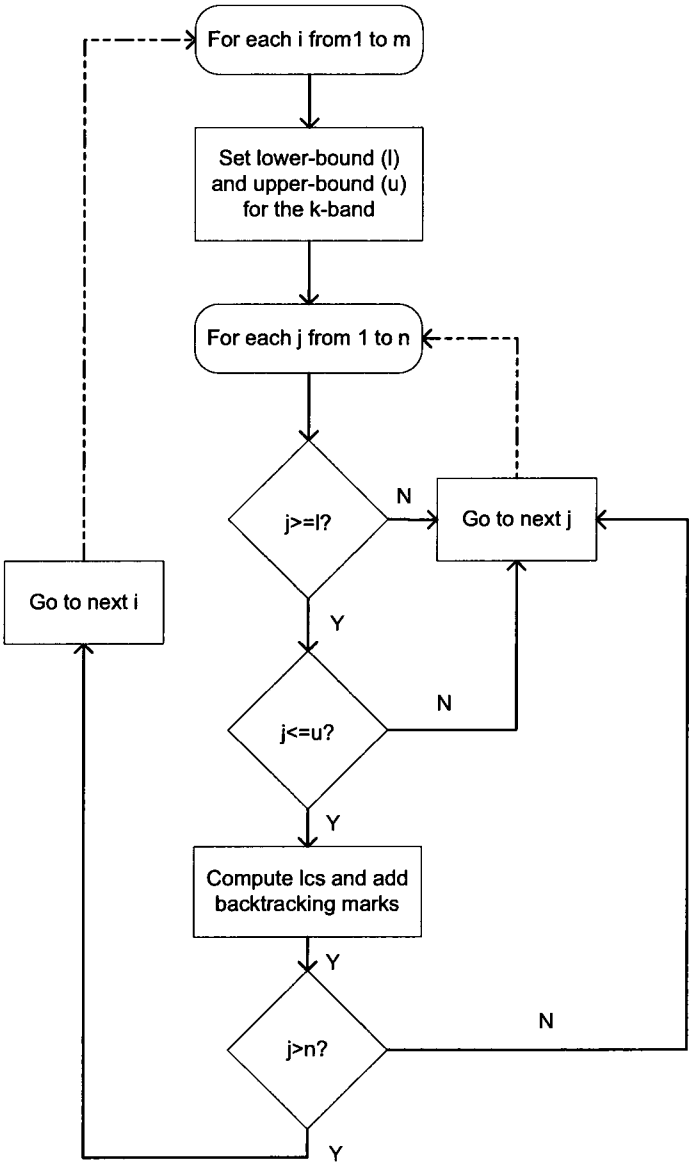


Figure 4.22. Modified DP algorithm to solve the LCS problem in TextComp.

The *k*-band method can narrow the search space and reduce the search complexity efficiently and can save, in most cases, a lot of comparisons that would have to be made if the standard DP algorithm is applied. For the inner loop, it is not traversing every

single point in j from 1 to n but only to the limited k band range, that is, to $j+k$ and $j-k$ (k>=0, and k<=n/2). When the band size is 10, $|A|$<=10 and $|B|$<=10, the band size limiting is not necessary. In the best case scenario, if one paragraph in a language is translated as one paragraph in another language, this band size optimization can be set to a very small number such as 25 to 30. In the worst case scenario, if the length difference between |A| and |B| is huge, the band size can be very large. Even if it is in the worst case, the maximum number of comparisons will not exceed the number of comparisons to be performed in the standard dynamic programming model. Figure 4.22 shows the modified dynamic programming algorithm to solve the LCS problem using the k-band method in TextComp.

For each of the lines in the English paragraph feature list A, we compute the k-band lower bound (l) and the k-band upper bound (u). The size of this list is m. For each line in the French paragraph feature list B, we check the line number to see if it falls in the range of band (j>=l and j<=u). The size of B is n. If the line number is out of the range, we skip the comparison operation; otherwise we compute the LCS and add trace-marks for the backtracking matrix. This inner loop process continues till j is greater than n.

The following is the pseudo-code for LCS-TextComp, our k-band algorithm that modifies the standard dynamic programming approach. Please note the difference between LCS-TextComp and the standard DP function LCS1 which we described in the previous section. In LCS-TextComp, we added sections to initialize band size k according to the length difference of the two sequences. We also added blocks to keep the matching process within the k-band and to reduce the number of iterations.

```
LCS-TextComp(a, b, m, n)
; A = English paragraph feature list
; B = French paragraph feature list
; m = size of A
; n = size of B

; initializing the matrix
for i ← 1 to m do
        pos(i,0) ← 0
for j ← 1 to n do
        pos(0,j) ← 0

; initializing band size k
gap ← |m - n|
if gap <=10 then
        gap ← 0
k ← 25 + gap

; outer loop
for i ← 1 to m do
        ; set lower_bound and upper_bound for the k-band range
        lower_bound ← i - k
        upper_bound ← i + k
```

```
if lower_bound <1 then
        lower_bound ← 1
if upper_bound >n then
        upper_bound ← n

; inner loop
for j ← 1 to n do
        if (j)  >= lower_bound and j <= upper_bound then
        ; compare within the k band
        ; compute the lcs and add trace-marks for the backtracking matrix.


                if a_i = b_j then
                        pos(i, j) ← pos(i-1, j-1) +1
                        trace(i, j) ← UP_AND_LEFT
                else
                        if pos(i-1, j) >= pos(i, j-1) then
                                pos(i, j) ← pos(i-1, j) + 0
                                trace(i, j) ← UP
                        else
                                pos(i, j) ← pos(i, j-1) + 0
                                trace(i, j) ← LEFT


return pos and trace
```

In selecting the k value for the k-band algorithm, we have to strike a balance between speed and precision. We have to make sure that the band is not too small. Small band size can make the program run faster, but will miss some exact matches or correct alignments. The band size parameter cannot be too large either. If it is, it becomes more likely to find the correct translation candidates, but the system will be making many unnecessary comparisons, and the system cannot cut down much search space. In view of the observed data we gathered about paragraph number deviations in the alignment of SDC data, we set the default k value to be 25. Normally, when the English paragraph is 100, the French translation should be able to be located in a range of paragraph 100-25 to paragraph 100+25. However, if the paragraph number difference between the two languages exceeds 10, we add the difference to the band size k.

For the k-band algorithm in TextComp, we have 2 feature lists A and B, which consist of vectors of extracted features. Each vector in the feature lists bears matching information about a paragraph. A vector can be viewed as a condensed character in a string sequence. In this way, the whole matter of paragraph alignment is reduced to an exercise of aligning characters in a sequence. We define the two lists A or B as two sequences containing vectors of selected features. Thus the two sequences in bilingual text alignment are $A = e_1 \ldots e_i \ldots e_m$ and $B = f_1 \ldots f_j \ldots f_n$. We assume that the difference between lengths $|A|$ and $|B|$ is not great. A subsequence P is a subgroup of the feature lists that can be found in sequences of A and B. This can be denoted by $P \subseteq B$ and $P \subseteq A$. The longest common subsequence is the longest succession of matched feature vectors that preserve the same relative order in the two sequences. They do not have to be contiguous.

The application of LCS in TextComp hinges upon the feature vectors extracted from the translation texts. Without the feature vectors, the implementation of LCS is almost impossible, because, in theory, using LCS to compare two strings which have nothing in common is meaningless. If we directly use LCS to compare texts of different languages, we are almost always comparing texts that share no common paragraphs. For the adapted LCS approach, we model our data sets before the application of the LCS model. A dot map for each paragraph of the two texts is built. The dot map consists of a skeleton of unified features such as numbers, cognates, punctuation marks etc. Elements in this dot map vector look like the interlingua that is shared in different languages in machine translation. To increase the comparability of the elements in the feature list, TextComp has a filtering mechanism to remove duplicates in order to reduce the sensitivity of the alignment algorithm. Then the individual elements are sorted to allow for exact string matching. When this is done, the dot map of the English text and the dot map of the French text are comparable. In concept, the skeleton feature vector for each paragraph can be compressed such that it can be treated as if it were a valid letter in the sequence alphabet. Thus a paragraph in bilingual text mapping can be represented by a letter and the problem of bilingual text comparison becomes a problem of sequence comparison.

In Figure 4.23, each point in the cell represents an element such as a cognate word, or a number in the feature vector that is shared between an English paragraph and a French paragraph. Paragraph alignment is derived from correspondences between a text S and its translation T; that is, between the respective feature lists $A= \{e_1, e_2, \ldots, e_9\}$ and $B= \{f_1, f_2, \ldots, f_{12}\}$. The paragraph alignment through dynamic programming is

{(e_1, f_1), (e_2, f_2), (e_3, f_3), (NULL, f_4), (e_4, f_5), (NULL, f_6), (e_5, f_7), (e_6, f_8), (e_7, f_9), (NULL, f_{10}), (e_8, f_{11}), (e_9, f_{12})}

which associates paragraph e_1 with paragraph f_1, paragraph e_2 with paragraph f_2 and so on.

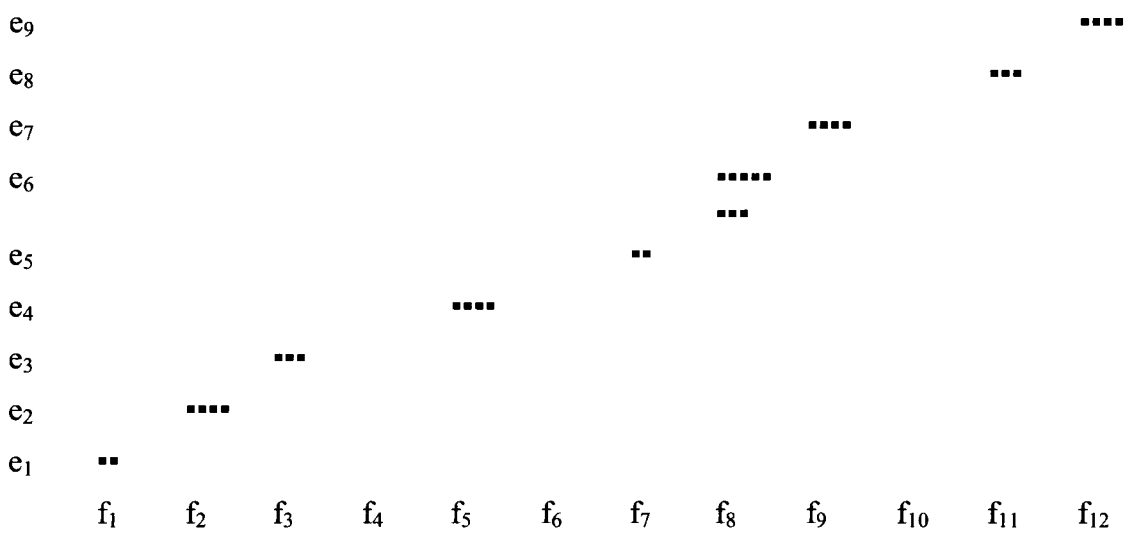


Figure 4.23. Projected feature vectors in the paragraph alignment grid in TextComp. Each point in the cell represents a shared feature element.

At the same time, bilingual text alignment can be considered as a special kind of LCS problem and some problems for bilingual text mapping may not be present in other LCS applications. For example, *indels* are uncommon in the bilingual text alignment, and the majority of the comparisons are 1:1 matches. In a common LCS application, we rarely combine two letters in one sequence to match one letter in the other sequence. We use a match plus an *indel*. In bilingual text mapping, alignment types such as 1:2 and 2:1 are acceptable. Also, a common LCS application usually does not allow alignment of letters

in a crossed order. For example, we do not usually say *ab* is an LCS of *abc* and *bad.* In bilingual text mapping, we can treat *ab* in *abc* as a letter unit, and similarly for *ba* in *bad.* Then when *ab* and *ba* are compared as if they were single letter units, they can match. This kind of matching is usually defined as the 2:2 alignment type in bilingual text mapping. It allows two adjacent characters of swapped order to be aligned.

4.2.2.2 Aligning Text Segments

When the optimal alignment path is established for paragraphs, we break the paragraphs into text segments, and do the alignment of text segments at the same time. This is indicated as Align 2 in Figure 4.18. The statistical model that TextComp is employing for the alignment of text segments is the least squares fitting technique in regression analysis. Prior to adopting such a technique, we have to make sure that there is strong correlation between the length distributions of the two texts. The technique cannot be properly applied if the correlation does not present evidence of strong association between length distributions.

Since the target data sets of TextComp in the BMIA model are mostly web-based bilingual texts in government organizations, we analyzed the aligned pairs of the StatCan Daily Corpus to see if we can observe some patterns of length variation in aligned chunks of bilingual texts. For sampling from SDC, we took an aligned pair from every 40 pairs and collected a total of 12,219 aligned translation pairs in SDC. We obtained the lengths for the two text segments in the translation pair, and divided the length range into bins such as lengths 1 to 10, 11 to 20, 21 to 30 and so on.

Some interesting observations can be made in Figures 4.24 and 4.25 about the sampled SDC data. The English and French texts share a very similar increasing or decreasing trend in the frequencies of the length range bins. For example, the length range of 1 to 49 is very frequent in both of the languages, as can be seen from the leftmost 4 bins in Figure 4.24. This means that in SDC, many aligned pairs consist of only short text segments of about one to eight words. For shorter length bins, English is more frequent, and for longer length bins, French is more frequent (see Figure 4.24). The switching point is at around bin range 14.

In Figure 4.25, when the bin range is around 25, the cumulative proportion for both languages is very close to 1, which means aligned text segments that are longer than 250 characters are only few and far between. Also, it seems that it takes more bins in French (for example, 390) to cover the same amount of cumulative proportion as can be covered in English (which is 300).

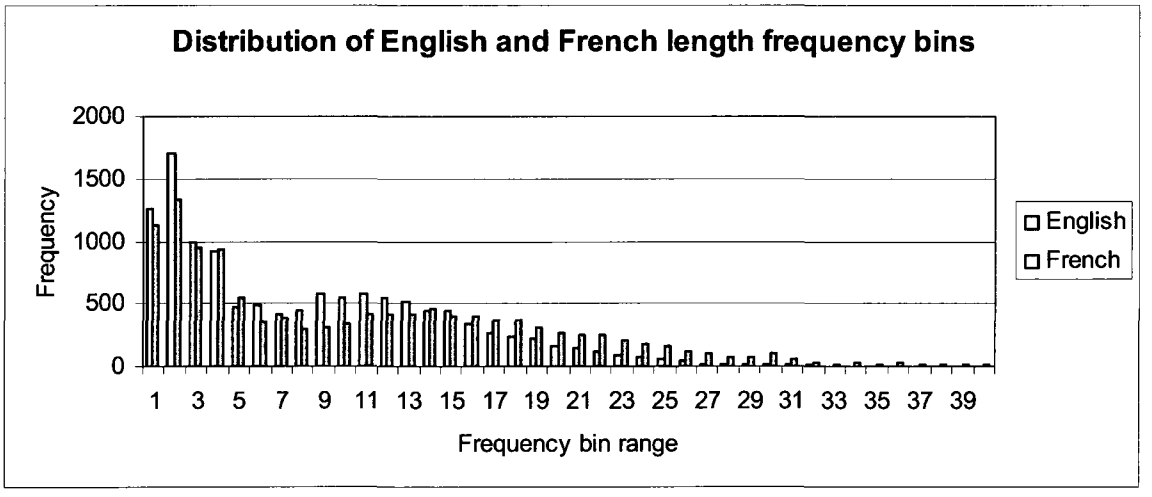


Figure 4.24. Distribution of English and French length frequency bins (length range = bin range * 10)

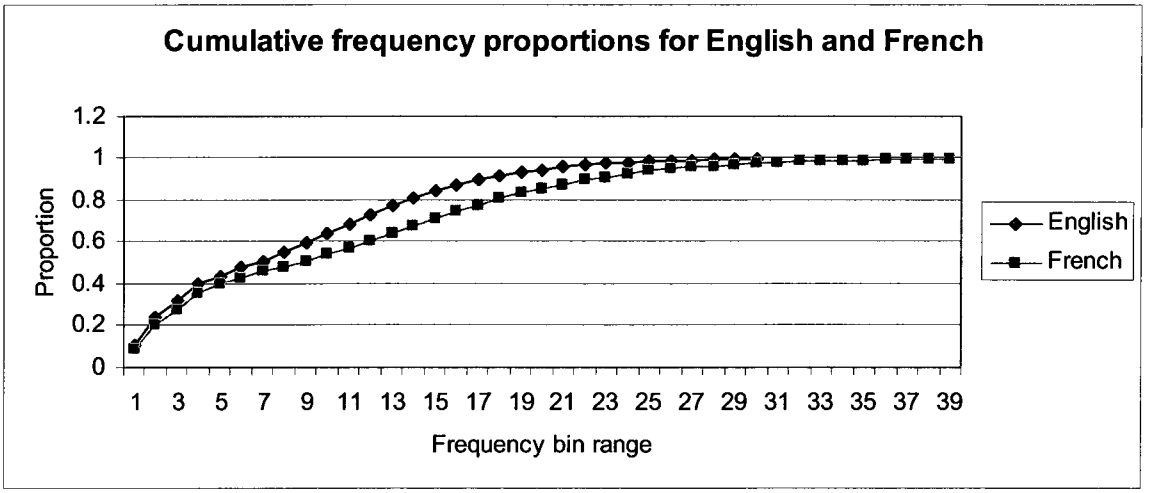


Figure 4.25. Cumulative proportions of the length bins in English and French (length range = bin range * 10)

When we closely examine the relationship between the upper bound of the length range bin and the cumulative proportion in Figures 4.25, we notice that length bins at a step of 7 in English covers a very similar proportion as the French length bins at an interval of 9. This is corroborated by the numbers in bold type in Table 4.9.

In general these observations can demonstrate that the association in length distribution indicates a strong relationship between individual length points in English text segments and the French text segments. Next, we use the correlation coefficient (denoted as Pearson's product-moment r) as a measurement of the relationship between of the two variables, the English text segment length x and the French text segment length y. We are interested in finding out if in most cases when the English text segment length is long, the French text segment length is long also. The higher the correlation coefficient, the stronger correlation relationship is indicated between the variables, and the better the linear regression model will fit the actual data points.

French			English		
Upper bound of the length range bin	Frequency count	Cumulative proportion	Upper bound of the length range bin	Frequency count	Cumulative proportion
10	1120	0.091661			
20	1339	0.201244			
30	943	0.278419	10	1258	0.102954
40	931	0.354612	20	1698	0.241918
50	553	0.399869	30	996	0.323431
60	350	0.428513	40	918	0.39856
70	385	0.460021	50	481	0.437925
80	295	0.484164	60	485	0.477617
90	**315**	**0.509944**	**70**	**409**	**0.511089**
100	347	0.538342			
110	408	0.571733			
120	420	0.606105	80	445	0.547508
130	421	0.64056	90	571	0.594238
140	461	0.678288	100	542	0.638596
150	398	0.71086	110	578	0.685899
160	406	0.744087	120	541	0.730174
170	376	0.774859	130	518	0.772567
180	**374**	**0.805467**	**140**	**446**	**0.809068**
190	315	0.831246			
200	265	0.852934			
210	257	0.873967	150	438	0.844914
220	247	0.894181	160	339	0.872657
230	201	0.910631	170	271	0.894836
240	184	0.925689	180	233	0.913905
250	165	0.939193	190	227	0.932482
260	113	0.948441	200	162	0.94574
270	**102**	**0.956789**	**210**	**145**	**0.957607**
280	78	0.963172			
290	73	0.969146			
300	99	0.977249	220	112	0.966773
310	58	0.981995	230	95	0.974548
320	34	0.984778	240	76	0.980768
330	21	0.986496	250	54	0.985187
340	32	0.989115	260	50	0.989279
350	21	0.990834	270	21	0.990998
360	**23**	**0.992716**	**280**	**21**	**0.992716**
370	17	0.994108			
380	10	0.994926	290	18	0.994189
390	10	0.995744	300	21	0.995908
400	10	0.996563			

Table 4.9. Length range bins at a step of 7 in English cover a very similar cumulative proportion to the French length bins at an interval of 9.

In computing the correlation coefficient for TextComp, the sum of squares of x is defined as

$$SSx = \sum (x - \bar{x})^2$$

$$= \sum x^2 - \frac{(\sum x)^2}{n}$$

the sum of squares of y is

$$SSy = \sum (y - \bar{y})^2$$

$$= \sum y^2 - \frac{(\sum y)^2}{n}$$

the sum of products of x and y is

$$SPxy = \sum (x - \bar{x})(y - \bar{y})$$

$$= \sum xy - \frac{(\sum x)(\sum y)}{n}$$

and the correlation coefficient r is

$$r = \frac{SPxy}{\sqrt{SSxSSy}}$$

$$= \frac{\sum (x-\bar{x})(y-\bar{y})}{\sqrt{\left[\sum (x-\bar{x})^2\right]\left[\sum (y-\bar{y})^2\right]}}$$

$$= \frac{\sum xy - \frac{(\sum x)(\sum y)}{n}}{\sqrt{\left[\sum x^2 - \frac{(\sum x)^2}{n}\right]\left[\sum y^2 - \frac{(\sum y)^2}{n}\right]}}$$

Applying the equation to the aligned text segment lengths of English and French in the TextComp training data, we have

$$r = \frac{170238067 - \frac{(985891)(1264148)}{12219}}{\sqrt{\left[133209971 - \frac{(985891)^2}{12219}\right]\left[222542528 - \frac{(1264148)^2}{12219}\right]}}$$

$$= 0.9725$$

The Pearson's product-moment r is .97, which is very close to 1. This indicates that there is a very strong association between the English text segment length and the French text segment length and the length data points cluster very closely around a straight line. The correlation coefficient shows that the linear regression model can be a very suitable model to apply for the length data in TextComp.

In bivariate or multivariate statistical methods, we have two types of variables: dependent variables and independent variables. For bivariate population we deal with only one independent variable. A commonly used statistical analysis of bivariate population is the linear least squares regression model that can provide a solution to the problem of finding the best fitting straight line through a set of points on a scatterplot. A regression equation can manifest the nature of the relationship between two or more variables algebraically. It can be used to indicate the extent to which one can predict some variables by knowing others, or the extent to which some variables are associated with others.

