Automated Construction of Arabic-English Parallel Corpus

Dr. Mohammed M. Sakre
Al Shorouk Academy
High Institute for Computers and Information Systems
m_sakre2001@yahoo.com

Prof. Dr. Mohammed M. Kouta
Arab Academy for Science, Technology & Maritime Transport
College of Computing and Information Technology
mmkouta2004@yahoo.com

Ali M. N. Allam
Arab Academy for Science, Technology & Maritime Transport College of Computing and Information Technology
Aliallam13@ hotmail

Abstract: *Large-scale parallel corpus has become a reliable resource to cross the language barriers between the user and the web. These parallel texts provide the primary training material for statistical translation models and testing machine translation systems. Arabic-English parallel texts are not available in sufficient quantities and manual construction is time consuming. Therefore, this paper presents a technique that aims to construct an Arabic-English corpus automatically through web mining. The proposed technique is straightforward, automated, and portable to any pair of languages.*

Keywords: Cross language information retrieval, parallel corpus construction, web mining, parallelism matching.

1. Introduction

Cross-language information retrieval (CLIR) is a subfield of information retrieval dealing with retrieving information written in a language different from the language of the user's query, in order to cross the language barriers. According to [7,15] 56% of the web contents are in English while the English web population does not exceed 35%. Therefore, CLIR has many useful applications. For example, multilingual searchers might want to issue a single query to a multilingual

collection. Also, searchers with a limited active vocabulary, but good reading comprehension in a second language (e.g. English), might prefer to issue queries in their most fluent language (e.g. Arabic) [18]. There are three main approaches to bridge the language gap between the web and users: Parallel and comparable corpus, machine-readable dictionaries and machine translation [1,14,16].

In corpus-based methods, translation knowledge is derived from multilingual text collections using various statistical methods. Therefore, large-scale parallel corpus plays an important role in cross-language information retrieval (CLIR) by providing the primary training data for statistical translational models [2].

The main obstacle is that English-Arabic parallel corpora are not available in sufficient quantities. Therefore, most previous work on parallel texts has been conducted on a few manually constructed parallel corpora such as Canadian Hansard Corpus and Linguistic Data Consortium (LDC) [9]. However, manual collection of large corpora is a tedious task, which is time and resource consuming. Therefore, the main objectives of the proposed technique are twofold:

1. Constructing the parallel corpus automatically through web mining.
2. Preparing the constructed corpus to be used as training material for the translation model.

The proposed technique will use English-Arabic parallel texts to construct the corpus. However, the technique can be easily applied to other language pairs in a very similar way.

2. Proposed System Architecture

English-Arabic parallel texts are collected using web mining, mainly from news websites, to construct an English-Arabic parallel corpus. First, a host crawling is performed on a specified domain, and thus all pages of the desired language pair are downloaded from that domain. The system extracts the language of the page from its URL if possible; otherwise, a simple language detector is required. Second, some rules are defined to quickly reject all false pages in order to create a set of candidate pairs. Finally, content-based matching is performed to calculate the parallelism similarity between each candidate pair using an English-Arabic dictionary to determine whether it is a match or not.

The output of this technique is an English-Arabic parallel corpus that is well-aligned at paragraph level with completely clean texts. The mining system architecture is illustrated in Figure (1). This approach is straightforward, fully automated, and easy to port to any other pair of languages.

