An Expectation Maximization Algorithm for Textual Unit Alignment

Radu Ion
Research Institute for AI
Calea 13 Septembrie nr. 13
Bucharest 050711, Romania
radu@racai.ro

Alexandru Ceauşu
Dublin City University
Glasnevin, Dublin 9, Ireland
[address3]
aceausu@computing.dcu.ie

Elena Irimia
Research Institute for AI
Calea 13 Septembrie nr. 13
Bucharest 050711, Romania
elena@racai.ro

Abstract

The paper presents an Expectation Maximization (EM) algorithm for automatic generation of parallel and quasi-parallel data from any degree of comparable corpora ranging from parallel to weakly comparable. Specifically, we address the problem of extracting related textual units (documents, paragraphs or sentences) relying on the hypothesis that, in a given corpus, certain pairs of translation equivalents are better indicators of a correct textual unit correspondence than other pairs of translation equivalents. We evaluate our method on mixed types of bilingual comparable corpora in six language pairs, obtaining state of the art accuracy figures.

1 Introduction

Statistical Machine Translation (SMT) is in a constant need of good quality training data both for translation models and for the language models. Regarding the latter, monolingual corpora is evidently easier to collect than parallel corpora and the truth of this statement is even more obvious when it comes to pairs of languages other than those both widely spoken and computationally well-treated around the world such as English, Spanish, French or German.

Comparable corpora came as a possible solution to the problem of scarcity of parallel corpora with the promise that it may serve as a seed for parallel data extraction. A general definition of comparability that we find operational is given by Munteanu and Marcu (2005). They say that a (bilingual) comparable corpus is a set of paired documents that, *while not parallel in the strict sense, are related and convey overlapping information*.

Current practices of automatically collecting domain-dependent bilingual comparable corpora from the Web usually begin with collecting a list of t terms as seed data in both the source and the target languages. Each term (in each language) is then queried on the most popular search engine and the first N document hits are retained. The final corpus will contain $t \times N$ documents in each language and in subsequent usage the document boundaries are often disregarded.

At this point, it is important to stress out the importance of the pairing of documents in a comparable corpus. Suppose that we want to word-align a bilingual comparable corpus consisting of M documents per language, each with k words, using the IBM-1 word alignment algorithm (Brown et al., 1993). This algorithm searches for each source word, the target words that have a maximum translation probability with the source word. Aligning all the words in our corpus with no regard to document boundaries, would yield a time complexity of k^2M^2 operations. The alternative would be in finding a 1:p (with p a small positive integer, usually 1, 2 or 3) document assignment (a set of aligned document pairs) that would enforce the "no search outside the document boundary" condition when doing word alignment with the advantage of reducing the time complexity to k^2Mp operations. When M is large, the reduction may actually be vital to getting a result in a reasonable amount of time. The downside of this simplification is the loss of information: two documents may not be correctly aligned thus depriving the word-alignment algorithm of the part of the search space that would have contained the right alignments.

Proceedings of the 4th Workshop on Building and Using Comparable Corpora, pages 128–135,
49th Annual Meeting of the Association for Computational Linguistics,
Portland, Oregon, 24 June 2011.

Word alignment forms the basis of the phrase alignment procedure which, in turn, is the basis of any statistical translation model. A comparable corpus differs essentially from a parallel corpus by the fact that textual units do not follow a translation order that otherwise greatly reduces the word alignment search space in a parallel corpus. Given this limitation of a comparable corpus in general and the sizes of the comparable corpora that we will have to deal with in particular, we have devised one variant of an Expectation Maximization (EM) algorithm (Dempster et al., 1977) that generates a 1:1 (p = 1) document assignment from a parallel and/or comparable corpus using only pre-existing translation lexicons. Its generality would permit it to perform the same task on other textual units such as paragraphs or sentences.

In what follows, we will briefly review the literature discussing document/paragraph alignment and then we will present the derivation of the EM algorithm that generates 1:1 document alignments. We will end the article with a thorough evaluation of the performances of this algorithm and the conclusions that arise from these evaluations.

