WebMining: An unsupervised parallel corpora web retrieval system

Jesús Tomás
Instituto Tecnológico de Informática
Universidad Politécnica de Valencia
jtomas@upv.es

Enrique Sánchez-Villamil
Dpto. de Lenguajes y Sist. Informáticos
Universidad de Alicante
esvillamil@dlsi.ua.es

Jaime Lloret
Dpto. de Comunicaciones
Universidad Politécnica de Valencia
jlloret@dcom.upv.es

Francisco Casacuberta
Instituto Tecnológico de Informática
Universidad Politécnica de Valencia
fcn@iti.upv.es

Abstract

Inductive methods in multilingual natural language processing require using parallel corpora, which is not often available. Besides, Internet is an important source of parallel texts and a increasingly number of multilingual websites can be found. In this paper we present a system able to automatically collect large multilingual corpora from the web. The input of the system is a list of multilingual websites that has been manually created. For each website, the system downloads all their webpages following their links, and each new webpage is compared to all previously downloaded ones looking for possible parallel webpages. To determine if two pages are parallel, the system uses four filters based on: size, language, HTML structure and plain text. The last one extracts sentences of both pages and tries to align them using a word-to-word statistical translation model, which is trained on the fly. The final output of the system is an aligned bilingual sentence corpus. Our system has been compared to three other similar systems. Evaluation results show the high precision obtained with our system.

1 Introduction

Inductive methods have proved to be valuable in tasks such as automatic speech recognition and natural language processing. The main limitation of these methods is that a corpus is needed to train their models. In multilingual tasks a specific kind of corpus is required: the parallel corpus, that is, a collection of pairs of texts which are translations between them. Parallel corpora are a necessary resource in

many tasks such as statistical machine translation [1][2], cross-lingual information retrieval [3], word sense disambiguation [4] or lexical acquisition [5].

However, parallel corpora are difficult to obtain and the few available corpora are not balanced. Some public organization like United Nations or European Union publish an important number of parallel documents, although this kind of corpus is especially useful only for administrative tasks. Parallel corpora available for minority languages are practically non-existent.

Nonetheless, Internet is an important source of parallel texts. A considerable number of web sites contain a multilingual version of their pages. The available number of webpages in different languages is increasing daily in the web and any kind of information can be found.

In this paper, we present a system able to automatically collect large bilingual text corpora from the web. The system searchs into a website looking for parallel pages. Two pages will be considered parallel only if their sizes are similar, their languages are different and interesting for our search and their HTML structures are similar. Once both pages have passed the previous filters, their sentences are extracted and the system tries to align them according to a word-to-word statistical translation model, which is trained on the fly.

The following section describes the system architecture. In section 3 we compare our approach to three other similar systems. In section 4 the evaluation results are presented. Finally, the conclusions and future work are discussed in section 5.

2 System Architecture

This section explains the mining algorithm that allows the system to collect parallel pages. Figure 1 shows a dataflow that summarizes the system architecture.

The GUI of the WebMining application allows the user to observe easily the parallel webpages that are being downloaded and analyzed to find parallel pages as can be seen in figure 2.

The application is able to stop each time a parallel page is found, so that the HTML format, the inner text or the sequence of tags of the two pages could be analyzed. Furthermore, this allows the checking of the sentence alignment of the inner text and the pages distances obtained by the system.

In addition, the evaluation of the system can be performed with a precision-coverage calculation module, using a manually built parallel corpus as a reference, so that different thresholds can be tested.

