Natural Language Engineering **22** (4): 549–573. © Cambridge University Press 2016
doi:10.1017/S1351324916000139

Extracting parallel phrases from comparable data for machine translation†

SANJIKA HEWAVITHARANA[1]
and STEPHAN VOGEL[2]
[1]*Raytheon BBN Technologies, Cambridge, MA 02138, USA*
email: shewavit@bbn.com
[2]*Qatar Computing Research Institute, Doha, Qatar*
email: svogel@qf.org.qa

(*Received 16 December 2014; revised 01 May 2015; accepted 12 July 2015*)

Abstract

Mining parallel data from comparable corpora is a promising approach for overcoming the data sparseness in statistical machine translation and other natural language processing applications. In this paper, we address the task of detecting parallel phrase pairs embedded in comparable sentence pairs. We present a novel phrase alignment approach that is designed to only align parallel sections bypassing non-parallel sections of the sentence. We compare the proposed approach with two other alignment methods: (1) the standard phrase extraction algorithm, which relies on the Viterbi path of the word alignment, (2) a binary classifier to detect parallel phrase pairs when presented with a large collection of phrase pair candidates. We evaluate the accuracy of these approaches using a manually aligned data set, and show that the proposed approach outperforms the other two approaches. Finally, we demonstrate the effectiveness of the extracted phrase pairs by using them in Arabic–English and Urdu–English translation systems, which resulted in improvements upto 1.2 Bleu over the baseline. The main contributions of this paper are two-fold: (1) novel phrase alignment algorithms to extract parallel phrase pairs from comparable sentences, (2) evaluating the utility of the extracted phrases by using them directly in the MT decoder.

1 Introduction

Statistical Machine Translation (SMT), like many natural language processing tasks, relies primarily on parallel corpora. The translation performance of SMT systems directly depends on the quantity and the quality of the available parallel data. However, such corpora are only available in large quantities for a handful of languages, including English, Arabic, Chinese and few other European languages. Much of the available parallel data is derived from parliamentary proceedings (e.g. Europarl, Canadian Hansard, etc.) and proceedings at the United Nations, though a limited amount of newswire text is also available. For most other

†Part of this work was conducted when the authors were affiliated to the Language Technologies Institute, Carnegie Mellon University, Pittsburgh, PA 15213, USA.

https://doi.org/10.1017/S1351324916000139 Published online by Cambridge University Press

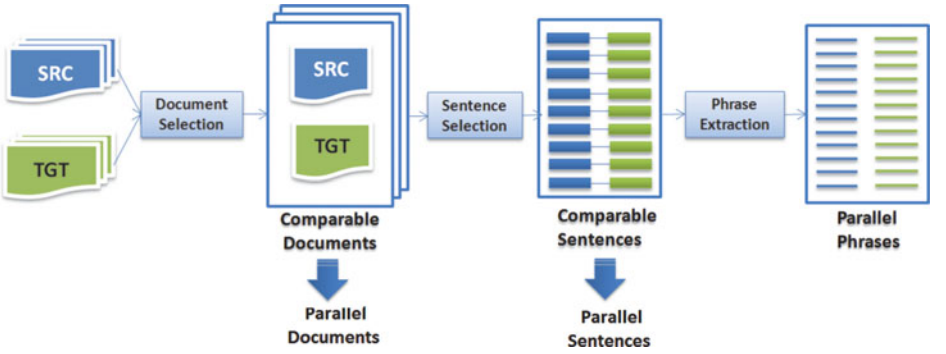


Fig. 1. (Colour online) Parallel data extraction framework.

languages, especially for less commonly used languages, parallel data is virtually non-existent.

Comparable corpora provide a possible solution to this data sparseness problem. Comparable documents are not strictly parallel, but are loose translations of each other, with overlapping information. A good example for comparable documents is the newswire text produced by multilingual news organizations such as AFP or Reuters. The degree of parallelism can vary greatly, ranging from *noisy parallel* documents that contain many parallel sentences, to *quasi parallel* documents that may cover different topics (Fung and Cheung 2004). The web is by far the largest source of comparable data. *Wikipedia* is becoming a more attractive source of comparable documents because of its structured nature.

Comparable corpora may contain parallel data in different levels of granularity. This includes: parallel documents, parallel sentences or parallel sub-sentential fragments. To simplify the process of extracting these different types of translational correspondences and to reduce the computational overhead, it can be divided into a cascade of tasks as shown in Figure 1. First, a document level alignment is identified between the two collections of comparable corpora. This results in a set of comparable document pairs, some of which may already be parallel. In the next step, a sentence level alignment is detected within the identified document pairs. Even if two comparable documents have few or no parallel sentence pairs, there could still be partially parallel sentences, which contain parallel sub-sentential fragments, including word translation pairs, named entities and long phrase pairs. These fragments are detected in the third step. The ability to identify these pairs would create a valuable resource for SMT, especially for low-resource languages.

In this paper, we present a novel phrase alignment approach to detect parallel phrase pairs embedded in comparable sentence pairs. We assume that a collection of comparable sentence pairs is already available at this stage (i.e. the first two steps in the framework in Figure 1), extracted for example using methods similar to Munteanu and Marcu (2005) and Tillmann and Hewavitharana (2013). Our intention is to use the extracted phrases directly in the translation process, along with other phrase pairs extracted from parallel corpora. We compare the proposed approach with two other alignment methods: (1) the standard phrase extraction approach for parallel data, which relies on the Viterbi path of the word alignment,

https://doi.org/10.1017/S1351324916000139 Published online by Cambridge University Press

(2) a binary classifier to detect parallel phrase pairs when presented with a large collection of phrase pair candidates. We evaluate the accuracy of these approaches in detecting alignments for phrase pairs that have a known translation in a comparable sentence pair. Finally, we demonstrate the effectiveness of the extracted phrase pairs by adding them to Arabic–English and Urdu–English translation systems.

