Indian Statistical Institute
Kolkata

M.Tech. (Computer Science) Dissertation

Bilingual Parallel Corpora Mining from the Web for Improving Hindi-English Machine Translation System

A dissertation submitted in partial fulfillment of the requirements
for the award of Master of Technology
in
Computer Science

Author:
Ravindra Kumar
Roll No: MTC 1304

Supervisor:
Dr. Utpal Garain
Computer Vision and
Pattern Recognition Unit

M.TECH(CS) DISSERTATION THESIS COMPLETION CERTIFICATE

Student : Ravindra Kumar (MTC1304)
Topic : Bilingual Parallel Corpora Mining from the Web for Improving Hindi-English Machine Translation System
Supervisor : Dr. Utpal Garain

This is to certify that the thesis titled "Bilingual Parallel Corpora Mining from the Web for Improving Hindi-English Machine Translation System" submitted by Ravindra Kumar in partial fulfillment for the award of the degree of Master of Technology is a bonafide record of work carried out by him under our supervision. The thesis has fulfilled all the requirements as per the regulations of this Institute and, in our opinion, has reached the standard needed for submission. The results embodied in this thesis have not been submitted to any other university for the award of any degree or diploma.

Utpal Garain

Date : 10^{th} *July*, 2015

Acknowledgements

I would like to thank my dissertation supervisor Dr. Utpal Garain for suggesting this topic to me in the first place, and for helping me with the challenges I have faced.

I would also like to thank Arjun Das, Abhisek Chakrabarty and all my classmates for always being willing to hold a discussion with me.

Abstract

Bilingual parallel corpora is used in many applications in Natural Language Processing (NLP) and beyond. Machine Translation System is a well known application to use bilingual parallel corpora. Our work presents a system to mine bilingual parallel corpora from the web. We first collected candidate sites that contain Hindi-English text by initially supplying Hindi-English language pair and a list of Hindi words to the system. Hindi-English parallel corpora is mined from these candidate websites. Although our system has space for improvements but the resultant parallel corpus is very accurate and good. We have not built a very big Hindi-English parallel corpus because our initial goal was to find an approach. We have shown the improvements in machine translation by this Hindi-English parallel corpus. Details of our Hindi-English parallel corpora, mined from the web till now are also given. In our system no manual efforts are required. Our system can be used to mine domain specific as well as general domain bilingual parallel corpora. As data are growing over the web with time, our system can be used to build larger parallel corpora in future.

Contents

Abstract **i**

List of Figures **iv**

List of Tables **v**

1 Introduction **1**
1.1 Background 1
1.2 Problem Statement 2
1.3 Previous Work 3
1.4 Contribution of This Work 4
1.5 Organization of The Thesis 5

2 Crawling of Government Websites **6**
2.1 Introduction 6
2.2 Crawling and Collecting URLs 6
2.2.1 Candidate sites gathering 7
2.2.2 Collecting URLs 7
2.3 Summary 8

3 Selection of the Pages **10**
3.1 Introduction 10
3.2 Selection Method 11
3.2.1 Bilingual URL Pair Selection 11
3.2.2 Downloading the Web Pages 12
3.2.3 Content Verification and Cleaning 12
3.3 Summary 13

4 Alignment of Sentences **14**
4.1 Introduction 14
4.2 Alignment Method 15
4.2.1 Sentence-Length-Based Alignment 15

4.2.2 Word Translation Model 15
4.2.3 Word-Correspondence-Based Alignment 16
4.3 Summary 17

5 Improving Statistical Machine Translation 18
5.1 Introduction 18
5.2 Experimental Protocol 18
5.2.1 Structure 19
5.2.2 Scoring Method 19
5.3 The Experimental Results 20

6 Conclusions and Future Work 21
6.1 Conclusions 21
6.2 Future Work 22

Bibliography 22

List of Figures

2.1 Crawling of Government Sites. 8

3.1 Selection of the Pages. 13

List of Tables

5.1 Results 20

Chapter 1

Introduction

Over the years in last two decade it is accepted by many researcher that bilingual parallel corpora is very useful and play an important role in the area of Natural Language Processing(NLP). Statistical Machine Translation(SMT), a sub domain of NLP is one of the best application where bilingual parallel corpora is used. SMT systems mainly depend on the bilingual parallel corpora apart from the translation models that they follow. An SMT system trained on a larger bilingual corpora gives batter translation since it learns the translation model parameters with a greater confidence. In the initial days of SMT, bilingual parallel copora were generated manually that required human resources or language experts. In the last few years many tried to find the ways to extract parallel corpora from the available multilingual text such as parliamentary proceedings. Some other think that web may be used as a big resource to mine bilingual parallel corpora. In our work we are mining a bilingual corpora from the web for improving Hindi-English SMT system.

