High-Accuracy Phrase Translation Acquisition Through Battle-Royale Selection

Lionel Nicolas Egon W. Stemle Klara Kranebitter Verena Lyding
Institute for Specialised Communication and Multilingualism,
European Academy of Bozen/Bolzano
{lionel.nicolas,egon.stemle,klara.kranebitter,verena.lyding}@eurac.edu

Abstract

In this paper, we report on an unsupervised greedy-style process for acquiring phrase translations from sentence-aligned parallel corpora. Thanks to innovative selection strategies, this process can acquire multiple translations without size criteria, i.e. phrases can have several translations, can be of any size, and their size is not considered when selecting their translations. Even though the process is in an early development stage and has much room for improvements, evaluation shows that it yields phrase translations of high precision that are relevant to machine translation but also to a wider set of applications including memory-based translation or multi-word acquisition.

1 Introduction

This paper reports on work in progress to acquire contiguous phrase translations from sentence-aligned parallel corpora in an unsupervised way.

The described process has three key features: it allows to acquire multiple translations for each phrase, the acquired translations can comprise phrases of any length,[1] and it does not rely on any relation between the sizes of the phrases (no *fertility* criteria). In addition, its performance, especially its precision, allows for competition with the state-of-the-art. Furthermore, the acquired phrase translations can be used for performing machine translation, and memory-based translation; phrase/word alignment; multi-word, paraphrase, and synonymy acquisition; and error correction.

The process starts by generating an exhaustive set of candidate translations and coarsely filters them. It then provides the remaining set to a greedy fine-grained selection that processes one candidate translation at each iteration. The iteration stops when no candidate translations remain.

The main contributions of this paper are (1) to introduce a set of filters for the coarse filtering of candidate translations, and (2) to describe a greedy-style process for performing a fine-grained selection of translations.

In section 2 and 6, we describe the state-of-the-art and, in section 3 and 4, the process itself. We then present its results in section 5, compare it with related work in section 6, highlight future works in section 7 and conclude in section 8.

2 Related works

The process described here can be considered in between two lines of approaches: bilingual lexicon acquisition and phrase translations extraction from word alignments or translations.

Methods performing bilingual lexicon acquisition focus on short phrases, mostly with one or two tokens. They generally use association measures to rank candidate translations and apply several thresholds to decide which ones to keep (Gale and Church, 1991; Melamed, 1995; Wu and Xia, 1994). Most association measures used focus on recurrent occurrences, except methods like Widdows et al. (2002) which apply measures from semantic similarity approaches. Some approaches rely on either or both part-of-speech knowledge (Tufis, 2002; Ma et al., 2011) and transliterations (Tsuji and Kageura, 2004). As explained in Melamed (1997), incorrect translations can be generated because some phrases co-occur too often with the correct translation of a phrase[2]. The commonly used counter-measure is to discard a candidate translation in a bitext if it competes with another one with a higher score (Moore, 2001; Melamed, 1997; Melamed, 2000;

[1] We only use a loose maximum length restriction in order to limit exponential computation

[2] These are usually named *indirect associations*.

Proceedings of Recent Advances in Natural Language Processing, pages 516–524,
Hissar, Bulgaria, 7-13 September 2013.

Tsuji and Kageura, 2004; Tufis, 2002; Yamamoto et al., 2003). The evaluation of the extracted lexica is mostly performed by classifying the generated translations into three categories: *wrong*, *correct* and *near misses*.

The line of approaches for extracting phrase translations from word alignments or translations are built on the outputs of the ones performing bilingual lexicon acquisition[3] (Neubig et al., 2011; Tillmann, 2003; Tambouratzis et al., 2012; Venugopal et al., 2003; Vogel, 2005; Moore and Quirk, 2007; Deng and Byrne, 2008; Koehn et al., 2003; DeNero and Klein, 2008). Some methods such as Zettlemoyer and Moore (2007) and Duan et al. (2011) work on top of the others by refining the phrase translations table acquired. While describing each of the numerous methods would go beyond the scope of this paper, we can summarize that most methods apply a similar set of ideas and combine them in a diversified manner. So as to evaluate a phrase translation, they usually combine features such as translation probabilities, expected size of the translation (often called *fertility*), expected position of the translation and number of word alignments included. Apart from the word alignments or translations, few methods rely on additional data such as part-of-speech. Performances are usually evaluated indirectly through the performance of a machine translation tool taking the phrase translations as input.

Since we could not find previous works for a direct comparison, a global one with related work is provided later in sect. 6.

3 Generation of candidates

3.1 Phrase collection

For each bitext $bit : sent\_l1 \parallel sent\_l2$ of the N available bitexts, we tokenize sentences $sent\_l1$ and $sent\_l2$, count their number of tokens and compute the two global values num_tok_l1 and num_tok_l2, i.e. the number of overall tokens in the $l1$ and $l2$ part of the corpus. Then, we add a start-of-sentence -*s*- token and e -*/s*-end-of-sentence -*/s*- one and generate all contiguous phrases in each bitext[4]. For each generated phrase ph of a language $lang$ we register four values.