The main contributions of this paper are two-fold: (1) novel phrase alignment algorithms to extract parallel phrase pairs from comparable sentences, (2) evaluating the utility of the extracted phrases by using them directly in the MT decoder. The rest of the paper is organized as follows: In Section 2, we discuss previous work on extracting parallel data from comparable corpora. Section 3 introduces the phrase alignment problem in comparable sentences and discusses some of the challenges involved. It presents the proposed novel phrase alignment approach along with several algorithms to achieve it. It also explains other alternative alignment approaches we explore. Section 4 presents the experimental setup, results of evaluating the alignment accuracy and results of translation experiments. We conclude, in Section 5, with an analysis of the results and an outline of the direction for future work.

2 Related work

Many of the early attempts to exploit comparable corpora focused on learning bilingual word translations (Rapp 1995; Fung and Yee 1998; Rapp 1999). The underlying assumption in these efforts is that a word and its translation appear in similar context in the respective languages, and therefore co-occurrence statistics can be used to detect them. Several context similarity measures such as term frequency (*tf*) and inverse document frequency (*idf*) have been studied by Fung and Yee (1998).

Cross-lingual information retrieval has been widely used to detect document level alignments in comparable corpora (Zhao and Vogel 2002b; Utiyama and Isahara 2003; Munteanu and Marcu 2005). Here, a source language document is translated into the target language using a dictionary, and the resulting translation is used as a query to match against a collection of target documents. Resnik and Smith (2003) find parallel article pairs from the web by exploiting the similarities in the URL structure and the structure of meta information in the articles among other clues.

Detecting sentence alignment within comparable documents has been tried in different ways. Zhao and Vogel (2002a) and Utiyama and Isahara (2003) extend parallel sentence alignment algorithms to identify parallel sentence pairs within comparable news corpora. Munteanu and Marcu (2005) and Fung and Cheung (2004) work on non-parallel documents, and do not assume that every sentence on source document has a corresponding translation on the target document. Each source sentence is matched against candidate target sentences and the sentence pairs above a certain similarity measure is picked as parallel. Fung and Cheung (2004) use cosine similarity measure for this task, while Munteanu and Marcu (2005) use a maximum entropy classifier. The former uses an iterative refinement where previously identified sentence pairs are added to the system in later iterations. Munteanu and

https://doi.org/10.1017/S1351324916000139 Published online by Cambridge University Press

Marcu (2005) demonstrates that this method can be successfully used to improve the performance of a machine translation system. Tillmann and Xu (2009) introduce a system that performs both the document selection and sentence alignment tasks in a single run without any document level pre-filtering. Such a system is useful when document level boundaries are not available in the comparable corpus.

The first attempt to detect sub-sentential fragments from comparable sentences was proposed by Munteanu and Marcu (2006). After identifying sentence pairs that have at least partial overlap, the parallel fragments are detected using an approach inspired by signal processing. Quirk, Udupa and Menezes (2007) later extended this work by introducing generative models that explicitly aligns comparable sentence pairs, including non-parallel sections. They showed improvements in translation performance when the method was applied on cross-domain test data. Kumano, Tanaka and Tokunaga (2007) have proposed a phrasal alignment approach for comparable corpora using the joint probability SMT model. While this approach is appealing for low-resource scenarios as it does not require any seed parallel corpus, the high computational cost is a deterrent in its applicability to large corpora. Tillmann and Hewavitharana (2011; 2013) present a single unified search algorithm that can be used flexibly on all the different extraction tasks handled by the various algorithms cited above. Translation pair extraction is handled as a chunk-alignment problem with no document-level pre-filtering, and data processed directly at the sentence level. The results indicate significant gains by using the extracted parallel data, especially when using the extracted fragments. Sub-sentential fragment extraction has been explored recently for low-resource languages (Gupta, Pal and Bandyopadhyay 2013).

In all the approaches cited above, the extracted fragment data was used as additional training data to train alignment models. In contrast, this work focuses on extracting phrase pairs that can be directly used in the translation process, complementing a phrase table generated with existing parallel data. In our previous work (Hewavitharana and Vogel 2011; Hewavitharana and Vogel 2013), we presented initial results on extracting phrase pairs from comparable sentences. This paper expands on previous work by presenting several phrase alignment algorithms that achieve this task. We also present results from translation experiments that use the extracted phrase pairs under several settings to improve translation performance.

3 Parallel phrase extraction

Figure 2 shows three sample sentences that were extracted from Gigaword Arabic and Gigaword English collections. For each comparable sentence pair, the Arabic sentence is shown first, followed by its correct English translation (in Italics). The extracted comparable English sentence is shown next. The parallel sections in each sentence are marked in boldface. In the first two sentences pairs, the English sentence contains the full translation of the Arabic sentence, but there are additional phrases on the English side that are not present on the Arabic sentence. These phrases appear at the beginning of sentence 1 and at the end of sentence 2. In sentence 3, there are parallel phrases as well as phrases that appear only on one side. The

https://doi.org/10.1017/S1351324916000139 Published online by Cambridge University Press

1.

واضاف انها تهدف لصرف انتباه الراي العام عن الاعمال الوحشية المتزايدة التي يرتكبها النظام الاسرائيلي ضد الفلسطينيين في الاراضي المحتلة

[He] added that it aims to divert public attention from the growing atrocities committed by the Israeli regime against the Palestinians in the occupied territories.

" Iran considers these remarks as interference in its internal affairs , " Kharazi said , **adding that they are aimed at detracting public opinion from heightened atrocities committed by the Israeli regime against the Palestinians in occupied lands .**

2.

واضاف " لكن حتي الان لم نواجه مشكلات "

But "Until now we did not have problems"

" but up to now , we didn't meet any problems ; the afghan people are very kind to us , " he said.

3.

تعد هذه هي اول زيارة لموسي على العراق منذ توليه الامانة العامة للجامعة العربية في مايو الماضي

This is the first visit by Moussa** to Iraq, **since he became the General Secretary of the Arab League in last May.

This was also the first such visit by Moussa himself , the former Egyptian foreign minister , **since he assumed the post as AL chief in may last year .**

Fig. 2. Sample comparable sentences that contain parallel phrases.

phrase 'to Iraq' appears only on the Arabic sentence while the phrase 'the former Egyptian foreign minister' appears only on the English side.