1.1 Background

Warren Weavers (1949) first attempted at machine translation (MT) using mechanical approaches then onwards different MT techniques emerged over the time. Primarily these techniques can be classified as either rule-based or data-driven. Although these two approaches are very different but today they borrow many sub techniques from each other.
MT is a longstanding goal of computer science. A system which translate one language (source language) string to another language (target language) string, should know the information of both sides. The system should know word-to-word, phrase-to-phrase translations and grammar rules of two languages. There is a possible way to make this type of system where a language

expert, who knows source and target languages, can program the system using word-to-word, phrase-to-phrase synonyms and grammar rules. Although this type of systems are capable to give reasonably good translation but constructing this type of system is very time consuming and labour intensive. There is an another way if somebody can make the system which learn from the bilingual parallel corpora and monolingual corpora. A bilingual parallel corpora are documents that are translation of each other and a monolingual corpora is a document of its own language. This type of system uses a different framework and this approach is called statistical approach that comes under data-driven or corpus based approach. A system built by using this approach is called statistical machine translation system.
Statistical machine translation(SMT) is an approach to MT that is characterized by the use of machine learning methods [9]. This approach is very promising over the last two decade. This approach requires a bilingual parallel corpora and a target language monolingual corpora. These type of systems uses bilingual parallel corpora to learn the relationship between both the languages and monolingual corpus to learn the nature of target language. It is accepted by many that this approach is very promising in the domain of machine translation. This approach increase the importance of bilingual parallel corpora and this importance attracts many researcher. To make a monolingual corpus is relatively easy as compare to the making of bilingual parallel corpora. To construct a bilingual parallel corpora, many tried different-different approaches. There are many ways that can be used to construct a bilingual parallel corpora. Earlier Canadian government released parliamentary proceeding in English and French that was used to construct bilingual parallel copora. Later some tried to extract parallel sentences from some other such multilingual texts. Some other have proposed that web multilingual data may be used to construct the bilingual parallel corpora.

1.2 Problem Statement

A parallel corpus is defined as follows:

Definition: Bodies of text in parallel translation is called a parallel corpus, it is also known as bitext [16].

In tradition, bilingual parallel corpora are created by employing human resources (language experts). Some bilingual parallel text such as Canadian parliamentary proceedings are also used to build parallel copora. The ICON[1]

[1] `http://ltrc.iiit.ac.in/icon2015/index.php`

released Hindi-English parallel corpora is domain specific i.e. it only contains sentences related to health and tourism. If a Hindi-English parallel corpora can be mined from the Web than it will solve many problems. This type of system will not require language experts and the resulted corpora would be general in domain. The problem statement is described in brief as follows:

To mine bilingual parallel corpora from the web to improve Hindi-English machine translation system. Our primary aim is to mine Hindi-English parallel corpora. In order to do that we need to build a unsupervised system that will mine Hindi-English parallel sentences from the web.

1.3 Previous Work

There are many approaches that came into the light over the time in last decade. Fung and Cheung [7] and later Munteanu and Marca [11] focused on extracting parallel sentences out of comparable corpora. Bilingual dictionaries and machine learning techniques such as classification are also used to find the pairs of sentences of different languages. This approach requires language specific resources and gives positive results, but the amount of parallel content in comparable data is very low.
One propose to extract parallel sentences from mixed language web pages, *i.e.*, pages that have more than one languages [21] . The major issue in this approach is the scarcity of this type of web pages. Wikipedia data is also targeted to extract parallel content [19] . It have parallel content in a great amount at document level but parallel content decreases in case of not major languages, *e.g.* Bulgarian as shown in their article.
In last few years many worked on a common approach : By looking for bitext in the websites containing bilingual content, finding the parallel text documents and then aligning the bilingual text at sentence level. There are two ways, either parallel websites can be found [13][16][23] or source can be previously fixed [3]. Document URL similarity [13], inter-language links and HTML structure [16] are some different features that can be used for document level alignment. Some also give dictionary based approaches for document level alignment by comparing the content of documents [6] or extracting named entities that are language independent [12].
PaCo2[17] : A fully automated tool for the required job, can be considered to be from the last group. This approach presents two main problems, one is that some of the features e.g. documents URLs and inter-language links are dependent on the structure of website. These features become very attractive when they are applicable and result good recall and precision. The

second problem, the bottleneck of this approach is the large number of document comparisons that need to be made. When the different features are combined, it solved both the problems because it provides a fast method for the websites that fully follow some specific features and for the other cases it reduced the comparisons.
PaCo2 working pipeline have three main phases. Any phase of them can be used by the user according to the need. First phase process searches the web and find the websites which have the bilingual documents. The second phase find the parallel documents from each websites documents that were gathered in the first phase. In the last and third phase bitext is aligned at sentence level and finally the bilingual parallel corpora is built [17].
Our approach is similar to PaCo2 upto some extent but there are some differences. In the first phase of system pipeline, we are using only Google search engine (Google search API) to gather candidate websites, it produced good result in our case. In the second phase of system pipeline they have used different different approaches *i.e.* Link follower filter, URL pattern search and HTML structure and content filter, to find the parallel documents but we have only used a modified version of URL pattern search. In the sentence alignment phase, third phase of the system pipeline, they used Hunalign [5] tool and we used Bilingual Sentence Aligner [4][10]. The main purpose to use Bilingual Sentence Aligner is this tool uses an approach where it does not require any kind of anchor points in parallel text.

1.4 Contribution of This Work

Our work contributes in many ways in Natural Language Processing (NLP) since bilingual corpora is very useful in this domain. One of the best application to exploit the bilingual parallel corpora is statistical machine translation (SMT). We have found an approach to mine bilingual parallel sentences from the web. In our work, we have built a Hindi-English parallel corpora by mining the web. Our bilingual parallel corpora contains 6949 pairs of sentences, we randomly picked 200 sentences from this corpora and in these sentences we found 198 (99%) correct translated pairs of sentences. Although we have managed to collect less number of sentences but this approach can give large number of sentences when applied to large number of websites. Our works also help to make a general domain parallel corpora since a mixture of websites of different domains gives general domain sentences.