Let us consider a regression line as a running series of means of the expected value of y for each value of x. Suppose we have a sample of n data points (x_1, y_1), (x_2, y_2), ..., (x_n, y_n). For a given data point, say the point (x_i, y_i), the observed value of y is y_i and the predicted value of y would be $\hat{y}_i = a + bx_i$ where

$$a = \bar{y} - b\bar{x}\,,\quad b = \frac{SS_{xy}}{SS_{xx}}$$

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i\,,\quad \bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$$

$$SS_{xy} = \sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})$$

$$SS_{xx} = \sum_{i=1}^{n}(x_i - \bar{x})^2$$

n = sample size.

This formula is a linear function for a straight-line graph with slope b and y-intercept a (often called a "constant"). We are interested in estimating b, which shows the effect of the independent variable on the dependent variable. When the English length is regarded as the independent variable x, then b can be the dependent variable. In our context, this can be interpreted as any increase or decrease in the English length having an effect of b on the French length.

Normally, the scatter of data around the linear regression line approximately follows a Gaussian distribution. The least squares line is a mathematical representation of reality and is usually approximate rather than exact. Moreover, the theoretical model permits random, unexplained deviations from this line of expected values (the error term ε) for individual independent variables. These factors can add uncertainty to our prediction. In the real world applications, it is not very common that every data point in a bivariate population falls exactly on the line of regression. So, in TextComp, we use a regression prediction interval to include a margin of error and to capture this uncertainty.

In computing a $100(1-\alpha)\%$ prediction interval for an individual y at a fixed x, we are trying to estimate the likely uncertainty in point forecasts. A common method of calculating the prediction interval is to use a theoretical formula conditional on a best-fitting model.

$$a + bx_i \pm t_{\alpha/2,n-2} \cdot s_\varepsilon \sqrt{1 + \frac{1}{n} + \frac{(x_i - \bar{x})^2}{SS_{xx}}}$$

TextComp is interested in predicting the upper limit of the prediction interval of the length of either English or French on the basis of the known value of one of them. For example, when the English length is 100, what is the maximum length we can consider for the French translation based on the best fit regression line? We can calculate the expected value of y using the linear regression equation, but we also want to know how far away from the expected point is the maximum acceptable distance measured in characters. Suppose the data set consists of pairs of values (x_1, y_1), (x_2, y_2), ... taken from SDC, x is the independent variable representing the English length, and y is the dependent variable representing the French length. Here is the data modeling process from which the TextComp linear regression forecasting model is derived to predict the expected upper bound length.

- Step 1. For each pair of sampled translation pairs in the StatCan Daily Corpus, calculate the logarithm of x for the English length and the logarithm of y for the French length.
- Step 2. Estimate unknown parameters such as intercept and slope in the distribution and build the linear regression equation.
- Step 3. Select a critical value t and compute the prediction interval to arrive at the predicted upper bound length u' for each pair of the sampled translation population.
- Step 4. Take the antilog of u' to obtain u.

- Step 5. Build a new linear regression equation using the bivariate data sets with x corresponding to u.
- Step 6. Use the new equation with x as the independent variable and u as the dependent variable. The u value thus obtained is the predicted maximal length allowed for French when the English length is x.
- Step 7. Empirically test and statistically check the adequacy of the model to see if it can cover most of the observed data. If not, go to step 3, tune the critical value t, and repeat steps 4, 5, 6 and 7 to set up a new linear regression equation.

Figure 4.26 demonstrates the t critical value parameters we used to tune the performance of the model for regression interval forecasting in the bilingual text comparison system.

This regression interval forecasting model can help us judge if a pair of text segments are potentially legitimate translations by the length criterion. For example, for the most common 1:1 alignment pattern, we have the very basic length condition to meet before they can be considered as a legitimate translation pair. Let *len1_1match[i,j]* be a function to judge if a text segment pair can be aligned for $E = (a_1, a_2, \ldots, a_i, a_{i+1}, \ldots, a_m)$ and $F = (b_1, b_2, \ldots, b_j, b_{j+1}, \ldots, b_n)$. a_i, a_{i+1} are the text segment lengths for English and b_j, b_{j+1} are the text segment lengths for French. We obtain the Boolean value of *len1_1match[i,j]* as follows:

$$len1\_1match[i,j] = \begin{cases} 1 & \text{if } a_i <= (pn21 * b_j + pn22) \text{ and } b_j <= (pn11 * a_i + pn12) \\ & \text{and} \\ & \text{if } a_{i+1} <= (pn21 * b_{j+1} + pn22) \text{ and } b_{j+1} <= (pn11 * a_{i+1} + pn12) \\ 0 & \text{otherwise} \end{cases}$$

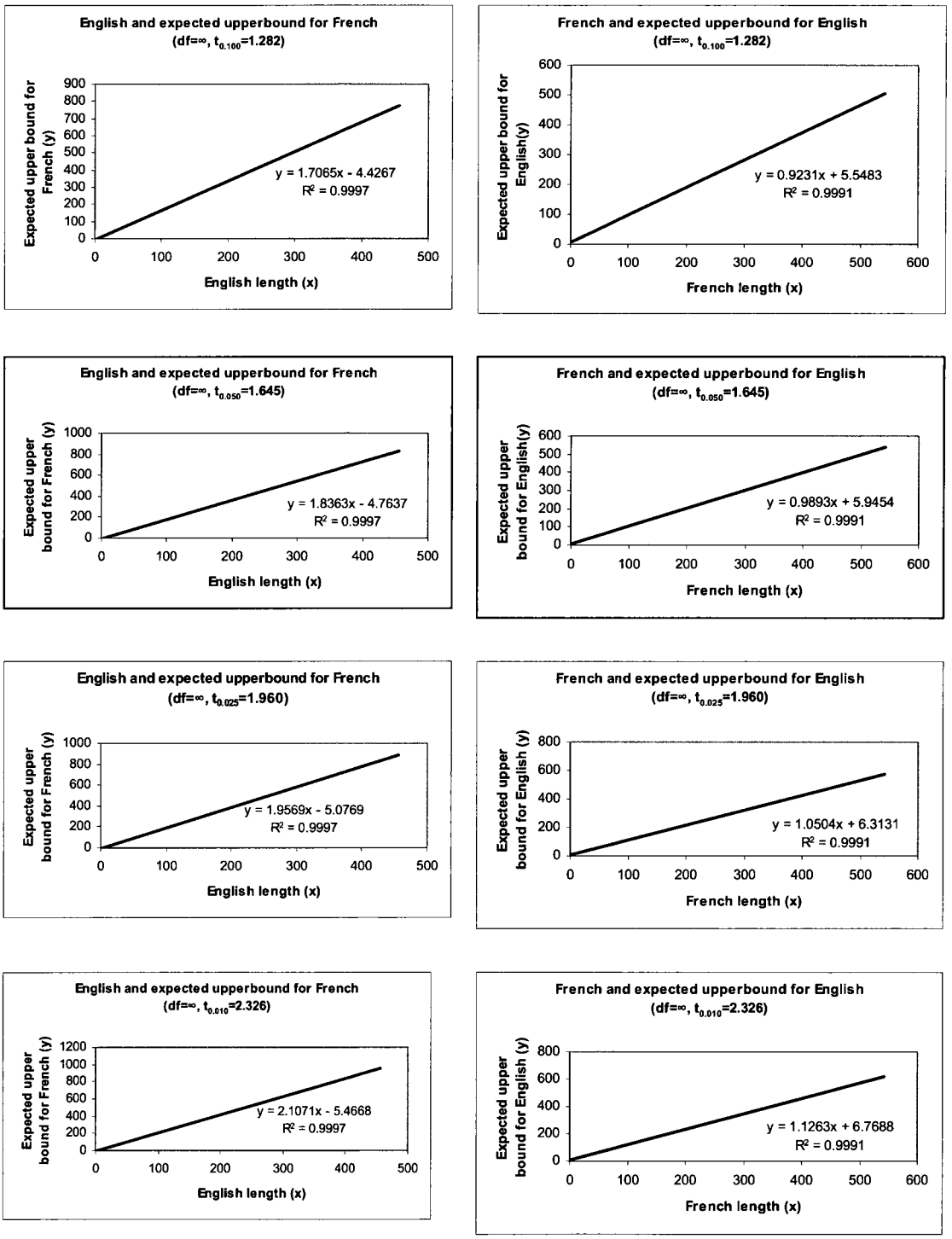


Figure 4.26. Tuning the *t* critical values for TextComp.

Here, we have two regression equations: one for the predicted upper bound length for English and the other for the predicted upper bound length for French. *pn11* and *pn12* are the slope and the intercept of the regression equation one when the English length is the dependent variable; *pn21* and *pn22* are the slope and the intercept of regression equation two in which the French length is the dependent variable. *pn12* and *pn22* can have negative values. For TextComp, these are the values we adopted:

$t = 2.576$ (99.9% confidence interval)

$pn11 = 1.2$

$pn12 = 7$

$pn21 = 2.2$

$pn22 = -6$

We obtained these values by testing the t critical value parameters with the SDC data. For noisy data of published translated texts or data sets that are supposed to be noisy such as those pre-publication translations to be checked, we need to set the critical value of t higher, so that most instances of alignments or varied lengths can be allowed and covered.

In aligning text segments, we employ a *forward-backward matching* approach. The assumption of this approach is that most of the translation text segments can be aligned as 1:1 matches. This assumption is based on the statistics from the analysis of SDC: 93% of the text alignment patterns are 1:1 alignments (see Figure 4.27). Thus for the majority of beads of text, we do not have to use the algorithm that we would use to find 1:2, 2:1, and 2:2 alignment types. This approach can save search space and alignment comparisons and

can improve the efficiency. The *forward-backward matching* algorithm first works forward: TextComp takes one text segment at a time from each language and compares their lengths. The system uses the linear regression forecasting model as a yardstick and checks to see if the candidate text segments form a good alignment pair. The algorithm can basically allow the following three types of length variations:

1. E ----------*i*

   F ----------*j*

The Tribunal's mandate is to provide fair, timely and effective disposition of international trade cases, government procurement review and government-mandated inquiries in various areas of the Tribunal's jurisdiction.	Le Tribunal a pour mandat de veiller au règlement équitable, opportun et efficace de dossiers commerciaux internationaux, des examens des marchés publics et des enquêtes menées sur instructions du gouvernement dans divers domaines relevant de la compétence du Tribunal.
The Tribunal conducts inquiries into complaints relating to unfair trade (i.e. dumping and subsidizing), requests for protection from import competition (safeguards) and complaints regarding federal government procurement.	Il mène des enquêtes sur des plaintes relatives à des pratiques commerciales déloyales (c. -à-d. dumping et subventionnement), sur des demandes de protection contre les importations (mesures de sauvegarde) et sur des plaintes concernant les marchés publics fédéraux.
The Tribunal hears appeals from decisions of the Canada Revenue Agency (CRA) and the Canada Border Services Agency (CBSA) under the Excise Tax Act and the Customs Act respectively.	Il entend les appels à l'égard des décisions rendues par l'Agence du revenu du Canada (ARC) et l'Agence des services frontaliers du Canada (ASFC) en vertu de la Loi sur la taxe d'accise et de la Loi sur les douanes, respectivement.
In its advisory role, the Tribunal undertakes general economic inquiries and tariff references for the Minister of Finance or the Governor in Council.	Dans son rôle consultatif, il entreprend des enquêtes sur des questions économiques et tarifaires de portée générale pour le ministre des Finances et le gouverneur en conseil.
In so doing, the Tribunal contributes to Canada's competitiveness.	Ce faisant, il contribue à assurer la compétitivité du Canada.

Figure 4.27. Default one-to-one text segment mappings within a paragraph in file RE3254

The lengths of two text segments are the same ($i>0$ and $j>0$ and $i=j$). However, when the translations are expository texts, paired translation text segments with exactly the same length are rare.

2. E ----------*i*

F ----------*j*

The French segment is longer than the English one ($i>0$ and $j>0$ and $i<j$), but it is within the range of the linear regression forecasting model. In the translation text data we collected, this pattern accounts for most of the aligned pairs of text segments.

3. E ----------*i*

F ----------*j*

The English segment is longer than the French ($i>0$ and $j>0$ and $i>j$), but is within the maximal acceptable range of the linear regression forecasting model. This pattern is not uncommon in the web-based bilingual materials from websites hosted by government departments and agencies in Canada. However, they are not as frequent as the second pattern described above.

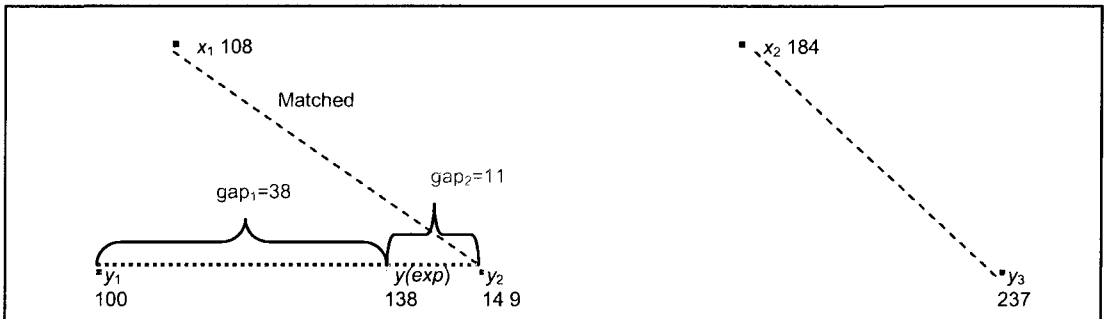
When a pair of text segments cannot pass the length criterion, the algorithm begins to work backwards, namely working from the last text segment in the paragraph to the first using the same length criterion. Based on the prior probability of the alignment patterns, when the algorithm hits a pattern that is not a 1:1 alignment type, it is most likely the place where we find a 2:1 or 1:2 or 2:2 pattern. To prevent the worst case alignment problems, the backward aligner includes a mechanism to deal with alignment patterns other than 1:1. This mechanism is used only when there is a problem in aligning the

candidate pair as the 1:1 alignment type. In this way, additional comparisons are performed only sparingly for alignment patterns that are not of the 1:1 alignment type.

One advantage of the *forward-backward matching* algorithm is that it can allow for alignment patterns like 1:3 or 3:1. In general, the *forward and backward matching* algorithm is like finding the AST in Kay and Röscheisen (1993), but we do not use the WAT to arrive at SAT. When BMIA finds the alignable text segment, the agent usually confirms the alignment by the lexical clues or the length criterion. If the pair is not alignable as a 1:1 type, the agent then tries other appropriate alignment types such as 2:1, 1:2, 2:2, 1:3 or 3:1 according to the length properties of the text segments. This process does not reiterate, and is thus much less expensive in computation.

There are times when two length points are competing for being included in an alignment pair, and the length criterion allows for two alignment options such as 1:1 or 1:2. In this case TextComp applies the basic linear regression equation to find the closest fit to the expected value of the dependent variable (Figure 4.28). Suppose the alignment agent in BMIA already knows the length of English sentence 1 (x_1), and is trying to find the matching point of the French texts from two candidate sentence boundary points (y_1, and y_2). Both y_1 and y_2 can be valid matching points using the linear regression forecasting model. The agent will first calculate the expected length offset position in the English text y_{exp}. Then it counts the gap distances of the expected offset point to y_1 and y_2 respectively, and compares the gap differences. The matching point is determined by finding out which one is closer to the expected theoretical matching point y_{exp}. Figure 4.28 illustrates the decision making process. In the example demonstrated in the figure, it

is found that the length criterion prefers a 1:2 alignment type because the gap $|y_{exp} - y_2|$ is much smaller than $|y_{exp}\text{-}y_1|$.



$x_1 = 108$ $x_2 = 184$ $y_1 = 100$ $y_2 = 149$ $y_3 = 237$	$y(\text{exp}) = 1.27x + 1$ $= 138.16$ $gap_1 = y(\text{exp}) - y_1$ $= 38$ $gap_2 = y_1 + y_2 - y(\text{exp})$ $= 11$	$Match(i,j) = \min\begin{cases} y(\text{exp}) - y_1 \\ y_1 + y_2 - y(\text{exp}) \end{cases}$ $= gap_2 = 11$ *so* $A = \{(x_1, y_2), (x_2, y_3)\}$

Figure 4.28. Alignment decision making based on the least squares fitting technique.

Experiments were conducted to check if the predicted region of the linear regression forecasting model can account for most of the length variability in text segment alignment. The empirical testing results are very encouraging. Figure 4.29 is a sample of the test that records the lower bound y, the observed y, and the upper bound y for the first 50 text segments of a randomly selected file pair. Almost all the observed y values fall between the upper bound and the lower bound.

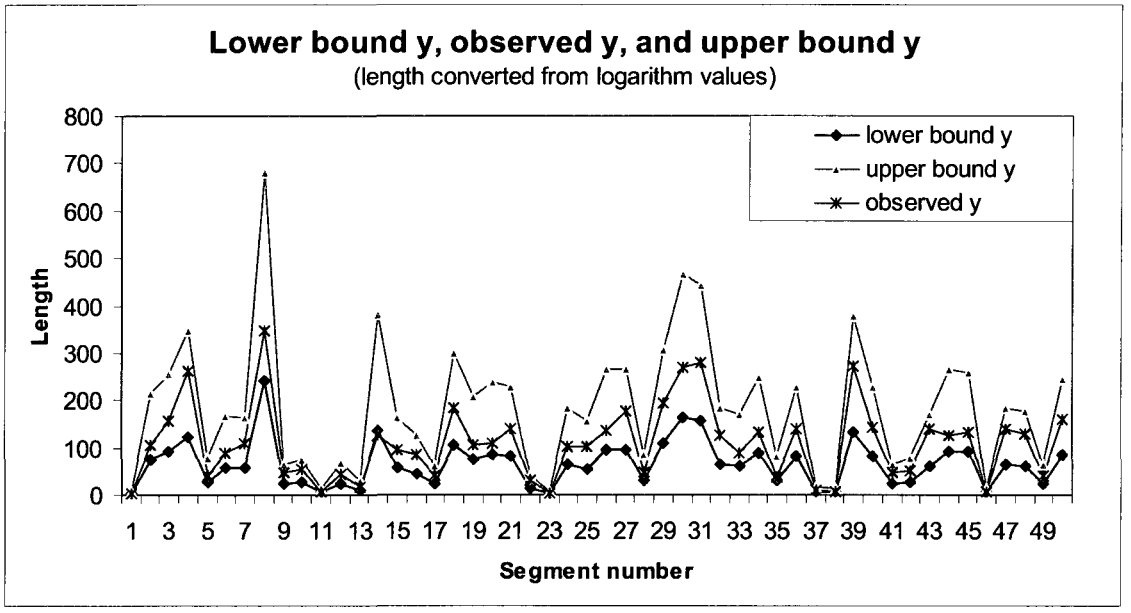


Figure 4.29. Comparison of lower bound y, observed y and upper bound y for the first 50 text segments of a randomly selected file pair.

4.2.2.3 Performance Evaluation

One frequent feedback about TextComp from people who are interested in the technical aspects of aligning bilingual texts is that an evaluation should be carried out using the naïve one-to-one match method as the baseline method. Some argue that possibly because government translations are clean and consistent, the majority of translations can already align very well by themselves. Knowing how well texts align by default will help understand whether TextComp algorithms are worthwhile for the gain in precision and recall.

Here, we set up two baseline methods to compare the quality of alignments for the three aligners, yielding baseline method 1 (B1), baseline method 2 (B2), and TextComp

(TC). We consider a text S and its translation T as two sets of text segments. $S = \{s_1, s_2, \ldots, s_m\}$ and $T=\{t_1, t_2, \ldots, t_n\}$. An alignment A between S and T can be defined as a subset of the Cartesian product $S \times T$ (Isabelle and Simard 1996). The triple (S, T, A) is a bitext. In baseline method 1 (B1), we assume that text segment number i in S aligns with text segment i in T. For example,

$S = \{s_1, s_2, s_3, s_4, s_5, s_6\}$

$T = \{t_1, t_2, t_3, t_4, t_5, t_6\}$

Baseline 1 alignment of the bitext is: $A = \{(s_1, t_1), (s_2, t_2), (s_3, t_3), (s_4, t_4), (s_5, t_5), (s_6, t_6)\}$.