Standard word alignment and phrase alignment algorithms are formulated to work on parallel sentence pairs. Therefore, these standard algorithms are ill suited to operate on partially parallel sentence pairs. Presence of non-parallel phrases may result in undesirable word alignments. Figure 3 illustrates this phenomenon. It compares a typical word alignment pattern in a parallel sentence pair (a) to one in a non-parallel sentence pair (b). The darkness of a square indicates the strength of the word alignment probability between the corresponding word pair (i.e. Viterbi path). In Figure 3(a), we observe high probability word-to-word alignments (dark squares) over the entire length of the sentences. In Figure 3(b), we see one dark area above 'weapons of mass destruction', corresponding to the parallel phrase pair, and some scattered dark spots, where high frequency English words pair with high frequency Arabic words. These spurious alignments pose problems to the phrase alignment, and indicate that word alignment probabilities alone might not be sufficient to produce reliable phrase alignments in comparable sentences pairs.

Our aim in this paper is to address the issue of aligning comparable sentence pairs that contain parallel as well as non-parallel sections. As illustrated in Figure 3, the standard phrase extraction algorithms designed for parallel sentences fail when faced with non-parallel sentences. In this regard, in the following sections of the paper, we propose a novel phrase alignment approach to detect parallel phrases in comparable sentence pairs and compare it against other alternative approaches.

3.1 Non-Viterbi (PESA) alignment

A phrase alignment algorithm called 'PESA' that does not rely on the Viterbi path is described in Vogel (2005). PESA identifies the boundaries of the target phrase

https://doi.org/10.1017/S1351324916000139 Published online by Cambridge University Press

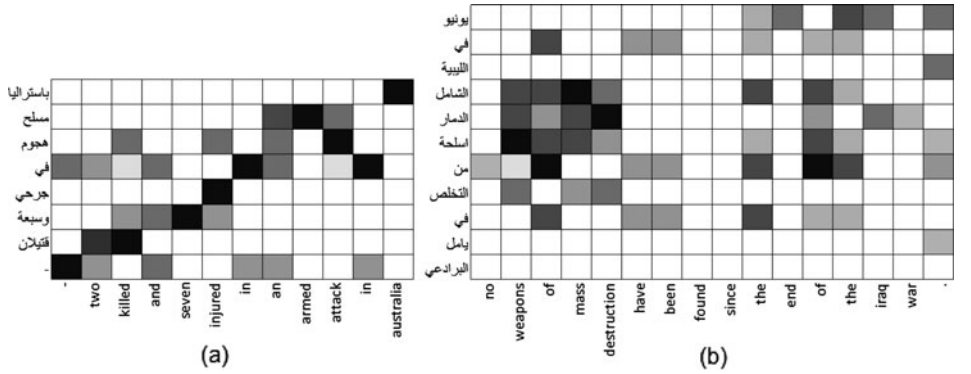


Fig. 3. Word-to-word alignment pattern for (a) a parallel sentence pair and (b) a non-parallel sentence pair.

by aligning words inside the source phrase with words inside the target phrase, and similarly for the words outside the boundaries of the phrase pair. It does not attempt to generate phrase alignments for the full sentence. Rather, it identifies the best target phrase that matches a given source phrase. PESA requires a statistical word-to-word lexicon. A seed parallel corpus is required to automatically build this lexicon.

The approach followed by PESA is particularly well suited in extracting phrase pairs from comparable sentences due to several reasons:

- It is designed to not generate a complete word alignment for the entire sentence, but to find only the target side of a source phrase embedded in the sentence. This provides the potential to devise alignment techniques, which only focuses on the parallel sections of a comparable sentence pair, ignoring non-parallel sections.
- The alignment algorithm can be bootstrapped using a lexical translation table or a bilingual dictionary. This is particularly appealing in low-resource scenarios, where available initial parallel resources may be limited.
- The ability to use as an on-demand phrase extraction method at run-time, which does not require prior extraction of phrase pairs. This method can readily be used as a way of finding auxiliary phrase pairs as needed by the translation system, when the baseline phrase table does not cover it.

We briefly explain the PESA alignment approach below. Instead of searching for all possible phrase alignments in a parallel sentence pair, this approach finds the alignment for a single source phrase $S = s_1 \dots s_l$. Assume that we have a parallel sentence pair (s_1^J, t_1^I) which contains the source phrase S in the source sentence s_1^J. We want to find the target phrase $T = t_1 \dots t_k$ in the target sentence t_1^I which is the translation of the source phrase.

A constrained IBM Model-1 alignment is now applied as follows:

- Source words inside the phrase boundary are aligned only with the target words inside the phrase boundary. Source words outside the phrase boundary are only aligned with target words outside the phrase boundary.

https://doi.org/10.1017/S1351324916000139 Published online by Cambridge University Press

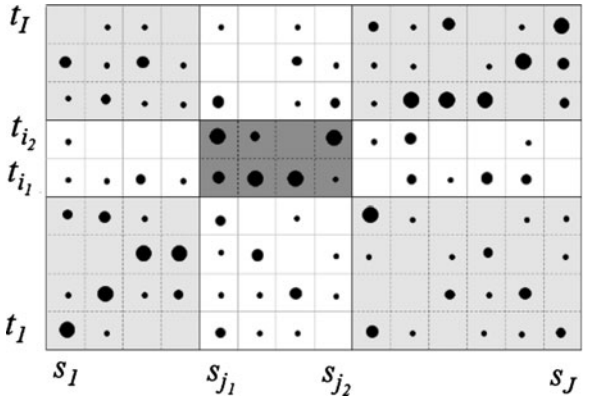


Fig. 4. PESA phrase alignment.

- Position alignment probability for the sentence, which is $1/I$ in IBM Model-1, is modified to be $1/k$ inside the source phrase and to $1/(I-k)$ outside the phrase.