1.5 Organization of The Thesis

The pipeline of the system is explained in the upcoming three chapters (i.e. chapter 2, 3 and 4). In chapter 5 it is explained how our system improve Hindi to English machine translation system. Conclusion of the work and future work is explained in chapter 6 and references are mentioned in last of the thesis.

In chapter 2, Crawling of Government Websites, first phase of the systems pipeline is explained in detail. The motivation behind the crowing of government websites is discussed in brief in this chapter. We shall also talked about the tool used in the crowing of web and the challenges we faced in this phase of our system.

In chapter 3, Selection of the Pages, second phase of the system pipeline is explained in detail. In introduction of this chapter, different approaches to select parallel pages from a website are described in brief. In our selection method, we shall describe our assumptions and procedure of selection.The result, we have got with those assumptions are also described .Finally we shall summaries the approach of selection of parallel pages.

The third and the last phase of the system, Alignment of Sentence Pairs is explained in chapter 4. The alignment tool that we have used in our work is described briefly in introduction and the methodology used by the alignment tool is described in detail in rest of the chapter. In the last we shall talk about the results that have been given by the system in our experiment and then summaries the whole work of sentence alignment.

In chapter 5, Improving Statistical Machine Translation, improvements that have been achieved by the bilingual parallel corpus, that is mined from the web by using our system, in the statistical machine translation will be shown. In this chapter first we shall discus the machine translation software setup in brief. Secondly, we shall talked about the available bilingual parallel corpora and our bilingual corpora. The scoring method of the statistical machine translation is also discussed in brief. In the last, the improvement results will be shown and discussed. In the last chapter, Conclusion and Future Work will be discussed in brief. We shall discussed the overall success and the restrictions of the system. In last we shall also talked about the future thoughts about the work.

Chapter 2

Crawling of Government Websites

2.1 Introduction

As it is mentioned in the previous chapter there are many researcher targeted the web to mine bilingual parallel corpora. We are finding the Hindi-English bilingual parallel corpora so we need to find those websites that have Hindi content as well as English translation of that Hindi content.
It is empirically measured that many of the Indian government websites have the content in English as well as Hindi translation of the English content. Our motivation behind crawling the Indian government websites is that we assume these websites have enough bilingual text to build the bilingual parallel corpora. If somehow we are able to collect the available bilingual data in the form of bilingual parallel corpora it will replace the required human resources (language experts).
A big benefit of crawling the Indian government's websites is the heterogeneity of data. ICON release a Hindi-English bilingual parallel corpora. This corpora have only 50000 pair of sentences and this bilingual corpora have only health and tourism related data *i.e.* it is a domain specific bilingual parallel corpora. On the other hand mining bilingual parallel corpora from the web would be general in domain.

2.2 Crawling and Collecting URLs

This job can be divided in two major subsections:

- Candidate sites gathering [17]
- Collecting URLs

2.2.1 Candidate sites gathering

In this subsection the module to find the candidate sites is explained. The approach followed here is similar to the way used in the building of PaCo2 [17] tool. We first make a list of words of one language preferably one with less content available on the web, in our case we made Hindi word list. Google search API[1] can be used to collect the candidate sites. Now to form a query to search engine three words from list are combined, and resulted websites for these queries are collected. Although it return many useless URLs such as blog platforms and social networks. These type of results can be restricted by using hints for the domain names *i.e.* gov.in , ac.in , org.in etc. In our work we used Hindi word list with gov.in and ac.in.

2.2.2 Collecting URLs

In this subsection the approach of collecting URLs for a given website is explained. There are different ways to do this job. One approach to do this, is all HTML pages are downloaded for the given website by using wget[2] and then URLs are listed out. This approach is space exhaustive as well as time exhaustive because all downloaded HTML pages might not be useful. The system requires only HTML pages that contains Hindi content and English translation of Hindi pages content *i.e.* HTML pages that form parallel pairs of Hindi English content.
Another approach to meet the goal is to use sitemap[3] generator tool. This approach comparably performs well as per the requirement of our system. A brief introduction of the sitemap generator tool is as follows:

Sitemap Generator: A sitemap is list of pages of a website available for a user or crawler. It can be in any form of a document which is supported by the search engine at which you want to put your website. Sitemap generator is a tool to generate sitemaps. In our work we are using an open source sitemap generator *sitemap_gen.py* [20] written by Vladimir Toncar.
This sitemap generator tool is a platform-independent software written in python. If you run this software with a website URL, it generates a sitemap

[1] `https://developers.google.com/web-search/docs/`
[2] `http://www.gnu.org/software/wget/`
[3] `http://www.sitemaps.org/`

starting from the given URL in XML format. The domain name of a website is given to the sitemap generator and it generates the sitemap for the corresponding website. This file contain all website pages URLs that are available for crawling.