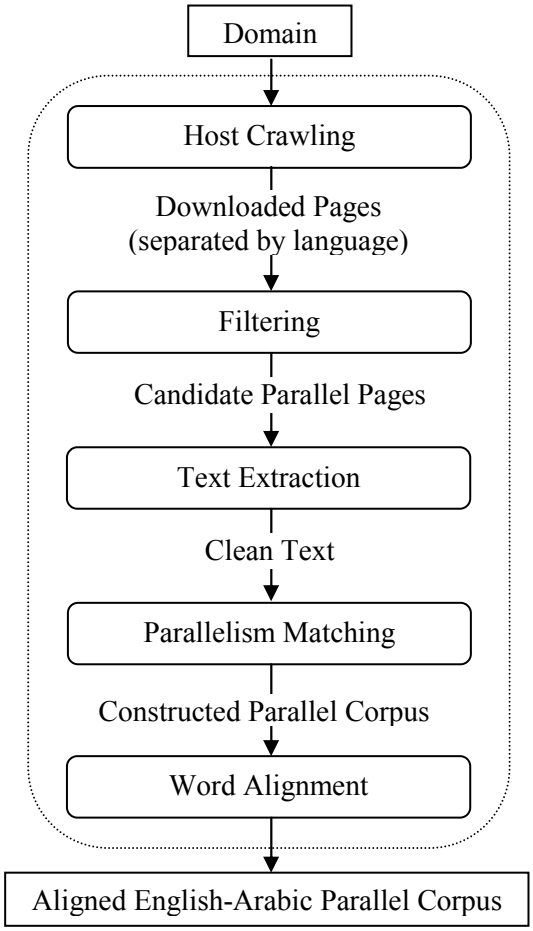

Figure 1: Proposed Mining System Architecture

2.1. Host Crawling

The crawling process is similar to that of "PTMiner" developed by Kraaij et al. (2003) [11]. Starting from a given URL on a specified host, the system crawls the host for all pages that are written in either Arabic or English. Webmasters usually keep parallel pages in different directories with respect to the name of the language. For example, (…/Arabic/…), (…/Ar/…), and (…/Ar_file.html) are more likely to be in Arabic, and the same phenomenon is observed for English pages. Therefore, a list of patterns is

used so that those pages in languages other than English and Arabic are rejected, without downloading them. This list of patterns can be easily modified to work with other language pairs. For instance, ("Fr", "French", and "Francais") is a list of patterns used for the French language.

Instead of implementing the host crawler, the technique uses the web crawler "**GNU Wget**" [4,6,12] since it supports full-featured recursion, and more importantly, it is well designed to work with configurable parameters including the list of patterns. For example, to recursively download the files in the "/Ar" and "/En" directories only, as well as excluding those files with suffixes (gif, jpg, wmv, rm, and mid), the following command is used:

```
C:\> Wget -r --reject gif,jpg,wmv,rm,mid --include /Ar,/En www.sis.gov.eg
```

The host crawling process is shown in Figure (2):

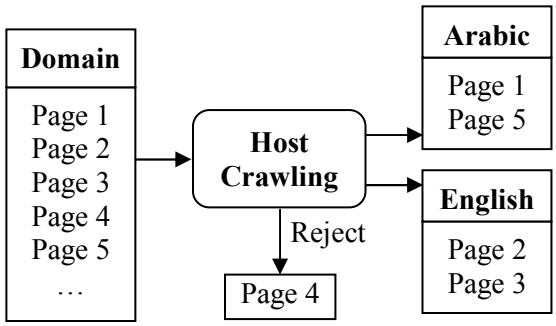

Figure 2 : Host crawling process

According to previous observations, the webpage language is the useful information that can be extracted from URLs. Moreover, Chen and Nie (2000) [3] stated that finding candidate pairs can only be done with content intervention. For instance, "www.sis.gov.eg/**Ar**/Politics/index.htm"and"www.sis.gov.eg/**En**/Politics/index.htm" are actually a pair at the website of Egypt State Information Service [10].

However, a language detector will be useful during the process of host crawling for two reasons. The first reason is that it is not always possible to extract the language of the page from its URL. The second reason is that the URL may contain a pattern that is meant to represent something else other than the language. For example, the pattern "Ar" does not necessarily always stand for Arabic; it may stand for the English word "Article". Similarly, the pattern "En" does not necessarily stand for English; it may stand for the French word "Enquête" (which means a survey or investigation). Therefore, a language detector is useful to detect and overcome such ambiguous patterns.

2.2. Filtering

As mentioned before, two pages can be considered a pair only after content-based similarity (parallelism) has been measured. However, calculating parallelism between all possible combinations (i.e. a full cross-product) of two sets of downloaded web pages, as suggested by Xiaoyi and Liberman (1999) in BITS [17], is very exhaustive and very time-consuming. Therefore, some criteria must be defined to quickly reject all false combinations in order to create a smaller set of candidate pairs.

For example, Resnik (2003) [14] presented the STRAND system which aligns parallel documents according to their HTML structures. Actually, there are several parallel pairs that have quite different HTML structures. Also, news websites have a large number of pages which do not parallelize with any others at all, yet they share same HTML structures. Thus, the three following criteria are suggested to create candidate pairs:

- **Path and filename similarity:**
In most cases, candidate pairs can be recognized by path and filename similarity comparison, since parallel pages usually share similar paths and filenames. Webmasters often stick to this policy in storing parallel pages to easily find and maintain them afterwards.