41:31	"It creates opportunities for our engineering students to gain practical experience, which is indispensable as they look toward entering Canada's workforce".	Elle permet en effet à nos étudiants en génie d'acquérir une expérience pratique, qui est indispensable pour leur entrée sur le marché du travail au Canada».
42:32	Today's repayable investment totalled $5,97 million.	L'investissement remboursable annoncé aujourd'hui s'élève à 5,97 millions de dollars et est effectué dans le cadre de l'Initiative stratégique pour l'aérospatiale et la défense (ISAD), qui soutient les projets stratégiques de recherche industrielle et de développement préconcurrentiel dans les industries de l'aérospatiale, de la défense, de l'espace et de la sécurité.
43:32	This investment is being made through the Strategic Aerospace and Defence Initiative (SADI), which supports strategic industrial research and pre-competitive development projects in the aerospace, defence, space and security industries.	L'ISAD est gérée par l'Office des technologies industrielles, un organisme de service spécial d'Industrie Canada ayant pour mandat de promouvoir la R-D de pointe des industries canadiennes.
44:33	SADI is managed by the Industrial Technologies Office, a special operating agency of Industry Canada with a mandate to advance leading-edge R&D by Canadian industries.	- 30 -
45:34	- 30 -	Renseignements (médias seulement) :
46:35	For further information (media only), please contact:	Pema Lhalungpa

Figure 4.30. Baseline method 1 evaluation: correct = 1, incorrect = 5

38:31	"It creates opportunities for our engineering students to gain practical experience, which is indispensable as they look toward entering Canada's workforce".	Elle permet en effet à nos étudiants en génie d'acquérir une expérience pratique, qui est indispensable pour leur entrée sur le marché du travail au Canada».
39:32	Today's repayable investment totalled $5,97 million.	L'investissement remboursable annoncé aujourd'hui s'élève à 5,97 millions de dollars et est effectué dans le cadre de l'Initiative stratégique pour l'aérospatiale et la défense (ISAD), qui soutient les projets stratégiques de recherche industrielle et de développement préconcurrentiel dans les industries de l'aérospatiale, de la défense, de l'espace et de la sécurité.
40:32	This investment is being made through the Strategic Aerospace and Defence Initiative (SADI), which supports strategic industrial research and pre-competitive development projects in the aerospace, defence, space and security industries.	L'ISAD est gérée par l'Office des technologies industrielles, un organisme de service spécial d'Industrie Canada ayant pour mandat de promouvoir la R-D de pointe des industries canadiennes.
41:32	SADI is managed by the Industrial Technologies Office, a special operating agency of Industry Canada with a mandate to advance leading-edge R&D by Canadian industries.	
42:33	- 30 -	- 30 -
43:34	For further information (media only), please contact:	Renseignements (médias seulement) :
44:35	Pema Lhalungpa	Pema Lhalungpa

Figure 4.31. Baseline method 2 evaluation: correct = 4, incorrect = 3

38:31	"It creates opportunities for our engineering students to gain practical experience, which is indispensable as they look toward entering Canada's workforce."	Elle permet en effet à nos étudiants en génie d'acquérir une expérience pratique, qui est indispensable pour leur entrée sur le marché du travail au Canada».
39:32	Today's repayable investment totalled $5.97 million. This investment is being made through the Strategic Aerospace and Defence Initiative (SADI), which supports strategic industrial research and pre-competitive development projects in the aerospace, defence, space and security industries.	L'investissement remboursable annoncé aujourd'hui s'élève à 5,97 millions de dollars et est effe ctué dans le cadre de l'Initiative stratégique pour l'aérospatiale et la défense (ISAD), qui soutient les projets stratégiques de recherche industrielle et de développement préconcurrentiel dans les industries de l'aérospatiale, de la défense, de l'espace et de la sécurité.
40:32	SADI is managed by the Industrial Technologies Office, a special operating agency of Industry Canada with a mandate to advance leading-edge R&D by Canadian industries.	L'ISAD est gérée par l'Office des technologies industrielles, un organisme de service spécial d'Industrie Canada ayant pour mandat de promouvoir la R-D de pointe des industries canadiennes.
41:33	- 30 -	- 30 -
42:34	For further information (media only), please contact:	Renseignements (médias seulement) :
43:35	Pema Lhalungpa	Pema Lhalungpa

Figure 4.32. TextComp evaluation: correct = 6, incorrect = 0

In baseline method 2, we assume that text S and text T are already aligned at the paragraph level by default. We consider that text segment j in S aligns with text segment j in T in the same paragraph. We evaluate the baseline algorithms relative to the TextComp alignment algorithm. Figures 4.30, 4.31, and 4.32 show three validation views for the same section of a file randomly chosen for evaluation.

As can be seen from the comparison of the three figures, our evaluation criteria reject partial matches in text segment alignments. We check the imperfectly aligned text segments, or text segment alignments that are partially correct, and count those alignments as incorrect. We use the same metrics for evaluation as described in Section 4.1.6:

$$P = \frac{|A \cap M|}{|A|} \qquad R = \frac{|A \cap M|}{|M|}$$

where M is the set of segments in the golden reference collection and A is the set of aligned segments proposed by the specific method. Precision (P) equals correct alignments divided by proposed alignment, and recall (R) equals correct alignments divided by reference alignments. We also compute the F-measure which is defined as follows:

$$F = 2\frac{R \cdot P}{R + P}$$

Files for evaluation are randomly assembled from online pages in Canadian government websites in March 2009. They include officially published materials from

more than 20 government departments and agencies. We put the selected files for the evaluation metrics into 6 groups according to the types of texts we assembled: 1. questions and answers (QA), 2. news releases (NE), 3. mandate introductions (IN), 4. speeches (SP), 5. acts and regulations (AC), and 6. government department reports (RE). There are 5 files in each group and a total of 30 files were used for the evaluation of the quality of alignments generated by the three aligners. Table 4.10 shows the results of the evaluation. In this table, TC is for the TextComp system; B1 is for baseline method 1 and B2 is for baseline method 2 with paragraphs aligned by default.

	F-measure			Precision			Recall		
	TC	B2	B1	TC	B2	B1	TC	B2	B1
QA	0.965	0.906	0.302	0.965	0.885	0.302	0.965	0.928	0.302
NE	0.973	0.921	0.56	0.97	0.901	0.544	0.975	0.944	0.578
IN	0.968	0.884	0.423	0.96	0.856	0.412	0.977	0.917	0.436
SP	0.979	0.931	0.266	0.979	0.914	0.263	0.979	0.949	0.271
AC	0.909	0.86	0.519	0.926	0.861	0.513	0.893	0.86	0.525
RE	0.985	0.903	0.254	0.985	0.886	0.249	0.985	0.921	0.259
Average	0.963	0.902	0.388	0.964	0.884	0.381	0.962	0.92	0.395

Table 4.10. Performance evaluation of TextComp, Baseline 1 and Baseline 2

The average precision, recall, and *F*-measure for TextComp are 0.964, 0.962, and 0.963 respectively. This demonstrates that the alignment component of TextComp in BMIA has achieved very good results. We notice that for the AC category texts, precision and recall are low on the whole for TextComp and Baseline method 2. This is contrary to what we have expected of legal documents such as acts and regulations where translation correspondences abound. We will explain this in the discussion section in 4.2.5.

When we compare the performance of TextComp with the other two aligners, we can see the gain TextComp has over the other two baseline methods (see Figure 4.33). When we use the naïve alignment method (B1) to align text segments in the selected files, the *F*-measure value is less than 0.4. However, if we use the DP algorithm to align paragraphs, and then align the text segments within the paragraphs sequentially one by one (B2), all three measures improve significantly.

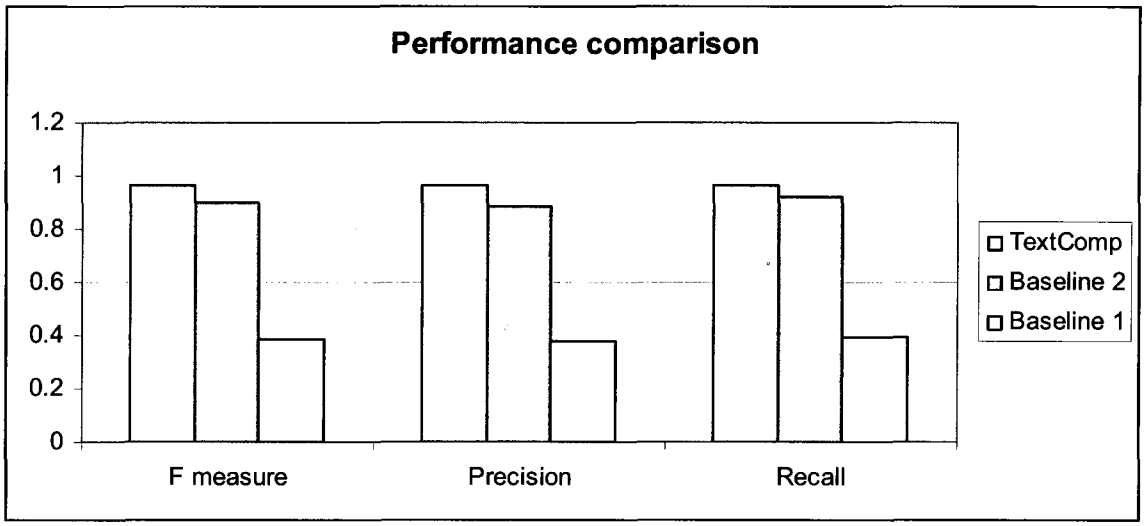


Figure 4.33. Performance comparison of TextComp, Baseline 1 and Baseline 2.

The difference between TextComp and B1 can be accounted for by the use of the DP algorithm for paragraph alignment and the linear regression forecasting model for text segment. Now, if we compare B2 with TextComp, we can check the improvement brought about by the linear regression forecasting model. Figures 4.34 and 4.35 show the results of the comparison.

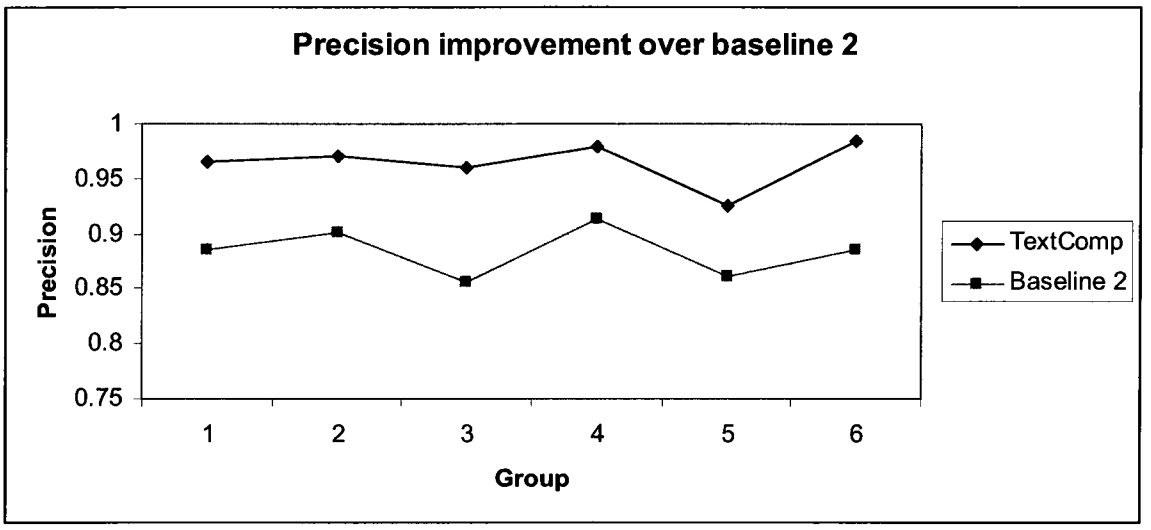


Figure 4.34. Precision improvement of TextComp over baseline method 2.

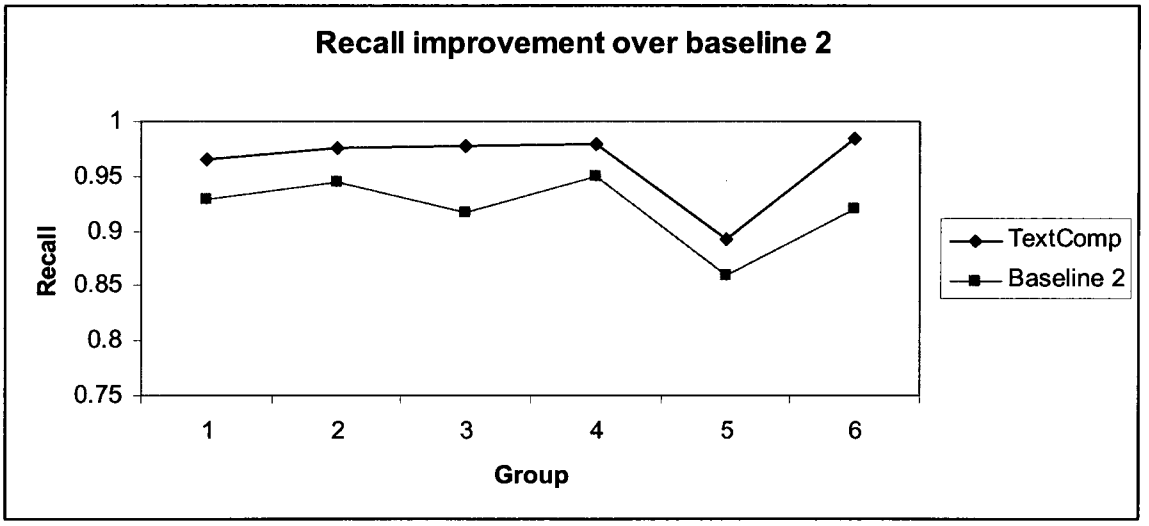


Figure 4.35. Recall improvement of TextComp over baseline method 2.

We can see the consistent gain of TextComp over baseline method 2. The improvement in precision is more significant (0.964 vs. 0.884) than that of recall, which is around .04. The improvement in alignment makes the aligned data much cleaner and more reliable. The performance gain is important if we want to use the aligned data for human

translation reference or as training data in machine translation or machine learning. In addition, one interesting finding in the performance evaluation is that for all the groups, baseline method 2 (B2) is doing very well (precision=.886 and recall=.921). This means that the default DP algorithm for aligning paragraphs is very effective and has a high rate of precision and recall. Secondly, it indicates that most sentences in officially published government materials on the internet can be matched on a one-to-one mapping basis. There are not so many alignment type variations within paragraphs. Even if an alignment type such as 2:1 or 1:3 is disrupting the alignment, the misalignment will not go far and can be rectified rather quickly if paragraphs are correctly aligned. When the average paragraph length is not long, there is no risk of massive misalignment as can be witnessed in B1. Thirdly, it can mean that for the alignment of government materials online, paragraph alignment is more than half the battle. When aligning texts where the average paragraph length is not so long and the 1:1 alignment type dominates, it is worth spending more computational and algorithmic resources on paragraph alignment. This at the same time gives good justification for the *forward-backward matching* algorithm (with 1:1 alignment type given first priority) TextComp adopts as a means of saving time when the system implements the linear regression forecasting model for alignment at the text segment level.

In addition, we aligned the same groups of files using the Moore sentence aligner (Moore 2002). Prior to using the Moore aligner, data files are formatted such that each line represents one text segment. As can be seen from Table 4.11, TextComp has a better F value (.963). Although the Moore method has a very good precision value (1.000), its recall is low because Moore's aligner only focuses on the 1:1 alignment type.

	Precision	Recall	F-measure
TextComp	0.964	0.962	0.963
Moore	1.000	0.882	0.937
Gale-Church with paragraphs hand-aligned	0.970	0.974	0.972
Gale-Church with naïve paragraph alignment	0.361	0.383	0.371

Table 4.11. Performance comparison of TextComp, Gale-Church and Moore

It is also useful to compare the performance of TextComp with that of the two-pass Gale-Church algorithm as adopted in SDTES. However, a fair comparison of the two methods is not very straightforward to implement. There are certain issues that have to be addressed to make the two alignment systems comparable. For example, in SDTES, the two-pass Gale and Church procedure works only with published web-based materials which have HTML tags in the input files, because it needs some specified main HTML tags (see Section 4.2.1) to help align the paragraphs in the first pass. On the other hand, TextComp is designed to deal with noisier data sets and a wider range of data formats. Texts to be compared can be web pages or raw text files without HTML tags. No matter whether the texts are from web-based materials or from text editors, when they are cut and pasted into the comparison box, they become HTML tag free. If we use the same groups of input files for evaluation, we would not be able to run the two-pass Gale-Church algorithm in the same manner as in SDTES because there are no HTML elements in the input texts. This makes direct comparison of TextComp and the SDTES two-pass Gale-Church algorithm difficult.

In evaluating and comparing the alignment results, we considered automating (with no hand alignment) the first pass in the two-pass Gale-Church algorithm without using HTML tags, aligning the paragraphs on a naive 1:1 match basis. For files that do not have the same number of paragraphs, the system adds empty lines at the end of the file that has

fewer paragraphs. We notice that in this configuration, the Gale and Church algorithm works rather poorly (see Gale-Church with naive paragraph alignment in Table 4.11) because of the automatic naive paragraph alignment. This shows that if the automated first pass does not employ some kind of anchors such as structural, lexical or statistical information, the system can easily fail on paragraph alignment and therefore can increase the error rate in sentence alignment. Applying the Gale and Church sentence alignment algorithm without a robust and reliable first pass for paragraph alignment can render poor results.

In another test, instead of employing the naive 1:1 match approach for paragraph alignment, we manually aligned the paragraphs and then proceed to sentence alignment using the Gale-Church length model. This seems to be a more fair use of the Gale and Church algorithm because the Gale-Church aligner is mostly a sentence level aligner that is based on matched paragraphs. From Table 4.11 we can see that the Gale and Church algorithm performs very well (F=0.972; see Gale-Church with hand-aligned paragraphs) because of the obvious initial boost due to manually aligned paragraphs. However, Gale-Church with initial manual alignment of paragraphs is not straightforwardly comparable to TextComp, because TextComp is fully automatic and the Gale-Church algorithm used in the evaluation is not. If we intend to replace TextComp with the Gale-Church algorithm, we will at least have to find a better way of automating the first pass.

The alignment algorithm we designed for translation error detection has an obvious advantage over the standard longest common subsequence algorithm. The UNIX utility *diff* is an implementation of the standard LCS algorithm. It can do some of the things that the Gale-Church model cannot do. For example, for the alignment of noisy texts, it may

have more 1:0 and 0:1 alignment patterns, but the ratio of wrongly aligned text segments will be low. However, there are two major problems in using the standard LCS model for bilingual text alignment. First, it does not take into account the length factor. This means, for example, that a paragraph of 100 words in English can be aligned with a paragraph of 1000 words in French. It has been observed in the computational linguistics community that directly applying the utility *diff* in bilingual text alignment can cause serious problems because the utility *diff* has no preference for aligning chunks of similar lengths (Resnik 1999). The second problem is that *diff* can usually do alignment of types 1:1, 1:0 and 0:1, but not of types such as 1:2, 2:1 and 2:2. Because of this limitation, it can either miss a good alignment or fail to identify true translations. In our model, which is based on the modified dynamic programming algorithm and the linear regression forecasting algorithm, we avoid shortcomings of massively misaligning noisier texts in the Gale-Church model. At the same time, our model can do various alignment patterns including 3:1, 1:3, 2:2, 2:1, 1:1, 1:0 and 0:1 (see Figure 4.36), some of which many other aligners cannot accomplish.

378:227	The applicant is given one final chance to introduce any new oral evidence to address any matters raised in the intervenors' evidence or cross-examination.	Le demandeur a ensuite une dernière occasion de présenter une nouvelle preuve de vive voix afin d'aborder les questions soulevées pendant la présentation des preuves d'intervenant et les contre-interrogatoires.
379:228	The panel issues the draft conditions for comment. (may occur before or during the hearing).	Le comité publie l'ébauche des conditions aux fins de commentaires (avant ou pendant l'audience).
380:229	Parties summarize their position in the final argument, beginning with the applicant followed by the intervenors and government participants.	Les parties résument leur point de vue au cours de leur plaidoirie finale. Le demandeur est le premier à prendre la parole. Il est suivi des intervenants et des participants du gouvernement.
381:230	The applicant is allowed a final reply argument.	Le demandeur a le droit de répliquer aux plaidoiries finales.

Figure 4.36. More difficult alignment patterns such as 2:1 and 1:3 in TextComp

Although the aligning algorithm in TextComp can outperform the aligning algorithm we used for SDTES, it does not mean that we can do away with the Gale-Church model in our study. The Gale and Church algorithm is important in that it is a convenient means and a fitting model for extracting translations from published texts for bilingual corpus building and corpus-based alignment data analysis. Without the statistical outcome of the analysis of the Gale-Church algorithm, we could not have built the statistical model for the alignment algorithm in TextComp. In turn, TextComp algorithms can be used to improve the alignment model in SDTES to make the process of translation extraction more accurate and more efficient.

4.2.3 Translation Discrepancy Detection

To understand the essential principles underlying the process of translation discrepancy detection, it is necessary to understand the cognitive process of translation quality assessment. There are at least three types of knowledge used in translation quality assessment:

1. Linguistic knowledge. The first stage in human translation checking is complete comprehension of the source language text. When you do not know what message is conveyed in the original text, you are not sure what type of translation to expect in the target language. On the other hand, linguistic knowledge makes it possible to judge if the translation reads well in the target language, or whether there are no blunders and mistakes at grammatical, syntactic and morphological levels.

2. Background knowledge. This includes knowledge of the subject field, especially the commonly used terminology in the field. Also, it includes knowledge of socio-cultural aspects, that is, of the customs and conventions of the source and target language cultures. With background knowledge, we can see if the translation is acceptable in a specific cultural and technical setting.
3. Professional knowledge in translation analysis. This includes a sense of translation correspondence and equivalence (whether the translations match in tone and styles). In addition, it helps ensure what you read in the target language is what you intend to convey in the source language.

Given the complexity of translation discrepancy detection, it would be absurd to claim that a machine could generate exactly the same assessment as that of a human evaluator. However, it is clear that even humans can make oversights in checking translation products and the machine can be designed to reduce the amount of work and the number of judgement errors. We can break up the process in two stages. The first stage is a quick but detailed formal comparison of the texts in both languages, and a report of what it can gather as potential discrepancies and errors in translation. At this stage, main flaws in translation will come to surface such as massive misinterpretations, deletions and insertions. The second stage is the decision-making process. The human evaluator can choose between merely making minor adjustments or implementing radical changes and thorough revisions on the basis to the report. What TextComp aims at is performing the first stage of this process in an automatic way, so that the human evaluator can proceed directly to the second which gives more meticulous attention to areas of texts with potential translation discrepancies.

The performance of the translation discrepancy detection component hinges on the quality of the parallel alignment. It is implemented at the text segment level and checks the translations segment by segment to see if they are a perfectly legitimate translation pair or if there are major translation divergences in the pair.

4.2.3.1 Feature Selection for Bilingual Text Comparison

In comparing bilingual texts, TextComp focuses on checking some key properties in translation equivalence studies such as verifying if all the feature information is included; nothing is added, omitted or different (Larson 1984). In a sense, the agent is checking the formal and structural features of the texts that relate to the paradigm of faithfulness in traditional translation studies. These features usually reveal the corresponding relations in translation, and can thus be indicative of problems in translation. Main selected features in TextComp include: alignment type, length constraint, numbers, punctuations, symbols and cognate words.

Once the report has been submitted, the work of the mediator is complete.	Une fois son rapport soumis, le travail du médiateur est terminé.
The responsible authority must take the mediator's report into consideration before determining the significance of the environmental effects of the project.	L'autorité responsable doit tenir compte du rapport du médiateur avant de prendre quelque décision que ce soit relativement au projet.
	Elle doit aussi répondre au rapport, avec l'approbation du Cabinet.
How can I get involved in a mediation?	Comment puis-je m'impliquer dans une médiation?

Figure 4.37. Alignment type 0:1 indicating translation insertion

Alignment type refers to the alignment patterns which commonly include 1:0 or 0:1, 1:1, 2:1 or 1:2, and 2:2. Translation pairs with the types of 1:0 and 0:1 are marked as discrepancy pairs because these alignment types indicate that one side of the translation is missing, or that there exists potential insertions or deletions (see Figure 4.37). The other alignment types are acceptable as far as they can pass the other validation constraints.