2 Related Work

Document alignment and other types of textual unit alignment have been attempted in various situations involving extracting parallel data from comparable corpora. The first case study is offered by Munteanu and Marcu (2002). They align sentences in an English-French comparable corpus of 1.3M of words per language by comparing suffix trees of the sentences. Each sentence from each part of the corpus is encoded as a suffix tree which is a tree that stores each possible suffix of a string from the last character to the full string. Using this method, Munteanu and Marcu are able to detect correct sentence alignments with a precision of 95% (out of 100 human-judged and randomly selected sentences from the generated output). The running time of their algorithm is approximately 100 hours for 50000 sentences in each of the languages.

A popular method of aligning sentences in a comparable corpus is by classifying pairs of sentences as parallel or not parallel. Munteanu and Marcu (2005) use a Maximum Entropy classifier for the job trained with the following features: sentence lengths and their differences and ratios, percentage of the words in a source sentence that have translations in a target sentence (translations are taken from pre-existing translation lexicons), the top three largest fertilities, length of the longest sequence of words that have translations, etc. The training data consisted of a small parallel corpus of 5000 sentences per language. Since the number of negative instances ($5000^2 - 5000$) is far more large than the number of positive ones (5000), the negative training instances were selected randomly out of instances that passed a certain word overlap filter (see the paper for details). The classifier precision is around 97% with a recall of 40% at the Chinese-English task and around 95% with a recall of 41% for the Arabic-English task.

Another case study of sentence alignment that we will present here is that of Chen (1993). He employs an EM algorithm that will find a sentence alignment in a parallel corpus which maximizes the translation probability for each sentence bead in the alignment. The translation probability to be maximized by the EM procedure considering each possible alignment $\mathcal{A}$ is given by

$$P(\mathcal{E}, \mathcal{F}, \mathcal{A}) = p(L) \prod_{k=1}^{L} P\left(\left[E_p^k; F_p^k\right]\right)$$

The following notations were used: $\mathcal{E}$ is the English corpus (a sequence of English sentences), $\mathcal{F}$ is the French corpus, $\left[E_p^k; F_p^k\right]$ is a sentence bead (a pairing of m sentences in English with n sentences in French), $\mathcal{A} = \left(\left[E_p^1; F_p^1\right], \ldots, \left[E_p^L; F_p^L\right]\right)$ is the sentence alignment (a sequence of sentence beads) and $p(L)$ is the probability that an alignment contains L beads. The obtained accuracy is around 96% and was computed indirectly by checking disagreement with the Brown sentence aligner (Brown et al., 1991) on randomly selected 500 disagreement cases.

The last case study of document and sentence alignment from "very-non-parallel corpora" is the work from Fung and Cheung (2004). Their contribution to the problem of textual unit alignment resides in devising a bootstrapping mechanism in which, after an initial document pairing and consequent sentence alignment using a lexical overlapping similarity measure, IBM-4 model (Brown et al., 1993) is employed to enrich the bilingual dictionary that is used by the similarity measure. The

process is repeated until the set of identified aligned sentences does not grow anymore. The precision of this method on English-Chinese sentence alignment is 65.7% (out of the top 2500 identified pairs).

3 EMACC

We propose *a specific instantiation of the well-known general EM algorithm* for aligning different types of textual units: documents, paragraphs, and sentences which we will name EMACC (an acronym for "Expectation Maximization Alignment for Comparable Corpora"). We draw our inspiration from the famous IBM models (specifically from the IBM-1 model) for word alignment (Brown et al., 1993) where the translation probability (eq. (5)) is modeled through an EM algorithm where the hidden variable **a** models the assignment (1:1 word alignments) from the French sequence of words (' indexes) to the English one.