- **Document length:**
The ratio of the lengths of a pair of parallel pages is usually comparable to the typical length ratio of the two languages (i.e. English and Arabic), especially when the text is long enough. Hence, a simple verification is to compare the lengths of the two documents. Since many web documents are quite short, a tolerance up to 40% from the typical ratio is considered acceptable.

- **Creation date:**
Naturally, parallel pages have a very near creation date, since web editors tend to create the translated version of a page right after it has been created. Consequently, this time difference will be rather short, especially for news sites since news always need to be the latest. The distance of one day after can be a suitable threshold for candidate pages on news sites. However, this time difference threshold can be longer in websites that contain archival material.

2.3. Text Extraction

Text can be extracted using "HTML Text Extractor", [8] a program that extracts text (i.e. without HTML or scripts) from any webpage, even those that have been protected. Figure (3) shows the text extracted from an English-Arabic pair at the homepage of Egypt State Information Service (www.sis.gov.eg).

Figure 3: Extracted text from an English-Arabic pair (http://www.sis.gov.eg)

2.4. Parallelism Matching:

After filtering has been performed and text has been extracted, the system should have a limited list of Arabic documents to scan for each English document (i.e. only those pages with similar paths and filenames that satisfy the length ratio and have a close

creation date). Content-based matching must be performed between two candidate documents, as a final step, in order to measure parallelism.

Any two parallel documents must contain some token pairs that are exact translations of each other. These token pairs are known as translational token pairs. For example, in the two sentences:

"Egypt marks the 35th anniversary of October victory" and

"مصر تشهد الذكرى الـ ٣٥ لنصر أكتوبر"

The translational token pairs are: ("Egypt" - "مصر"), ("anniversary" - "ذكرى"), ("October" - "أكتوبر"), and ("victory" - "نصر").

Therefore, a simple method to calculate parallelism for a pair of documents is to scan them for translated token pairs, and then use the number of translated pairs found as a value of parallelism similarity. However, a pair in which the position of the two translational token pairs is far from each other is rarely to be a correct translation pair. The approach proposed in this technique does not need to search all possible translational tokens extracted from each pair of documents.

A reliable threshold θ_d is set to conclude whether a pair of documents is a translation pair or not. Thus, for each English document, the Arabic documents are scanned until one pair with similarity exceeding θ_d is found. The value of θ_d can be determined empirically. Similarity between a pair of documents (A, E) is defined as:

$$Sim(A,E) = \frac{2N}{\sum_{A,E} number\ of\ tokens} \qquad \textbf{(1)}$$

where N is the number of translational token pairs found between A and E.

It is assumed that the difference in position between a good pair of paragraphs varies from (1) to (-1). Therefore, the number of translation pairs N between two documents is based on the total number of translation pairs between paragraphs n_k. Each English paragraph $p_{e,k}$ will be compared to its 3 neighbor Arabic paragraphs $p_{a,k-1}$, $p_{a,k}$, $p_{a,k+1}$, and the one with the maximum value of translation pairs n_k together with $p_{e,k}$ will form a translation pair of paragraphs.

$$N = \sum n_k \qquad \textbf{(2)}$$

The following algorithm shows how to form a translation pair of paragraphs:

For each English paragraph $p_{e,k}$
Tokenize $p_{e,k}$ with Porter-Stemming Algorithm
S_{max}=0;
For each of 3 Arabic neighbor paragraphs $p_{a,j}$, $j \in \{k-1,k,k+1\}$

$S_j = Sim(p_{e,k}, p_{a,j})$
$if(S_j > S_{max})$
$S_{max} = S_j$

The technique should use an English-Arabic dictionary of stemmed words. Translational token pairs are found by first stemming the English words with Porter Stemming Algorithm [13], and then looked up in the dictionary for all possible Arabic words.

2.5. Word Alignment

The output of the previous section is an English-Arabic parallel corpus aligned at paragraph level. Bilingual pairs of documents collected from the web are used as training material for the statistical translation models. In practice, this material must be organized into a set of smaller pairs (typically, sentences rather than paragraphs), each consisting of a sequence of word tokens. Therefore, the corpus must first be prepared for the translation model. This preparatory step requires aligning the extracted text at sentence level.