URLs with specific extension can be excluded and the maximum number of URLs in sitemap can be restricted by the configuration of the tool [20]. As we required only web text contents embedded in web pages, in our system we excluded all URLs having extension .png, .jpg, .avi, .pdf and for the termination of long crawling, we limited maximum number of URLs to 5000. Now using a simple XML parsing program, all URLs can be extracted from a given XML file. We listed all the active pages URLs in a separate file for each candidate site. The whole process described in this chapter is shown in the following figure:

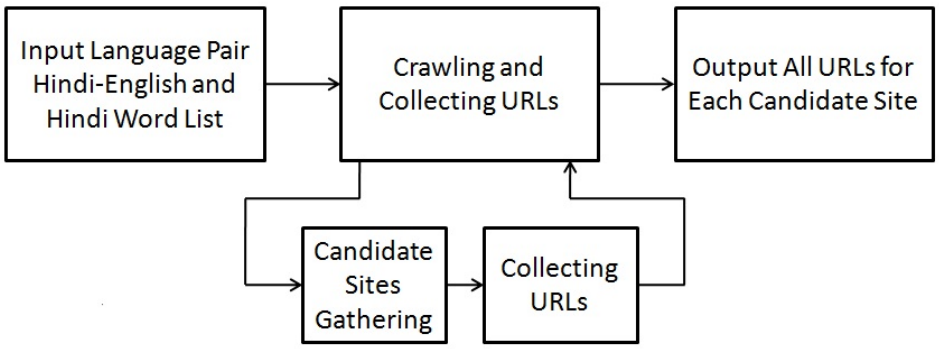

Figure 2.1: Crawling of Government Sites.

2.3 Summary

As compared to the work followed in the building PaCo2 we left the parallel site detection in this phase of our system. This system does not care about Alt Texts , Interlingual Links, Combo Structures and Page Language [17]. We can consider these techniques in future improvements of the system. This system only play with URLs and highly rely on them.
Using Google search API it is possible to find the candidate sites that contains bilingual content. In our system experiment we found about 1000 unique candidates sites. All the candidate sites are belonging to Indian government

websites including websites of schools, institutes, state governments and other government organizations. These all websites have Hindi-English parallel text with the higher probability.

Crawling of the candidate sites is done using an open source software *sitemap_gen.py*. It helps the system to crawl the candidate site to collect the active pages URLs available on the given candidate site in the form of sitemap in XML. A simple parsing program is used to parse the XML file. In the end of this phase, our system produces all the active pages URLs for each candidate site.

Chapter 3

Selection of the Pages

3.1 Introduction

When all URLs of a candidate site are in hand then system attempts to find the pair of bilingual parallel pages URLs. This job is described including the brief description to useful tools in this chapter. Regarding this problem authors came with many solutions. There are three main approaches Link follower filter, URL pattern search and HTML structure and Content filters [17]. First two of these are highly dependent on the structure of the website and time efficient in processing . We skipped first approach because we found some of Hind-English bilingual websites using automatic English to Hindi translator. This type of bilingual Hindi-English content will degrade the quality of the resulted bilingual corpora. On the other hand we also skipped the third approach based HTML structure of web pages for our works because it is little more time consuming. Although If this technique is also used in this system it will increase the recall.
In our system only a restricted version of second approach *i.e.* URL pattern search is followed to find the parallel pairs of pages. Our approach is also similar to one that is used by Ying Zhang *et. al.*, they call it candidate pair extraction [23]. In the previous chapter, we have used the name "candidate websites" to point the websites, we choosed to call these URl pairs as bilingual URL pairs. After finding the bilingual URLs pairs all the parallel pages can be downloaded by using wget[1] tool. Paragraph text of downloaded web pages will be extracted, verified and put in the parallel pages text format. The detailed description of these steps will be explained in rest of this chapter.

[1] `http://www.gnu.org/software/wget/`

3.2 Selection Method

Selection method is one of the most important method that is highly responsible for the recall and precision. This work can be divided into subsections given as the following:

- Bilingual URL Pair Selection
- Downloading the Web Pages
- Content Verification and Cleaning

3.2.1 Bilingual URL Pair Selection

A URL can be break up in parts such as protocol prefix, domain name, path name and file name. Web-masters tend to name the pages similar if they are translation of each other [23]. We assume that Hindi pages of many websites are put in the separate directories. Although there are many possibilities of the URL structure for a candidate site and it is up to the web-master. But one structure we noticed that many of the website have only a minor difference *i.e.* Hindi page URL have an extra key substring. These extra key substring are not available in corresponding English page URL.
Generally these key substring are related to the language name *i.e.* for Hindi these segments may be "hi", "hin", "hindi", "Hi", "Hin", "Hindi", we use a general term **key_substring** to represent an instance of one of these. Some of the general URL of this nature can be described as:

http://www.AAAA.com/bb/cc/**key_substring**/dd.html
http://www.AAAA.com/bb/**key_substring**/dd.html
http://www.AAAA.com/bb/cc/**key_substring**/ee/dd.html

We remove the **key_subtring** to drive corresponding English content page URL. It seems very straightforward and simple but it gave us a few thousands of valid pair of pages. It is also noted that some of the derived English page URLs are wrong, our system skipped all those pairs. It is also found that key substring may be in the form of "Key_substring/.*/key_substring", we also modified our system to deal with this type of URLs. Bilingual URL pairs can be validated using curl[2] tool. Now we have bilingual URL pairs corresponding to Hindi-English parallel pages.

[2] `http://linux.about.com/od/commands/l/blcmdl1_curl.htm`

3.2.2 Downloading the Web Pages

Downloading of the web pages is done using wget tool. This is the most time consuming sub step of of this phase. There are many website that have high security and does not allowed for downloading. For each candidate site all the valid bilingual URL pair are put in two separate text files and using wget tool all parallel pages are dumped.