By analogy, we imagined that between two sets of documents (from now on, we will refer to documents as our textual units but what we present here is equally applicable – but with different performance penalties – to paragraphs and/or sentences) – let's call them $\mathbf{E}$ and $\mathbf{F}$, there is *an assignment* (a sequence of 1:1 document correspondences[1]), the distribution of which can be modeled by a hidden variable *z* taking values in the set {true, false}. This assignment will be largely determined by the existence of word translations between a pair of documents, translations that can differentiate between one another in their ability to indicate a correct document alignment versus an incorrect one. In other words, we hypothesize that there are certain pairs of translation equivalents that are better indicators of a correct document correspondence than other translation equivalents pairs.

We take the general formulation and derivation of the EM optimization problem from (Borman, 2009). The general goal is to optimize $P(X|\Theta)$, that is to find the parameter(s) Θ for which $P(X|\Theta)$ is maximum. In a sequence of derivations that we are not going to repeat here, the general EM equation is given by:

$$\Theta_{n+1} = \underset{\Theta}{\operatorname{argmax}} \sum_{z} P(z|X,\Theta_n) \ln P(X,z|\Theta) \qquad (1)$$

where $\sum_z P(z|X,\Theta_n) = 1$. At step *n*+1, we try to obtain a new parameter Θ_{n+1} that is going to maximize (the maximization step) the sum over *z* (the expectation step) that in its turn depends on the best parameter Θ_n obtained at step *n*. Thus, in principle, the algorithm *should iterate over the set of all possible Θ parameters*, compute the expectation expression for each of these parameters and choose the parameter(s) for which the expression has the largest value. But as we will see, in practice, the set of all possible parameters has a dimension that is exponential in terms of the number of parameters. This renders the problem intractable and one should back off to heuristic searches in order to find a near-optimal solution.

We now introduce a few notations that we will operate with from this point forward. We suggest to the reader *to frequently refer to this section* in order to properly understand the next equations:

- $\mathbf{E}$ is the set of source documents, $|\mathbf{E}|$ is the cardinal of this set;
- $\mathbf{F}$ is the set of target documents with $|\mathbf{F}|$ its cardinal;
- d_{ij} is a pair of documents, $d_i \in \mathbf{E}$ and $d_j \in \mathbf{F}$;
- w_{ij} is a pair of translation equivalents $\langle w_i, w_j \rangle$ such that w_i is a lexical item that belongs to d_i and w_j is a lexical item that belongs to d_j;
- $\mathbf{T}$ is the set of all existing translation equivalents pairs $\langle w_{ij}, p \rangle$. p is the translation probability score (as the one given for instance by GIZA++ (Gao and Vogel, 2008)). We assume that GIZA++ translation lexicons already exist for the pair of languages of interest.

In order to tie equation 1 to our problem, we define its variables as follows:

- Θ is the sequence of 1:1 document alignments of the form $D_{i_1 j_1}, D_{i_2 j_2}, \ldots, D_{ij} \in \{d_{ij} | d_i \in \mathbf{E}, d_j \in \mathbf{F}\}$. We call Θ *an assignment* which is basically a sequence of 1:1 document alignments. If there are $|\mathbf{E}|$ 1:1 document alignments in Θ and if $|\mathbf{E}| \leq |\mathbf{F}|$, then the set of all possible assignments has

[1] Or "alignments" or "pairs". These terms will be used with the same meaning throughout the presentation.

the cardinal equal to $|\mathbf{E}|!\binom{|\mathbf{F}|}{|\mathbf{E}|}$ where *n*! is the factorial function of the integer *n* and $\binom{n}{k}$ is the binomial coefficient. It is clear now that with this kind of dimension of the set of all possible assignments (or Θ parameters), we cannot simply iterate over it in order to choose the assignment that maximizes the expectation;

- $z \in \{\text{true,false}\}$ is the hidden variable that signals if a pair of documents d_{ij} represents a correct alignment (true) or not (false);
- X is the sequence of translation equivalents pairs W_{ij} from $\mathbf{T}$ in the order they appear in each document pair from Θ.

Having defined the variables in equation 1 this way, we aim at maximizing the translation equivalents probability over a given assignment, $P(X|\Theta)$. In doing so, through the use of the hidden variable *z*, we are also able to find the 1:1 document alignments that attest for this maximization.