Figure 4 shows the different regions. Given the source sentence and the source phrase from position j_1 to j_2, we want to find the boundaries of the target phrase, i_1 and i_2. The dark area in the middle is the phrase we want to align. The size of the blobs in each box indicates the lexical strength of the word pair.
The constrained alignment probability is calculated as follows:

$$\begin{aligned} p(s|t) = & \left(\prod_{j=1}^{j_1-1} \sum_{i \notin (i_1 \ldots i_2)} \frac{1}{I-k} p(s_j|t_i) \right) \\ & \times \left(\prod_{j=j_1}^{j_2} \sum_{i=i_1}^{i_2} \frac{1}{k} p(s_j|t_i) \right) \\ & \times \left(\prod_{j=j_2+1}^{J} \sum_{i \notin (i_1 \ldots i_2)} \frac{1}{I-k} p(s_j|t_i) \right) \end{aligned} \tag{1}$$

$p(t|s)$ is similarly calculated by switching source and target sides in equation (1):

$$\begin{aligned} p(t|s) = & \left(\prod_{i=1}^{i_1-1} \sum_{i \notin (j_1 \ldots j_2)} \frac{1}{J-l} p(t_i|s_j) \right) \\ & \times \left(\prod_{i=i_1}^{i_2} \sum_{j=j_1}^{j_2} \frac{1}{l} p(t_i|s_j) \right) \\ & \times \left(\prod_{i=i_2+1}^{I} \sum_{j \notin (j_1 \ldots j_2)} \frac{1}{J-l} p(t_i|s_j) \right) \end{aligned} \tag{2}$$

To find the optimal target phrase boundaries, we interpolate the two probabilities in equations (1) and (2) and select the boundary (i_1, i_2) that gives the highest probability,

https://doi.org/10.1017/S1351324916000139 Published online by Cambridge University Press

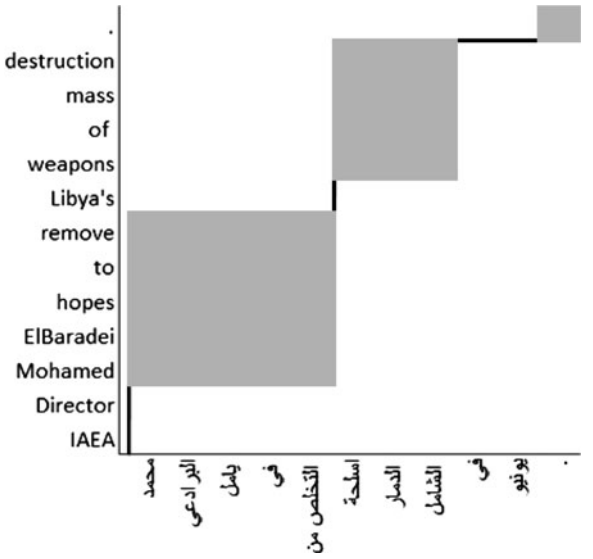


Fig. 5. Phrase alignment for comparable sentences.

as in equation (3).

$$(i_1, i_2) = \underset{i_1, i_2}{\text{argmax}} \ \{(1-\lambda)\ log(p(s|t)) + \ \lambda\ log(p(t|s))\} \tag{3}$$

The value of λ is estimated using held-out data.

PESA can be used to identify all possible phrase pairs in a given parallel sentence pair by iterating over all possible n-gram in the source sentence. An important difference is that each phrase is found independently of any other phrase pair, whereas in the standard phrase extraction they are tied through the word alignment of the sentence pair.

3.2 Extending PESA alignment for comparable sentences

We want to extract parallel phrases from comparable sentence pairs using the non-Viterbi PESA alignment. The sentence pair (f_1^J, e_1^I) now contains parallel sections as well as non-parallel sections. Figure 5 illustrates an example of such a sentence. The gray squares indicate parallel phrases that are common to both sentences. Black lines indicate non-parallel phrases that are present only on one side.

We extend PESA phrase alignment algorithm to comparable sentences, by explicitly modeling the non-parallel sections of the sentence.

Following Quirk *et al.* (2007), we hypothesize that a monolingual model, such as an n-gram language model, is better suited to model the non-parallel sections of the comparable sentence pair. Conversely, a bilingual translation model will better model the parallel sections than a language model. i.e. if the phrase pair $(\tilde{f}, \tilde{e})$ is parallel, then $P(\tilde{f}, \tilde{e}) > P(\tilde{f}) \cdot P(\tilde{e})$.

We use two monolingual models, L_S for the source language and L_T for the target language, and two translation models, X_{ST} for source-to-target direction and X_{TS} for target-to-source direction.

https://doi.org/10.1017/S1351324916000139 Published online by Cambridge University Press

Using Bayes' rule, we can break down the join probability $Pr(\tilde{f},\tilde{e})$ into two decompositions of translation model and language model.

$$Pr(\tilde{f},\tilde{e}) = Pr(\tilde{e}) \cdot Pr(\tilde{f}|\tilde{e}) = Pr(\tilde{f}) \cdot Pr(\tilde{e}|\tilde{f})$$

However, our models trained on limited amounts of data are only approximations. Therefore, the equation will not strictly be true and the expressions $Pr(\tilde{e}) \cdot Pr(\tilde{f}|\tilde{e})$ and $Pr(\tilde{f}) \cdot Pr(\tilde{e}|\tilde{f})$ may differ. Hence, we take the average of the two estimates. Thus, the probability of aligning the phrase pair $(f_{j_1}^{j_2}, e_{i_1}^{i_2})$ is as follows:

$$Pr(f_{j_1}^{j_2}, e_{i_1}^{i_2}) = \frac{1}{2}\left[L_S(f_{j_1}^{j_2}) \cdot X_{TS}(e_{i_1}^{i_2}|f_{j_1}^{j_2})\right] + \frac{1}{2}\left[L_T(e_{i_1}^{i_2}) \cdot X_{ST}(f_{j_1}^{j_2}|e_{i_1}^{i_2})\right] \tag{4}$$

where

$$L_S(f_{j_1}^{j_2}) = \prod_{k=j_1}^{j_2} L_S(f_k|f_1^{k-1}) \tag{5}$$

and

$$X_{TS}(e_{i_1}^{i_2}|f_{j_1}^{j_2}) = \prod_{j=j_1}^{j_2} \sum_{i=i_1}^{i_2} \frac{1}{(i_2 - i_1)} X_{TS}(f_j|e_i) \tag{6}$$

$L_T(e_{i_1}^{i_2})$ and $X_{ST}(f_{j_1}^{j_2}|e_{i_1}^{i_2})$ are defined similarly.