3.2.3 Content Verification and Cleaning

When the bilingual parallel pages are in hand, we need to verify the content. It is done by using the char-set because both languages have different char-sets. The procedure from verification to getting the plain text in parallel files can be discussed as the following:

- **Web pages parsing**: There are many libraries available to this task. In our work we used jsoup[3] java library. It is a Java library for working with real-world HTML. It provides a very convenient API for extracting and manipulating data, using the best of DOM, CSS, and jquery-like methods[3]. Using jsoup library we only extract paragraph text because maximum of the text data, that is useful for the bilingual parallel corpora, available in a web pages are in the form of paragraph text.

- **Language content verification**: Each language script has a specific char-set and a specific unicode range corresponding to its char-set. For example Devanagari script unicode range is 0900 to 097F[4] in hexadecimal, it can be used to verify the content of the Hindi web pages. On the other hand to verify the English content we used ascii values of English alphabets.

- **Cleaning and paragraph splitting**: In the end of previous step we had verified bilingual parallel pages that have text in the form of paragraphs. Now paragraphs have to be splitted into sentences. For english language we used PTBTokenizer[5] tool of Stanford Parser to split paragraphs into sentences. PTBTokenizer is a an efficient, fast, deterministic tokenizer[5]. On the other hand for Hindi we wrote our own simple ruled base tool to split paragraph into sentences. Hindi sentences end boundaries are assumed as '—' and '?'.

[3] `http://jsoup.org/`

[4] `http://unicode.org/charts/PDF/U0900.pdf`

[5] `http://nlp.stanford.edu/software/tokenizer.shtml`

At the end of these sub steps we have verified and cleaned sentences in bilingual parallel pages. The following figure shows the whole process described in this chapter:

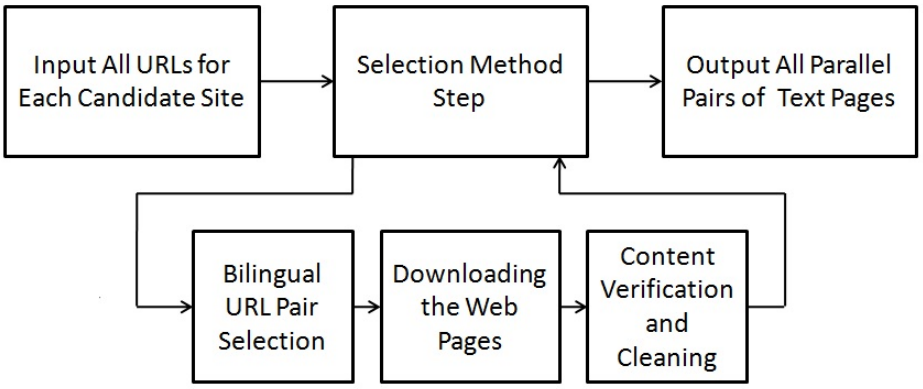

Figure 3.1: Selection of the Pages.

3.3 Summary

In our experiment of the system we have found around 3000 pairs of HTML pages. We have assumed that maximum text of any language in the HTML web pages are surrounded by paragraph tags. As our goal is to mine sentences, we have considered only paragraph text. Firstly, data is to be extracted from the HTML pages, that is done by using *jsoup* library. Secondly, it is required to verify the content language, that is done by using the unicode char-set range. In the last, parallel pages text has been verified and paragraph text splitted into sentences. When we look at the samples of pages we found that some pair of pages are very bad *i.e.* they do not have translation of each other, on the other hand some pair of pages are exactly translation of each other. It can also be seen that many pages of a particular website have some common data.

This phase of the system pipeline can be improved to increase the recall. In our system we are not able to find all the parallel pages of websites. This is a drawback of our system's selection method. Although our page selection method is only follow URL pattern search but it gives fairly good result and it is one of the fast approach.

Chapter 4

Alignment of Sentences

4.1 Introduction

Sentence-aligned parallel bilingual corpora have been proved very useful for applying machine learning to machine translation [10] but generating them in sentence-aligned form is not a trivial task. This non trivial task attracts many researcher and many of them introduce different-different approaches to solve this problem. The ideal approach to solve this problem should be fast, highly accurate and require no domain specific knowledge of both languages.
All the previous approaches can be divided into two major classes one is sentence length based approaches and the other is word-correspondence-based models. Some of the recent approaches combined these both to perform fast and accurate. The first effective approach for the large corpora was based on sentence length based and similar type of approaches were developed by Gale and Church [8] and Brown *et. al.* [1]. Melamed [18] method was based on word correspondence which gives the accuracy slightly better than Gale and Church. Wu [22] used a small corpus specific bilingual lexicon to increase the accuracy of a version of Gale and Church method. The purely length based approaches require corpus-dependent anchor points or paragraph boundaries to restrict the search space but no special knowledge about concerned languages. On the other hand word-correspondence-based model requires either language based bilingual lexicon or they depend on finding cognates in the two languages to suggest the word correspondences.

We are using a method described by Robert C. Moore as Fast and accurate sentence alignment of bilingual corpora [10]. This method is the combination of these two approaches defined so far. This hybrid approach is relatively faster, accurate and does not require any domain specific information about

the concerned languages.

4.2 Alignment Method

The alignment algorithm is hybrid which combines both length based and word-correspondence-based techniques. Three major steps of the algorithm are briefly described in the following subsection.