We proceed by reducing equation 1 to a form that is readily amenable to software coding. That is, we aim at obtaining some distinct probability tables that are going to be (re-)estimated by the EM procedure. Due to the lack of space, we omit the full derivation and directly give the general form of the derived EM equation

$$\Theta_{n+1} = \underset{\Theta}{\operatorname{argmax}}[\ln P(X|\Theta) + \ln P(\text{true}|\Theta)] \quad (2)$$

Equation 2 suggests a method of updating the assignment probability $P(\text{true}|\Theta)$ with the lexical alignment probability $P(X|\Theta)$ in an effort to provide the alignment clues that will "guide" the assignment probability towards the correct assignment. All it remains to do now is to define the two probabilities.

The **lexical document alignment probability** $P(X|\Theta)$ is defined as follows:

$$P(X|\Theta) = \prod_{d_{ab}\in\Theta} \frac{\sum_{w_{ij}\in X} P(d_{ab}|w_{ij})}{|\mathbf{E}||\mathbf{F}|} \quad (3)$$

where $P(d_{ab}|w_{ij})$ is the simplified lexical document alignment probability which is initially equal to $P(w_{ij})$ from the set $\mathbf{T}$. This probability is to be read as "the contribution w_{ij} makes to the correctness of the d_{ab} alignment". We want that the alignment contribution of one translation equivalents pair w_{ij} to distribute over the set of all possible document pairs thus enforcing that

$$\sum_{d_{ab}\in\{d_{xy}|d_x\in\mathbf{E},\, d_y\in\mathbf{F}\}} P(d_{ab}|w_{ij}) = 1 \quad (4)$$

The summation over X in equation 3 is actually over all translation equivalents pairs that are to be found only in the current d_{ab} document pair and the presence of the product $|\mathbf{E}||\mathbf{F}|$ ensures that we still have a probability value.

The **assignment probability** $P(\text{true}|\Theta)$ is also defined in the following way:

$$P(\text{true}|\Theta) = \prod_{d_{ab}\in\Theta} P(d_{ab}|\text{true}) \quad (5)$$

for which we enforce the condition:

$$\sum_{d_{ab}\in\{d_{xy}|d_x\in\mathbf{E},\, d_y\in\mathbf{F}\}} P(d_{ab}|\text{true}) = 1 \quad (6)$$

Using equations 2, 3 and 5 we deduce the final, computation-ready EM equation

$$\begin{aligned}\Theta_{n+1} &= \\ &= \underset{\Theta}{\operatorname{argmax}}\left[\ln \prod_{d_{ab}\in\Theta} \frac{\sum_{w_{ij}\in X} P(d_{ab}|w_{ij})}{|\mathbf{E}||\mathbf{F}|}\right. \\ &\left. + \ln \prod_{d_{ab}\in\Theta} P(d_{ab}|\text{true})\right] \\ &= \underset{\Theta}{\operatorname{argmax}} \sum_{d_{ab}\in\Theta}\left[\ln \frac{\sum_{w_{ij}\in X} P(d_{ab}|w_{ij})}{|\mathbf{E}||\mathbf{F}|}\right. \\ &\left. + \ln P(d_{ab}|\text{true})\right]\end{aligned} \quad (7)$$

As it is, equation 7 suggests an exhaustive search *in the set of all possible Θ parameters*, in order to find the parameter(s) for which the expression that is the argument of "argmax" is maximum. But, as we know from section 3, the size of this this set is prohibitive to the attempt of enumerating each Θ assignment and computing the expectation expression. Our quick solution to this problem was to directly construct the "best" Θ assignment[2] using a

[2] We did not attempt to find the mathematical maximum of the expression from equation 7 and we realize that the conse-

greedy algorithm: simply iterate over all possible 1:1 document pairs and for each document pair $d_{ab} \in \{d_{xy} | d_x \in \mathbf{E},\ d_y \in \mathbf{F}\}$, compute the alignment count (it's not a probability so we call it a "count" following IBM-1 model's terminology)

$$\ln \frac{\sum_{w_{ij} \in X} P(d_{ab}|w_{ij})}{|\mathbf{E}||\mathbf{F}|} + \ln P(d_{ab}|\text{true})$$