(1) The number of tokens $size\_ph(ph)$.

(2) The global number of occurrences

$occ\_ph(ph) = \sum_{i=1}^{N\_bit} occ\_b\_ph(bit_i, ph)$

where $occ\_b\_ph(bit, ph)$ is the number of occurrences of ph in a bitext bit.

(3) The left and right diversity $left\_div\_ph(ph)$ and $right\_div\_ph(ph)$, i.e. the size of the set of different tokens/1-grams that occur next to ph.

(4) The value $num\_tok\_opp(ph)$ that corresponds to the number of tokens in the sentences of the other language (not $lang$) for the bitexts in which ph occurs.

We then discard phrases occurring less than min_occ times, i.e. when $occ\_ph(ph) < min\_occ$, and all l1 phrases with more than max_size_l1 tokens[5], i.e. when $size\_ph(ph\_l1) > max\_size\_l1$.

3.2 Candidate translations building

For every bitext $bit : sent\_l1 \parallel sent\_l2$ with l1 phrases $ph\_l1_1..ph\_l1_j$ and l2 phrases $ph\_l2_1..ph\_l2_l$, we compute the Cartesian product $[ct_1 : ph\_l1_1 \parallel ph\_l2_1], .., [ct_k : ph\_l1_j \parallel ph\_l2_l]$. A generated candidate translation $[ct : ph\_l1 \parallel ph\_l2]$ is said to occur in bit and two values are registered.

(1) The set of 1-grams occurring before and after ph_l1 and ph_l2 in the bitext.

(2) The number of occurrences $occ\_ct(bit, ct)$

$$occ\_ct(bit, ct) = min(occ\_bit\_ph(bit, ph\_l1), occ\_bit\_ph(bit, ph\_l2))$$

Once every bitext has been processed, we compute the following values for every candidate translation $[ct : ph\_l1 \parallel ph\_l2]$.

(1) The size of ct.

$size\_ct(ct) = size\_ct(ph\_l1) + size\_ph(ph\_l2)$

(2) The global number of occurrences.

$glob\_occ\_ct(ct) = \sum_{i=1}^{N_{bit}} occ\_ct(bit_i, ct)$

(3) The original relative frequency of ct.

$orig\_freq\_ct(ct) = \frac{occ\_ph(ph\_l1) * occ\_ph(ph\_l2)}{num\_tok\_l1 * num\_tok\_l2}$

(4) The values $num\_occ\_ph(ct, ph\_l1)$ and $num\_occ\_ph(ct, ph\_l2)$, which correspond to the number of occurrences of ph_l1 and ph_l2 in the set of bitexts where ct occurs.

(5) The conditional relative frequency of ct over the set of bitexts where it occurs.

$cond\_freq\_ct(ct) = \frac{num\_occ\_ph(ct, ph\_l1) * num\_occ\_ph(ct, ph\_l2)}{num\_tok\_opp(ph\_l2) * num\_tok\_opp(ph\_l1)}$

(6) The "strength", between 0 and 1, of ct, i.e. the likeliness of ct to be valid.

$str\_ct(ct) = cond\_freq\_ct(ct) - orig\_freq\_ct(ct)$

(7) The values $left\_div\_ct(ph, ct)$ and $right\_div\_ct(ph, ct)$ of ph_l1 and ph_l2, which

[3] The well known IBM models are a popular choice.

[4] The shortest phrase being one token and the longest phrase being the sentence itself.

[5] We do not apply such limits on the l2 phrases so as to not discard valid translations of the kept l1 phrases.

represent the size of the set of the different 1-grams occurring at their left or right side in all bitexts where ct occurs.

(8) The "context diversity" of ct[6].

$$context\_div(ct) = min(left\_div\_ct(ph\_l1, ct), right\_div\_ct(ph\_l1, ct), left\_div\_ct(ph\_l2, ct), right\_div\_ct(ph\_l2, ct))$$

3.3 Coarse filtering

Each candidate translation is submitted to four filters that aims at limiting computation by discarding the least likely ones [7] while leaving the selection of the remaining ones to the more sophisticated and computationally intense *battle-royale* method (see sect. 4).

Occurrence. This filter aims at dealing with candidate translations that combine completely unrelated phrases, i.e. candidate translations resulting from randomness[8]. A candidate translation ct is discarded if:
(1) it occurs in less than min_co_occ bitexts,
(2) in the bitexts where ct occurs, ph_l1 or ph_l2 occurs less than min_co_freq percents of their global number of occurrences.

If $occ\_ph(ph\_l1) * min\_co\_freq > num\_occ\_ph(ct, ph\_l1)$
Or $occ\_ph(ph\_l2) * min\_co\_freq > num\_occ\_ph(ct, ph\_l2)$

Context diversity. This filter has been designed to discard candidate translations that imply occurrences of either ph_l1 or ph_l2 with a limited left or right context.