The length constraint here is geared to identify those translation pairs that cannot pass the maximal allowable length ratio criterion. When the length ratio is out of proportion for the source language text and the target language text, the text segment pair is highlighted to indicate that there is possible translation discrepancy in the pair.

In most cases, numbers are literally translated as numbers. As a result, if there are number discrepancies in the translation, TextComp labels them as potential sources of translation problems.

It is apparent that in published government texts, most punctuation in translation strings matches up consistently. Some punctuations match more nicely than others, but in general they can give a convincing indication about the correspondence between the two texts in the translation pair. Some divergences in the use of punctuation can be easily identified and considered as indicators of problematic or erroneous translations. For example, if one side contains question marks, and they are missing on the other side, it is worth a warning message so that a closer examination can be conducted by the human evaluator. In TextComp, the punctuation correspondence constraint is not measured as an integer value, but as a Boolean value indicating if certain punctuation is present or not.

Some symbols can be anchors in establishing associative links between the two segments in a translation pair. For example, if in a table cell there is a negative symbol

"–" preceding a number on one side, and it is not present on the other side, it is very likely that something is wrong with the translation.

Cognate words checking in the system is normally across segments. If a cognate word is not present in the translation pair, but is present in one of its immediate neighbouring segments, very likely the translation pair only represents partial translation, or it can be a misaligned translation pair.

4.2.3.2 Identification of Translation Discrepancies

As a computational model of human behavior, TextComp in BMIA is supposed to have the capacity of locating the following types of problems that human checkers usually try to capture in checking translations:

- Insertion or deletion in translation. This is where forgotten, missing, or empty translations are detected. In some cases the target texts and source texts have a comparatively imbalanced length ratio, a ratio that indicates the texts in two languages are not parallel or are not mutual translations (see Figure 4.38).
- Flawed correspondences. In some cases, although the texts align well by the length criteria, they may not match in contents. It can be due to swapping of sentences or paragraphs without valid reasons.
- Inconsistency of translation. The problem can reveal itself in the use of punctuations, the capitalization of initial words, in abbreviations and other types of formal features in texts. It can also be the inconsistent use of cognate words.

- Mismatches in invariable elements of translation. For example, the numbers or symbols do not match.

The Canadian Environmental Assessment Act is the legal basis for the federal environmental assessment process.	La Loi canadienne sur l'évaluation environnementale constitue le fondement juridique du processus fédéral d'évaluation environnementale.
The Act sets out the responsibilities and procedures for carrying out the environmental assessments of projects which involve federal government decision making. A number of regulations have been established under the Act. Some are essential to the functioning of the Act. Others apply in special circumstances.	La Loi définit les responsabilités et les procédures pour mettre en ouvre les évaluations environnementales soumises au pouvoir de décision du gouvernement fédéral.
The four essential regulations are the:	Les quatre règlements indispensables sont :
Inclusion List Regulations	le Règlement sur la liste d'inclusion;

Figure 4.38. Detecting translation discrepancies within the aligned text segment by length criteria

In the problem identification and assessment schemata, the agent uses the text of one language as a point of departure and detects how far the text of the other language is deviant as far as the features comparison is concerned. TextComp detects both major and minor rules infringed. Major rule errors refer to failure to observe the length criteria in translation, and problems in translating numbers. Minor rules can be rules relating to the use of punctuation. It is good for punctuation to match, but if it does not, the penalty cost will not be that great. TextComp tries to diagnose and scale the severity of the problems and mark up the problem regions of text with hints as to what type of problem it is. The system will not attempt to detect mistakes arising from incorrect or incomplete

understanding of the source text, or inappropriate choice of language register. These tasks require more subjective judgment on the part of the individual translator. For these types of errors, the gravity of error levels can vary in the eyes of different reviewers.

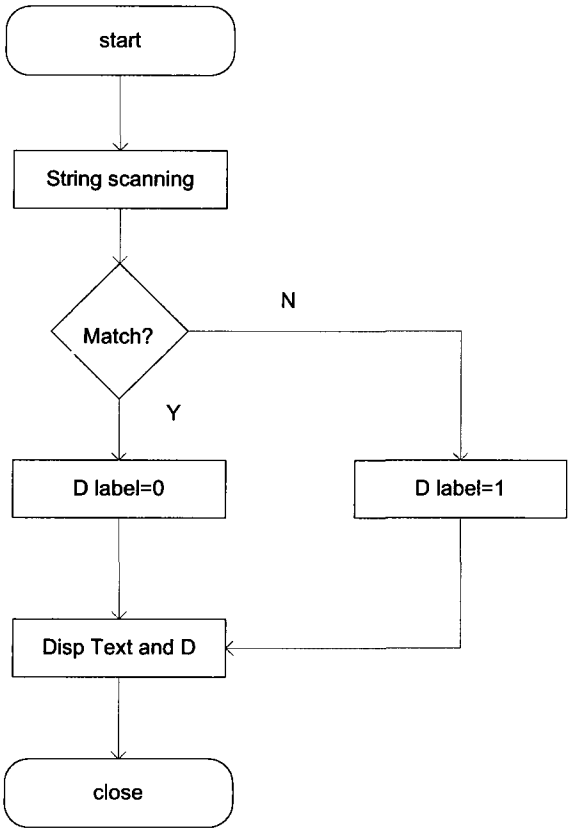


Figure 4.39. Shared subroutine in TextComp for translation discrepancy detection

There is a shared subroutine (Figure 4.39) in comparing all the main features in the translation texts. In this common subroutine, the software agent scans the pair of text segments and compares the feature properties in the strings. If the features match, TextComp assigns a zero to the D (for discrepancy) label, which means that the system

will print the string as it is. If the feature parameters do not correspond, the D label is equal to 1. This means that the system will mark the specific features in the context of the string that do not match, and highlight the divergence according to the discrepancy category. Then the system prints out the results of the computation and closes the subroutine.

TextComp employs three modes for viewing the results (see also comparison 1 in Figure 4.18). There are links in each display format so that the user can easily switch from one view to another. The first one is text segments view (see Figures 4.40). This is also the default TextComp view mode. This mode of viewing enables the user to focus on the text segments, and to check the translations segment by segment to verify if there is anything wrong with the translation in light of the information provided in the context.

31:31	Proactive Disclosure	Divulgation proactive
32:32	CONTENT|CONTENUWho We Are and What We Do What Are Consular Services?	CONTENT|CONTENUQui sommes-nous et quel est notre rôleEn quoi consistent les services consulaires?
33:33	Information Before You Leave	Renseignez-vous avant de partir
34:34	Assistance Once You Are Abroad: What We Can & Cannot Do for You	Obtenir de l'aide à l'étranger
35:35	Emergency Assistance	Aide en cas d'urgence
37:37	The Consular Affairs Bureau is committed to helping Canadians prepare for foreign travel and to providing you with a variety of services once you are abroad.	La Direction générale des affaires consulaires d'Affaires étrangères et Commerce international Canada a à coeur d'aider les Canadiens à préparer leurs voyages à l'étranger et de leur offrir tout un éventail de services pendant leur séjour à l'étranger.
38:37	We believe that preparation is the key to successful travel.	Selon nous, l'ingrédient indispensable d'un voyage réussi est la préparation.

Figure 4.40. Text segment view for translation discrepancy detection. See also *Segment* in Figure 4.18

The second view mode gives a larger context area for the discrepancies found: the texts are aligned paragraph by paragraph. However, the discrepancy region is marked

only at the text segment level so that the user can see the discrepancy area as it appears in a paragraph (Figure 4.41). It is an easier way to check the surrounding sentences to see if the highlighted discrepancy represents real problems in translation, or in translation alignment.

24	A Statement of Facts, signed by prosecutors, respective defense counsel for Mr. Cheeseman and Mr. Hennessey, along with agreement and signatures from Mr. Hennessey and Mr. Cheeseman themselves, was then read into evidence in Court of Queen's Bench. As the result of the facts presented the guilty pleas were accepted by the court. The "Agreed Statement of Fact", testimony and related evidence provided at the Preliminary Inquiry, and previously undisclosed evidence and information, has now been made public to all Canadians.	Un énoncé conjoint des faits signé par les procureurs, les avocats de la défense et les deux accusés a alors été présenté en preuve devant la Cour du Banc de la Reine. À la suite de cet exposé des faits, le tribunal a accepté les plaidoyers des accusés. L'énoncé conjoint des faits, les témoignages et les éléments de preuve présentés lors de l'enquête préliminaire, ainsi que d'autres renseignements jusqu'alors non divulgués, sont donc maintenant accessibles à tous les Canadiens.
25	The twenty-eight month criminal investigation into the murder of four young police officers on March 3rd, 2005, has been the subject of extraordinary speculation, uninformed innuendo and unfounded accusation, that has largely focused on the RCMP -- but has caused immeasurable pain to the families and loved ones of the four murdered RCMP Officers. This speculation, innuendo and accusation can now be judged against the factual evidence. Facts that are only now, available to the public in accordance with the law - -the law that we, the RCMP, have been and continue to be duty bound to follow. These facts provide for the context and scope of the four homicides and the extent of the criminal investigation that was to follow.	L'enquête criminelle de 28 mois sur le meurtre de ces quatre jeunes policiers a fait l'objet de conjectures débridées, d'allusions douteuses et d'accusations sans fondement qui ont surtout visé la GRC, mais qui ont aussi causé une douleur incommensurable aux familles et aux proches des victimes. Ces conjectures, allusions et accusations peuvent maintenant être examinées à la lumière des faits tout juste rendus publics, comme le veut la loi, celle que nous, à la GRC, avons depuis toujours le devoir de respecter. Ces faits établissent le contexte et l'ampleur des quatre homicides et l'étendue de l'enquête criminelle subséquente.

Figure 4.41. Paragraph level view for translation discrepancy detection. See also *Paragraph* in Figure 4.18

The third view mode lists all the translation pairs where translation discrepancies are found. This is a summary list for potential translation problems (Figure 4.42). By examining the items in this list we can have a very quick idea of how many potential spots we have to check in the translation and what kind of discrepancy problems they represent.

221:221	(Motions deemed adopted, bill read the first time and printed)	Adoption des motions; première lecture et impression du projet de loi
367:356	The simple fact that the consumption of alcohol during pregnancy is the one and only cause, FAS is 100% preventable by abstaining from the consumption of alcohol during pregnancy.	Du seul fait que la consommation d'alcool pendant la grossesse en est la seule cause, le SAF peut être évité complètement si la future mère s'abstient de consommer de l'alcool pendant sa grossesse.
2018:1230	The Conservatives tried to hide the truth, but now we know that detainees were tortured.	Ce sont seulement eux qui pensent qu'il n'y a pas de torture; tout le monde sait qu'il y en a.
2019:1230	They are the only ones who think that nobody was tortured.	Les conservateurs ont tenté de cacher la vérité.
2020:1230	The whole world knows people were.	Maintenant, on sait qu'il y a de la torture.
2617:1681	I am going to make this very short, but I am going to draw it into the question.	

Figure 4.42. Summary list view for translation discrepancy detection. See also *Summary* under *Comparison 1* in Figure 4.18

62:48	Despite this slowdown, gasoline continued to be the primary upward contributor for transportation costs.	Malgré ce ralentissement, l'essence a continué d'être le principal facteur déterminant de la hausse des coûts de transport.
63:49		La hausse des prix de l'assurance des véhicules automobiles et du transport aérien a également contribué à l'augmentation des coûts de transport en octobre.
64:50	Chart 3	Graphique 3
65:51	Evolution of gasoline prices	Évolution des prix de l'essence
66:52	Increasing prices for passenger vehicle insurance and air transportation also contributed to the rise in transportation costs in October.	
67:53	A 9.0% decline in prices to purchase and lease passenger vehicles was the most significant downward contributor for transportation costs.	Une baisse de 9,0 % des prix d'achat et de location à bail de véhicules automobiles a été le principal facteur exerçant des pressions à la baisse sur les coûts de transport.

Figure 4.43. Swapped paragraphs detected in TextComp.

All in all, the bilingual text comparison system is capable of identifying formal correspondence problems in alignment and in translation as sentences are being aligned. The software agent can mark up the potentially problematic text regions and indicate

what types of problems are identified. Although some errors identified can be false errors in translation, most of the problem regions identified deserve a quality check revisit (see Figure 4.43). The system is very quick in spotting problems that stem from inconsistent translations or inappropriate translation alignment.

4.2.4 Translation Correspondence Profiling

The question of faithfulness vs. transparency in translation studies has sometimes been described as "literal translation" and "free translation" (House 1981). It has also been discussed in terms of formal equivalence and dynamic equivalence (Nida 1964; Wilss 1982). They represent two opposite poles in translation studies. In reality, style distinctions in translation are not always polar distinctions: there are many grades and levels between the two extremes. Good translation often entails a judicious blending of faithfulness and transparency, and a translation can be both dynamically and formally equivalent to the original text at the same time. On the one hand it is simply wrong to think that there is always a word for word relation between languages and translation is only a straightforward and mechanical word mapping process. On the other hand we should not worry too much about the use of idioms, metaphors, individual linguistic hybrids, and other special expressions in translation. What Translation Correspondence Profiling (TCPro) aims to achieve is to identify whatever correspondences it can find, and "make explicit all the correspondences between *S* and *T*" (Isabelle *et al.* 1993). We believe that by mapping formal equivalences we can identify instances of dynamic equivalences. At the same time, when profiling different levels of translation

correspondence, we can capture text areas that are fraught with discrepancies and possible errors in translation.

To this end, the TCPro mechanism is not intended to replace human evaluators, but significantly reduce the intensity of their assessment work. The software agent scans translations, assembles evidence of correspondent constituents in translation, and gains ideas about constituent structures of the translated segments. It is not giving error counts as a negative assessment, but reporting the ratio of correspondence of translation constituents as a positive assessment factor. It is felt that the quantification in translation correspondence profiling lends objectivity to the assessment of the quality of translations.

Generally speaking, there can be two primary cases where the lack of translation correspondence does not mean errors or problems in translation. The first is that the text units are actually mutual translations, but TextComp has not been able to recognize the correspondence relationship. The second is that although the target language texts do not correspond to the source languages texts formally, they are perfectly natural or good translations in a freer translation style, or even creative ways of rendering the underlying meaning. The second case more or less relates to what Dorr (1994) described as "machine translation divergences". However, in a Canadian government translation publication setting, the rule of thumb is that translation non-correspondences are found mostly where there is a problem using corresponding translation units. Translators generally exercise due flexibility in seeking equivalents --- paraphrase the message where it is necessary and preserve the order of the original text where possible. A casual look at the web-based government publications in Canada reveals that there is a noticeable proliferation of translation correspondences. For bilingual materials published or to be

published by government departments and agencies in Canada, if the TCPro scores are high, chances of serious translation errors such as omissions, insertions, massive misinterpretations are relatively small. On the other hand, if the TCPro scores are abnormally low, it may well indicate that the texts compared are either not mutual translations of each other, or the translations have to be carefully examined for serious errors, for discrepancies, and for translation style prior to official publication.

4.2.4.1 Word Correspondence Identification

At the core of the translation correspondence profiling is the mapping of word correspondences in translation. For TCPro in TextComp, the bilingual text mapping and comparison are conducted at a more fine-grained level: checking words and other basic translation units such as numbers and symbols in the aligned text segment. The purpose is to see what proportion of the text has the structural correspondence. The parameters used for the formal correspondence mapping include words, numbers and symbols extracted from the translation text. Individual neighbouring word correspondences are then assimilated into short translation constituents where possible. Instead of highlighting the lack of correspondence in translation as in bitext comparison described in 4.2.3, we mark the translation constituents that have evidence of formal correspondence.

Numbers and symbols matching is the first step in TCPro. For example, if a number in English matches a number in French, the number pair is marked as a corresponding unit.

Word correspondence mapping in TCPro is divided into several stages. *Frequency-list based word matching in limited text range* is for those words that appear on the top ranking positions of the most frequent word lists for English and French in the StatCan

Daily Corpus. The selected words from the lists should match in lexical meaning (e.g. *avec* and *with*) and approximately in ranking positions. This is a short list of words covering different grammatical or syntactical categories such as prepositions, time adverbials, coordinating conjunctions, modals, pronouns, negative constructions, content words and others. This mapping process is skipped if any text segment in the aligned translation pair is longer than 70 characters. *Customized match list overwriting* is a convenient appendix to the matching list for special terminologies. This customized match list can be zero in length. *Cognate word matching* identifies and maps the cognates on the fly. In the word mapping process, we take the maximum match first. For example, the French word "Canada" will match the English word "Canada" before it is matched to the English word "Canadian". The system has rules for proximity match relaxation. The relaxation is generally in proportion with the text window range. The narrower the range the more relaxed the rules are. For example for a text window of only 20 characters in length, that is, if a word pair falls in a range of *i-20, i, i+20*, and if the pair has only the first three characters matching, it is considered a legitimate cognate pair.

Statistical word correspondence identification refers to the process of using a statistical metric to help map word correspondences for the remaining pool of words (stop words are excluded) after the above mentioned word mapping operations. In most other studies, this could have included every word in the paired translation text segments. For many years, word association similarity and word correspondence identification have been hot areas of research. Many measures have been proposed, discussed and experimented with for bi-grams in the same language (Inkpen and Hirst 2002; Inkpen and Hirst 2006) and for translation word pairs in two languages (Church and Gale 1991;

Dunning 1993; Smadja *et al.* 1996; Martin *et al.* 2003; Och and Ney 2003). Although the popular GIZA++ package (Al-Onaizan *et al.* 1999; Och and Ney 2003) includes a training program for IBM models and the HMM model for word alignment and is freely available, we did not choose it for further experiments because our initial test revealed that the algorithms can be too slow for our purpose in a web-based application. In our study, we opted for a few of the statistical measures for experimentation so that we can select a best fitting metric for TCPro in TextComp.

For the aligned text segments, let x and y be words in the English text and the French text respectively. We define the following frequencies:

freq(x,y) = counts of text segments where x and y co-occur;

freq(x) = frequency of x;

freq(y) = frequency of y;

N = total number of aligned text segments.

	y	¬ y	total
x	a	b	*freq(x)*
¬ x	c	d	
total	*freq(y)*		N

Table 4.12. A co-occurrence contingency table

In the contingency table (Table 4.12),

a = freq(x, y)

b = freq(x) – a

c = freq(y) – a

d = N – a – b – c

Based on the frequency counts in the contingency table, Gale and Church (1991) tested the association strength for each candidate word translation pair using phi-square coefficient:

$$\Phi^2 = \frac{(ad - bc)^2}{(a+b)(a+c)(b+d)(c+d)}$$

In their earlier attempts to find word pairs which are most probably alignable on the basis of similar distribution of text sectors, Kay and Röscheisen (1993) used the Dice coefficient to quantify the probability. Dice coefficient is in fact the harmonic mean of the two conditional probabilities. It produces values between 0 and 1, where 1 refers to the strongest correspondence.

$$Dice(x, y) = \frac{2a}{(a+b)(a+c)}$$

Point-wise mutual information (PMI) is a statistical association measure derived from information theory. It "measures the reduction of uncertainty about the occurrence of one word when we are told about the occurrence of the other" (Manning and Schütze 1999). Martin *et al.* (2003) found that when using the lower bound as a measure of association,

the PMI measure (Pedersen and Varma 2002) can outperform Chi-squared on the data they were experimenting. They also found that PMI can make the correct judgment on Gale and Church's well-known *near miss* problem such as "chamber-commune" (Gale and Church 1991).

$$PMI(x, y) = \log_2 \frac{a}{(a+b)(a+c)}$$

Gao (1997) noted that other similarity measures such as the Jaccard Coefficient and the Cosine Coefficient that are widely used in information retrieval could also be applied as measures to calculate the similarity of the distributions of a proposed translation word pair. He tested various measures with parallel English-Chinese texts, and found the Jaccard Coefficient the best measure.

$$Jaccard = \frac{c}{a+b-c}$$

Inkpen and Hirst (2002) used some popular hypothesis testing methods as statistical association measures to test if two words of the same language co-occur significantly more often than it would be expected if they would co-occur purely by chance. Pearson's Chi-squared (χ^2) and Log Likelihood Ratio (LL) are two of the metrics they experimented with for bi-grams in the same language. The two metrics are used to test whether we can reject the null hypothesis that the two words occurred together only by chance. "The higher the score, the less evidence there is in favor of concluding that the

words are independent" (Pedersen 2009). They found that Log Likelihood Ratio (Dunning 1993) is more appropriate for sparse data than the Chi-squared measure.

$$\chi^2 = \sum_{i,j} \frac{(n_{ij} - m_{ij})^2}{m_{ij}}$$

$$LL = 2\sum_{i,j} \frac{\log_2 n^2{}_{ij}}{m_{ij}}$$

Where $m_{ij} = \frac{n_{i+} n_{+j}}{n_{++}}$

In this study of translation word correspondence, we wrote programs to test all the six statistical methods with the weighted counts of word pairs. For χ^2 and *LL* testing, we used the same statistical information as specified in the 2×2 contingency table for the phi-coefficient except the counts for *N*. *N* in the two metrics is estimated as:

$$N = \frac{\text{No. of types in English} \bullet \text{No. of types in French}}{\text{No. of aligned text segments}}$$

Our tests show that for the type of bilingual materials TextComp uses, statistical measures that arise from significance testing such as Chi-Square and Log Likelihood Ratio perform better than the others. Log Likelihood Ratio gives the best results. Although some of the other methods are straightforward and simple to calculate, they sometimes yield results that do not agree with our intuition. For example in *PMI* result lists, sometimes infrequent pairs which are usually noisy pairs simply pop to the front of

the list, while real common translation pairs are left behind. When we examine the word count lists that have been collected from aligned pairs in TextComp, we found few words with very high frequencies and many words with very low frequencies. Typically for the very low frequency terms, statistical measures that are based on significance testing are not so reliable. However, since Log Likelihood Ratio is used in TextComp in conjunction with a frequency threshold, this basically overcomes the shortcoming of Log Likelihood Ratio in being sensitive to the sparseness of data. Therefore it seems plausible that we directly adopt the *LL* score as a criterion in deciding if a pair of words can be considered as a legitimate candidate translation pair. In TextComp, we apply the following two filtering algorithms.