Then, construct the best 1:1 assignment Θ_{n+1} by choosing those pairs d_{ab} for which we have counts with the maximum values. Before this cycle (which is the basic EM cycle) is resumed, we perform the following updates:

$$P(d_{ab}|\text{true}) \leftarrow P(d_{ab}|\text{true}) + \frac{\sum_{w_{ij} \in X} P(d_{ab}|w_{ij})}{|\mathbf{E}||\mathbf{F}|} \quad (7a)$$

$$P(d_{ab}|w_{ij}) \leftarrow \sum_{d_{xy} \in \Theta_{n+1}} P(d_{xy}|w_{ij}) \quad (7b)$$

and normalize the two probability tables with equations 6 and 4. The first update is to be interpreted as the contribution the lexical document alignment probability makes to the alignment probability. The second update equation aims at boosting the probability of a translation equivalent if and only if it is found in a pair of documents belonging to the best assignment so far. In this way, we hope that the updated translation equivalent will make a better contribution to the discovery of a correct document alignment that has not yet been discovered at step *n* + 1.

Before we start the EM iterations, we need to initialize the probability tables $P(d_{ab}|\text{true})$ and $P(d_{ab}|w_{ij})$. For the second table we used the GIZA++ scores that we have for the w_{ij} translation equivalents pairs and normalized the table with equation 4. For the first probability table we have (and tried) two choices:

- (**D1**) a uniform distribution: $\frac{1}{|\mathbf{E}||\mathbf{F}|}$;
- (**D2**) a lexical document alignment measure $L(d_{ab})$ (values between 0 and 1) that is computed directly from a pair of documents d_{ab} using the w_{ij} translation equivalents pairs from the dictionary **T**:

$$L(d_{ab}) = \frac{\sum_{w_i \text{ in } d_a} f_{d_a}(w_i) \sum_{w_j \text{ in } d_b} f_{d_b}(w_j)}{|d_a||d_b|} \quad (8)$$

where $|d_a|$ is the number of words in document d_a and $f_{d_a}(w_i)$ is the frequency of word w_i in document d_a (please note that, according to section 3, w_{ij} *is not* a random pair of words, but a pair of *translation equivalents*). If every word in the source document has at least one translation (of a given threshold probability score) in the target document, then this measure is 1. We normalize the table initialized using this measure with equation 6.

EMACC finds only 1:1 textual units alignments in its present form but a document pair d_{ab} can be easily extended to a document bead following the example from (Chen, 1993). The main difference between the algorithm described by Chen and ours is that the search procedure reported there is invalid for comparable corpora in which no pruning is available due to the nature of the corpus. A second very important difference is that Chen only relies on lexical alignment information, on the parallel nature of the corpus and on sentence lengths correlations while we add the probability of the whole assignment which, when initially set to the D2 distribution, produces a significant boost of the precision of the alignment.

4 Experiments and Evaluations

The test data for document alignment was compiled from the corpora that was previously collected in the ACCURAT project[3] and that is known to the project members as the "Initial Comparable Corpora" or ICC for short. It is important to know the fact that ICC contains all types of comparable corpora from parallel to weakly comparable documents but we classified document pairs in three classes: parallel (class name: **p**), strongly comparable (**cs**) and weakly comparable (**cw**). We have considered the following pairs of languages: English-Romanian (en-ro), English-Latvian (en-lv), English-Lithuanian (en-lt), English-Estonian (en-et), English-Slovene (en-sl) and English-Greek

quence of this choice and of the greedy search procedure is not finding the true optimum.

[3] http://www.accurat-project.eu/

(en-el). For each pair of languages, ICC also contains a Gold Standard list of document alignments that were compiled by hand for testing purposes.