This usually happens with indirect associations (Melamed, 1997) or candidate translations that combine a phrase with another one that is not the correct translation but includes the correct one. For example, for most occurrences of a candidate translation $[ct : the\ big \parallel la\ grande\ casa]$, the occurrences of $the\ big$ will have a low variability on its right context, i.e. it will almost always be followed by $house$. In order to detect that the context of a phrase ph is limited, we build on the assumption that values $left\_div\_ph(ph)$ and $right\_div\_ph(ph)$ follow a logarithmic curve as $occ\_ph(ph)$ augments. Therefore, the coefficient obtained from dividing the number of different contexts over the number of occurrences should decrease as the number of occurrences increases. Since $occ\_ct(ct)$ is either inferior or equal to both $occ\_ph(ph\_l1)$ and $occ\_ph(ph\_l2)$, the following conditions should be fulfilled:

$$\frac{left\_div\_ct(ph\_l1, ct)}{glob\_occ\_ct(ct)} \geq \frac{left\_div\_ph(ph\_l1)}{glob\_occ\_ph(ph\_l1)}$$
$$\frac{left\_div\_ct(ph\_l2, ct)}{glob\_occ\_ct(ct)} \geq \frac{left\_div\_ph(ph\_l2)}{glob\_occ\_ph(ph\_l2)}$$
$$\frac{right\_div\_ct(ph\_l1, ct)}{glob\_occ\_ct(ct)} \geq \frac{right\_div\_ph(ph\_l1)}{glob\_occ\_ph(ph\_l1)}$$
$$\frac{right\_div\_ct(ph\_l2, ct)}{glob\_occ\_ct(ct)} \geq \frac{right\_div\_ph(ph\_l2)}{glob\_occ\_ph(ph\_l2)}$$

Conditional frequency. This filter relies on the idea that the occurrence of a phrase ph_l1 triggers the occurrence of a translation ph_l2 in the same bitext and vice-versa. The relative frequencies over the bitexts where ct occurs for both phrases should thus be greater than their global frequency. A candidate translation is thus discarded when:

$$\text{If } \frac{num\_occ\_ct\_ph(ct, ph\_l2)}{num\_tok\_opp(ph\_l1)} \leq \frac{glob\_occ\_ph(ph\_l2)}{num\_tok\_l1}$$
$$\text{Or } \frac{num\_occ\_ct\_ph(ct, ph\_l1)}{num\_tok\_opp(ph\_l2)} \leq \frac{glob\_occ\_ph(ph\_l1)}{num\_tok\_l2}$$

Maximum number of translations. This filters limits the number of candidate translations covering a given phrase ph to the $max\_translations$ best ones in term of strength str_ct.

4 Battle-royale selection

This core part of our approach is named after a 2000 Japanese film, the story of which metaphorically matches the approach applied for performing the selection of candidate translations. In this movie, young people are involved in a deadly game where only one is meant to survive. This results in group alliances and group conflicts that evolve as the game progresses. The same idea is applied here, conflicts and alliances are spotted among candidate translations and a greedy algorithm processes one candidate translation at a time. Depending on which one gets processed first, the situation of the remaining related ones can evolve drastically.

So as to illustrate how we spot conflicts and alliances, we provide candidate translations over the dummy English-Italian bitext:
$[the\ big\ house\ is\ new \parallel la\ grande\ casa\ \grave{e}\ nuova]$

4.1 Detecting conflicts

We consider two candidate translations
$ct_a : ph\_l1_a = t_i..t_j \parallel ph\_l2_a = T_k..T_l$

[6] The higher it is, the more likely ct is to be valid.

[7] The values we used for configuration are provided in sect. 5.1.

[8] Usually one of the two phrases is a frequent one.

$ct_b : ph\_l1_b = t_m..t_n \parallel ph\_l2_b = T_o..T_p$
as being in conflict over one or several phrases $confl\_ph$ in a bitext bit when one of the following conditions is not met.

Non-concurrency condition. Two candidate translations should not cover the same phrase. E.g. $[the\ big \parallel la\ grande]$ and $[the\ big \parallel grande]$ conflict over $the\ big$.

$*$ If $ph\_l1_a = ph\_l1_b$ and $ph\_l2_a != ph\_l2_b$
Then $confl\_ph = ph\_l1_a$
$*$ If $ph\_l2_a = ph\_l2_b$ and $ph\_l1_a != ph\_l1_b$
Then $confl\_ph = ph\_l2_a$

Consistent inclusion condition. If a phrase in one language covered by a first candidate translation includes the phrase in the same language covered by a second candidate translation, then the two phrases in the other language should have the same relation. E.g. the two candidate translations $[the\ big \parallel la\ grande]$ and $[the\ big\ house \parallel grande\ casa]$ conflict since $the\ big\ house$ includes $the\ big$ but $la\ grande$ does not include $grande\ casa$.