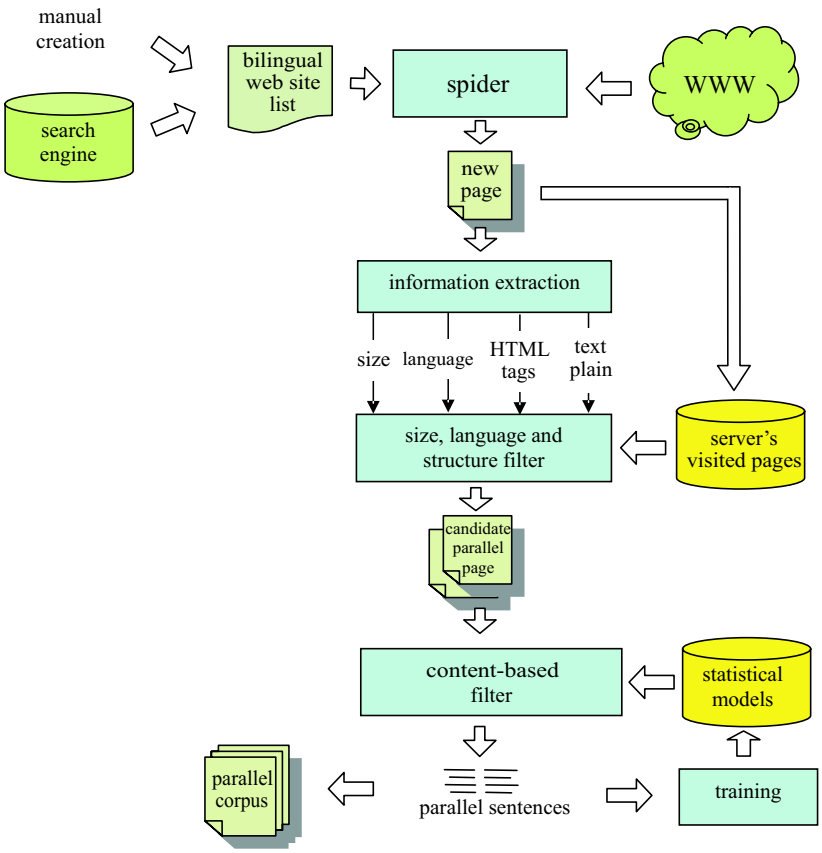

Figure 1: Dataflow of the system.

2.1 Bilingual Website Localization

First of all, we need to know what websites contain bilingual webpages. Several approaches have been proposed to create the list of websites: [6] proposes including all servers in a certain domain; [7] and [3] propose to use web search engines to locate these websites (see section 3 for more details).

In our experiments we have manually created the list of webpages, using a combination of those two techniques. For example, in order to obtain a Basque-Spanish parallel text corpus we can use "Altavista" engine to search websites that contain pages in Basque, and we presume that many of these websites contain this pages in Spanish. In order to obtain a Catalan-Spanish corpus we can include all servers in the domain of catalan universities.

2.2 Candidate Parallel Webpage Localization

For each website in the previous list, a *spider* is used to download all their webpages following their links. Once a new page is downloaded, it is stored and compared to all previously downloaded ones looking for possible parallel webpages. In order to determine if they are parallel, the system initially uses three filters based on: sizes, languages and HTML structures.

The first filter consists in comparing some statistical values such as the file size. Two parallel webpages should have similar file sizes. Three different comparisons are performed between both pages: file size, number of HTML tags and number of paragraphs.

The second filter ensures that both webpages are written in different languages and that they are languages of our interest. For this purpose, our system uses a bigram statistical language identification model [8]. We have trained these models for several European languages, using a corpus obtained from the European Union Parliament.

If both webpages pass the previous filters, their HTML structures are compared. We assume that two parallel webpages show similar appearance, as it is shown in figure 2. Usually, when a web designer translates a webpage, keeps the HTML markup information and replaces only plain text. In order to implement this filter, a sequence of HTML tags is extracted from the two webpages. Then, the parameters of the tags are erased and the plain text is replaced by a special tag $< text\_n >$, where n is the number of words that have been replaced. After that, the edit distance [1] between both sequences is computed. The weight of in-

[1]The edit distance is the minimum number of insertions, deletions, and substitutions required