We trained GIZA++ translation lexicons for every language pair using the DGT-TM[4] corpus. The input texts were converted from their Unicode encoding to UTF-8 and were tokenized using a tokenizer web service described by Ceauşu (2009). Then, we applied a parallel version of GIZA++ (Gao and Vogel, 2008) that gave us the translation dictionaries of content words only (nouns, verbs, adjective and adverbs) at wordform level. For Romanian, Lithuanian, Latvian, Greek and English, we had lists of inflectional suffixes which we used to stem entries in respective dictionaries and processed documents. Slovene remained the only language which involved wordform level processing.

The accuracy of EMACC is influenced by three parameters whose values have been experimentally set:

- the threshold over which we use translation equivalents from the dictionary **T** for textual unit alignment; values for this threshold (let's name it **ThrGiza**) are from the ordered set {0.001,0.4,0.8};
- the threshold over which we decide to update the probabilities of translation equivalents with equation 7b; values for this threshold (named **ThrUpdate**) are from the same ordered set {0.001,0.4,0.8};
- the top **ThrOut**% alignments from the best assignment found by EMACC. This parameter will introduce precision and recall with the "perfect" value for recall equal to **ThrOut**%. Values for this parameter are from the set {0.3,0.7,1}.

We ran EMACC (10 EM steps) on every possible combination of these parameters for the pairs of languages in question on both initial distributions D1 and D2. For comparison, we also performed a baseline document alignment using the greedy algorithm of EMACC with the equation 8 supplying the document similarity measure. The following 4 tables report a synthesis of the results we have obtained which, because of the lack of space, we cannot give in full. We omit the results of EMACC with D1 initial distribution because the accuracy figures (both precision and recall) are always lower (10-20%) than those of EMACC with D2.

cs	**P**/R	Prms.	P/**R**	Prms.	#
en-ro	1/ 0.69047	0.4 0.4 0.7	0.85714/ 0.85714	0.4 0.4 1	42
en-sl	0.96666/ 0.28807	0.4 0.4 0.3	0.83112/ 0.83112	0.4 0.4 1	302
en-el	0.97540/ 0.29238	0.001 0.8 0.3	0.80098/ 0.80098	0.001 0.4 1	407
en-lt	0.97368/ 0.29191	0.4 0.8 0.3	0.72978/ 0.72978	0.4 0.4 1	507
en-lv	0.95757/ 0.28675	0.4 0.4 0.3	0.79854/ 0.79854	0.001 0.8 1	560
en-et	0.88135/ 0.26442	0.4 0.8 0.3	0.55182/ 0.55182	0.4 0.4 1	987

Table 1: EMACC with D2 initial distribution on strongly comparable corpora

cs	**P**/R	Prms.	P/**R**	Prms.	#
en-ro	1/ 0.69047	0.4 0.7	0.85714/ 0.85714	0.4 1	42
en-sl	0.97777/ 0.29139	0.001 0.3	0.81456/ 0.81456	0.4 0.1	302
en-el	0.94124/ 0.28148	0.001 0.3	0.71851/ 0.71851	0.001 1	407
en-lt	0.95364/ 0.28514	0.001 0.3	0.72673/ 0.72673	0.001 1	507
en-lv	0.91463/ 0.27322	0.001 0.3	0.80692/ 0.80692	0.001 1	560
en-et	0.87030/ 0.26100	0.4 0.3	0.57727/ 0.57727	0.4 1	987

Table 2: D2 baseline algorithm on strongly comparable corpora

cw	**P**/R	Prms.	P/**R**	Prms.	#
en-ro	1/ 0.29411	0.4 0.001 0.3	0.66176/ 0.66176	0.4 0.001 1	68
en-sl	0.73958/ 0.22164	0.4 0.4 0.3	0.42767/ 0.42767	0.4 0.4 1	961
en-el	0.15238/ 0.04545	0.001 0.8 0.3	0.07670/ 0.07670	0.001 0.8 1	352
en-lt	0.55670/ 0.16615	0.4 0.8 0.3	0.28307/ 0.28307	0.4 0.8 1	325
en-lv	0.23529/ 0.07045	0.4 0.4 0.3	0.10176/ 0.10176	0.4 0.4 1	511
en-et	0.59027/ 0.17634	0.4 0.8 0.3	0.27800/ 0.27800	0.4 0.8 1	483