$*$ If $incl(ph\_l1_a, ph\_l1_b)$ and $!incl(ph\_l2_a, ph\_l2_b)$
Then $confl\_ph = ph\_l1_b$
$*$ If $incl(ph\_l2_a, ph\_l2_b)$ and $!incl(ph\_l1_a, ph\_l1_b)$
Then $confl\_ph = ph\_l2_b$
$*$ If $incl(ph\_l1_b, ph\_l1_a)$ and $!incl(ph\_l2_b, ph\_l2_a)$
Then $confl\_ph = ph\_l1_a$
$*$ If $incl(ph\_l2_b, ph\_l2_a)$ and $!incl(ph\_l1_b, ph\_l1_a)$
Then $confl\_ph = ph\_l2_a$

Consistent overlap condition. We say that two phrases overlap when they share a sub-phrase that spans either the left-most or the right-most token of both phrases. For two candidate translations, if two phrases of the same language overlap then the two phrases in the other language should also overlap. E.g. $[the\ big\ house \parallel la\ grande\ casa]$ and $[house\ is\ new \parallel casa\ è\ nuova]$ do not conflict since they both overlap on $house$ and $casa$ but $[the\ big\ house \parallel la\ grande\ casa]$ and $[house\ is\ new \parallel è\ nuova]$ do conflict since they only overlap on $house$.

$*$ If $exists(t_q..t_r)$
with $(q = m$ and $r = j)$ xor $(q = i$ and $r = n)$
and $incl(ph\_l1_a, t_q..t_r)$ and $incl(ph\_l1_b, t_q..t_r)$
and $!exist(T_s..T_t)$
with $(s = o$ and $t = l)$ xor $(s = k$ and $t = p)$
and $incl(ph\_l2_a, T_s..T_t)$ and $incl(ph\_l2_b, T_s..T_t)$
Then $confl\_ph = t_q..t_r$.

$*$ If $exists(T_s..T_t)$
with $(s = o$ and $t = l)$ xor $(s = k$ and $t = p)$
and $incl(ph\_l2_a, T_s..T_t)$ and $incl(ph\_l2_b, T_s..T_t)$
and $!exist(t_q..t_r)$
with $(q = m$ and $r = j)$ xor $(q = i$ and $r = n)$
and $incl(ph\_l1_a, t_q..t_r)$ and $incl(ph\_l1_b, t_q..t_r)$
Then $confl\_ph = T_s..T_t$.

4.2 Detecting alliances

We consider two candidate translations ct_a and ct_b as being in alliance in a bitext bit if there exist pairs of phrases $[al\_ph_l1, al\_ph_l2]$ that are included or equal to the phrases combined by ct_a and ct_b and if ct_a and ct_b are not in conflict. For example, $[the\ big \parallel la\ grande]$ and $[big\ house \parallel grande\ casa]$ are in alliance because they do not conflict and their phrases both include big and $grande$.

4.3 Rating conflicts

If there are two candidate translations ct_a and ct_b conflicting over a phrase $confl\_ph$, and it occurs more than once in a bitext bit (i.e. $occ(bit, confl\_ph) > 1$), then, as we do not perform word/phrase alignment beforehand, we have no certainty that ct_a and ct_b do conflict over the same occurrences of $confl\_ph$.

For example, if in an English sentence the word car occurs twice but is translated to $macchina$ and $auto$ in the Italian counterpart, the candidate translations $[car \parallel macchina]$ and $[car \parallel auto]$ will be considered as conflicting over car even though they are both correct and cover two different occurrences.

For evaluating the strength of a conflict $conf$ between two candidate translations over a set of phrases $confl\_ph$ in a bitext bit, we compute the probability $ap\_cf(bit, ct, confl)$ that each candidate translation ct does apply on the phrases they conflict over.

$ap\_cf(bit, ct, confl) = max(\frac{occ\_ct(bit,ct)}{occ\_ph(bit,confl\_ph)})$

For two candidate translations ct_a and ct_b with a conflict $confl$ in a bitext bit, if $ap\_cf(bit, ct_b, confl) = 1$, we say that ct_a has a hard-conflict (is fully-incompatible) with ct_b.

For a conflict $confl$ in a bitext bit, we compute the impacts over ct_a and ct_b as:

$imp\_cf(bit, confl, ct_a) = ap\_cf(bit, ct_b, confl) * str\_ct(ct_b)$
$imp\_cf(bit, confl, ct_b) = ap\_cf(bit, ct_a, confl) * str\_ct(ct_a)$

Once all local conflicts of a candidate translation ct are rated, we calculate:

(1) the value $nb\_hard\_confl(ct)$ corresponding to the number of bitexts in which ct has at least one hard-conflict,
(2) the sum $sum\_confl(ct)$ of all imp_cf values of the local conflicts it is involved in,
(3) the value $avg\_confl(ct)$ indicating how much, in average, ct conflicts with other candidate translations.
$avg\_confl(ct) = \frac{sum\_confl(ct)}{occ\_ct(ct)}$