4.2.1 Sentence-Length-Based Alignment

It was assumed by Brown *et. al.* that every parallel corpus can be aligned in terms of a sequence of minimal alignment segments.These segments are called beads and in these beads sentences aligned 1-to-0, 0-to-1, 1-to-1, 1-to-2 or 2-to-1.
The model assumes that each bead in the sequence is generated by using a fixed probability distribution over bead types, and each bead has a sub model to generate length of the sentences forming that type of bead. In 1-to-0 and 0-to-1 bead types, there is only one sentence in each bead and it's length is generated according to the observed sentence length distribution in corresponding language text. In 1-to-1, 1-to-2 and 2-to-1 type of beads, length(s) of source language sentence(s) is generated according to the distribution defined for 1-to-0 case and the total length of target sentence(s) is assumed according to a model conditioned on the total length of sentence(s) of source language in that bead. Let l_t is the length of sentence(s) of target language, l_s is the length of corresponding sentence(s) in source language and r is the ratio of mean sentence length in target language to mean sentence length in source language. Then the model assumes that l_t varies according to the Poisson distribution with mean l_s times the ratio r:

$$P\big(l_t \mid l_s\big) = \frac{\exp\big(l_s r\big)\big(l_s r\big)^{l_t}}{l_t!} \tag{4.1}$$

This step of algorithm is used by system to find the high probability 1-to-1 beads to train the word-translation model that is described in next step. These beads are found by performing backward-forward probability computation [15].

4.2.2 Word Translation Model

In this phase of algorithm highest probability 1-to-1 beads are used to train a word-translation model. The threshold of .99 probability of correct alignment

is used for this word-translation model. This threshold ensure the reliability of the model. In this word-alignment model a modified version of well known IBM Model 1 [2] is used.
Let the source sentence of length s contains l words s_1, s_2, ..., s_l ,then target language sentence t is generated in the IBM translation models is as follow: Target sentence length m is picked and for each word position in t a word from $l + 1$ words (where l words comes from s and one extra word is the null word s_0) is selected. Now to fill the each position is target sentence t, a target language word is selected for each pair of position in t and its generating word in s. It is assumed in this model that all possible t length are bounded to an arbitrary upper bound and have a uniform probability q ; all possible choices of source language generating words are equally likely and the probability $tr(t_j|s_i)$ of the generated target word are only depends only on generating source language words. This model is defined as follow:

$$P\big(t \mid s\big) = \frac{\epsilon}{\big(l+1\big)^m} \prod_{j=1}^{m} \sum_{i=0}^{l} tr\big(t_j \mid s_i\big) \tag{4.2}$$

Two minor modification are done in this existing model. The translation probabilities are removed for rare words, it would not make a much loss. The threshold each language is set in such a way that each language should contain 5000 unique words conditioned to minimum 2 occurrence per word. The second modification is related to accumulating the fractional counts for word translation pairs after first iteration of EM. In a given sentence,if the accumulating fractional count for a word translation pair is not greater than the one obtained by using a fully random choice then this count would not be added to the count of that particular translation pair.

4.2.3 Word-Correspondence-Based Alignment

In the final alignment model, a modified version of first length based model is used along with the IBM model 1. In this modified model it is assumed that the bead types and the sentence lengths are generated using the same probability distribution as used in sentence length-based model. The probability for the sequence of words in each bead can be estimated using the instance of IBM model 1 that is estimated from the initial alignment. These probability estimates are multiplied to the probability estimates based on bead types and sentence length.
There are five types of beads in our consideration i.e. 1-to-0 , 0-to-1 , 1-to-1 , 1-to-2 and 2-to-1. For 1-to-0 and 0-to-1 each word is generated independently according to the observed relative frequency f_u of that word in correspond-

ing language corpus. For 1-to-1 , 2-to-1 and 1-to-2 beads, a word in source language sentence(s) is generated independently by the same way as word generated in case of 1-to-0 beads and a word in target language sentence(s) is generated depending on words of source language sentence(s) according to the probability estimates in the instance of IBM model 1. When the IBM model 1 is applied in this task the assumption of having a uniform distribution on the target sentence length because the sentence length is considered in the first length-based model.

Let $P_{1-1}(l, m)$ is the probability given by the initial model to the 1-to-1 alignment of the a source sentence s of length l and the target sentence t of length m. Then the combined model that will estimate the probability of 1-to-1 bead as:

$$P(s,t) = \frac{P_{1-1}(l,m)}{(l+1)^m}\left(\prod_{j=1}^{m}\sum_{i=0}^{l} tr(t_j \mid s_i)\right)\left(\prod_{i=1}^{l} f_u(s_i)\right) \tag{4.3}$$

This is the final model used to give the final alignment of the sentences. The beads that passed the threshold of the probability (probable 0.5 to 0.9) are selected in the final selection and included in the result.

This alignment method explained so far in this section is already implemented as Bilingual Sentence Aligner [10]. So we have not implemented it separately. In our work Bilingual Sentence Aligner produce a very accurate bilingual corpora. To check the accuracy, we randomly picked 200 sentence pairs and found accuracy of 99%.

4.3 Summary

Although in our work it was not possible to check the recall and precision error in the alignment of the bilingual parallel corpora because the parallel bilingual text at page level is coming from the web and this web data is totally unknown to us. But using this alignment method we have found almost perfect alignment.This method does not require anchor points and any type of language based lexicon, these features make it more attractive. This hybrid method is the combination of sentence-length-based and the word-correspondence-based model, and the minor modification used in this method makes it faster and space effective, at the less cost, in comparison to the other methods described in the previous works for the same job.