Table 3: EMACC with D2 initial distribution on weakly comparable corpora

[4] http://langtech.jrc.it/DGT-TM.html

cw	**P**/R	Prms.	P/**R**	Prms.	#
en-ro	0.85/ 0.25	0.4 0.3	0.61764/ 0.61764	0.4 1	68
en-sl	0.65505/ 0.19624	0.4 0.3	0.39874/ 0.39874	0.4 1	961
en-el	0.11428/ 0.03428	0.4 0.3	0.06285/ 0.06285	0.4 1	352
en-lt	0.60416/ 0.18012	0.4 0.3	0.24844/ 0.24844	0.4 1	325
en-lv	0.13071/ 0.03921	0.4 0.3	0.09803/ 0.09803	0.4 1	511
en-et	0.48611/ 0.14522	0.001 0.3	0.25678/ 0.25678	0.4 1	483

Table 4: D2 baseline algorithm on weakly comparable corpora

In every table above, the **P**/R column gives the maximum precision and the associated recall EMACC was able to obtain for the corresponding pair of languages using the parameters (**Prms.**) from the next column. The P/**R** column gives the maximum recall with the associated precision that we obtained for that pair of languages.

The **Prms.** columns contain parameter settings for EMACC (see Tables 1 and 3) and for the D2 baseline algorithm (Tables 2 and 4): in Tables 1 and 3 values for `ThrGiza`, `ThrUpdate` and `ThrOut` are given from the top (of the cell) to the bottom and in Tables 2 and 4 values of `ThrGiza` and `ThrOut` are also given from top to bottom (the `ThrUpdate` parameter is missing because the D2 baseline algorithm does not do re-estimation). The # column contains the size of the test set: the number of documents in each language that have to be paired. The search space is # * # and the gold standard contains # pairs of human aligned document pairs.

To ease comparison between EMACC and the D2 baseline for each type of corpora (strongly and weakly comparable), we grayed maximal values between the two: either the precision in the **P**/R column or the recall in the P/**R** column.

In the case of strongly comparable corpora (Tables 1 and 2), we see that the benefits of re-estimating the probabilities of the translation equivalents (based on which we judge document alignments) begin to emerge with precisions for all pairs of languages (except en-sl) being better than those obtained with the D2 baseline. But the real benefit of re-estimating the probabilities of translation equivalents along the EM procedure is visible from the comparison between Tables 3 and 4. Thus, in the case of weakly comparable corpora, in which EMACC with the D2 distribution is clearly better than the baseline (with the only exception of en-lt precision), due to the significant decrease in the lexical overlap, the EM procedure is able to produce important alignment clues in the form of re-estimated (bigger) probabilities of translation equivalents that, otherwise, would have been ignored.

It is important to mention the fact that the results we obtained varied a lot with values of the parameters `ThrGiza` and `ThrUpdate`. We observed, for the majority of studied language pairs, that lowering the value for `ThrGiza` and/or `ThrUpdate` (0.1, 0.01, 0.001…), would negatively impact the performance of EMACC due to the fact of *introducing noise* in the initial computation of the D2 distribution and also on *re-estimating (increasing) probabilities for irrelevant translation equivalents*. At the other end, increasing the threshold for these parameters (0.8, 0.85, 0.9…) would also result in performance decreasing due to the fact that *too few translation equivalents (be they all correct) are not enough to pinpoint correct document alignments* since there are great chances for them to actually appear in all document pairs.

So, we have experimentally found that there is a certain balance between *the degree of correctness of translation equivalents* and *their ability to pinpoint correct document alignments*. In other words, the paradox resides in the fact that if a certain pair of translation equivalents is not correct but the respective words appear only in documents which correctly align to one another, that pair is very important to the alignment process. Conversely, if a pair of translation equivalents has a very high probability score (thus being correct) but appears in almost every possible pair of documents, that pair is not informative to the alignment process and must be excluded. We see now that the EMACC aims at finding the set of translation equivalents that is maximally informative with respect to the set of document alignments.