4.4 Rating alliances

For evaluating the strength of an alliance between two candidate translations regarding pairs of phrases $[al\_ph_{l1}, al\_ph_{l2}]$ in a bitext bit, we also compute the probability $ap\_al(bit, ct, al)$ that each candidate translation ct does apply on the phrases on which they are in alliance.
$ap\_al(bit, ct, al) = max(\frac{2 * occ\_ct(bit,ct)}{occ\_ph(bit,al\_ph_{l1}) * occ\_ph(bit,al\_ph_{l2})})$

For an alliance al in a bitext bit, we compute the impacts over ct_a and ct_b as:
$imp\_al(bit, al, ct_a) = ap\_al(bit, ct_b, al) * str\_ct(ct_b)$
$imp\_al(bit, al, ct_b) = ap\_al(bit, ct_a, al) * str\_ct(ct_a)$

Once all local alliances of each candidate translation ct are rated, we calculate:
(1) the sum $sum\_al(ct)$ of all imp_al values of the local alliances ct is involved in,
(2) the value $avg\_al(ct)$ indicating how much, in average, ct is in alliance with other candidate translations.
$avg\_al(ct) = \frac{sum\_al(ct)}{occ\_ct(ct)}$

4.5 Greedy-style selection

We start by computing the value $popularity(ct)$ of each candidate translation ct in order to perform the final selection.
$popularity(ct) = avg\_confl(ct) - str\_ct(ct) - avg\_al(ct)$

We then order the candidate translations according to, by order of importance, their $popularity$ (decrementally), str_c (incrementally), $context\_div$ (incrementally) an $size\_ct$ (incrementally) values[9].

Making use of this sorting procedure, a greedy-style selection is applied to the list of translation candidates that iterates as follows.
(1) Sort the list of candidate translations.
(2) Remove the first candidate translation ct.
(3) If $nb\_hard\_confl(ct) < \frac{occ\_ct(ct)}{2}$, then consider ct as valid and output it.
(4) Regardless of step 3, nullify its conflicts and alliances and update accordingly the avg_confl, avg_al and nb_hard_confl values of the related candidate translations.

At any iteration, even though a correct candidate translation can be ordered among the next candidates to be processed (and thus to be removed), its processing will be postponed as long as the ones with which it conflicts get selected before. Indeed, the more the values avg_confl and nb_hard_confl are updated, the more the candidate translation goes towards the end of the list. The exact opposite behaviour applies to the alliances: the more the values avg_al are updated, the more a candidate translation goes towards the beginning of the list. The later a candidate translation gets selected, the more likely it is to be considered as valid and kept in step 3.

5 Evaluation

5.1 Input corpora and configuration

To perform the evaluation, we used the 90345 bitexts of the *Catex* Corpus (Streiter et al., 2004). This bilingual corpus is a collection of Italian legal texts sentence-aligned with their German translations. Italian and German are a challenging pair since they have distinct word orders and handle gender, number and case in a rather different manner. The average length of Italian and German sentences are 23.2 and 21.8 tokens.

Regarding the thresholds used to coarsely limit the candidate translation generation (see sect. 3.1 and sect. 3.3), we chose very loose thresholds in order to evaluate the potential of the process. Therefore, a phrase had to occur only twice to be considered ($min\_occ = 2$), and, if German, could not have more than 10 tokens ($max\_size\_l1 = 10$). So as to be considered as possible translations, two phrases needed to co-occur in at least two bitexts ($min\_co\_occ = 2$) and co-occur in at least 5% of the bitexts of one another ($min\_co\_app = 0.05$). A phrase was allowed to have at maximum 20 possible translations ($max\_translations = 20$).

The process required 5 days of computation on a modern computer and the memory consumption raised up to 30 GB.

5.2 Formal Evaluation protocol

We decided to evaluate the phrase translations acquired with two metrics: an evaluation metric that we call hereafter *Scalable precision* that intends

[9] If two candidate translations have the same value for a given criterion, the next one is used for sorting.

to be as similar to the measures for evaluating the bilingual lexicon extraction methods described in Melamed (2000) and Moore (2001) and the well-known BLEU metric (Papineni et al., 2002).

We started from a manual evaluation where the evaluator, when necessary, corrects a candidate translation and count the minimum number of tokens $errors(ct)$ that are to be added or deleted in both phrases. For example, a candidate translation $[ct_b : landesgesetz\ vom\ 8.\ november \parallel provinciale\ 8\ novembre]$ requires to add $legge$ at the beginning of the Italian phrase and thus receives a score $errors(ct_b) = 1$. A total of 1000 randomly chosen candidate translations have been evaluated by a trained translator.

We then used the manually corrected candidate translations as gold standard to compute the *BLEU precision* both ways ($l1 \rightarrow l2$ and $l2 \rightarrow l1$) and the $errors(ct)$ values to compute the *Scalable precision* as follows.
$sca\_prec = 1 - \frac{errors(ct)}{size\_ct(ct)}$

5.3 Results

74771 candidate translations were considered as valid by the *battle-royale* selection.