To find the best parallel phrase pair in a comparable sentence pair, we assume that the rest of the sentence outside the phrase boundaries is non-parallel. This assumption is not always true, as there can be multiple parallel phrase pairs in the sentence pair, as illustrated in Figure 5. At this point, however, we only focus on finding the best parallel phrase pair in the given comparable sentence pair. We expand the search for remaining parallel sections later.

We compute the joint probability of the sentence pairs $P(f,e)$ as follows:

$$\begin{aligned} P(f,e) = {} & L_S(f_1^{j_1-1}) \cdot L_T(e_1^{i_1-1}) \\ & \times Pr(f_{j_1}^{j_2}, e_{i_1}^{i_2}) \\ & \times L_s(f_{j_2+1}^{J}) \cdot L_T(e_{i_2+1}^{I}) \end{aligned} \tag{7}$$

and find the boundaries (j_1, j_2, i_1, i_2) of the best pair by optimizing $P(f,e)$:

$$(i,j,k,l) = \underset{j_1,j_2,i_1,i_2}{\operatorname{argmax}} \; P(f,e) \tag{8}$$

3.2.1 Phrase extraction

To extract parallel phrases from the comparable sentence pairs, we adopt the following procedure:

- **Find the best parallel phrase pair**: First assume that there is only one parallel phrase embedded in the comparable sentence pair, and find the best parallel phrase. This process divides the sentence pair into several regions as illustrated in Figure 4.
- **Find the remaining parallel phrase pairs**: Perform a greedy search in the non-blocked regions to identify remaining parallel phrases.

https://doi.org/10.1017/S1351324916000139 Published online by Cambridge University Press

The following sections explain the proposed algorithm. It is important to point out that the phrase extraction algorithm does not restrict the word alignment within a phrase pair to be monotone. There can be non-monotone word alignment, as well as unaligned words within a phrase pair.

3.2.2 Find the best parallel phrase pair

The straightforward way to find the best phrase pair is by computing the alignment for every value of the boundary pairs (j_1, j_2) and (i_1, i_2) within the boundaries of the sentences and selecting the boundaries with the highest probability. This method is computationally expensive and has runtime complexity $O(J^2I^2)$. However, most of the boundary pairs in this calculation will not produce any meaningful phase pairs. For example, the boundary pair (1,1) and (1,15) is unlikely to be a reasonable phrase alignment as it aligns a single source word with fifteen target words. Therefore, the search space can be restricted to those boundary values that have the potential to produce meaningful phrases.

We introduce minimum and maximum phrase lengths to both the source phrase (l_{min}, l_{max}) and the target phrase (k_{min}, k_{max}). The best parallel phrase is then identified using the algorithm in Algorithm 1.

Algorithm 1 Identify the best phrase translation pair in a comparable sentence pair

Input: Source sentence f_1^J, target sentence e_1^I, phrase length limits $(l_{min},l_{max},k_{min},k_{max})$
Output: The boundaries of the best phrase translation pair $(j_1, j_2, i_1, i_2)_{best}$

$BestProb = 0$
$(j_1, j_2, i_1, i_2)_{best} = (0, 0, 0, 0)$
for $l = l_{min}$ to l_{max} **do**
  for $j = 1$ to $(J - l + 1)$ **do**
    for $k = k_{min}$ to k_{max} **do**
      for $i = 1$ to $(I - k + 1)$ **do**
        if $NewProb = Pr_{(j,(j+l-1),i,(i+k-1))}(f, e) > BestProb$ **then**
          $BestProb = NewProb$
          $(j_1, j_2, i_1, i_2)_{best} = (j, (j + l - 1), i, (i + k - 1))$
return $(j_1, j_2, i_1, i_2)_{best}$

This search process can also be thought of as follows: From among a set of allowed phrase patterns of different sizes (e.g. 3×8, 4×6, etc.), we want to find the best placement for that pattern in the sentence that would produce the best parallel phrase pair.

The length restriction reduces the run-time complexity of the algorithm to $O(c.JI)$, where c is the number of different source and target phrase length combinations allowed. By restricting (l_{min}, l_{max}) and (k_{min}, k_{max}), we can keep c in a manageable level. For example, if we are interested in phrases of three to seven words long in both source and target sides, then c is 25. Similar phrase length restrictions are typically enforced when extracting phrase translation pairs from parallel sentences

https://doi.org/10.1017/S1351324916000139 Published online by Cambridge University Press

using various heuristics to combine word-alignments (e.g. in Moses (Koehn *et al.* 2007)).

The relationship between source phrase and target phrase is not uniform. Morphologically rich languages, such as Arabic, typically have fewer words in a sentence than English. In most Arabic–English parallel corpora (e.g. LDC), the sentence length ratio (in terms of number of words) between Arabic and English is between 1 and 1.5. We use this ratio to further restrict the search with the following conditions:

$$\frac{1}{c_1} \cdot l - c_2 \leq k \leq c_1 \cdot l + c_2 \tag{9}$$

where c_1 is the length ratio and c_2 is an additive constant. Both c_1 and c_2 are selected empirically using a held-out data set.

Lexical translation models used in the alignment process may contain erroneous alignments that will affect the phrase extraction process. The models could give highest score to an alignment that is not the best, or the correct one. Therefore, we select not only the best phrase pair, but all candidate phrase pairs whose scores lie within a margin of the best one. There margin boundaries are selected empirically, using a held-out data set.