Filtering algorithm I:

Suppose for each aligned text segment a_i the mapping of each English word x to each French word y is the Cartesian Product $X \times Y$, $X \times Y = \{(x, y) \mid x \in X, y \in Y\}$, here is the filtering algorithm:

; filtering while building lists and hashes

LOOP1: for each pair (a_i) of aligned text segment

LOOP2: for each word (x_j) in the English part of a_i

next LOOP2 if the length of x_j<=3

next LOOP2 if x_j is a stop word

next LOOP2 if x_j has been processed for this line before

LOOP3: for each word (y_k) in the French part of a_i

next LOOP3 if the length of y_k<=3

next LOOP3 if y_k is a stop word

next LOOP3 if y_k has been processed for this line before

accumulate counts of freq(x_j), freq(y_k) and freq(x_j, y_k)

store i, x_j and y_k to different hashes

k++

next LOOP3

j++

next LOOP2

i++

LOOP1

; Frequency count filtering. Moore's threshold (Moore 2002) is 2.

; Martin's (Martin *et al.* 2003) is 3. Our tests show that 3 works better with our data sets.

LOOP4: for each word (x_j) in the English word count list

if (freq(x_j)<3){

delete from relevant lists and hashes

}

next LOOP4

LOOP5: for each word (y_k) in the French word count list

if (freq(y_k)<3){

delete from relevant lists and hashes

}

next LOOP5

LOOP6: for each word pair (x,y) in the word pair count list

if (freq(x,y)<3){

delete from relevant lists and hashes

next LOOP6

}

if (freq(y) >freq(x,y) * 3 or freq(x)>freq(x,y)*3){

delete from relevant lists and hashes

}

next LOOP6

Table 4.13 shows the translation fertility list for top ranking words in English for file RE3438. The list is sorted and ranked by co-occurrence frequency after the initial filtering. Words longer than 12 characters are truncated to 12. Thresholds for this list are f(x)>4, f(y)>=3, f(x,y)>=3. From this table, we observe that, after the steps of pruning, the filtered list is very close to the list that we can intuitively judge as legitimate

translation pairs: in 74% of the cases, the top-ranking French word matches the English word. Performance elevates as a result, and noise is greatly reduced. There are not so many instances of "indirect associations" (Melamed 2000) in the list after this initial filtering process.

w	f(x,y)	f(y)	w	f(x,y)	f(y)	w	f(x,y)	f(y)
government (f(x)=28)			**audit (f(x)=8)**			**such (f(x)=6)**		
gouvernement	26	27	vérification	7	9	comme	3	5
renseignemen	5	7	**found (f(x)=8)**			**some (f(x)=6)**		
gestion	5	10	constaté	8	9	gouvernement	5	27
fonds	5	6	ses	3	13	certains	5	6
ses	4	13	**year (f(x)=7)**			**areas (f(x)=6)**		
fiducies	4	4	année	3	3	domaines	5	5
compte	4	4	**health (f(x)=7)**			gouvernement	3	27
autres	4	5	santé	7	7	**food (f(x)=6)**		
rapport	4	6	cet	3	4	aliments	4	4
certains	4	6	population	3	6	établissemen	3	4
comme	3	5	**also (f(x)=7)**			**expected (f(x)=5)**		
population	3	6	aussi	4	6	fiducie	3	3
vie	3	5	ses	3	13	fonds	3	6
protection	3	5	gouvernement	3	27	gouvernement	3	27
travaux	3	4	**information (f(x)=6)**			attendus	3	3
fédérales	3	4	renseignemen	6	7	**manages (f(x)=5)**		
présent	3	4	institutions	4	4	gestion	4	10
uvre	3	4	vie	3	5	**csc (f(x)=5)**		
aussi	3	6	protection	3	5	service	4	4
matière	3	7	**management (f(x)=6)**			établissemen	3	4
domaines	3	5	gestion	5	10	**all (f(x)=5)**		
son	3	5	new 6			tous	5	5
reporting (f(x)=9)			ses	4	13	**small (f(x)=5)**		
rapport	6	6	**sustainable (f(x)=6)**			petites	3	3
rapports	4	5	durable	6	6	gouvernement	3	27
gouvernement	4	27	**needs (f(x)=6)**					
population	3	6	ses	3	13			
santé	3	7	besoins	3	3			
fonds	3	6	doit	3	3			

Table 4.13. Translation fertility list for top ranking words in English for file RE3438

Filtering algorithm II:

Let us define $C(x,y)$ as the number of aligned segments G for two words x and y in two languages ($x \in G_i^s$ and $y \in G_i^t$). For multiple occurrences within an aligned translation text segment, we count them only as one.

x	y	f(x)	f(y)	f(x,y)	ll	Rank
government	gouvernement	29	28	27	267.259	1
found	constaté	7.99	8.99	7.99	99.201	2
health	santé	7	7	7	94.264	3
sustainable	durable	6.05	6.05	6.05	83.238	4
audit	vérification	7.99	8.99	6.99	78.595	5
information	renseignemen	6.04	7.04	6.04	77.367	6
areas	domaines	6.05	5.05	5.05	65.882	7
some	certains	6	6	5	59.888	8
act	loi	4	4	4	58.347	9
reporting	rapport	8.05	6.05	5.05	55.253	10
csc	service	5	4	4	53.343	11
all	tous	5	4	4	53.343	12
office	bureau	3.99	4.99	3.99	53.222	13
management	gestion	5.99	9.99	4.99	51.332	14
food	aliments	6	4	4	50.709	15
fund	fiducie	3.05	3.05	3.05	46.145	16
diseases	maladies	3.05	3.05	3.05	46.145	17
two	deux	3	3	3	45.488	18
staffing	dotation	3	3	3	45.488	19
spending	dépenses	3	3	3	45.488	20
oversight	surveillance	3	3	3	45.488	21
also	aussi	7.05	6.05	4.05	41.687	22
manages	gestion	4.99	9.99	3.99	39.786	23
small	petites	5	3	3	38.758	24
expected	attendus	5	3	3	38.758	25
needs	besoins	6.05	3.05	3.05	37.759	26
other	autres	4	4	3	36.492	27
year	année	7	3	3	35.927	28
such	comme	6	5	3	30.446	29

Table 4.14. Word correspondence list after the second filtering process in TextComp for file RE3438

- Step 1: compute Log Likelihood Ratio score $ll(x, y)$ for word pairs $\langle x, y \rangle \in S \times T$.

- Step 2: sort all the *ll*(*x*,*y*) scores from the highest to the lowest.
- Step 3: select the word pair if $x \in G_i^s$ and $y \in G_i^t$, and $\neg\exists x' \in G_i^s : A(x', y) > A(x, y)$, and $\neg\exists y' \in G_i^t : A(x, y') > A(x, y)$.
- Step 4: select the word pair if $x \in G_i^s$ and $y \in G_i^t$, and $freq(x, y) + freq(x, y'') <= freq(x)$ and $freq(x, y) + freq(x'', y) <= freq(y)$.

Table 4.14 shows the result of the two rounds of filtering in TextComp. In the table, words longer than 12 characters are truncated to 12. The counts are adjusted weights.

Another interesting finding is that after the two filtering processes, the differences among the end result selected lists for all the six statistical methods become very small, varying only in ranking positions. Here are the individual score sets for different statistical similarity metrics we selected for the test data in file IN5648.

Gale and Church: three, trois (1.00); seats, sièges (1.00); prime, premier (1.00); bill, projet (1.00); accountabili, obligation (1.00); house, chambre (0.84); members, députés (0.80); laws, lois (0.79); government, gouvernement (0.79); number, nombre (0.74); crown, couronne (0.74); accountabili, comptes (0.74); usually, normalement (0.70); other, autres (0.65); accountabili, rendre (0.59); appointed, nommés (0.55); prime, conseil (0.49); all, toutes (0.41); power, pouvoir (0.36); both, deux (0.34); bill, loi (0.31); authority, pouvoir (0.26); members, appui (0.22); law, loi (0.22); government, comptes (0.16); usually, pouvoir (0.13); seats, chambre (0.13); commons, chambre (0.13); government, obligation (0.12); house, sièges (0.11); governments, gouvernement (0.11); government, rendre (0.11); crown, pouvoir (0.11); members, chambre (0.08); house,

députés (0.08); laws, gouvernement (0.07); house, appui (0.07); government, lois (0.05); house, premier (0.03); house, doivent (0.03); government, pouvoir (0.02); members, gouvernement (0.01); house, gouvernement (0.01); government, loi (0.01); government, chambre (0.00).

Kay and Röscheisen: three, trois (1.00); seats, sièges (1.00); prime, premier (1.00); bill, projet (1.00); accountabili, obligation (1.00); house, chambre (0.93); government, gouvernement (0.91); members, députés (0.90); laws, lois (0.89); number, nombre (0.86); crown, couronne (0.86); accountabili, comptes (0.86); usually, normalement (0.83); other, autres (0.80); appointed, nommés (0.75); accountabili, rendre (0.75); prime, conseil (0.67); both, deux (0.60); all, toutes (0.60); power, pouvoir (0.55); bill, loi (0.50); authority, pouvoir (0.50); members, appui (0.46); law, loi (0.46); usually, pouvoir (0.40); crown, pouvoir (0.38); house, députés (0.35); members, chambre (0.34); government, comptes (0.32); government, rendre (0.31); house, gouvernement (0.30); seats, chambre (0.26); commons, chambre (0.26); government, obligation (0.25); government, chambre (0.24); house, sièges (0.23); governments, gouvernement (0.23); government, lois (0.23); laws, gouvernement (0.22); house, appui (0.22); house, premier (0.21); house, doivent (0.21); government, pouvoir (0.21); government, loi (0.20); members, gouvernement (0.19).

PMI: bill, projet (7.46); accountabili, obligation (7.46); three, trois (7.05); number, nombre (7.05); accountabili, comptes (7.05); laws, lois (6.73); accountabili, rendre (6.73); appointed, nommés (6.63); prime, premier (6.46); prime, conseil (6.46); other,

autres (6.46); usually, normalement (6.24); all, toutes (6.24); power, pouvoir (6.05); crown, couronne (6.05); both, deux (5.99); top, début (5.88); bill, loi (5.88); authority, pouvoir (5.63); members, députés (5.59); members, appui (5.46); law, loi (5.46); usually, pouvoir (4.82); seats, chambre (4.73); commons, chambre (4.73); government, obligation (4.65); government, comptes (4.65); crown, pouvoir (4.63); house, sièges (4.52); house, chambre (4.52); governments, gouvernement (4.52); government, gouvernement (4.45); government, rendre (4.33); laws, gouvernement (4.11); house, appui (4.11); government, lois (3.92); members, chambre (3.88); house, députés (3.65); house, premier (3.52); house, doivent (3.52); government, pouvoir (3.24); government, loi (3.07); members, gouvernement (2.94); house, gouvernement (2.81); government, chambre (2.65).

Jaccard: three, trois (1.00); seats, sièges (1.00); prime, premier (1.00); bill, projet (1.00); accountabili, obligation (1.00); house, chambre (0.87); government, gouvernement (0.83); members, députés (0.82); laws, lois (0.80); number, nombre (0.75); crown, couronne (0.75); accountabili, comptes (0.75); usually, normalement (0.71); other, autres (0.67); appointed, nommés (0.60); accountabili, rendre (0.60); prime, conseil (0.50); both, deux (0.43); all, toutes (0.43); power, pouvoir (0.38); bill, loi (0.33); authority, pouvoir (0.33); members, appui (0.30); law, loi (0.30); usually, pouvoir (0.25); crown, pouvoir (0.23); members, chambre (0.21); house, députés (0.21); government, comptes (0.19); house, gouvernement (0.18); government, rendre (0.18); seats, chambre (0.15); commons, chambre (0.15); government, obligation (0.14); government, chambre (0.14); laws, gouvernement (0.13); house, sièges (0.13); house, appui (0.13); governments, gouvernement (0.13); government, lois (0.13); house, premier (0.12);

house, doivent (0.12); government, pouvoir (0.12); government, loi (0.11); members, gouvernement (0.10).

LL: government, gouvernement (144.40); top, début (91.17); members, députés (80.74); prime, premier (65.68); crown, couronne (56.69); usually, normalement (48.19); three, trois (47.05); laws, lois (42.04); other, autres (39.41); seats, sièges (37.02); bill, projet (37.02); accountabili, obligation (37.02); number, nombre (32.52); accountabili, comptes (32.52); accountabili, rendre (30.29); prime, conseil (28.70); appointed, nommés (28.02); all, toutes (27.46); government, comptes (26.60); power, pouvoir (26.43); bill, loi (25.56); house, députés (24.45); both, deux (23.57); authority, pouvoir (21.95); members, chambre (21.94); government, rendre (21.66); members, appui (21.08); law, loi (21.08); seats, chambre (20.11); commons, chambre (20.11); government, obligation (19.79); house, sièges (19.21); governments, gouvernement (19.21); house, gouvernement (18.94); usually, pouvoir (16.95); crown, pouvoir (15.94); laws, gouvernement (14.78); house, appui (14.78); government, lois (13.20); government, chambre (11.97); house, premier (11.12); house, doivent (11.12); government, pouvoir (9.56); government, loi (8.76); members, gouvernement (8.22).

χ^2: bill, projet (523.03); accountabili, obligation (523.03); three, trois (521.06); prime, premier (517.14); top, début (511.31); house, chambre (438.16); government, gouvernement (434.39); members, députés (422.34); laws, lois (418.43); number, nombre (393.02); accountabili, comptes (393.02); crown, couronne (389.32); usually, normalement (371.48); other, autres (347.37); accountabili, rendre (313.22); appointed,

nommés (296.05); prime, conseil (260.03); all, toutes (222.03); power, pouvoir (193.53); both, deux (188.02); bill, loi (171.37); authority, pouvoir (146.14); members, appui (129.48); law, loi (129.48); government, comptes (93.66); usually, pouvoir (81.42); government, rendre (76.54); seats, chambre (73.90); commons, chambre (73.90); crown, pouvoir (70.63); government, obligation (70.10); house, députés (68.06); members, chambre (67.47); house, sièges (63.51); governments, gouvernement (63.51); laws, gouvernement (48.38); house, appui (48.38); government, lois (41.57); house, gouvernement (39.35); house, premier (30.41); house, doivent (30.41); government, chambre (24.11); government, pouvoir (23.96); government, loi (20.71); members, gouvernement (18.50).

Table 4.15 contains the list of shared word correspondences that all the six metrics arrive at after the two filtering processes. We notice an obvious improvement in the performance after the filtering processes. Much of the noise is eliminated, while legitimate translation pairs with low scores that would have been excluded by similarity scores have been included in the final list. The difference lies only in the ranking positions of translation pairs. We also tested with other data sets, and they all show that after filtering, the finally recruited word pairs vary only slightly in the legitimate candidate pairs being selected, although ranking positions for selected word pairs can be different. Without these filtering mechanisms, for some legitimate word pairs with low scores to be included, we would have to set a threshold that could have introduced a lot of noise.

x , y	f(x)	f(y)	f(x,y)	ll
government, gouvernement	24.1	26.1	23.1	162.9
house, chambre	23.05	20.05	20.05	152.74
top, début	9	9	9	91.17
crown, couronne	11.3	9.3	9.3	83.05
members, députés	9.05	11.05	9.05	81.13
prime, premier	7	7	7	74.46
number, nombre	6.1	7.1	6.1	60.8
usually, normalement	7	5	5	48.19
three, trois	4.05	4.05	4.05	47.53
laws, lois	4.05	5.05	4.05	42.51
other, autres	4	6	4	39.41
seats, sièges	3.05	3.05	3.05	37.53
bill, projet	3	3	3	37.02
accountabili, obligation	3	3	3	37.02
power, pouvoir	4	9	4	34.68
appointed, nommés	4	4	3	28.02
all, toutes	7	3	3	27.46
both, deux	5	5	3	23.57
authority, pouvoir	4	8	3	21.95
law, loi	4.05	9.05	3.05	21.46

Table 4.15. Selected non-cognate word correspondences for file IN5648 (counts reflect adjusted weights)

4.2.4.2 From Word Correspondences to Correspondences of Translation Constituents

By translation constituents, we mean word sequences or word groups which are structural units, and in most cases conceptual units in the language. The constituents can be small (one word), or large (multi-word units). Most of them are semantic units such as noun combinations, verb phrases, or clause constructions. For some clusters of structural units, although they are not complete sense groups in their own right, we still merge them and treat them as translation constituents if correspondence associations can be established between them. We also call this type of translation correspondence "phrase

correspondence", although the conventional boundary of phrases as used here is somewhat blurred. Multi-word translation constituents are more or less like combinatorial morpho-syntactic units or constructions in Construction Grammar that can be specified to a greater or lesser extent for form and meaning (Fillmore 1988; Goldberg 1995; Kay and Fillmore 1999; Goldberg 2006; Asudeh *et al.* 2008). They are an interesting and curious type of linguistic construction in translation: as a complete linguistic form unit, they are usually replaceable with a similar construction in most contexts; but when standing alone, some of them do not necessarily sound right or conform to general rules of grammar.

TextComp does not use part-of-speech parsers for phrase structure analysis or dictionaries for collocations and phrase identification for two reasons. The first is that the current way of extracting phrase translations in TextComp is already good for the purpose of computing the degree of and coverage of mutual correspondences. The second is that parsing texts for clear phrase boundary identification in both languages and comparing candidate string sequences with dictionaries can be time-consuming and computationally too expensive for a web-based application. Also, Russell (1999) argued that using bilingual dictionaries for establishing translation word associations had its drawbacks. In the following, we describe how TextComp finds relevant mapping relations for translation constituents on the bases of the word correspondence information. The methods resulted in translation constituent links that form the backbone of translation correspondence profiling in TextComp.

N-gram annealing means merging individual adjacent mapped translation words into multi-word translation constituents. It is a procedure to transform sequential word translations into phrase translations. There are two types of annealing in TextComp: 3-

gram annealing and 2-gram annealing, both of them group the scattered, paired words into translation constituents. In 3-gram annealing, when three individual words are mapped as correspondences in close proximity, they are fused into one translation phrase. This is a progressive merging process working from left to right. The three translation correspondences should be very close in similarity configurations: they have to be of the same order when bound to a complementary sequence. A 2-gram annealing procedure entails coalescence changes with a following adjacent mapped translation word. Figures 4.44 and 4.45 show the changes in mapping the correspondences before and after the annealing process.

Figure 4.44. Translation constituents before bi-gram annealing.

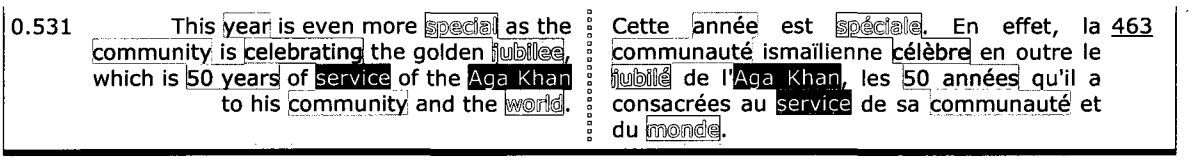


Figure 4.45. Translation constituents after bi-gram annealing.

Neighborhood assimilation refers to the process that combines neighbouring translation constituents into larger translation units or constituents. The neighbourhood assimilation operation can allow for word order changes in translation correspondence. For example, for two neighbouring translation words or constituents, when they are considered individually one after another, there is not necessarily a one-to-one correspondence. However, there exists a binding semantic relation between them. When

they are blended sequentially and are considered as a semantic unit, they are reciprocal equivalent translation constructions. The process of matching "Canadian Identity" in English with "Identité canadienne" in French in Figures 4.46 and 4.47 demonstrates this assimilation operation. Another type of neighbourhood assimilation involves the incorporation of short, unmapped translations in the neighbourhood into a larger translation constituent (see Figures 4.48 and 4.49). When neighbouring constituents get mixed, the combinatory constituent pair should observe the length ratio criterion of maximal allowable difference between the two matching translation units so that very long range translation dependencies can be avoided.

0.432	The Secretary of State for Multiculturalism and Canadian Identity sent out a video greeting to the Ismaili community, a first, I believe, for a government minister.	le secrétaire d'État (Multiculturalisme et Identité canadienne) a transmis ses meilleurs voeux sur bande vidéo à la communauté ismaïlienne. C'est une première. À ma connaissance, aucun autre ministre n'avait posé un tel geste auparavant.	465

Figure 4.46. Translation constituents before neighbourhood assimilation.

0.473	The Secretary of State for Multiculturalism and Canadian Identity sent out a video greeting to the Ismaili community, a first, I believe, for a government minister.	le secrétaire d'État (Multiculturalisme et Identité canadienne) a transmis ses meilleurs voeux sur bande vidéo à la communauté ismaïlienne. C'est une première. À ma connaissance, aucun autre ministre n'avait posé un tel geste auparavant.	465

Figure 4.47. Translation constituents after neighbourhood assimilation.

0.531	This year is even more special as the community is celebrating the golden jubilee, which is 50 years of service of the Aga Khan to his community and the world.	Cette année est spéciale. En effet, la communauté ismaïlienne célèbre en outre le jubilé de l'Aga Khan, les 50 années qu'il a consacrées au service de sa communauté et du monde.	463

Figure 4.48. Translation constituents before neighbourhood assimilation (type 2).

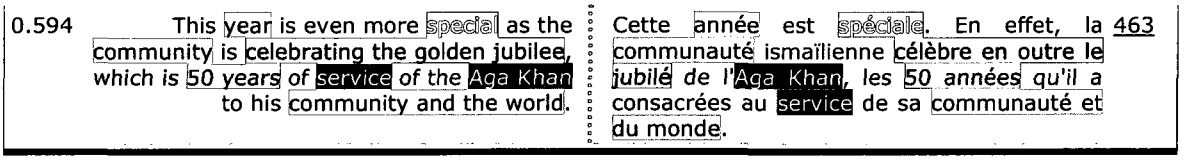


Figure 4.49. Second type of neighbourhood assimilation at work.