As we can see in Table 1, among the phrases selected (see sect. 3.1), the coverage of the phrases, i.e. the number of phrases with at least one translation, drops quickly as the size of the phrases increases. The coverage is rather equivalent for small phrases of both languages. However, because of the max_size_l1 length threshold that filters out ($l1$) German phrases only, coverage is less important for Italian ($l2$) as the size of the phrases increases.

When studying the results more closely, we observe two phenomena limiting coverage. The first one is when all the translations of a phrase are not originally selected (see sect. 3.1). This happens with low frequency phrases with several translations due, overall, to the different way Italian and German handle gender, number and case. Dealing with lemmas instead of forms would avoid such issue. The second phenomenon limiting coverage is related to word order: contiguous phrases in one language are translated to non-contiguous ones in the other language. Our method does not yet cope with such aspect.

The vast majority of the phrases in both languages were associated with only one translation. However, 2857 phrases in German and 5131 Italian phrases have been associated with multiple translations (respectively 2.3 in average for both language).

As we can see in Table 2, of the candidate translations manually evaluated, 54.6% were perfect and correcting the other ones required to add or delete 2.3 tokens in average. The *Scalable* and the *BLEU precision* are very similar: when considering all candidate translations equivalent in weight ($weight(ct) = 1$), both metrics score an average precision around $83 \sim 85\%$. When we consider the weight of a candidate translation equal to its size multiplied by its number of occurrences ($weight(ct) = size(ct) * glob\_occ\_ct(ct)$), $Scalable_{bis}$ and $BLEU_{bis}$ values, average precision raises up to $93 \sim 94\%$[10].

5.4 Evaluating improvements

As it is designed, the process has the useful property that improving the selection improves both precision and coverage. Indeed, so as to illustrate the idea, we could compare it to the tetris-like task of ordering the content of a box: the more ordered the objects inside the box are, the more objects fit in this limited space. Since the number of phrases to be covered is also finite and since the biggest set of non-conflicting candidate translations should be the set including all correct ones, comparing two versions of the method can be straightforwardly estimated with no gold-standard, by observing if the number of candidate translations acquired has raised.

6 Comparison with related work

As layed out in sect. 2, the approach described here can be situated midway between methods for acquiring bilingual lexicon and methods for extracting phrase translations from word translations and/or alignments.

Comparing our method, on a global perspective, with the ones for acquiring bilingual lexica, we see five main aspects to highlight. First, we are able to acquire much longer phrases. Second, the step of our approach performing candidate translation generation and coarse filtering is similar to the other methods. Third, the threshold we use to validate or discard a candidate translation is

[10] Since we acquire translations instead of generating some, we don't have to deal with word order issues. This also explain why *BLEU* scores are way higher than usually reported in litterature

Phrase Size	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	≥ 20
German cov.	38.2	14.6	10.1	8.5	7.7	7.2	5.3	4.5	2.7	3.5	-	-	-	-	-	-	-	-	-	-
Italian cov.	43	13	8	6.6	6	5.4	4.1	3.3	2.2	1.5	1	0.8	0.4	0.3	0.2	0.1	< 0.1	< 0.1	< 0.1	< 0.1

Table 1: Coverage

Size	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	≥ 18	**Total**
Nb evaluated	115	65	168	87	108	75	79	51	61	40	41	24	22	19	15	6	24	**1000**
Nb perfect	78.3	16.9	76.2	36.8	63.0	41.3	62.0	25.5	50.8	35.0	75.6	45.8	50.0	52.6	13.3	33.3	50.0	**54.6**
Avg errors	1.7	1.5	1.9	2.0	2.2	1.8	3.3	2.7	2.5	3.0	4.0	2.9	2.7	1.7	2.9	2.8	3.9	**2.3**
Scalable	81.7	57.9	88.8	75.2	86.6	84.6	84.5	77.8	87.5	82.5	91.9	87.8	90.3	94.7	84.2	89.2	89.1	**83.2**
Scalable$_{bis}$	99.1	60.3	93.0	79.2	91.2	89.3	86.6	79.3	93.3	84.6	94.5	91.3	91.5	95.5	87.4	89.2	88.7	**93.6**
Bleu	84.5	71.6	91.8	79.5	88.3	84.2	83.9	76.9	85.9	82.1	91.2	86.9	89.0	94.2	86.2	88.2	90.2	**85.2**
Bleu$_{bis}$	99.2	74.1	95.0	83.0	92.5	89.1	85.7	78.1	92.5	84.3	94.1	90.5	90.3	95.4	88.8	88.2	90.0	**94.1**

Table 2: Candidate translations statistics and evaluation

dynamically adjusted and therefore less restrictive and prone to bias than manually set thresholds. Fourth, our *battle-royale* selection implements the selection algorithm used by other methods where concurrency conflicts are considered (Moore, 2001; Melamed, 1997; Melamed, 2000; Tsuji and Kageura, 2004; Tufis, 2002; Yamamoto et al., 2003) and extends it to a more sophisticated level. Fifth, even though a straight comparison with reported results is irrelevant, ours seem competitive and promising both in term of coverage and precision.