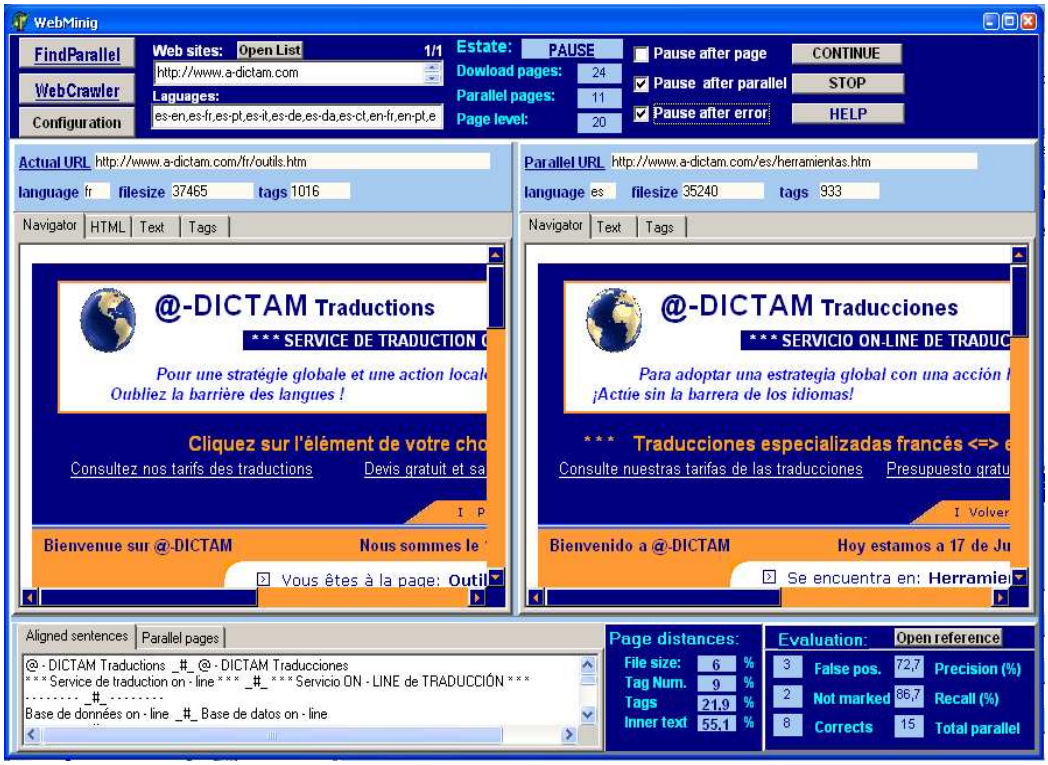

Figure 2: Example of the interactive GUI.

sertion, deletion and substitution are set to one, except the weight of substitution of $< text\_n >$ for $< text\_m >$ that is set to $abs(n-m)/10$. Only if the edit distance between both sequences of tags is lower than a threshold, both webpages are considered as parallel candidates.

2.3 Content-Based Matching

The previously mentioned filters are accurate enough to obtain good results in many cases, as it is reported in [7] and [3]. However, in other cases, they do not work properly, maybe because some webpages include mainly text and little markup information [6], or because both webpages have been created using two different HTML editors [3]. In order to solve this problem, our system uses an additional filter based on the plain text of pages.

to transform one sequence into the other.

Other approaches use cognates [6] or translation lexicons [7][6] to implement the content-based matching. The inconvenience of these approaches is that they usually are language dependent.

Statistical Machine translation has been proved to be a valuable resource to solve this kind of problem [9][10], even if linguistic information is not available. However, statistical methods require a bilingual corpus to train the models and it is just the goal of the system. To solve the problem of aligning sentences in bilingual corpora [11], in [10] it is proposed the following iteration procedure: Firstly, the system uses only structural matching, then the corpus obtained is used to train the statistical model. The process starts again but using the statistical model obtained in the previous iteration. Chen [9] proposes an interesting alternative: training the translation model on the fly during alignment, which is the approach used in our system.

For each pair of candidate parallel webpages their sentences are fragmented. Then, the system tries to align the sentences of both webpages using a dynamic programming technique [11][9][10]. For this task, a score is needed to indicate if a sentence is the translation of the other one. We use a estimation of a statistical translation model, which is similar to IBM-1 model[1]:

$$Pr(\mathbf{f}|\mathbf{e}) = Pr(m|l) \prod_{j=1}^{m} Pr(f_j|\mathbf{e}) = \alpha^{|m-l|} \prod_{j=1}^{m} \sum_{i=0}^{l} p(f_j|e_i), \qquad (1)$$

where, $\mathbf{f} = f_1...f_m$ is the source language sentence, $\mathbf{e} = e_1...e_l$ is the target sentence, m is the number of words in the source sentence and l is the number of words in the target sentence.