Algorithm 2 Search for the remaining parallel phrases in a comparable sentence pair

Input: Source sentence f_1^J, target sentence e_1^I, list of phrase patterns $(m \times n)$, best phrase pair boundaries $(j_1, j_2, i_1, i_2)_{best}$ found using Algorithm 1

Output: List of phrase pairs (f_j^{j+m}, e_i^{i+n})

$OldProb = Pr_{(j_1,j_2,i_1,i_2)}(f, e)$
$\mathscr{A} = \{(j,i) | j_1 \leqslant j \leqslant j_2; i_1 \leqslant i \leqslant i_2\}$
while *Continue* **do**
  $BestProb = 0$
  for each phrase pattern $(m \times n)$ **do**
    for $j = 1$ to $(J - m + 1)$ **do**
      for $i = 1$ to $(I - n + 1)$ **do**
        if $(x,y) \notin \mathscr{A}\ \forall \{(x,y) | j \leqslant x \leqslant j+m; i \leqslant y \leqslant i+n\}$ **then**
          compute $Pr(f_j^{j+m}, e_i^{i+n})$ using equation (4)
          compute $L_S(f_j^{j+m})$ using equation (5)
          similarly compute $L_T(e_i^{i+n})$
          $NewProb = \frac{OldProb \cdot Pr(f_j^{j+m}, e_i^{i+n})}{L_S(f_j^{j+m}) \cdot L_T(e_i^{i+n})}$
          if $NewProb > BestProb$ **then**
            $BestProb = NewProb$
  if $BestProb > OldProb$ **then**
    $OldProb = BestProb$
    $\mathscr{A} = \mathscr{A} \cup \{(j,i) | j_{best} \leqslant j \leqslant j_{best} + m; i_{best} \leqslant i \leqslant i_{best} + n\}$
    Output the phrase pair $(f_{j_{best}}^{j_{best}+m}, e_{i_{best}}^{i_{best}+n})$
    $Continue = True$
  else
    $Continue = False$

https://doi.org/10.1017/S1351324916000139 Published online by Cambridge University Press

3.2.3 Find remaining parallel phrases in the sentence

There are four possible regions (regions 1, 3, 5 and 7 in Figure 4, counting clockwise starting from bottom left region) in which the remaining parallel phrases can be present. To detect them, a greedy search is performed as follows:

(1) Define a set $\mathscr{A}$ which contains tuples (i, j) to indicate the word positions that are already covered by previously identified phrase pairs. This is initialized with values from the best parallel phrase pair identified as in the previous section.
(2) For each of the possible phrase pair regions of remaining word pairs, iterate over all possible placements of them in the sentence pair. They are considered as candidates only if they can be placed completed on an unaligned area. i.e. the word alignment pairs corresponding to the phrase pair are not included in $\mathscr{A}$.
(3) Out of all possible such phrase pair candidates, pick the pair that gives the best overall sentence alignment.
(4) The phrase pair is extracted as parallel only if the alignment probability is better than the previous sentence alignment probability. i.e. the phrase pair is better explained using the translation model than the language model. $\mathscr{A}$ is updated accordingly.
(5) This process is iterated until no further parallel phrases can be identified.

The search algorithm is as shown in Algorithm 2. Several computational steps in this algorithm can be reduced by incremental calculation of the partial sums of probabilities.

3.2.4 Find target boundaries for a given source phrase

Algorithms 1 and 2 above together extract all possible parallel phrase pairs from a given comparable sentence pair. This approach is appropriate when we attempt to build a phrase table from all available corpora before decoding a test set. Alternatively, PESA performs real-time phrase extraction at decoding time, where no pre-computed phrase table exits. At run-time, an appropriate target phrase needs to be extracted for the source phrase under consideration. We present a modified version of the search algorithm in Algorithm 3, where the target phrase boundaries are obtained for a given source phrase.

When extracting phrases for a test set, we first generate all possible source phrases (with phrase length ranging between l_{min} and l_{max}), and extract the desired target phrases using algorithm 3 above. By restricting the search of target boundaries for only a given source phrase, we reduce the run-time complexity of the algorithm to $O(c.I)$, where c is the number of different source and target phrase length combinations allowed. This is faster than aligning all the words in the sentence pairs as in the Viterbi alignment.

3.2.5 Additional features

In our initial experiments, we found that the phrases extracted using the equation (7) tend to have incorrect word boundaries. Most of the time, the extracted phrase

https://doi.org/10.1017/S1351324916000139 Published online by Cambridge University Press

Algorithm 3 Algorithm to identify the best target boundaries for a given source phrase

Input: Source sentence f_1^J, target sentence e_1^I, source phrase boundary (j_1, j_2), target phrase length limits (k_{min}, k_{max})
Output: The boundaries of the best target phrase translation pair $(i_1, i_2)_{best}$

```
BestProb = 0
(i1, i2)best = (0, 0, 0, 0)
for each source phrase ẽ do
    Select comparable sentence pairs (f_1^J, e_1^I) such that e contains ẽ
    for each (f_1^J, e_1^I) do
        for k = k_min to k_max do
            for i = 1 to (I − k + 1) do
                Compute NewProb = Pr(f, e) using the log-linear model
                if NewProb > BestProb then
                    BestProb = NewProb
                    (i1, i2)best = (i, (i + k − 1))
        return (i1, i2)best
```

contained more words on either side of the boundary. This is due to the effect of two language model scores for the left and right of the parallel phrase, where the model prefers a smaller language model score. We introduce the following additional features to better balance the phrase length. We defined the following additional features:

- Phrase Length Ratio (one feature): This is the ratio between the source phrase length and the target phrase length. i.e. $|j_2 - j_1|/|i_2 - i_1|$.
- Phrase Length Difference (one feature): Difference in source phrase length and the target phrase length. i.e. $|j_2 - j_1| - |i_2 - i_1|$.
- Target Language Model Score for outside region (two features): These scores are normalized with respect to the source or target phrase length.
- Phrase alignment score for the inside region (one feature): This is computed as in equation (4).

The phrase alignment score in equation (7) is combined with the additional features in a weighted log-linear model as follows:

$$(i_1, i_2)_{best} = \underset{(i_1,i_2)}{\operatorname{argmax}} \sum_{k=1}^{6} \lambda_k F_k(f_1^J, e_1^I, j_1, j_2) \tag{10}$$

where F_ks are feature functions as defined above and λ_ks are the weights associated with feature function which are optimized, using a held-out data set.