Chapter 5

Improving Statistical Machine Translation

5.1 Introduction

As it was told in the very first chapter that bilingual parallel corpora plays a vital role in statistical machine translation (SMT), the improvement in the Hindi-English machine translation system will be shown in this chapter. Some of the well known SMT tools i.e. mosesdecoder[1], GIZA++[2], IRSTLM[3] are used for the experiments. We will briefly discuss about these tools. Firstly, the structure of experiment will be discussed and then BLEU [14] score of each system will be compared with each other. We shall compare our Hindi-English parallel corpora with ICON Hindi-English parallel corpora. We also shall combine web and ICON corpora to check the improvements.

5.2 Experimental Protocol

The experiments to check the quality of the bilingual parallel corpora is done by training a SMT system by using our bilingual corpora. The experimental protocol is described in two sections as follows:

- Structure
- Scoring Method

[1] `http://www.statmt.org/moses/`
[2] `https://github.com/moses-smt/giza-pp`
[3] `http://sourceforge.net/projects/irstlm/`

In the first section, we shall describe the required tools and the bilingual corpora divisions for training sets, development sets and testing sets. On the other hand in second section scoring metric will be explained.

5.2.1 Structure

To build an SMT system, we used some open source tools, these tools are freely available for experiments. These tools are described in brief as follows:

- **Moses**: Moses is a SMT system that allows you to automatically train translation models for any language pair. All you need is a collection of translated texts (parallel corpus)[1].
- **GIZA++**: GIZA++ is a statistical machine translation toolkit that is used to train IBM Models 1-5 and an HMM word alignment model[2].
- **IRSTLM**: A tool for the estimation, representation, and computation of statistical language models[3].

The division of the bilingual parallel corpora into training set , development set and the testing set can be given as the following:

- **Web bilingual parallel corpora**: 5400 pairs of sentences for training, 1000 pairs of sentences for development (tuning) and 549 pairs of sentences for testing are used. All of them are taken randomly. Our full bilingual parallel corpora have 6949 pairs of sentences.
- **ICON bilingual parallel corpora**: 48000 pairs of sentences for training, 1000 pairs of sentences for development (tuning) and 1000 pairs of sentences of testing are used. ICON full bilingual parallel corpora have 50,000 pairs of sentences.

5.2.2 Scoring Method

To check the contribution of our web bilingual parallel corpora in the SMT systems, we need to check the performance of these systems. BLEU [14] a method for automatic evaluation of machine translation, is used to evaluate the SMT systems. A brief description of this metric is given as the following:

BLEU

Let c be the length of candidate translation [14] and r be the effective reference [14] corpus length then the brevity penalty BP [14] is calculated as:

$$BP = \begin{cases} 1, & \text{if } c > r; \\ \exp(1 - r/c), & \text{otherwise.} \end{cases} . \tag{5.1}$$

Then, $BLEU$ is calculated as:

$$BLEU = BP \cdot \exp\left(\sum_{n=1}^{N} w_n \cdot \log p_n\right). \tag{5.2}$$

where w_n are the positive weights summing to one and p_n is the n-gram precision [14]. Our system is using $N = 4$ and $w_n = 1/N$ for all values of n.

5.3 The Experimental Results

To compare the results we have built three systems trained by using web bilingual corpora, ICON corpora and combination of both respectively. To test translation of these systems, we only used the web bilingual parallel corpora sentences because these are general in domain as per our assumption. As described in subsection 5.2.1 we randomly partitioned the web bilingual parallel corpora two times to form two different-different training sets, development sets and testing sets. We call these sets set1 and set2 and the experiments corresponding to them Experiment1 and Experiment2. ICON Corpora SMT System, Web Corpora SMT System and Combined Corpora SMT System are trained by using ICON corpora, web corpora, and combination of both respectively. The $BLEU$ score multiply by 100 of each system is shown in the following table:

	ICON Corpora SMT System	Web Corpora SMT System	Combined Corpora SMT System
Experiment1	05.42	09.31	10.62
Experiment2	05.58	08.03	09.38

Table 5.1: Results

It can be seen that system trained on web parallel corpora gave batter BLEU score in comparison to system trained on ICON parallel copora and it also improves the BLEU score when both are combined. Although we have not done a k-fold validation of the corpora evaluation, but as it shown in the the table both experiments perform very similar it shows that web bilingual corpora improved the system.

Chapter 6

Conclusions and Future Work

6.1 Conclusions

We have built an approach for mining bilingual parallel corpora from the web to improve Hindi-English MT system. Although we have used some existing algorithms and tools as per our requirement for our work but our work makes clear that web can be used for mining the bilingual parallel corpora.
Mining a bilingual parallel corpora from the web is a very time efficient as compare to the traditional ways. The longest time taken part of the system is downloading the web pages. Since we have to download all pages regardless they contain parallel text or not. Another non trivial work is to alignment the sentences because it is difficult to know about the data in parallel pages. For a really bad data a common alignment approach does not work batter. The main reason behind getting the less number of sentence pairs is that we have reached to less number of parallel pages. It is observed that the sentence aligner works very well and gives accuracy close to 100% on large data.