Comparing our method, on a global perspective, with the ones for extracting phrase translations from word translations and/or alignments, we see three main aspects to highlight. First, we do not take word alignments or translations as input. We believe that identifying word translations first would lead to diminished results. Indeed, in addition to the size issue, i.e. the translation of a word can have several tokens, translations of longer phrases are sometimes easier to identify than the translations of the phrases they contain. An example of this would be a non-ambiguous phrase containing a polysemous word. We thus aim at considering them all together at the same time. The second aspect to highlight is that we do use a feature similar to translation probabilities (i.e. $strength$ value) but do not directly intend to evaluate the expected size and position of the translation or the alignment of the sub-phrases included. We however indirectly rely on the *battle-royale* selection to exploit these concepts. If the size of a candidate translation, its position or the sub-phrases it includes are not compatible with the other candidate translations, conflicts will arise instead of alliances. The third aspect to highlight does not regard the method itself but the way to evaluate it. Indeed, no methods assessed directly, as we did, the quality of the phrase translations acquired. They were generally evaluated with respect to the differences in performance of a machine translation system. Thus the phrase translations are not themselves evaluated but their impact on a tool is. Unfortunately, evaluating phrase translations with machine translation only allows to evaluate how well machine translation systems manage to take advantage of this data at decoding time. However, it does not allow to evaluate how adequate such data would be for the other tasks that can benefit from such data (see sect. 7.2).

Last but not least, no methods mention the use of the left and right 1-gram of the phrases to filter or select candidate translations.

7 Future work

7.1 Planned improvements

Evaluation. We consider evaluating as we did for ours phrase translations generated by state-of-the-art tools. Also, as in most of the state-of-the-art, we strongly consider evaluating the phrase translations generated through a sophisticated machine translation system such as Moses (Koehn et al., 2007).

Performance. Depending on the configuration and the size of the input corpus, time and memory consumption can easily be a challenge even for modern computers and represent a scalability issue[11]. Parallelising the approach and adapting it to an incremental behavior could help tackling this aspect.

[11] However, since such data should not be generated often and modern HDDs provide decent swapping memory, these aspects are more drawbacks than issues.

Lemmatization. A pre-processing step that converts an input form-based parallel corpus into a lemmatized enhanced one could be added. All occurrences of different phrases with the same sequences of lemmas would be grouped and thus, both the average number of occurrences and the total number of occurrences would be higher[12]. Such improvement should increase both precision and coverage.

Beam-search. The greedy-style *battle-royale* selection can straightforwardly be adapted to a beam search driven by the sum of all the $popularity$ values.

Non-contiguous phrases. The approach could already cope with non-contiguous phrases. However, this would drastically increase the search space.

7.2 Possible applications

Thanks to the high precision achieved, a wider spectrum of applications than mentioned in the related work can be considered.

Machine translation. As proposed in most of the state-of-the-art, the candidate translations generated could be used to achieve machine translation.

Memory based translation. This task could be enhanced by using the candidate translations in a t9-style/auto-completion algorithms and propose typing suggestions. Such tools could both help saving time and standardizing translations.

Word/phrase alignment. Since high precision translations of both words and phrases are generated, a bottom-up or a top-down approach could take advantage of such data.

Multiword detection. A multiword in one language often corresponds to an unique word in another language[13]. Detecting multiwords could thus be achieved by selecting the candidate translation combining a single-token with a multi-token phrase matching certain part-of-speech patterns[14].

Paraphrase/synonyms acquisition. Two phrases that can be translated to the same phrase are possibly semantically equivalent. However, false positives can be generated from polysemous phrases.

Error acquisition. Error correction can be seen as the translation of an incorrect sentence into a correct one. Any parallel corpus for this task could thus be used as input and candidate translations combining two different phrases would represent errors.

[12]The occurrences of non selected phrases could be taken into account in their lemmatized version.

[13]E.g. $pomme\ de\ terre\ (French) \parallel potato\ (English)$ or $landesgesetz\ (German) \parallel legge\ provinciale\ (Italian)$

[14]We expect high precision and, depending on the pair of languages considered, low recall. However, recall could be boosted by combining several pairs of languages, and a phrase labeled as multi-word can always be used in an ad-hoc fashion for further detection.

8 Conclusion

In this paper, we have presented an unsupervised approach that is able to acquire phrase translations with great flexibility.

As it is a recent and on-going work, it has still much room for improvement. However, its performance already allows it to compete with the state-of-the-art.

We provided several tracks for improving it and described a set of applications that can be considered thanks to the precision achieved.

The evaluation performed confirms both its relevance and its potential.