Using this statistical model the best sentence alignment that the model offers is obtained. Then, we compute the percentage of sentence pairs with a translation probability higher than a threshold, to check if both pages are mutual translations.

In order to estimate quickly the parameters of the model, we use an incremental variation of the expectation-maximization (EM) algorithm [12]. The EM algorithm comprises two phases[9]: in the expectation phase, counts on the corpus is taken using the current parameter estimation; in the expectation phase, the parameters are re-estimated based on the previous counts. Both phases are done several times.

In the incremental version, each time two webpages are detected as parallel, the system applies expectation-maximization to improve the estimated parameters. In the expectation phase, the counts obtained in the previous expectation are keeped and incremented applying the actual estimation of the parameters to

the new parallel page. After that, in the estimation phase, the parameters are re-estimated using the counts just taken.

Initially, all word-to-word translation probabilities are initialized equiprobably, except when the source and target words are the same that are initialized to 1. Using this information and the sentence lengths only, our system obtains reasonably good sentence alignments.

When using this filter in the first webpages, if two webpages are parallel, a reasonably good sentence alignment is obtained. However, if they are not truly parallel we do not have enough information to detect it. This problem was solved setting very restricted thresholds in previous filters, until we considered there was a good estimation of the statistical model.

3 Related Works

Recently, several systems have been developed to find parallel webpages in the web. In this section we describe three of these systems: STRAND [7], PTMiner [3] and BITS [6]. All systems present a similar architecture. We compare the differences using the same structure as the description of our system.

3.1 Bilingual Website Localization

Different alternatives have been proposed to locate bilingual websites: STRANDS and PTMiner use a Internet search engine to locate webpages that contain links to other webpages in different languages. For example, Resnik [7] proposes the following query in ”Altavista” engine lo locate English-French pages: (anchor:”english” OR anchor:”anglais”) AND (anchor:”french” OR anchor:”français”). BITS propose to explore all web servers given a certain domain. To obtain this list, it queries a DNS server finding servers that their urls begin with www.

3.2 Parallel Webpage Localization

PTMiner, BITS and our system use a similar strategy to locate parallel webpages. A *spider* is used to obtain all webpages in the server while comparing them to the previously downloaded ones.

On the contrary, STRAND remains using ”Altavista” engine for this task. Only the webpages obtained as a result of a query similar to the previous example are considered as possible parallel webpages. We have verified that many parallel webpages do not have links to the their parallel ones or do not satisfy the

query condition (for instance if the links are represented by images without text). Thus, the coverage of this method is lower.

3.3 Candidate Parallel Page Searching

The three compared systems start using a filter based on filename and path URL similarity comparison. The webmasters usually name the parallel webpages with similar names, i.e.: $http://www.host.com/english/indexen.html$ and $http://www.host.com/french/indexfr.html$. This criterion is useful in the majority of the cases. However, the way a webmaster names the files is difficult to predict. Therefore, our system do not use this filter and compare all webpages beween them in the website. The filters that the system uses are fast enough and do not require to reduce the search space.

PTMiner, BITS and our system use language identification based on an N-gram technique [8], and all systems use a criterion based on file length. In addition, BITS uses the number of paragraph and the number of links. The criterion based on HTML structure comparison is implemented by all systems but BITS.

3.4 Content-Based Matching

The last version of STRAND [7] incorporates a content-based filter. It uses a generic score of translational similarity based upon any word-to-word translation lexicon. The entries in this lexicon are not weighted. To compute this measure, STRAND firstly obtains the best word-to-word alignment using the lexicon. Then, it uses the next equation:

$$similarity(A, B) = \frac{two - word\_alignments}{two - word\_alignments + words\_not\_aligned} \quad (2)$$

PTMiner do not use Content-Based Matching but content-based matching is the main filter used in BITS. It considers the number of words in a webpage that have a translation in the other one. It uses the next equation.

$$similarity(A, B) = \frac{two - word\_alignments}{words\_in\_A} \quad (3)$$

In order to find a pair of aligned words, it proposes two alternatives: for lexically similar language pairs, such as French-English, the alignment between

words can be found using cognates. That is, a pair of tokens that share phonological or orthographic properties (for example: functionality and funktionalität). For other language pairs, such a Chinese-English, it proposes to use a lexicon.