Astray-match filtering. Sometimes one word in the source language may match two or three words in the target language. When one of the translations has gone through the annealing or assimilating operation, and has been merged with other word translations to form translation constituents, the remaining matches will become extra matchless words that have no corresponding match counterparts in the other language. These leftover elements that become pair-wise disjoint are no longer of interest to us in translation correspondences mapping. These singletons are negligible and are thus removed from the translation correspondence list by *astray-match filtering*.

Figure 4.50 shows identified instances of translation constituents that contain two or more words. The results are generated while the system maps text units as translation equivalents or close translation equivalents. Translation problems can be quickly spotted from this look-up list of translations. Through a quick look at the list of approximations in formal translations correspondences, we can know if certain phrases are translated consistently or if the diversity in translation expressions is justified. The links in the right column can lead to other modes of viewing the results in TextComp in case we need more contextual information for verification.

41	canadian residents and non-residents	résidents du canada et les non-résidents	12
42	canadian residents	résidents canadiens	18
43	canadian stocks	actions canadiennes	159
44	canadian travellers to the united states	voyageurs canadiens allant aux Etats-Unis	56
45	canadians on overseas trips	canadiens dans les voyages	58
46	canadians travelling by car and staying at least one night in	canadiens voyageant par automobile et restant au moins une nuit aux	57
47	capital account	compte capital	141
48	capital and financial account	compte capital et financier	140
49	capital and financial	capital et financier	16
50	case for the entire year	cas pour toute l'année	94
51	client services	services à la clientèle	104
52	commercial services	services commerciaux	220
53	consecutive record	record consécutif	53
54	consumer price index	indice des prix à la consommation	264
55	contact us	contactez-nous	268
56	covers all	retrace l'ensemble des	12
57	current account balance	solde du compte courant	139
58	current account payments	paiements du compte courant	129
59	current account receipts	recettes du compte courant	119
60	current account surplus or deficit	surplus ou un déficit au compte courant	15
61	current account surplus	compte courant en surplus	20
62	current account	compte courant	233
63	current transfers	transferts courants	230
65	data are compiled	données étant compilées	22
66	data for the first quarter of	données du premier trimestre de	103
67	data sources	source de données	101
68	date modified	date de modification	278
69	decline in payments exceeded that of	diminution des paiements a excédé celle des	62
70	Deficit corresponds	déficit correspond	20
71	Deficit on investment income remains stable	le déficit des revenus d'investissement reste stable	61
72	Deficit on investment income	déficit des revenus d'investissement	62

Figure 4.50. Summary view of phrase translations in TCPro. See also *Summary* under *Comparison 2* in Figure 4.18

The n-gram based annealing and neighbourhood assimilating are of a type of approximate string matching. By combining a sequence of identified translation units into a larger matching constituent, it can allow the combined sequence to be compared to its

translation counterpart in an easier and more efficient manner. With the help of the maximal allowance criterion for length differences in translation strings, many translation constituents at the phrase level can be identified using this methodology.

4.2.4.3 TCPro Scores

Starting from the results of the annealing and assimilation operations, TextComp computes a TCPro score for each of the aligned text segments. The main idea for TCPro scoring is to measure how well the identified translation constituents overlap in the texts of two languages. The design of TCPro score metrics is motivated by the BLEU (Papineni *et al.* 2001) scores, but a different set of variables are computed. The TCPro score employs a weighted average of the basic statistics obtained while TextComp is mapping translation constituents. Part of it is based on the geometric mean of the precision of the variables evaluated. The variables include:

- v_1 = number of bi-gram translation constituents that match;
- v_2 = number of translation constituents that match;
- v_3 = number of words in the matched translation constituents;
- v_4 = number of characters in the matched translation constituents.

For example, the precision for the number of constituents that match is given by:

$$pre(v_2) = \frac{(0.1 + v_2)}{(0.1 + v_2 + \neg v_2)}$$

0.1 is added here for smoothing in case no translation constituent is found in the aligned text segment. TCPro score also takes into account the factors for penalty. The penalty score (*ps*) components include:

- Align-type (p_1) : the alignment type 1:1 has the best score. Other types such as 1:2 or 2:1 will be penalized.
- Unmatched text area (p_2): this penalizes text chunks, particularly large chunks that are longer than 45 characters where translation correspondences are not found. In the data we selected for testing, if the unmapped area in one half of the translation pair exceeds approximately 45 characters, and if the unmapped area is not generally balanced on the other half of the translation pair, it usually indicates discrepancies in translation.
- Length criterion (p_3): aligned text segments that are out of proportion, and that cannot pass the linear regression matching criterion are penalized.

Given the precision pre_n of variables of size up to N (here N=4), and the penalty score ps_m , the TCPro score is computed as follows:

$$TCPscore = ps_m \cdot \exp\left(\sum_{n=1}^{N} w_n \log pre_n \right)$$

Where

$$pre_n = \sum_{i=1}^{4} pre(v_i)$$

$$ps_m = \prod_{i=1}^{3} ps(p_i)$$

w_n is the weighting factor, and it is set at $1/N$.

Our experiments show that for published official translations on the government website, the average TCPro scores are in the range of around .50 to .95. Most of the online texts, when aligned by TextComp, have a TCPro score of .6 to .8. If the average TCPro score is below .4, usually there is something to be found such as omissions, discrepancies in translation or misalignment. Low average TCPro scores as a result of free style translations and idiomatic expressions in translations are not so common in officially published web-based bilingual materials.

Text segments mapped with translation constituent correspondences and displayed sequentially along with TCPro scores (Figure 4.51) make explicit the levels of formal correspondence in translation in the real text. It can provide us with context clues for certain translation correspondences when we want to refer to the neighbouring text segments. Observing these corresponding links in the background of the aligned text segments in proximity, we may decide if there is enough evidence of association to validate that the text segments are a true translation pair, or if some of the discrepancies are erroneous translations that derive from the misalignment of neighbouring translation pairs.

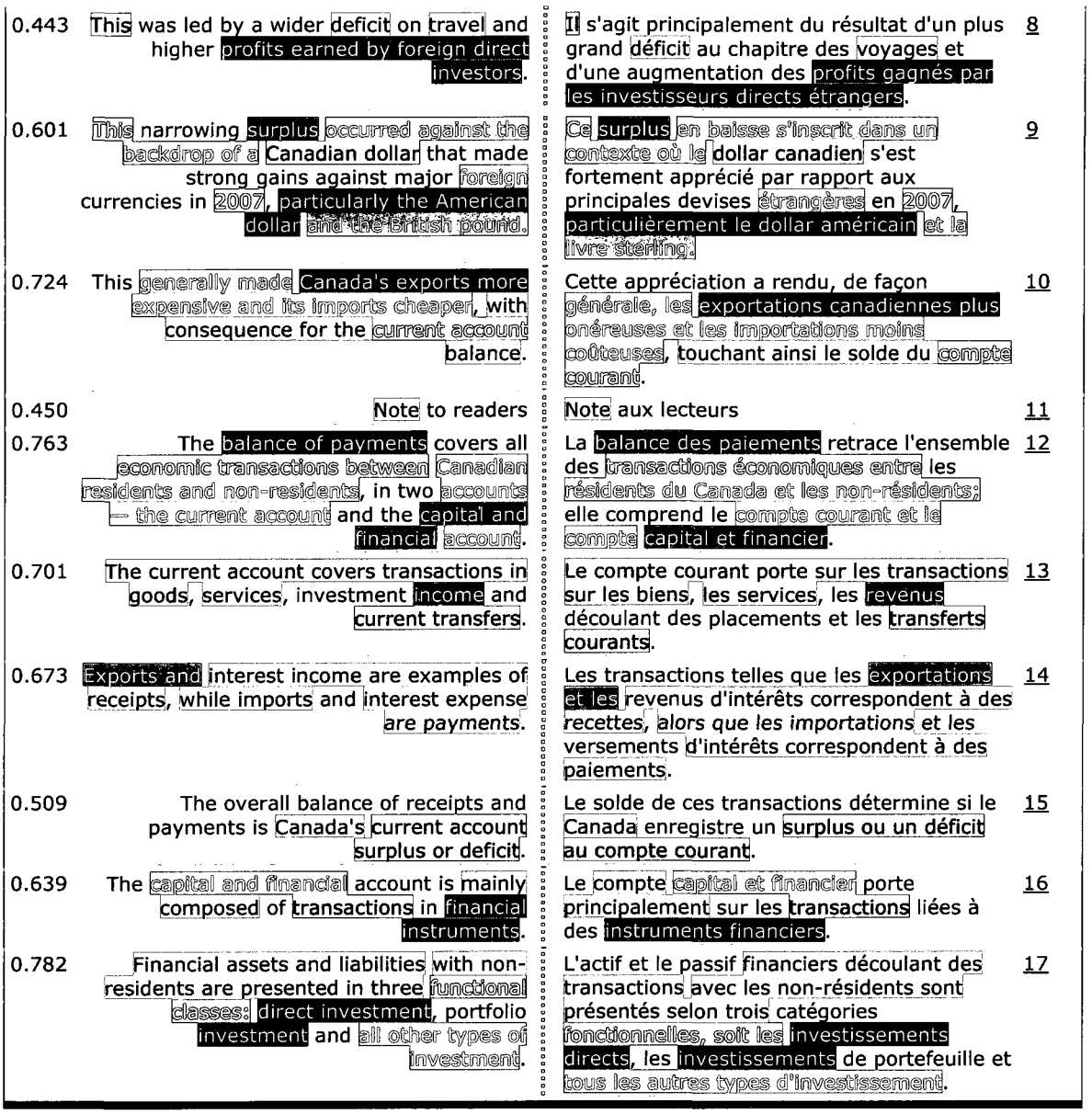
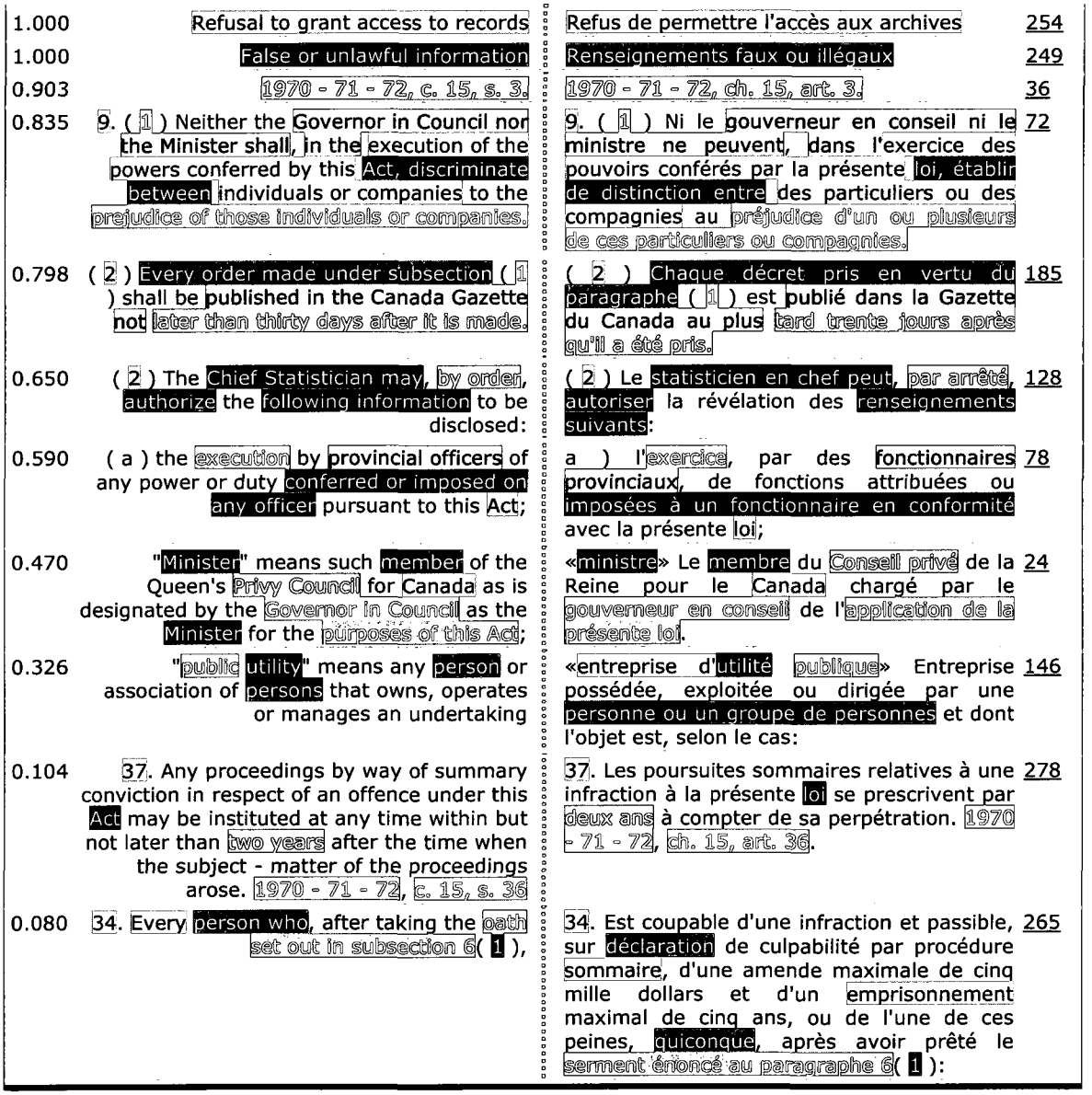


Figure 4.51. Sequential view of TCPro. See also *Serial* in Figure 4.18

Text segments can also be sorted by TCPro scores and displayed on a scale-basis (Figure 4.52). Here the system can group aligned text segments according to their translation correspondence scores. Considerable information can be gleaned from this mode of view about the proportions of translation correspondence in the aligned

translation text segments where translation discrepancies exist. In the areas where translation correspondence scores are extremely low, we are more likely to find translation discrepancies that indicate translation errors or translations that need to be adjusted.



1.000	Refusal to grant access to records	Refus de permettre l'accès aux archives	254
1.000	False or unlawful information	Renseignements faux ou illégaux	249
0.903	1970 - 71 - 72, c. 15, s. 3.	1970 - 71 - 72, ch. 15, art. 3.	36
0.835	9. (1) Neither the Governor in Council nor the Minister shall, in the execution of the powers conferred by this Act, discriminate between individuals or companies to the prejudice of those individuals or companies.	9. (1) Ni le gouverneur en conseil ni le ministre ne peuvent, dans l'exercice des pouvoirs conférés par la présente loi, établir de distinction entre des particuliers ou des compagnies au préjudice d'un ou plusieurs de ces particuliers ou compagnies.	72
0.798	(2) Every order made under subsection (1) shall be published in the Canada Gazette not later than thirty days after it is made.	(2) Chaque décret pris en vertu du paragraphe (1) est publié dans la Gazette du Canada au plus tard trente jours après qu'il a été pris.	185
0.650	(2) The Chief Statistician may, by order, authorize the following information to be disclosed:	(2) Le statisticien en chef peut, par arrêté, autoriser la révélation des renseignements suivants:	128
0.590	(a) the execution by provincial officers of any power or duty conferred or imposed on any officer pursuant to this Act;	a) l'exercice, par des fonctionnaires provinciaux, de fonctions attribuées ou imposées à un fonctionnaire en conformité avec la présente loi;	78
0.470	"Minister" means such member of the Queen's Privy Council for Canada as is designated by the Governor in Council as the Minister for the purposes of this Act;	«ministre» Le membre du Conseil privé de la Reine pour le Canada chargé par le gouverneur en conseil de l'application de la présente loi.	24
0.326	"public utility" means any person or association of persons that owns, operates or manages an undertaking	«entreprise d'utilité publique» Entreprise possédée, exploitée ou dirigée par une personne ou un groupe de personnes et dont l'objet est, selon le cas:	146
0.104	37. Any proceedings by way of summary conviction in respect of an offence under this Act may be instituted at any time within but not later than two years after the time when the subject - matter of the proceedings arose. 1970 - 71 - 72, c. 15, s. 36	37. Les poursuites sommaires relatives à une infraction à la présente loi se prescrivent par deux ans à compter de sa perpétration. 1970 - 71 - 72, ch. 15, art. 36.	278
0.080	34. Every person who, after taking the oath set out in subsection 6(1),	34. Est coupable d'une infraction et passible, sur déclaration de culpabilité par procédure sommaire, d'une amende maximale de cinq mille dollars et d'un emprisonnement maximal de cinq ans, ou de l'une de ces peines, quiconque, après avoir prêté le serment énoncé au paragraphe 6(1):	265

Figure 4.52. TCPro on a scale-basis. See also *Scaling* in Figure 4.18

4.2.5 Discussion

Translation correspondence profiling in TextComp offers exciting new information about different types of translation correspondences in different text segments. Overall, the translation correspondence mapping algorithms produce reliable, paired translation constituent profiles. The profiling represents an important diagnostic tool to detect areas of translation discrepancy. From the profiling information, we can further examine the areas where the correspondences are lacking. Then we can generate new hypotheses about the potential translation discrepancies in the region and determine if they are creative translations or erroneous translations. For example, in the paired translation text segments in Figure 4.53, the word and phrase order changes can be detected by a quick look at the coloring schemes that indicate translation correspondences. In the translation pair in Figure 4.54, we notice the blank color stretch of text that demonstrates the absence of translation correspondence. A closer look reveals the insertion and deletion present in the translation pair.

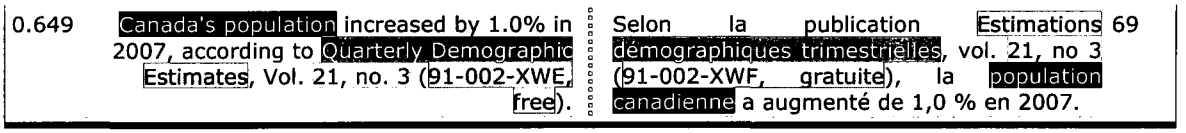


Figure 4.53. Word and phrase order changes detected by TCPro.

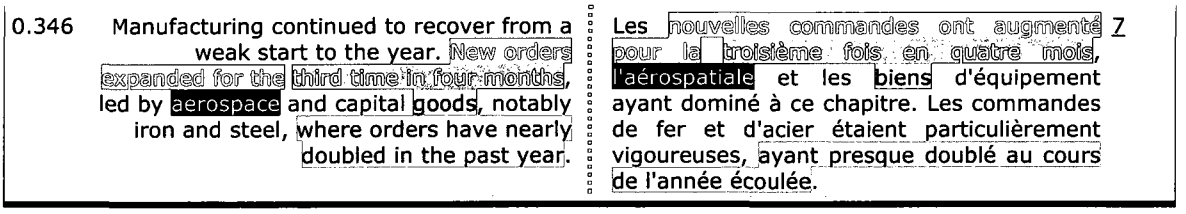


Figure 4.54. Insertion and deletion detected by TCPro.

BMIA is a helpful tool for spotting discrepancies and potential translation errors that benefits even experienced translators operating under well-developed quality assurance protocols, as evidenced by usage of BMIA at Statistics Canada. The system can help decrease the number of mistakes and improve the overall quality of translation. For example, when BMIA finds a number discrepancy in translation, such as 1990 and 90 in the following alignment, or any of the disputable inconsistent uses of numbers in Table 4.16, it is good to alert the translator for further attention.

> *e1*. In the early 1990s, despite larger declines in earnings in the North than in Canada, employment income remained higher.
>
> *f1*. Au début des années 90, malgré une baisse des gains plus prononcée dans le Nord que dans l'ensemble du Canada, le revenu d'emploi y est demeuré plus élevé.

English	French
23-Jun-08	23 juin 2008
1995/6 - 1997/8	1995-1996 - 1997-1998
9-1-1	911
2003/2004	2003-2004
three percent	3 p. 100
10 days	dix jours
08:15	8 h 15
Four hundred years ago	Il y a 400 ans
During the 1920s,	Pendant les années 20,

Table 4.16. Examples of potentially disputable discrepancies in number translations.

However, perceiving translation discrepancies and errors is not as easy a task as it may appear. It is true that formal mistakes are usually indicative of quality problems in a translation, but sometimes a formal mistake can be a sanctioned false error rather than a

true oversight. On some occasions, the kind of features for error detection does not necessarily determine the existence of a translation error. For example, when the agent scans sentences with different contact information, different email addresses and different phone numbers in different languages, it gives the alignment some penalties. In translation problem detection, the system penalizes instances of mismatched numbers. But actually, the texts may have no translation problems at all. Or if there is a problem, it is only minor in nature. For different phone numbers of contacts for different languages, the discrepancy in numbers does not mean that the sentences are misaligned or the translation is problematic. It is the right way to say that for different languages, call different numbers or contact different people. The same is true with the use of numbers to indicate a particular year. When we say 2006 in French and say 'this year' in English, it becomes a 'problem' match because of different ways of expressing the same content meaning. But the lack of the number '2006' in English does not constitute a misalignment, nor a breach in the translation style. These confusing cases of translation discrepancies about the mapping of numbers can cause misjudgments in translation error detection for BMIA.

Although there are guidelines and standards to follow in the editing of bilingual materials for publication on government websites, sometimes the bilingual text editor has to consciously deviate in translation for the sake of fulfilling demands of readability and fluency. In this case, BMIA should be appreciative and lenient rather than critical and punitive, unless the circumstances are not appropriate. Ideally, items and features that cause the mismatch should be filtered in specified contexts. The knowledge can be stored in memory by an inherent learning mechanism of the agent. This will enable the agent to

act more intelligently in different situations and make the feature lists more refined and discriminating. The direct benefit will be a more accurate and reliable detection operation.