3.2.6 Parameter optimization

To optimize the parameters of the extended phrase extraction model, we use a manually word-aligned parallel corpus. From this corpus, we extract parallel phrase

https://doi.org/10.1017/S1351324916000139 Published online by Cambridge University Press

pairs that are consistent with the underlying word alignments. Each source phrase was allowed to have multiple reference target phrases. The details of the extraction process are explained in Section 4.1.

We use the online margin infused relaxed algorithm (MIRA) (Crammer *et al.* 2006) to obtain the optimized model weights. MIRA is defined by an update rule, which is subject to max-margin constraints with respect to a loss function that is computed for the predicted hypothesis against the reference phrase. The weights are updated after seeing each training phrase pair in each iteration. Change in the weight vector is constrained so that the margin between the hypothesis and the reference should be at least as large as the loss value. Formally, the update rule in MIRA is as follows:

$$\underset{w}{\text{argmax}} \;\; ||w^{i+1} - w|| + C \cdot \sum_{k} \xi_k(y_t, y') \tag{11}$$

$$s.t., \quad s(x_t, y_t) - s(x_t, y_k) + \xi_k(y_t, y') \geq L(y_t, y'); \;\; y' \in best_k(x_t, w^i); \;\; \xi_k(y_t, y') \geq 0$$

where w is the weight vector, y_t is the target candidate closest to the reference phrase, y' is in the k-best candidate list, $s(x, y)$ is the scoring function, $L(y_t, y')$ is the computed loss. ξ_k is the slack variable whose value is always non-negative and C is the slack constant the determines how aggressively the weight vector is updated after each instance. The weights obtained after each update are averaged at the end of the training step in order to avoid overfitting.

We optimize the model performance toward an oracle target phrase which is selected from a k-best list of candidate target phrase translations. Reference target phrases obtained from the manual alignment may not always be reachable by the phrase extraction algorithm. This is due to the fact that we extract target candidate phrases from sentence pairs, which contain the source phrase in the source sentence, but may not always contain the reference phrase in the target sentence (i.e. an alternative translation). Thus, from the k-best list of candidate target phrases, we select an m-best list of target phrases that are closest to one or multiple references. In the experiments, we set m=1 (i.e. only the first-best oracle phrase is considered).

To compute the loss function, we use a modified smoothed version of BLEU (*mBLEU*). The BLEU metric is defined on the document level which considers n-gram precision (typically up to four). To compute BLEU-like scores at the phrase level, we define a smoothed version that considers only unigram matches in its computation. The smoothing technique is similar to the NIST BLEU version.[1] It is computed by adding a partial count of $(1/2^k)$ for each precision score whose matching n-gram count is zero, where $k = 1$ for the first n values for which the n-gram match count is zero. The loss function using the smoothed-BLEU is defined as follows:

$$L(y_t, y' = 1 - mBLEU(y_t, y')$$

The error rates in training and testing sets for MIRA is evaluated in terms of normalized Levenshtein distance at word level (i.e. word error rate). This metric

[1] ftp://jaguar.ncsl.nist.gov/mt/resources/mteval-v13.pl

https://doi.org/10.1017/S1351324916000139 Published online by Cambridge University Press

awards partial 'credit' for candidate target phrases when the reference phrase is not always reachable.

With the extracted phrase pairs from the manually aligned corpus, we used a seventy five/twenty five per cent split for training/testing. Each source phrase is allowed to have multiple reference phrase pairs in the setup. We generate a ten-best target phrase candidate list for each source phrase and evaluated the error against the first best candidate. We run MIRA for ten iterations. The final averaged weight vector obtained after the optimization is used for the phrase extraction experiments in Section 4.2. Further details of the optimization is described in Gupta, Hewavitharana and Vogel (2011).

3.3 Other approaches

We compare the proposed phrase extraction with two other phrase alignment approaches.

3.3.1 Standard Viterbi alignment

We use the standard phrase extraction approach used by most phrase-based SMT systems: obtain word alignment models for both directions using IBM models (source to target and target to source), combine the Viterbi paths using one of many heuristics and extract phrase pairs from the combined alignment. We used the Moses toolkit (Koehn *et al.* 2007) for this task. To obtain the word alignments for comparable sentence pairs, we performed a forced alignment using the models trained on parallel data.

3.3.2 Binary classifier

We use a Maximum Entropy classifier as our second alternative approach to extract parallel phrase pairs from comparable sentences. Such classifiers have been used in the past to detect parallel sentence pairs in large collections of comparable documents (Munteanu and Marcu 2005; Tillmann and Hewavitharana 2013). Our classifier is similar, but we apply it at phrase level rather than at sentence level. The classifier probability is defined as

$$p(c|S,T) = \frac{exp\left(\sum_{i=1}^{n} \lambda_i f_i(c,S,T)\right)}{Z(S,T)} \tag{12}$$

where $S = s_1^L$ is a source phrase of length L and $T = t_1^K$ is a target phrase of length K. $c \in \{0,1\}$ is a binary variable representing the two classes of phrases: *parallel* and *not parallel*. $p(c|S,T) \in [0,1]$ is the probability where a value $p(c=1|S,T)$ close to 1.0 indicates that S and T are translations of each other. $f_i(c,S,T)$ are feature functions that are co-indexed with respect to the class variable c. The parameters λ_i are the weights for the feature functions obtained during training. $Z(S,T)$ is the normalization factor. In the feature vector for phrase pair (S,T), each feature appears twice, once for each class $c \in \{0,1\}$.

https://doi.org/10.1017/S1351324916000139 Published online by Cambridge University Press

The feature set we use is inspired by Munteanu (2005), who define the features based on IBM Model-1 (Brown *et al.* 1993) alignments for source and target pairs. However, in our experiments, the features are computed primarily on IBM Model-1 probabilities (i.e. lexicon). We do not explicitly compute IBM Model-1 alignments. To compute coverage features, we identify alignment points for which IBM Model-1 probability is above a threshold. We produce two sets of features based on IBM Model-1 probabilities obtained by training in both directions. All the features have been normalized with respect to the source phrase length L or the target phrase length K. We use the following eleven features:

(1) Lexical probability (two features): IBM Model-1 log probabilities $p(S|T)$ and $p(T|S)$.
(2) Phrase length ratio (two features): source length ratio K/L and target length ratio L/K.
(3) Phrase length difference (one feature): source length minus target length, $L-K$.
(4) Number of words covered (two features): A source word s is said to be covered if there is a target word $t \in T$ such that $p(s|t) > \epsilon$, where $\epsilon = 0.5$. Target word coverage is defined accordingly.
(5) Number of words not covered (two features): This is computed similarly to (4) above, but this time counting the number of positions that are not covered.
(6) Length of the longest covered sequence of words (two features).