4 Experimental Results

We have selected 8 multilingual websites to carry out the evaluation, of our system. To measure the precision and recall we have manually retrieved the parallel webpages of these websites. As a result, 652 parallel webpages were found. Table 1 shows the details of the evaluation test.

We set the parameters of the system as follows: the threshold of difference size length between files was set to 30%, the difference of tags was set to 20% and the difference of paragraphs was set to 40%.

To set the edit distance threshold we used a training set of parallel webpages from five websites. We manually marked the parallel pages in the same way as in the text. We carried out an experiment with these websites using only the size and the language filter and this experiment obtained a precision of 100%, but it had very low recall. Then, we depicted in figure 3 each parallel webpages detected by the system in function of the edit distance and the number of tags of the biggest one. As figure 3 shows, if the size of the webpages is large the edit distance threshold can be increased. Thus, we decided to use a threshold that depends on the number of tags, as it is shown in the figure.

Table 1 shows that using structural information is enough to obtain high precision results at the first six websites. However, the last two websites contain webpages with a very similar HTML structure, which decrease the precision of our system.

Another experiment was carried out to evaluate the influence of introducing content-based matching. This experiment consisted in detecting only English-Spanish parallel webpages from previous websites. Table 2 shows the results. Precision improves significantly, especially in the last two websites.

5 Conclusions

In this paper we have presented an unsupervised system to obtain parallel corpus from bilingual webpages. The system uses the following information to determine if two pages are parallel: file size, page language, HTML structure and content sentences of the webpage.

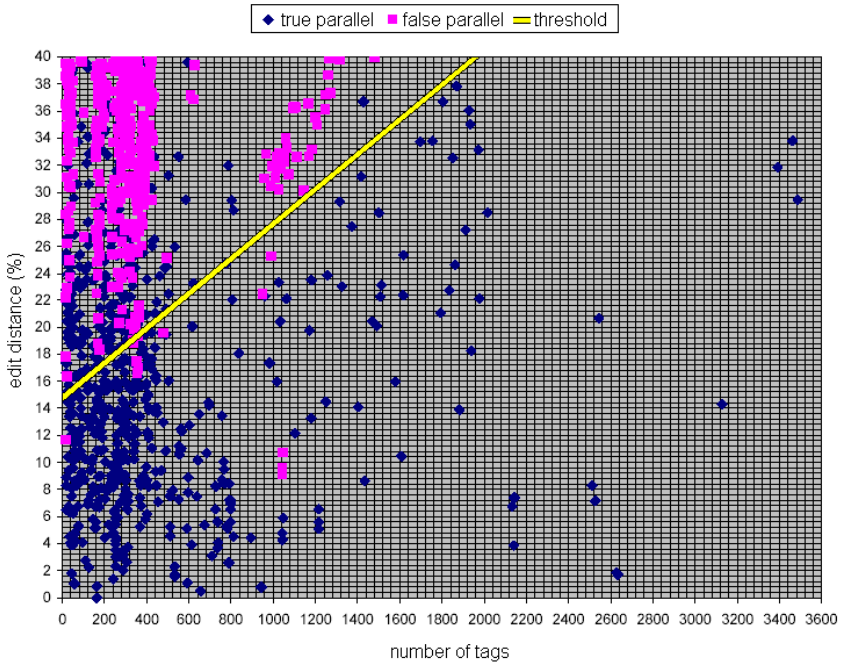

Figure 3: The edit distance threshold.