2. In this Act,	**2.** Les définitions qui suivent s'appliquent à la présente loi.
"Board" «*conseil* »	«agriculture» "*farming*"
"Board" means the Board of Directors of the Corporation;	«agriculture » Outre la culture du sol ou l'élevage du bétail, l'apiculture, la production laitière, la culture des fruits, ainsi que toute activité agricole ayant fait l'objet d'une décision au titre du paragraphe 4(3).
"business related to farming" «*entreprise liée à l'agriculture* »	«conseil» "*Board*"
"business related to farming" means a business that primarily produces, transports, stores, distributes, supplies, processes or adds value to inputs to or outputs from farming operations;	«conseil » Le conseil d'administration de la Société.
"Chairperson" «*Version anglaise seulement* »	«entreprise liée à l'agriculture» "*business related to farming*"
"Chairperson" means the Chairperson of the Board appointed under subsection 7(1);	«entreprise liée à l'agriculture » S'entend d'une entreprise dont l'activité principale est la production, le transport, l'entreposage, la distribution, l'approvisionnement ou la transformation soit de moyens de production destinés à des exploitations agricoles, soit de produits de ces exploitations, ou l'adjonction de valeur à ceux-ci.
"Corporation" «*Société* »	«filiale» "*subsidiary*"
"Corporation" means Farm Credit Canada continued by subsection 3(1);	«filiale » S'entend au sens du paragraphe 83(6) de la *Loi sur la gestion des finances publiques*.
"director" «*Version anglaise seulement* »	
"director" means a member of the Board;	
"farming" «*agriculture* »	
"farming" includes livestock raising, bee-keeping, dairying, fruit growing, tillage of the soil and any other husbandry activity that the Corporation determines, under subsection 4(3), to be farming for the purposes of this definition;	

Figure 4.55. Some items in definition lists in different languages do not correspond on a text segment basis.

When we looked at the files that do not align very well in TextComp, we found that, to the contrary of what we had expected, legal documents do not necessarily align better than other types of texts (see also the *AC* row in Table 4.10 in Section 4.2.2.3). The main reason is that for most of the laws, acts and regulations, there is a section to explain special terminology. The terms and definitions in the section are usually arranged in alphabetical order. Items sorted in the alphabetical lists do not correspond to each other. For example, in aligning text segments in Figure 4.55, "agriculture" in French is on the top of the list, while "farming" in English is not. This large scale swapping of sentences and definitions can cause problems in paragraph alignment, and result in massive text segment misalignment for TextComp.

Since the main targeted data sets for this study are web-based bilingual materials, the positioning of translated paragraphs and the consistency of using the same HTML styles for both languages all have an impact on SDTES' performance in automatic translation extraction. TextComp in BMIA sometimes has the same problem with tables and contents that are formatted in a radically different manner in English and French. When the web pages are cut and pasted for comparison, they return different presentation formats that can disrupt the chain of correct text segment alignment.

For translation discrepancy detection and translation correspondence profiling, although techniques based on lexical information offer a safer and more reliable alignment for noisier data sets, they often have the additional cost of increased computational complexity. Inexpensive acquisition of specific lexical information that can be used as anchoring points is still to be sought. In the word correspondence matching model in TextComp, although the filtering mechanisms can greatly reduce

noises, they cannot eliminate all cases of near misses (Gale and Church 1991) or indirect associations (Melamed 2000). Cognate-based lexical information, when infused into the hybrid BMIA model, can achieve good results for targeted bilingual texts, but not necessarily for texts of a free translation style or texts of another language pair. In translation texts where none of the translation constituent correspondence can be found, translation discrepancy detection and translation correspondence profiling can be difficult.

There is no doubt that BMIA can never replace an experienced translator who is engaged in the problem detection process. Although a sound translation evaluation should go beyond intuition to achieve objectivity and accuracy (Kupsch-Losereit 1985), the practical quality evaluation operation inevitably involves personal judgment and cannot be a mere mechanical process. What BMIA detects as translation problems are errors that indicate the more formal translation inconsistencies and mistakes. The adopted approach to translation problem prediction does not entail all the complicated subjective elements used by human translators with solid theoretical as well as practical background in translation. There are some translation problems that cannot be handled by the translation error detection mechanism of BMIA, such as the tone of the translated text, the natural flow of texts, better choice of words, better ways of expressing the same content meaning, idiomatic usage of the target language, and thorough understanding of the source text. It is currently a typical AI complete problem to produce an objective assessment measurement that can compute the severity of translation problems in a way which meets the full expectation of experienced translators.

4.3 Summary

In this chapter, we have described the major component systems of BMIA: the StatCan Daily Translation Extraction System (SDTES) and the StatCan Bilingual Text Comparison System (TextComp). SDTES and TextComp are designed and developed for two types of bilingual text mapping tasks, one for translation corpora building and the other for translation discrepancy detection. We found that, in SDTES, the Gale-Church algorithm with the help of some main HTML tags and other structural properties was a good fit for aligning published web-based materials. This is particularly so when the purpose of bilingual text mapping is to extract translations for bilingual corpus building. However in TextComp, when the goal of bilingual text mapping is for translation discrepancy detection in a web-based application, and when the input data sets are noisier texts such as those in the process of being edited or checked prior to publication, the approach that integrates dynamic programming with linear regression forecasting has proved to be a good choice.

For SDTES, we proposed a set of protocols and algorithms that combined the Gale-Church algorithm and the *k*-vec algorithm with our Acceptable Matching Sequence (AMS) algorithm. AMS integrates the straightforwardness of the cognate matching algorithm of Simard *et al.* (1992) with the strength of the no-crossing-links constraint in the Longest Common Subsequence Ratio algorithm (Melamed 1999). Tests and evaluation show that these procedures and methods are very efficient in mapping officially published web materials for automatic translation extraction. We also presented some results of SDTES --- the StatCan Daily Corpus (SDC) and the translation concordance search system (TransConcord) to access SDC. Impressive features of

TransConcord include its capacity to instantly retrieve translations from large data banks and corpora, the use of monolingual KWIC format to narrow down the query choice range, and the ability to identify translation contexts so that the returned results can be shown in a bilingual KWIC format.

TextComp is a web-based application that is targeted at mapping bilingual data sets for translation discrepancy detection and Translation Correspondence Profiling (TCPro). The input texts can be noisier in that they can contain insertions and deletions, translation discrepancies and other problems. In this chapter, we presented our algorithms for aligning and comparing this type of bilingual texts. We found that our *k*-band algorithm narrows down the search space significantly, and makes the Dynamic Programming (DP) approach a good and practical choice for paragraph alignment. We modeled the sampled data sets from SDC and designed a linear regression forecasting model to estimate the maximal allowable length constraint for text segment alignment. Results show that this approach, when used with the *forward-backword matching* algorithm, can be a great time-saver for mapping the translation text segments. Translation Correspondence Profiling (TCPro) is a new concept that we proposed in this thesis. We argue that profiling the structural associations in translation texts can help objectively measure and assess the faithfulness of translations, and thus can aid the process of detecting translation divergences and discrepancies. We designed methods to establish similarity links between words, and then between *translation constituents*. Most of the algorithms are unsupervised algorithms that do not rely on external lexical, syntactic or semantic resources, which makes them a good fit for time-critical, web-based applications. Examples shown by the different display modes of TCPro demonstrated that algorithms

in TextComp such as *n-gram annealing, neighbourhood assimilation, astray-match filtering*, and the TCPro score metric all yielded very desirable results.

Chapter 5

Conclusion and Future Work

A computational model of a Bitext Mapping Intelligent Agent (BMIA) has been proposed in this thesis for the StatCan Daily Translation Extraction System (SDTES) and the StatCan Bilingual Text Comparison System (TextComp). The BMIA model can be deemed as an attempt to creatively implement some key ideas (Isabelle *et al.* 1993; Isabelle and Church 1997; Kay 1997) in machine translation, bilingual text mapping and computer assisted human translation. Meanwhile, the work intends to provide new approaches and algorithms to better meet the practical needs of collecting, mapping, and comparing translation texts for translation corpus building, machine translation, translation analysis, translation evaluation, and translation quality assessments.

The thesis reviewed the background of corpus development, the theoretical issues in corpus-based research, and challenges and issues in bilingual corpus building. We also investigated major previous work of core technologies underlying all these models and systems in BMIA: bilingual text alignment and word correspondence mapping. Four relevant text alignment models were studied and described in this thesis: the Gale and Church length-based model, IBM Model 1, the Kay and Röscheisen lexical model, and the K-vec model.

We explained the algorithms and procedures for bilingual text mapping in the StatCan Daily Translation Extraction System (SDTES) and the StatCan Bilingual Text Comparison System (TextComp), two major systems in the BMIA model. We

demonstrated how SDTES was used to build the StatCan Daily Corpus (SDC), and how the translation concordance system (TransConcord) was employed to retrieve data from SDC. We presented algorithms that are developed for the StatCan Bilingual Text Comparison System (TextComp), their features, applications and results for translation error detection and Translation Correspondence Profiling (TCPro).

In SDTES, we used the Gale-Church algorithm for a two-pass alignment at the paragraph and text segment levels. We designed the AMS algorithm for automatic cognate identification on the basis of the candidate translation word pairs produced by the K-vec algorithm. BMIA used various types of anchor points such as length, numbers, cognate words and HTML markups for misalignment detection. When identified misaligned pairs are filtered, the clean translations are assembled for the compilation of the bilingual corpus: the StatCan Daily Corpus (SDC) which consists of more than 488,000 aligned text segments. SDTES has also been tested with web-based materials from 5 other government websites, which produced tens of thousands aligned translation pairs that can be readily fed into translation memory systems. For easy access to the bilingual corpora, we developed a brand new translation concordance search system (TransConcord) which features bilingual KWIC indexing and automatic query translation identification in target language texts.

The TextComp system included mechanisms for translation discrepancy detection and Translation Correspondence Profiling (TCPro). Texts have to be aligned before they are compared. For paragraph level alignment, we designed a *k*-band algorithm which is based on the dynamic programming algorithm. After analyzing the aligned SDC data sets, we proposed a linear regression forecasting model for text segment level alignment for

TextComp. In mapping translation equivalents for translation correspondence profiling, we introduced approaches such as *n-gram annealing* and *neighbourhood assimilation* for expanding translation correspondence from word level to the level of what we call *translation constituents*. TCPro metrics were also designed in this thesis for measuring and distinguishing correspondence levels of translation equivalents.

Relevant evaluation methods were proposed together with the experimental results. Evaluation results and user response at Statistics Canada show that, for the task of automatic identification of translation discrepancies, TextComp in BMIA is robust, practical and suitable. TextComp not only aligns the translation pairs and finds translation correspondences, but also perceives the context of sentence alignment and judges if there are problems or discrepancies with the translation at the text segment level or at the translation constituent level. Also, TextComp algorithms are autonomous in that no human pre-aligning intervention is needed. Most of the methods and algorithms proposed are unsupervised and on-the-fly implementations that are fast and efficient for web-based applications.

The protocols, algorithms, and procedures developed in this thesis for parallel text mapping will add to the literature in bitext mapping and data mining, translation equivalence studies and analysis, machine translation and machine assisted translation, translation evaluation and translation quality assessment. Unique features in these fully functioning systems contribute to the design and development of translation support software and tools. At the same time, the systems described in the BMIA model have a diversity of applications. They can be independently used for bilingual corpus building and translation discrepancy detection. They also have the potential of being integrated

into text editors, translation memories, and other applications for bilingual text comparison, translation error checking, translation correspondence analysis, bilingual information retrieval and bilingual text navigation. The mechanisms for detecting and identifying potential translation errors can be readily helpful to translation quality assurance and proof-reading. As a computational model, major components and algorithms of BMIA can be employed to help towards word and phrase alignment, machine translation, and natural language text modeling.

5.1 Future Work

There is still a lot of room for improvement in each of the component systems developed in this thesis. We plan to look at new and original methods for better performance. At the same time, we need to further explore existing algorithms and build on our experimentations with the algorithms to boost the performance. For example, although TransConcord constitutes progress in query word matching for bilingual concordancing, the query word correspondence identification process can be improved to generate more exact results, especially in the case of near misses. For this, we may need to continue our experimentation with IBM Model 1. Our testing with IBM Model 1 yielded impressive results, but a discouraging aspect is the slow speed of iterations in EM. In many previous applications of IBM Model 1, an implicitly recognized approach to overcome the slowness of the training process is "the widespread adoption of early stopping in estimating the parameters of Model 1" (Moore 2004). However, reducing the number of iterations in EM is usually at the cost of increasing the alignment error rate. There is an issue in incorporating IBM Model 1 in the BMIA model: it has to be fast because our

system is web-based and it requires real-time, unsupervised data training to accommodate data sets of a more arbitrary nature for each user and for each pair of documents in bilingual text alignment and comparison. In our current experiment with the standard IBM Model 1 on a regular StatCan server that can host the agency's busy web-based applications such as *Summary Tables*, the alignment of 100 text segment pairs with approximately 10 to 20 iterations each usually requires a waiting time that is not acceptable to a normal web interface user. Recently there has been encouraging research investigating methods for improving IBM Model 1 (Moore 2004), particularly in tackling parameter estimation problems in the EM process (Callison-Burch et al. 2004; Talbot 2005). Our future research will be along the line of constrained EM (Talbot 2005) or alternative approaches that allow fast optimization of model parameters as proposed by Moore (2005).

We plan to use the SDTES system in BMIA to build more bilingual corpora from more Canadian government websites, and then do contrastive translation analysis between the corpora. Hopefully by studying some quantitative patterns such as the type-token ratio, the lexical density and other structural and statistical properties of the translation texts, we can gain more insights about the translation universals shared among them and prominent dissimilarities between them.

We would like to try the BMIA model with some more difficult types of texts such as novels, dialogues, plays, essays etc. We intend to see what parameters need to be adjusted or redesigned to yield good results for those genres of bilingual texts. Although we know that some mechanisms would not work well in other language pairs, we still plan to try the BMIA model with other language pairs, especially disparate language pairs such as

English and Chinese. It would be interesting to see what algorithms would work and what would not, and what key factors would have an impact on the performance.

Translation correspondence profiling as proposed in this thesis can be used to provide information on translation behaviours of specific texts. The capacity of highlighting equivalence relationships between translation constituents in source and target texts can open doors for many related studies such as in machine translation evaluation, objective assessment of translations, translation divergences (Dorr 1994), lexical gaps, and semantic-syntactic correspondences (Zhu 2009). When more data is available, a next step in TCPro research is to analyze bilingual texts of different genres on a multi-dimensional scale, as proposed by Biber (1993b). We predict that factor analysis and other statistics of the aligned bilingual texts will reveal interesting findings about different text genres with different translation styles ranging from literal translation to free translation.

References

Aarts, J. 1990. Corpus linguistics: an appraisal. In *Computers in Literary and Linguistic Research*, pages 13-28, Champion Slatkine, Paris-Genève.

Adamson, G. W. and J. Boreham. 1974. The use of an association measure based on character structure to identify semantically related pairs of words and document titles. In *Information Storage and Retrieval*, 10:253-260.

Al-Onaizan, Y., J. Curin, M. Jahr, K. Knight, J. Lafferty, D. Melamed, F. Och, D. Purdy, N. Smith, and D. Yarowsky. 1999. Statistical machine translation. Technical report, Johns Hopkins University.

Arthern, P. J. 1978. Machine translation and computerized terminology systems: a translator's viewpoint. In B. M. Snell, editor. *Translating and the Computer: Proceedings of a Seminar, London, 14th November 1978*, pages 77–108, North Holland, Amsterdam.

Asudeh, A. and I. Toivonen. 2009 (in press). Lexical-Functional Grammar. In B. Heine and H. Narrog, editors, *The Oxford Handbook of Linguistic Analysis*, Oxford University Press, Oxford.

Asudeh, A., M. Dalrymple and I. Toivonen. 2008. Constructions with lexical integrity: Templates as the lexicon-syntax interface. In M. Butt and T. H. King, editors, *Proceedings of the LFG08 Conference*, pages 68–88, CSLI Publications, Stanford, CA.

Baker, M. 1992. *In Other Words*, Routledge, London, p 304.

Baker, M. 2000. Towards a methodology for investigating the style of a literary translator. *Target* (12)2: 241-266.

Barlow, M. 2000. Parallel texts in language teaching. In S. Botley, T. McEnery and A. Wilson, editors, *Multilingual Corpora in Teaching and Research*, pages 106-115, Rodopi, Amsterdam.

Barlow, M. 2002. ParaConc. Concordance software for multilingual parallel corpora. In *LREC-2002: Third International Conference on Language Resources and Evaluation. Workshop: Language Resources for Translation Work and Research,* pages 20-24, Las Palmas, Canary Islands.

Biber, D. 1993a. Representativeness in corpus design. In *Literary and Linguistic Computing*, Issue 4, pages 243-257, Oxford University Press, Oxford.

Biber, D. 1993b. The multi-dimensional approach to linguistic analyses of genre variation: an overview of methodology and findings. *Computers and the Humanities* 26, pages 331-345.

Bowker, L. and M. Barlow. 2004. Bilingual concordancers and translation memories: a comparative evaluation. In *Proceedings of the Second International Workshop on Language Resources for Translation Work, Research and Training*, pages 52-61.

Brown, P. F., J. C. Lai, and R. L. Mercer. 1991. Aligning sentences in parallel corpora. In *Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics*, pages 169-176, Berkeley, CA.

Brown, P. F., S. A. Della Pietra, V. J. Della Pietra, and R. L. Mercer. 1993. The mathematics of statistical machine translation: parameter estimation. *Computational Linguistics*, 19 (2): 263–311.

Brown, R. D., G. J. Carbonell, and Y. Yang. 2000. Automatic dictionary extraction for cross-language information retrieval. In J. Véronis, editor, *Parallel Text Processing*, pages 275-298, Kluwer Academic Publishers, Dordrecht.

Callison-Burch, C., D. Talbot, and M. Osborne. 2004. Statistical machine translation with word- and sentence-aligned parallel corpora. In *Proceedings of the 42nd Meeting of the Association for Computational Linguistics (ACL 2004)*, pages 175–182, Barcelona, Spain.

Callison-Burch, C., C. Bannard, and J. Schroeder. 2005. A compact data structure for searchable translation memories. In *10th European Association for Machine Translation Conference: Building Applications of Machine Translation*, pages 59–65.

Carroll, J. B., P. Davies, and B. Richman. 1971. *The American Heritage Word Frequency Book*. Houghton Mifflin, Boston.

Chen, J. and J. Y. Nie. 2000. Parallel web text mining for cross-language IR. In *Proceedings of RIAO 2000: Content-Based Multimedia Information Access*, vol. 1, pages 62-78, Paris, France.

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

Chen, S. F. 1996. *Building Probabilistic Models for Natural Language*. Ph.D. thesis, Harvard University, Cambridge, MA.

Chomsky, N. 1957. *Syntactic Structures*. Mouton, The Hague.

Chomsky, N. 1980. *Rules and Representations*, Columbia University Press, New York (Excerpted in *The Behavioral and Brain Sciences* 3 (1980): 1-61, 1980).

Chomsky, N. 1986. *Knowledge of Language: Its Nature, Origin, and Use.* Praeger Publishers, New York.

Chomsky, N. 1995. *The Minimalist Program*, MIT Press, Cambridge, MA.

Choueka, Y., E. S. Conley and I. Dagan. 2000. A comprehensive bilingual word alignment system: application to disparate languages - Hebrew and English. In J. Veronis, editor, *Parallel Text Processing*, pages 69-96, Kluwer Academic Publishers.

Church, K. W. and W. A. Gale. 1991. Concordances for parallel texts. In *Using Corpora, Proceedings of the 7th Annual Conference of the UW Centre for the New OED and Text Research*, pages 40–62, Oxford.

Church, K. W. 1993. Char_align: A program for aligning parallel texts at the character level. In *Proceedings of the Workshop on Very Large Corpora: Academic and Industrial Perspectives, ACL*. Association for Computational Linguistics.

Clear, J. 1992. Corpus sampling. In G. Leitner, editor, *New Directions in English Language Corpora*, pages 21-31, Mouton-de-Gruyter, Berlin.

Clerck, B. D. 2003. Review of new frontiers of corpus research. Pam Peters, Peter Collins and Adam Smith (eds.). *ICAME*, 27:76-82.

Dagan, I., K. W. Church, and W. A. Gale. 1993. Robust bilingual word alignment for machine aided translation. In *Proceedings. of the Workshop on Very Large Corpora: Academic and Industrial Perspectives*, pages 1–8.

Danielsson, P. and K. Mühlenbock. 2000. The misconception of high-frequency words in Scandinavian translation. In *Envisioning machine translation in the information future, 4th conference of the Association for Machine Translation in the Americas, AMTA 2000*, pages 158–168, Cuernavaca, Mexico.

Debili, F. and E. Sammouda. 1992. Appariement des phrases de textes bilingues. In *Proceedings of the 14th International Conference on Computational Linguistics (COLING'92)*, pages 517-538, Nantes, France.

Deng, Y., S. Kumar, and W. Byrne. 2006. Segmentation and alignment of parallel text for statistical machine translation. *Natural Language Engineering*, 12(4):235-260.

Diab, M. and P. Resnik. 2002. An unsupervised method for word sense tagging using parallel corpora. In *40th ACL Meeting*, Philadelphia.

Dice, L. R. 1945. Measures of the amount of ecologic association between species. *Ecology*, 26: 297-302.

Ding, Y., D. Gildea and M. Palmer. 2003. An algorithm for word-level alignment of parallel dependency trees. In *MT Summit IX*, pages 95-101, New Orleans, LO.

Dominic W., B. Dorow and C. Chan. 2002. Using parallel corpora to enrich multilingual lexical resources. In *Third International Conference on Language Resources and Evaluation (LREC 3)*, pages 240-245, Las Palmas.

Dorr, B. J. 1994. Machine translation divergences: a formal description and proposed solution. *Computational Linguistics*, 20(4):597-633.

Dorr, B. J. and C. Monz. 2004. Introduction to computational linguistics. Available at http://www.umiacs.umd.edu/~christof/courses/cmsc723-fall04/lecture-notes /Lecture8-statmt.ppt.

Dunning, T. 1993. Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, 19(1): 61-74.

Eijk, P. V. D. 1993. Automating the acquisition of bilingual terminology. In *Proceedings of the 6th Conference of the European Chapter of the ACL*, Utrecht / The Netherlands.

Fillmore, C. J. 1988. The mechanisms of 'Construction Grammar'. In *Proceedings of the Berkeley Linguistics Society,* 14:35-55, Berkeley, CA.

Fillmore, C. J. 1992. "Corpus linguistics" vs. "Computer-aided armchair linguistics", In *Directions in Corpus Linguistics: Proceedings from a 1991 Nobel Symposium on Corpus Linguistics*, pages 35-66, Mouton de Gruyter, Stockholm.

Fillmore, C. J., N. Ide, D. Jurafsky, and C. Macleod. 1998. An American National Corpus: a proposal. In *Proceedings of the First Annual Conference on Language Resources and Evaluation*, pages 965-969, European Language Resources Association, Paris.