References

John DeNero and Dan Klein. 2008. The complexity of phrase alignment problems. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Short Papers*, pages 25–28. Association for Computational Linguistics.

Yonggang Deng and William Byrne. 2008. Hmm word and phrase alignment for statistical machine translation. volume 16, pages 494–507. IEEE.

Nan Duan, Mu Li, Ming Zhou, and Lei Cui. 2011. Improving phrase extraction via mbr phrase scoring and pruning. In *Proceedings of MT Summit*, volume 13, pages 189–197.

William A. Gale and Kenneth W. Church. 1991. Identifying word correspondence in parallel texts. In *Proceedings of the workshop on Speech and Natural Language*, HLT '91, pages 152–157, Stroudsburg, PA, USA. Association for Computational Linguistics.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 48–54. Association for Computational Linguistics.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi,

Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, ACL '07, pages 177–180, Stroudsburg, PA, USA. Association for Computational Linguistics.

Qing Ma, Shinya Sakagami, and Masaki Murata. 2011. Extraction of broad-scale, high-precision japanese-english parallel translation expressions using lexical information and rules. In *PACLIC*, volume 25, pages 577–586.

I. Dan Melamed. 1995. Automatic evaluation and uniform filter cascades for inducing n-best translation lexicons. In *Proceedings of the Third Workshop on Very Large Corpora*, pages 184–198.

I. Dan Melamed. 1997. Automatic discovery of non-compositional compounds in parallel data. In *Proceedings of the 2nd Conference on Empirical Methods in Natural Language Processing*.

I. Dan Melamed. 2000. Models of translational equivalence among words. *Computational Linguistics*, 26:221–249.

Robert C Moore and Chris Quirk. 2007. An iteratively-trained segmentation-free phrase translation model for statistical machine translation. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 112–119. Association for Computational Linguistics.

Robert C. Moore. 2001. Towards a simple and accurate statistical approach to learning translation relationships among words. In *Proceedings of the workshop on Data-driven methods in machine translation - Volume 14*, DMMT '01, pages 1–8, Stroudsburg, PA, USA. Association for Computational Linguistics.

Graham Neubig, Taro Watanabe, Eiichiro Sumita, Shinsuke Mori, and Tatsuya Kawahara. 2011. An unsupervised model for joint phrase alignment and extraction. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 632–641. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. 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.

Oliver Streiter, Mathias Stuflesser, and Isabella Ties. 2004. Cle, an aligned tri-lingual ladin-italian-german corpus. corpus design and interface. *First Steps in Language Documentation for Minority Languages*, page 84.

George Tambouratzis, Michalis Troullinos, Sokratis Sofianopoulos, and Marina Vassiliou. 2012. Accurate phrase alignment in a bilingual corpus for ebmt systems. In *Proceedings of the 5th BUCC Workshop, held within the LREC2012 Conference*, volume 26, pages 104–111.

Christoph Tillmann. 2003. A projection extension algorithm for statistical machine translation. In *Proceedings of the 2003 conference on Empirical methods in natural language processing*, pages 1–8. Association for Computational Linguistics.

Keita Tsuji and Kyo Kageura. 2004. Extracting low-frequency translation pairs from japanese-english bilingual corpora. In Sophia Ananadiou and Pierre Zweigenbaum, editors, *COLING 2004 CompuTerm 2004: 3rd International Workshop on Computational Terminology*, pages 23–30, Geneva, Switzerland, August 29. COLING.

Dan Tufis. 2002. A cheap and fast way to build useful translation lexicons. In *Proceedings of the 19th International Conference on Computational Linguistics, COLING2002*, pages 1030–1036.

Ashish Venugopal, Stephan Vogel, and Alex Waibel. 2003. Effective phrase translation extraction from alignment models. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, pages 319–326. Association for Computational Linguistics.

Stephan Vogel. 2005. Pesa: Phrase pair extraction as sentence splitting. In *Proceedings of the Machine Translation Summit X*, pages 251–258.

Dominic Widdows, Beate Dorow, and Chiu ki Chan. 2002. Using parallel corpora to enrich multilingual lexical resources. In *Third International Conference on Language Resources and Evaluation*, pages 240–245.

Dekai Wu and Xuanyin Xia. 1994. Learning an english-chinese lexicon from a parallel corpus. In *Proceedings of the First Conference of the Association for Machine Translation in the Americas*, pages 206–213.

Kaoru Yamamoto, Taku Kudo, Yuta Tsuboi, and Yuji Matsumoto. 2003. Learning sequence-to-sequence correspondences from parallel corpora via sequential pattern mining. In *Proceedings of the HLT-NAACL 2003 Workshop on Building and using parallel texts: data driven machine translation and beyond-Volume 3*, pages 73–80. Association for Computational Linguistics.

Luke S Zettlemoyer and Robert C Moore. 2007. Selective phrase pair extraction for improved statistical machine translation. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Companion Volume, Short Papers*, pages 209–212. Association for Computational Linguistics.