Once textual data have been extracted and have been neatly segmented into paragraphs, word alignments are carried out by a parallel corpus aligner such as "Cairo", a word alignment tool available in the "EGYPT Toolkit 1.0" [5]. If two files are known to be translations of each other, Cairo can be used to automatically align them (word-by-word). The word alignments are used for future reference of statistical translation training using "Giza", a statistical-model training tool in the "EGYPT toolkit".

3. Experimental Results

The experiments are carried out using the news website: Egypt State Information Service (www.sis.gov.eg). This website provides parallel pages in English, Arabic and French. However, the experiments utilize only English and Arabic web pages.

However, content-based matching (parallelism) between a pair of documents is affected by the threshold parameter θ_d as well as the number of neighbor paragraphs to be matched. Therefore, the following experimental strategy was used to estimate these two parameters:

i. Estimating the ideal value for the threshold θ_d (i.e. finding the minimum parallelism similarity value between a pair of truly parallel documents).
ii. Estimating the ideal value for parameter k (number of neighbor paragraphs to be scanned) using the ideal threshold value of θ_d.

3.1. Estimating θ_d

For content-based matching evaluation, the experiment was first carried out to determine the ideal threshold value of θ_d. This threshold value determines whether a pair of documents is considered parallel or not; if their similarity exceeds θ_d, then they are considered a parallel pair, otherwise they are not. In the process of evaluating

parallelism similarity between a pair of documents, the total number of translation pairs N between those two documents was based on any number of translational token pairs n_k found between two paragraphs.

The results of this experiment showed that the minimum similarity value between two truly parallel documents was 0.0363. This very low similarity value was because the Arabic document contained more paragraphs than that contained within the English document and in a slightly different order. However, this low value of θd is not accepted as a threshold value because it will result in a large number of false pairs, and therefore it will affect precision. In addition, the results of this experiment showed the maximum similarity value between a false parallel pair of documents was 0.0975.

Therefore, setting θd to 0.1 would be very reasonable, since only the pairs with a content similarity of 10% or more will be considered parallel. Although this threshold value excluded truly parallel pages, yet it reduced the false pairs as well. Table (1) shows the distribution of the parallel documents within the corpus with respect to their similarity values:

Sim(A,E)	Parallel Documents
[0.0 , 0.1 [	24%
[0.1 , 0.2 [	12%
[0.2 , 0.3 [	12%
[0.3 , 0.4 [	16%
[0.4 , 0.5 [	22%
[0.5 , 0.6 [	8%
[0.6 , 0.7 [	4%
[0.7 , 0.8 [	2%
[0.8 , 0.9 [	0%
[0.9 , 1.0 [	0%

Table 1: Number of parallel documents according to similarity

3.2. Estimating k

As mentioned previously, parallel documents have been aligned at paragraph level with a maximum difference in paragraphs position of one at k =3. A final experiment was carried out for the distance k=5 (i.e. examine p_k of d_e with p_{k-2}, p_{k-1}, p_k, p_{k+1}, p_{k+2} of d_a) and observed very similar results but the whole process lasted much longer. Therefore, the default value (k =3) is accepted as the ideal value for the position difference

between a pair of paragraphs. Figure (4) shows a sample of five pairs of English-Arabic paragraphs in two parallel documents with a similarity of 0.595.

Figure 4: A sample of paragraph matching pairs

Paragraphs (P_1, P_3, P_4 and P_5) in the English document matched with paragraphs (P_1, P_2, P_4, and P_5) in the Arabic document, respectively. However, paragraph (P_2) in the English document did not match with any of the three neighbor paragraphs in the Arabic document. In addition, paragraph (P_3) in the Arabic document did not match with any paragraph in the English document.

4. Conclusion and Future Work

Parallel corpora played an important role in the cross-language information retrieval (CLIR), by providing the primary training data for the known statistical translational models. The main traditional obstacle was that parallel corpora were not available in sufficient quantities. As a result, most previous work has been conducted on a few manually constructed parallel corpora.

Therefore, the proposed technique aims to automatically construct the parallel corpus through web mining, and to prepare the corpus for the translation model. However, this paper did not aim to test or analyze the statistical translation model.

First, the system crawled the host using GNU Wget in order to obtain English and Arabic web pages, and then created candidate parallel pairs of documents by filtering them according to their similarity in path, filename, creation date and length. Finally, the technique measured the parallelism similarity between these candidate pairs according to the number of translational tokens found between an English paragraph and its three Arabic neighbor paragraphs. The parallelism similarity achieved the highest precision at θ_d=0.1.