Foster, G., P. Langlais, E. Macklovitch, and G. Lapalme. 2002. TransType: text prediction for translators. Demonstration description. In *40th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 148-155, Philadelphia.

Fries C. and A. Traver. 1940. *English Word Lists: a Study of Their Adaptability and Instruction*, American Council of Education, Washington, DC.

Fung, P. and K. W. Church. 1994. K-vec: A new approach for aligning parallel texts. In *Proceedings of the 15th International Conference on Computational Linguistics*, pages 1,096-1,102, Kyoto, Japan.

Fung, P. and K. McKeown. 1994. Aligning noisy parallel corpora across language groups: word pair feature matching by dynamic time warping. In *Technology*

Partnerships for Crossing the Language Barrier: Proceedings of the First Conference of the Association for Machine Translation in the Americas, pages 81–88, Columbia, MD.

Fung, P. and K. McKeown. 1997. A technical word and term translation aid using noisy parallel corpora across language groups. *Machine Translation*, 12 (1–2):53–87.

Gale, W. A. and K. W. Church 1991. A program for aligning sentences in bilingual corpora. In *Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 177-184, Berkeley, CA.

Gale, W. A. and K. W. Church. 1993. A program for aligning sentences in bilingual corpora. *Computational Linguistics*, 19(1): 75-102.

Gao, Z. M. 1997. *Automatic Extraction of Translation Equivalents from a Parallel Chinese–English Corpus*. Ph.D thesis, UMIST, Manchester, England.

Garside, R. 1987. The CLAWS word-tagging system. In R. Garside, G. Leech and G. Sampson, editors, *The Computational Analysis of English: A Corpus-based Approach*. Longman, London.

Gey, F. C., A. Chen, M. K. Buckland, and R. R. Larson. 2002. Translingual vocabulary mappings for multilingual information access. In *25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2002)*, pages 455–456, Tampere, Finland.

Giguet, E. and P. Luquet. 2005. Multi-lingual lexical database generation from parallel texts with endogenous resources. In *PAPILLON-2005 Workshop on Multilingual Lexical Databases*, Chiang Rai, Thailand.

Goldberg, A.. 1995. *Constructions: A Construction Grammar Approach to Argument Structure*. University of Chicago Press, Chicago.

Goldberg, A.. 2006. *Constructions at Work: The Nature of Generalization in Language*. Oxford University Press.

Halliday, M. A. K. 1966. Lexis as a linguistic level. In C. E. Bazell, J. C. Catford, M. A. K. Halliday and R. H. Robins, editors, *In Memory of J. R. Firth*. Longmans, London.

Haruno, M. and T. Yamazaki. 1996. High-performance bilingual text alignment using statistical and dictionary information. In *Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics*, pages 131-138, University of California, Santa Cruz, California,

Hofland, K. 1996. A program for aligning English and Norwegian sentences. In S. Hockey, N. Ide and G. Perissinotto, editors, *Research In Humanities Computing*, pages 165-178, Oxford University Press, Oxford.

Horton, D. 1998. Translation assessment: notes on the interlingual transfer of an advertising text. In *IRAL*, Vol. XXXVI/ 2, pages 95-119.

House, J. 1981. *A Model for Translation Quality Assessment*, Tübingen, Gunter Narr.

Hutchins, J. 2005. Towards a definition of example-based machine translation. In *MT Summit X. Workshop: Second Workshop on Example-Based Machine Translation*, pages 63-70, Phuket, Thailand.

Ide, N. 1999. Parallel translations as sense discriminators. In *SIGLEX99 Workshop: Standardizing Lexical Resources*, pages 52-61, Maryland.

Inkpen, D. and G. Hirst. 2002. Acquiring collocations for lexical choice between near-synonyms. *ACL 2002 Workshop on Unsupervised Lexical Acquisition*, Philadelphia.

Inkpen, D. and G. Hirst. 2006. Building and using a lexical knowledge-base of near-synonym differences. *Computational Linguistics* 32(2): 223-262.

Inkpen, D., O. Frunza, and G. Kondrak. 2005. Automatic identification of cognates and false friends in French and English. In *Proceedings of RANLP'05*, pages 251–257, Borovets, Bulgaria.

Isabelle, P. 1992. Bi-textual aids for translators. In *Screening Words: User Interfaces for Text, Proceedings of the 8th Annual Conference of the UW Centre for the New OED and Text Research*, Waterloo, Ont., available at http://rali.iro.umontreal.ca/Publications/urls /bi_textual_aids.ps

Isabelle, P., M. Dymetman, G. Foster, J-M. Jutras, E. Macklovitch, F. Perrault, X. Ren, and M. Simard. 1993. Translation analysis and translation automation. In *Proceedings of the Fifth International Conference on Theoretical and Methodological Issues in Machine Translation*, Kyoto, Japan.

Isabelle, P. and M. Simard. 1996. Propositions pour la représentation et l'évaluation des alignements de textes parallèles dans l'ARC A2. Rapport technique, CITI, Laval, Canada.

Isabelle, P. and K. Church. 1997. Preface. *Machine Translation* 12(1-2): 1-2.

Johansson, S. 1980. The LOB corpus of British English texts: Presentation and comments. *ALLC Journal*, 1(1): 25-36.

Johns, T. 1986. Microconcord: a language learner's research tool. *System* 14(2):151-162.

Jones, D. B. and H. Somers. 1995. Bilingual vocabulary estimation from noisy parallel corpora using variable bag estimation. In *JADT 1995: III Giornate Internazionali di Analisi Statistica dei Dati Testuali*, vol. I, pages 255-262, Rome.

Julapalli, M. and S. Dhond. 2003. Word alignment in bilingual parallel corpora. Available at http://nlp.stanford.edu/courses/cs224n/2003/fp/mohith/paper.pdf

Jutras, J-M. 2000. An automatic reviser: the TransCheck system. In *Proceedings of Applied Natural Language Processing*, pages 127-134, Seattle, WA.

Kaplan, R. M. 1987. Three Seductions of Computational Psycholinguistics. In P. Whitelock, M. McGee Wood, H. L. Somers, R. Johnson and P. Bennett, editors. *Linguistic Theory and Computer Applications*, 149–181. London: Academic Press, London. Reprinted in Dalrymple et al. (1995: 339–367).

Kaplan, R. M. 1989. The Formal Architecture of Lexical-Functional Grammar. In C. Huang and K. Chen, editors. *Proceedings of ROCLING II*, pages 3–18. Reprinted in Dalrymple et al. (1995: 7–27).

Kay, M. 1997. The proper place of men and machines in language translation. *Machine Translation* 12(1-2):3-23.

Kay, M., and M. Röscheisen. 1993. Text-translation alignment. *Computational Linguistics,* 19(1): 121-142.

Kay, P. and C. J. Fillmore. 1999. Grammatical constructions and linguistic generalization: the *What's X doing Y?* construction. *Language*. 75: 1-33.

Kennedy, G. 1998. *An Introduction to Corpus Linguistics*. Addison Wesley Longman, Harlow.

Koehn, P. 2005. Europarl: A parallel corpus for statistical machine translation. In *MT Summit 2005.*

Kondrak, G. and B. Dorr. 2004. Identification of confusable drug names: A new approach and evaluation methodology. In *Proceedings of the 20th International Conference on Computational Linguistics*, vol. II, pages 952–958, Geneva, Switzerland.

Kraaij, W., J.-Y. Nie. and M. Simard. 2003. Embedding web-based statistical translation models in cross-language information retrieval. *Computational Linguistics*, 29(3): 381-419.

Kucera, H. and W. Francis. 1967. *Computational Analysis of Present-Day American English*, Brown University Press, Providence.

Kucera, H. and W. Francis. 1979. *Brown Corpus Manual*, Brown University Press, Providence.

Kupiec, J. 1993. An algorithm for finding noun phrase correspondences in bilingual corpora. In *ACL93*.

Kupsch-Losereit, S. 1985. The problem of translation error evaluation. In C. Titford and A. E. Hieke, editors, *Translation in Foreign Language Teaching and Testing*, pages 169-179, Tübingen, Narr.

Larson, Mildred L. 1984. *Meaning-Based Translation: A Guide to Cross-Language Equivalence*, pages 489-490, University Press of America, Lanham.

Leech, G. 1991. The state-of-the-art in corpus linguistics. In K. Aijmer and B. Altenberg, editors, *English Corpus Linguistics, Studies in Honour of Jan Svartvik*, Longman, London/New York.

Leech G. 1992. Corpora and theories of linguistic performance. In J. Svartvik, editor, *Directions in Corpus Linguistics*. Mouton de Gruyter, Berlin.

Leech, G. 1993. 100 million words of English: A description of the background, nature and prospects of the British National Corpus project. In *English Today*, Vol. 9, No. 1, Cambridge University Press.

Lewandowska-Tomaszczyk, B., M. P. Oakes, and M. Wynne. 1999. Automatic alignment of Polish and English texts. In B. Lewandowska-Tomaszczyk and P. J. Melia, editors, *PALC'99: Practical Applications In Language Corpora*, pages 77-86, Peter Lang, Frankfurt.

Macklovitch, E., M. Simard, and P. Langlais. 2000. TransSearch. A free translation memory on the World Wide Web. In *Proceedings of LREC*, pages 1201-1208, vol. 3, Athens, Greece.

Macklovitch, E., G. Lapalme, and F. Gotti. 2008. TransSearch: What are translators looking for. In *AMTA'2008- The Eighth Conference of the Association for Machine Translation in the Americas*, pages 1-10, Waikiki, Hawaii.

Malouf, R. 2006. Review of Corpus Linguistics: Readings in a Widening discipline, Geoffrey Sampson and Diana McCarthy (eds.). *Computational Linguistics*, 32(1):153-155.

Manning, C. and H. Schütze. 1999. *Foundations of Statistical Natural Language Processing*, MIT Press, Cambridge, MA.

Martin, J., H. Johnson, B. Farley, and A. Maclachlan. 2003. Aligning and using an English-Inuktitut parallel corpus. In *Building and Using Parallel Texts: Data Driven Machine Translation and Beyond*, pages 115–118, Edmonton, Canada.

McArthur, T., editor. 1992. *The Oxford Companion to the English Language.* Oxford University Press, Oxford.

McEnery, A. M. and M. P. Oakes. 1996, Sentence and word alignment in the Crater project. In J. Thomas and M. Short, editors, *Using Corpora for Language Research*, pages 211-231, Longman, London.

Melamed, I. D. 1996. A geometric approach to mapping bitext correspondence. In *First Conference on Empirical Methods in Natural Language Processing (EMNLP'96)*, pages 1-12, Philadelphia, PA.

Melamed, I. D. 1999. Bitext maps and alignment via pattern recognition. *Computational Linguistics*, 25(1): 107-130.

Melamed, I. D. 2000. Models of translational equivalence among words. *Computational Linguistics,* 26(2): 221-249

Melby, A. 1981. A bilingual concordance system and its use in linguistic studies. In W. Gutwinski and G. Jolly, editors, *LACUS 8: the 8th Lacus Forum, Glendon College, York University, Canada, August 1981*, pages 541–554, Hornbeam Press, Columbia, SC.

Meyer C. F. 2002. *English Corpus Linguistics : An Introduction*, Cambridge University Press, Cambridge.

Mihalcea, R. and M. Simard. 2005. Parallel texts. *Natural Language Engineering*, 11(3):239-246.

Mindt, D. 1986. Corpus, grammar and teaching English as a foreign language. In G. Leitner, editor, *The English Reference Grammar: Language and Linguistics, Writers and Readers*, pages 125-139, Niemeyer, Tübingen.

Moore, R. C. 2002. Fast and accurate sentence alignment of bilingual corpora. In *Machine Translation: From Research to Real Users, 5th Conference of the Association for Machine Translation in the Americas*, pages 135-144, Springer-Verlag, Berlin.

Moore, R. C. 2004. Improving IBM word-alignment model 1. In *Proceedings of the 42nd Meeting of the Association for Computational Linguistics (ACL 2004)*, page 518-525, Barcelona, Spain.

Moore, R. C. 2005. A discriminative framework for bilingual word alignment. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing (HLT-EMNLP 2005),* pages 81–88, Vancouver, British Columbia, Canada.

Munteanu D. S, A. Fraser, and D. Marcu. 2004. Improved machine translation performance via parallel sentence extraction from comparable corpora. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics,* HLT-NAACL 2004, pages 265-272.

Murison-Bowie, S. 1993. *Micro-Concord Manual: an Introduction to the Practices and Principles of Concordancing in Language Teaching,* Oxford University Press.

Nagao, M. 1984. A framework of a mechanical translation between Japanese and English by analogy principle. In A. Elithorn and R. Banerji, editors, *Artificial and Human Intelligence,* pages 173-180, North-Holland, Amsterdam.

Nerbonne, J., L. Karttunen, E. Paskaleva, G. Proszeky, and T. Roosmaa. 1997. Reading more into foreign languages. In *Fifth Conference on Applied Natural Language Processing*, Washington, DC.

Neumann, S. and S. Hansen-Schirra. 2005. The CroCo project. Cross-linguistic corpora for the investigation of explicitation in translations. In *Proceedings from the Corpus Linguistics Conference Series, Corpus Linguistics 2005*, Birmingham, UK, vol 1, no. 1. Available online: http://www.corpus.bham.ac.uk/PCLC/cl-134-pap.pdf

Nida, E. A. 1964. *Toward a Science of Translating*. E.J. Brill, Leiden.

Och, F. J., D. Gildea, S. Khudanpur, A. Sarkar, K. Yamada, A. Fraser, S. Kumar, L. Shen, D. Smith, K. Eng, V. Jain, Z. Jin, and D. Radev. 2004. A smorgasbord of features for statistical machine translation. In *Proceedings. of the Human Language Technology Conference NAACL*, pages 161–168, Boston, MA.

Och, Franz J. and H. Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19-51.

Olohan, M. 2004. *Introducing Corpora in Translation Studies*. Routledge, London.

Orr, T. 2006. Introduction to the special issue: Insights from corpus linguistics for professional communication. In *IEEE Transactions on Professional Communication*, 49 (3): 213-216.

Palmer, H. 1933. *Second Interim Report on English Collocations*. Institute for Research in English Teaching, Tokyo.

Papineni, K., S. Roukos, T. Ward, and W. Zhu. 2001. BLEU: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, PA.

Pearson, K. 2003. Using parallel texts in the transla-tor training environment. In F. Zanettin, S. Bernardini and D. Stewart, editors, *Corpora in Translator Education*, pages 15-24, St Jerome, Manchester.

Pedersen, T. and N. Varma. 2002. K-vec++: Approach for finding word correspondences. Available online: http://www.d.umn.edu/ ~tpederse/parallel.html.

Pedersen, T. 2009. Text::NSP::Measures::2D::CHI::x2 - Perl module that implements Pearson's chi squared measure of association for bigrams. Available online: http://search.cpan.org/~tpederse/Text-NSP-1.03/lib/Text/NSP/Measures/2D/CHI/x2.pm.

Quirk, R. 1960. Towards a description of English usage. In *Transactions of the Philological Society*, pages 40-61.

Renouf, A. 1987. Corpus development. In J. M. Sinclair, editor, *Looking Up, An Account of the COBUILD Project in Lexical Computing and the Development of the Collins COBUILD English Language Dictionary*, pages 1-40.

Resnik, P. 1999. Mining the web for bilingual text. In *Proceedings of the 37th Meeting of ACL*, pages 527-534, College Park, MD.

Resnik, P. and N. A. Smith. 2003. The web as a parallel corpus. *Computational Linguistics*, 29 (3): 349-380.

Resnik, P. and D. Yarowsky. 1997. A perspective on word sense disambiguation methods and their evaluating. In *ACL-SIGLEX Workshop Tagging Texts with Lexical Semantics: Why, What and How*?, Washington.

Ribeiro, A., G. Dias, G. Lopes, and J. Mexia. 2001. Cognates alignment. In *MT Summit VIII, Machine Translation in the Information Age*, pages 287–292, Santiago de Compostela, Spain.

Romary, L., N. Mehl, and D. Woolls. 1995. The Lingua parallel concordancing project: managing multilingual texts for educational purposes. *Text Technology*, 5 (3): 206-220.

Russell, G. 1999. Errors of omission in translation. In *Proceedings of the Eighth International Conference on Theoretical and Methodological Issues in Machine Translation (TMI-99),* pages 128-138, Chester.

Sanchez-Villamil, E., S. Santos-Anton, S. Ortiz-Rojas, and M. L. Forcada. 2006. Evaluation of alignment methods for HTML parallel text. In *Advances in Natural Language Processing, Proceedings of FinTAL 2006, 5th International Conference on Natural Language Processing*, pages 280–290, Turku, Finland, LNCS 4139. Springer, Berlin.

Sankoff, D. 2001. Statistics in sociolinguistics. In R. Mesthrie, editor, *Concise Encyclopedia of Sociolinguistics*, pages 828-834, Elsevier.

Sankoff, D. and J. Kruskal, editors. 1983. *Time Warps, String Edits, and Macromolecules: the Theory and Practice of Sequence Comparison*, Addison--Wesley.

Sankoff, D. and G. Sankoff. 1973. Sample survey methods and computer-assisted analysis in the study of grammatical variation. In R. Darnell, editor, *Canadian Languages in their Social Context*, pages 1-64, Linguistic Research Incorporated, Edmonton.

Searle, J. 1972. Chomsky's revolution in linguistics. In G. Harman, editor, *On Noam Chomsky: Critical Essays*, University of Massachusetts Press, Amherst, MA.

Sellers, P. H. 1980. The theory and computation of evolutionary distances: pattern recognition. *Journal of Algorithms*, 1 (4): 359-373.

Simard, M., G. Foster, and P. Isabelle. 1992. Using cognates to align sentences in bilingual corpora. In *Proceedings of the Fourth International Congress on Theoretical and Methodological Issues in Machine Translation*, pages 67-81, Montreal, Canada.

Simard, M., G. Foster, and F. Perrault. 1993. TransSearch: A bilingual concordance tool. Technical report, Centre d'innovation en technologies de l'information, Laval, Canada.

Simões A. M. and J. J. Almeida. 2006. Combinatory examples extraction for machine translation. In *11th Annual Conference of the European Association for Machine Translation*, pages 27–32, Oslo, Norway.

Sinclair, J. M., editor. 1987a. *Collins COBUILD English Language Dictionary*, Harper Collins Publishers, London.

Sinclair, J. M., editor. 1987b. *Looking Up, An account of the COBUILD Project in Lexical Computing and the Development of the Collins COBUILD English Language Dictionary*, Collins ELT, London.

Sinclair, J. M. 1991. *Corpus, Concordance, Collocation*. Oxford University Press.

Sinclair, J. M. 1996. Preliminary recommendations on corpus typology. Technical report, EAGLES.

Sinclair, J. M. 2003. *Reading Concordances*. Longman, London.

Smadja, F., K. McKeown and V. Hatzivassiloglou. 1996. Translating collocations for bilingual lexicons: a statistical approach. *Computational Linguistics,* 22(1): 1-38.

Svartvik, J. and R. Quirk, editors. 1980. *A Corpus of English Conversation.* Gleerup, Lund.

Talbot, D. 2005. Constrained EM for parallel text alignment. *Natural Language Engineering*, 11(3):263–277.

Thorndike, E. L. 1921. *A Teacher's Wordbook.* Columbia Teachers College, New York.

Thorndike, E. L. and I. Lorge. 1938. *Semantic Counts of English Words.* Columbia University Press, New York.

Thorndike, E. L. and I. Lorge. 1944. *The Teacher's Word Book of 30,000 Word,* Columbia University Press, New York.

Tufis, D., A. M. Barbu and R. Ion. 2004. Extracting multilingual lexicons from parallel corpora. *Computers and the Humanities*, 38 (2):163-189.

Venugopal, A., S. Vogel, and A. Waibel. 2003. Effective phrase translation extraction from alignment models. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 319–326, Sapporo, Japan.

Véronis, J. 2000a. Alignement de corpus multilingue. In J-M Pierrel, editor, *Ingénierie des langues, Traité IC2-Série Informatique et SI.* Èditions Hermes Science, Paris.

Véronis, J. 2000b. From the Rosetta stone to the information society: A survey of parallel text processing. In J. Véronis, editor, *Parallel Text Processing: Alignment and Use of Translation Corpora*, pages 1-24, Kluwer Academic Publishers, Dordrecht, The Netherlands.

Wang, Y. and A. Waibel. 1998. Modeling with structures in statistical machine translation. In *Proceedings. of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics*, Montreal, Canada.

Wilss, W. 1982. *The Science of Translation.* Gunter Narr, Tübingen.

Wu, D. 1994. Aligning a parallel English-Chinese corpus statistically with lexical criteria. In *Proceedings of the 32nd Annual Meeting of the ACL*, pages 80–87, Las Cruces, NM.

Wu, J., K. Yeh, T. C. Chuang, W. C. Shei, and J. S. Chang. 2003. TotalRecall. A bilingual concordance for computer assisted translation and language learning, In *Companion Volume to the Proceedings of ACL*, pages 201-204.

Wu, J., T. C. Chuang, W. C. Shei, and J. S. Chang. 2004. Subsentential translation memory for computer assisted writing and translation. In *Proceedings of the 42nd Annual Meeting of Association for Computational Linguistics*, vol. 2004.

Zhao, B., K. Zechner, S. Vogel, and A. Waibel. 2003. Efficient optimization for bilingual sentence alignment based on linear regression. In *HLT-NAACL-WPT-03*, Edmonton. Canada.

Zhu, Q. 1989. A quantitative look at the Guangzhou petroleum English corpus. *ICAME Journal*, 13: 28–38.

Zhu, Q. 1991. *Word Frequency Book of Petroleum English*, Petroleum University Press, DongYing.

Zhu, Q. 1999. Automatic alignment of an English-Chinese novel text. Technical report, Berkeley Chinese language working group, University of California at Berkeley.

Zhu, Q. 2009. A corpus-based analysis of argument realization by preposition structures. *Natural Language Engineering*, 15(3): 379-414.

Zhu, Q., D. Inkpen and A. Asudeh. 2007. Automatic extraction of translations from web-based bilingual materials. *Machine Translation,* 21(3): 139–163.

Zipf, G. K. 1935. *The Psycho-biology of language: An Introduction to Dynamic Biology*, MIT Press, Cambridge, MA.