We can crawl more websites to get more sentence pairs. It requires high bandwidth Internet because downloading the web pages is most time consuming sub step in the system. Although our system improves Hindi-English SMT system at a small level but the bilingual parallel corpus is almost accurate. Our web bilingual parallel corpora contains 6949 pair of sentences. The correctness of the parallel corpora attract us to make improvements in the system to get more parallel pages to build a larger parallel corpora.

6.2 Future Work

As we discussed in the last section about the system and it's results, it is clear that we have an approach for mining a bilingual parallel corpora from the web. But there are many steps where we can improve our system for batter results. Here is the list of future works and challenges:

- Improvements can be done in page selection step. We are only using URL pattern matching approach so far but in future we will approach some other machine learning techniques such as clustering of parallel pages.
- Alignment approach used in our system is works well for fairly parallel content. In web pages some times data is not fairly parallel. Improvements in alignment algorithm in this regards would be appreciable.
- Existing alignment algorithm prunes some pairs of parallel sentences. We can improve the algorithm in such a way that these pruned parallel pair of sentences can be extracted in the second iteration, by using first iteration resulted parallel corpora for word-correspondence model.
- A fully automated tool for mining bilingual parallel corpora from web can be designed.

Bibliography

[1] Peter F Brown, Jennifer C Lai, and Robert L Mercer. Aligning sentences in parallel corpora. In *Proceedings of the 29th annual meeting on Association for Computational Linguistics*, pages 169–176. Association for Computational Linguistics, 1991.

[2] Peter F Brown, Vincent J Della Pietra, Stephen A Della Pietra, and Robert L Mercer. The mathematics of statistical machine translation: Parameter estimation. *Computational linguistics*, 19(2):263–311, 1993.

[3] Jisong Chen, Rowena Chau, and Chung-Hsing Yeh. Discovering parallel text from the world wide web. In *Proceedings of the second workshop on Australasian information security, Data Mining and Web Intelligence, and Software Internationalisation-Volume 32*, pages 157–161. Australian Computer Society, Inc., 2004.

[4] Microsoft Reseach Corporation. Bilingual sentence aligner. `http://research.microsoft.com/en-us/downloads/aafd5dcf-4dcc-49b2-8a22-f7055113e656/`. Online; accessed 19-May-2015.

[5] L Nmeth D Varga, A Kornai P Halcsy, and V Nagy V Trn. Hunalign-sentence aligner. `http://mokk.bme.hu/en/resources/hunalign/`. Online; accessed 19-May-2015.

[6] Ken'ichi Fukushima, Kenjiro Taura, and Takashi Chikayama. A fast and accurate method for detecting english-japanese parallel texts. In *Proceedings of the Workshop on Multilingual Language Resources and Interoperability*, pages 60–67. Association for Computational Linguistics, 2006.

[7] Pascale Fung and Percy Cheung. Multi-level bootstrapping for extracting parallel sentences from a quasi-comparable corpus. In *Proceedings of the 20th international conference on Computational Linguistics*, page 1051. Association for Computational Linguistics, 2004.

[8] William A Gale and Kenneth W Church. A program for aligning sentences in bilingual corpora. *Computational linguistics*, 19(1):75–102, 1993.

[9] Adam Lopez. Statistical machine translation. *ACM Computing Surveys (CSUR)*, 40(3):8, 2008.

[10] Robert C Moore. *Fast and accurate sentence alignment of bilingual corpora*. Springer, 2002.

[11] Dragos Stefan Munteanu and Daniel Marcu. Improving machine translation performance by exploiting non-parallel corpora. *Computational Linguistics*, 31(4):477–504, 2005.

[12] David Nadeau and George Foster. Real-time identification of parallel texts from bilingual newsfeed. *CLINE 2004, Computational Linguistics in the North East*, 2004.

[13] Jian-Yun Nie, Michel Simard, Pierre Isabelle, and Richard Durand. Cross-language information retrieval based on parallel texts and automatic mining of parallel texts from the web. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 74–81. ACM, 1999.

[14] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics, 2002.

[15] Lawrence R Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

[16] Philip Resnik and Noah A Smith. The web as a parallel corpus. *Computational Linguistics*, 29(3):349–380, 2003.

[17] Inaki San Vicente and Iker Manterola. Paco2: A fully automated tool for gathering parallel corpora from the web. In *LREC*, pages 1–6, 2012.

[18] Michel Simard and Pierre Plamondon. Bilingual sentence alignment: Balancing robustness and accuracy. *Machine Translation*, 13(1):59–80, 1998.

[19] Jason R Smith, Chris Quirk, and Kristina Toutanova. Extracting parallel sentences from comparable corpora using document level alignment.

In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 403–411. Association for Computational Linguistics, 2010.

[20] Vladimir Toncar. Site map generator. `http://toncar.cz/opensource/sitemap_gen.html`. Online; accessed 19-May-2015.

[21] Masao Utiyama, Daisuke Kawahara, Keiji Yasuda, and Eiichiro Sumita. Mining parallel texts from mixed-language web pages. *Proceedings of the XII Machine Translation Summit*, 2009.

[22] Dekai Wu. Aligning a parallel english-chinese corpus statistically with lexical criteria. In *Proceedings of the 32nd annual meeting on Association for Computational Linguistics*, pages 80–87. Association for Computational Linguistics, 1994.

[23] Ying Zhang, Ke Wu, Jianfeng Gao, and Phil Vines. Automatic acquisition of chinese–english parallel corpus from the web. In *Advances in Information Retrieval*, pages 420–431. Springer, 2006.