The process of constructing the English-Arabic parallel corpus automatically provided very promising results. Moreover, the technique is adaptable and easy to apply to other pair of languages by changing the bilingual dictionary and using the same filtering rules.

In this paper, however, the process of constructing the parallel corpus did not take into consideration the domain to which the document belongs, although parallel corpus used in translation is highly domain specific (e.g. business, medical, martial, legislative, etc.). Therefore, integrating a domain-detector could optimize the current technique. This would provide the ability to extract a portion of the corpus for a given document.

The technique presented in this paper searched only for the parallel pages that were good translations for each other. Therefore, from a different point of view, this technique could be substituted by another technique that uses other unrestricted equations to measure the similarity of parallel documents. This alternative technique would enrich the parallel corpus and make it huger, but on the other hand, it would have many false parallel documents that in turn would result in a worse translation quality.

Finally, this paper did not test or compare the different models of statistical translation training using the constructed prepared parallel corpus. Therefore, more future work could be done in this point of research.

5. References

[1] Aljlayl, M. & Frieder, O., "Effective Arabic-English Cross-Language Information Retrieval via Machine Readable Dictionaries and Machine Translation". ACM 10th Conference on Information and Knowledge Management, p.295-302,2001.

[2] Brown, P., Pietra, S.A.D., Pietra, V.J.D. & Mercer, R.L., "The Mathematics of Statistical Machine Translation: Parameter Estimation. Computational Linguistics", 19 (2), p.263-311,1993.

[3] Chen, J. & Nie, J., "Automatic Construction of Parallel English-Chinese Corpus for Cross-Language Information Retrieval". Proceedings of ANLP, Seattle, p.21-28,2000.

[4] Dang, V.B. & Ho, B., "Automatic Construction of English-Vietnamese Parallel Corpus through Web Mining. Innovation and Vision for the Future", IEEE International Conference, p.261-266,2007.

[5] EGYPT Toolkit 1.0. [Software]. Center for Language and Speech Processing at Johns-Hopkins University (CLSP/JHU). Available at: http://www.clsp.jhu.edu/ws99/projects/mt/toolkit/ [Accessed 26 February 2008].

[6] GNU Wget 1.10.2. [Software]. Available at: http://ftp.gnu.org/gnu/wget/ [Accessed 26 February 2008].

[7] Google, 2002. "Internet Statistics: Distribution of languages on the Internet. [Online] ". Available at: http://www.netz-tipp.de/languages.html [Accessed 26 February 2008].

[8] HTML Text Extractor 1.5. [Software]. Available at: http://www.iconico.com/HTMLExtractor [Accessed 26 February 2008].

[9] http://www.ldc.upenn.edu/

[10] http://www.sis.gov.eg

[11] Kraaij, W., Nie, J. & Simard, M., 2003. "Embedding Web-Based Statistical Translation Models in Cross-Language Information Retrieval. Computational Linguistics", 29 (3), p.381-419. 62

[12] Niksic, H. et al, April 2005. GNU Wget 1.10. Free Software Foundation. Available at: http://www.gnu.org/software/wget/manual [Accessed 26 February 2008]

[13] Porter, M., 2006. Porter Stemming Algorithm. [Online]. Available at http://tartarus.org/martin/PorterStemmer/ [accessed at 29 February 2008].

[14] Resnik, P. & Smith, N.A., 2003. The Web as Parallel Corpus. Computational Linguistics, 29 (3), p.349-380.

[15] Sigurbjörnsson, B., Kamps, J. & Rijke, M., Blueprint of a Cross-Lingual Web Retrieval Collection. Journal of Digital Information Management, 3 (4),2005.

[16] Talvensaari, T. et al, "Creating and Exploiting a Comparable Corpus in Cross-Language Information Retrieval", ACM Transactions on Information Systems, 25 (1), Article 4,2007

[17] Xiaoyi, M. & Liberman, M. Y., "BITS: A Method for Bilingual Text Search over the Web. Proceedings of Machine Translation Summit" VII, p. 538-542,1999.

[18] Youssef, M., "Cross Language Information Retrieval. Universal Usability in Practice", Department of Computer Science, University of Maryland,2001.