We have introduced the `ThrOut` parameter in order to have better precision. This parameter actually instructs EMACC to output only the top (according to the alignment score probability $P(d_{ab}|\text{true})$) `ThrOut`% of the document alignments it has found. This means that, if all are correct, the maximum recall can only be `ThrOut`%.

But another important function of `ThrOut` is to restrict the translation equivalents re-estimation (equation 7b) for only the top `ThrOut`% alignments. In other words, only the probabilities of translation equivalents that are to be found in top `ThrOut`% best alignments in the current EM step are re-estimated. We introduced this restriction in order to confine translation equivalents probability re-estimation to correct document alignments found so far.

Regarding the running time of EMACC, we can report that on a cluster with a total of 32 CPU cores (4 nodes) with 6-8 GB of RAM per node, the total running time is between 12h and 48h per language pair (about 2000 documents per language) depending on the setting of the various parameters.

5 Conclusions

The whole point in developing textual unit alignment algorithms for comparable corpora is to be able to provide good quality quasi-aligned data to programs that are specialized in extracting parallel data from these alignments. In the context of this paper, the most important result to note is that translation probability re-estimation is a good tool in discovering new correct textual unit alignments in the case of weakly related documents. We also tested EMACC at the alignment of 200 parallel paragraphs (small texts of no more than 50 words) for all pairs of languages that we have considered here. We can briefly report that the results are better than the strongly comparable document alignments from Tables 1 and 2 which is a promising result because one would think that a significant reduction in textual unit size would negatively impact the alignment accuracy.

Acknowledgements

This work has been supported by the ACCURAT project (http://www.accurat-project.eu/) funded by the European Community's Seventh Framework Program (FP7/2007-2013) under the Grant Agreement n° 248347. It has also been partially supported by the Romanian Ministry of Education and Research through the STAR project (no. 742/19.01.2009).

References

Borman, S. 2009. The Expectation Maximization Algorithm. A short tutorial. Online at: http://www.isi.edu/natural-language/teaching/cs562/2009/readings/B06.pdf

Brown, P. F., Lai, J. C., and Mercer, R. L. 1991. Aligning sentences in parallel corpora. In Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics, pp. 169–176, June 8-21, 1991, University of California, Berkeley, California, USA.

Brown, P. F., Pietra, S. A. D., Pietra, V. J. D., and Mercer, R. L. 1993. The Mathematics of Statistical Machine Translation: Parameter Estimation. Computational Linguistics, 19(2): 263—311.

Ceaușu, A. 2009. Statistical Machine Translation for Romanian. PhD Thesis, Romanian Academy (in Romanian).

Chen, S. F. 1993. Aligning Sentences in Bilingual Corpora Using Lexical Information. In Proceedings of the 31st Annual Meeting on Association for Computational Linguistics, pp. 9—16, Columbus, Ohio, USA.

Dempster, A. P., Laird, N. M., and Rubin, D. B. 1977. Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society, 39(B):1—38.

Fung, P., and Cheung, P. 2004. Mining Very-Non-Parallel Corpora: Parallel Sentence and Lexicon Extraction via Bootstrapping and EM. In Proceedings of EMNLP 2004, Barcelona, Spain: July 2004.

Gao, Q., and Vogel, S. 2008. Parallel implementations of word alignment tool. ACL-08 HLT: Software Engineering, Testing, and Quality Assurance for Natural Language Processing, pp. 49—57, June 20, 2008, The Ohio State University, Columbus, Ohio, USA.

Munteanu, D. S., and Marcu, D. 2002. Processing comparable corpora with bilingual suffix trees. In Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002), pp. 289–295, July 6-7, 2002, University of Pennsylvania, Philadelphia, USA.

Munteanu, D. S., and Marcu, D. 2005. Improving machine translation performance by exploiting non-parallel corpora. Computational Linguistics, 31(4):477–504.