Table 1: Results obtained in 8 multilingual web sites using structural information of the pages (es-Spanish, fr-French, en-English, ct-Catalan, de-German, it-Italian, pt-Portuguese)

Web Site	languages	total pages	parallel pages	precision	recall
www.puc-rio.br/louvre	es, fr, en	290	242	97%	82%
www.a-dictam.com	es, fr	34	17	88%	94%
www.covax.org	es, ct, en, de, it	71	110	100%	64%
www.faocopemed.org	es, fr, en	193	94	100%	87%
www.eapn.org	fr, en	124	46	100%	68%
www.fsmed.info	es, ct, en, de, it	36	63	100%	92%
www.sphereproject.org	es, fr, en	315	56	11%	55%
www.rocksolid.com	es, pt, en	24	24	32%	71%
total:		1087	652	89%	78%

Table 2: Results obtained in 6 English-Spanish web sites using structural information and content-based matching.

	precision	recall
only structural information	85%	82%
with content-based matching	96%	89%

Our system does not use a web search engine to find the multilingual parallel pages [7][3]. The search engines cover only a part of the existing websites and do not index all the webpages of a website. [3].

The rest of compared systems use a first filter to determine if two webpages are parallel based on their URLs. As it was shown before, that filter does not work properly in all cases. That is why we prefer to explore all the webpages in the website.

In order to implement the content-based matching, we propose not to use lexicons [7][6]. Instead, our system learns this information from the same webpages that it is collecting. For this task, a statistical translation model trained on the fly is used.

In summary, the main property of our system is that it is language independent and it does not require linguistic resources (except a small text of each language, 100k is enough, to train the language identification module).

In the future, we plan to use this system to collect big corpora of minority languages, such a Spanish-Basque. We are also interested in the study of the automation of multilingual websites localization. Another interesting point is to test if the system is accurate enough while obtaining parallel corpus from very different languages, such a English-Chinese.

Acknowledgements

This work has been partially supported by the Spanish project TIC2003-08681-C02-01/02, by the IST Programme of the European Union IST-2001-32091 and by *Agencia Valenciana de Ciencia y Tecnología* under contract GRUPOS03/031.

References

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

[2] Melamed, I.D. (1997) A word-to-word model of translational equivalence. In: Proc. 35th Annual Conf. of the Association for Computational Linguistics, Madrid, Spain

[3] Chen, J., Nie, J.Y. (2000) Parallel web text mining for cross-language information retrieval. *Recherche d'Informations Assistée par Ordinateur (RIAO)* 62-77

[4] Brown, P.F., Della Pietra, V.J., Della Pietra, S.A., , Mercer, R.L. (1991) Word sense disambiguation using statistical methods. In: Procs. of 29th Annual Meeting of the ACL, Berkeley, CA 264-270

[5] Melamed, I.D. (1997) Automatic discovery of non-compositional compounds in parallel data. In Cardie, C., Weischedel, R., eds.: Proceedings of the Second Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, Somerset, New Jersey 97-108

[6] Ma, X., Liberman, M. (1999) Bits: A method for bilingual text search over the web. In: Machine Translation Summit VII.

[7] Resnik, P., Smith, N.A. (2003) The web as a parallel corpus. *Computational Linguistics 29*, 349-380

[8] Grefenstette, G. (1995) Comparing two language identification schemes. In: Proc. of the 3rd International Conference on Statistical Analysis of Textual Data (JADT'95), Rome, Italy

[9] Chen, S.F. (1993) Aligning sentences in bilingual corpora using lexical information. In: Conf. of the Association for Computational Linguistics, Columbus, Ohio, 9-16

[10] Tomás, J., Fabregat, F., del Val, J., Casacuberta, F., Picó, D., Sanchís, A., Vidal, E. (2001) Automatic development of spanish-catalan corpora for machine translation. In: Procs. of the Second International Workshop on Spanish Language Processing and Language Technologies, Jaén, Spain

[11] Gale,W., Church, K. (1993) A program for aligning sentences in bilingual corpora. *Computational Linguistics 19,* 75-90

[12] Dempster, A.P., Laird, N.M., Rubin, D.B. (1977) Maximum likelihood from incomplete data via the EM algorithm. *J. Royal Statist. Soc. Ser. B 39,* 1-22