To train the classifier, we used parallel phrase pairs extracted from a manually word-aligned corpus. In selecting negative examples, we followed the same approach as in Munteanu and Marcu (2005): pairing all source phrases with all target phrases, but filter out the parallel pairs and those that have high length difference or a low lexical overlap, and then randomly select a subset of phrase pairs as the negative training set. The model parameters are estimated using the GIS algorithm.

4 Experiments

4.1 Alignment accuracy

To compare the performance of the three phrase alignment methods discussed so far, we need a comparable corpus containing sentences with a known parallel phrase pair embedded in them. Using a manually aligned parallel corpus, and two monolingual corpora, we obtained a test corpus as follows: For each parallel phrase pair, we search a source language corpus for sentences that contain the source phrase, and a target language corpus for sentences that contain the target phrase. We measure phrase-level exact matches, as well of unigram level precision, recall and F measure. The scores are computed for each phrase pair, and then aggregated to produce a score for the test set.

For this evaluation, we used Arabic–English language pair. We obtained a manually word aligned corpus containing 663 Arabic–English sentence pairs. From this, we selected 300 sentences, and extracted phrase pairs (up to ten words long) that are consistent with the underlying word alignment. From the resulting list of phrase pairs, we removed the fifty most frequently occurring pairs, as well as those

https://doi.org/10.1017/S1351324916000139 Published online by Cambridge University Press

Table 1. *n-gram type distribution of manually aligned phrase set*

	1	2	3	4	5	6	7	8	9	10	All
Test set	2.8 K	3.7 K	3.4 K	3 K	2.7 K	2.4 K	2 K	1.6 K	1.5 K	1.4 K	24.9 K
Test set (found)	2.7 K	2.7 K	1.2 K	373	87	29	7	2	1	0	7 K

only consisting of punctuations. Almost all high frequency phrases are function words, which are typically covered by the translation lexicon. Line 1 in Table 1 gives the n-gram type distribution for the source phrases.

Using the phrase pairs extracted from the manually aligned sentences, we constructed a comparable corpus as follows:

(1) For each Arabic phrase, we search the Arabic Gigaword[2] corpus for sentences that contain the phrase and select up to five sentences. Similarly, for each corresponding English phrase, we select up to five sentences from English Gigaword.[3]
(2) For each phrase pair, we generate the Cartesian product of the sentences and produce a sentence pair collection. i.e. up to twenty five comparable sentence pairs were constructed for each phrase pair.
(3) Sentences that have more than hundred words were rejected. Not all phrases had a matching sentence pair in the corpus. The final comparable corpus contained 170 K sentence pairs.

Line 2 in Table 1 gives the n-gram type distribution for the phrase pairs for which we found both a source sentence and a target sentence in the monolingual corpora. As expected, the longer the phrases, the less likely it is to find them in even larger corpora. We consider the resulting set as our comparable corpus which we use to evaluate the three alignment approaches. In most sentence pairs, except for the phrase pair that we are interested in, the rest of the sentence does not typically match the other side.

We obtained the Viterbi alignment using standard word alignment techniques: IBM4 word alignment for both directions, Viterbi path combination using heuristics (grow-diag-final) and phrase extraction from two-sided training, as implemented in the Moses package (Koehn *et al.* 2007). Because the non-parallel segments will lead the word alignment astray, this may have a negative effect on the alignment in the parallel sections. Alignment models trained on parallel data are used to generate the Viterbi alignment for the comparable sentences. We then extract the target phrases that are aligned to the embedded source phrases. A phrase pair is extracted only when the alignment does not conflict with other word alignments in the sentence pair. The alignments are not constrained to produce contiguous phrases. We allow

[2] Arabic Gigaword Fourth Edition (LDC2009T30).
[3] English Gigaword Fourth Edition (LDC2009T13).

https://doi.org/10.1017/S1351324916000139 Published online by Cambridge University Press

Table 2. *Alignment accuracy for the three approaches: standard Viterbi, classifier and the extended-PESA approach*

	Viterbi				Classifier				Extended-PESA			
Lexicon	Exact	P	R	F	Exact	P	R	F	Exact	P	R	F
Lex-Full	43.56	65.71	57.99	61.61	54.46	81.79	89.10	85.29	67.94	93.61	86.59	89.96
Lex-Large	42.95	65.68	56.69	60.85	53.57	81.32	88.34	84.69	67.28	92.91	85.96	89.30
Lex-Medium	41.10	63.60	51.15	56.70	52.38	80.30	86.64	83.35	65.81	91.95	84.52	87,93
Lex-Small	41.02	62.10	49.38	55.01	52.51	80.51	83.84	82.14	63.23	89,08	81.84	85.31
Lex-BTEC	19.10	26.94	23.63	25.18	18.76	45.90	36.17	40.46	17.45	46.36	36.07	40.57

unaligned words to be present in the phrase pair. For each source phrase, we selected the target phrase that has the least number of unaligned words.

The classifier (Section 3.3.2) is applied at the phrase level. We generate the phrase pair candidates as follows: For a given target sentence, we generate all n-grams up to length 10. We pair each n-gram with the source phrase embedded in the corresponding source sentence to generate a phrase pair. From the 170 K sentence pairs, we obtained 15.6 million phrase pair candidates. The maximum entropy classifier is then applied to the phrase pairs. For each source phrase, we pick the target candidate for which $p(c = 1, S, T)$ has the highest value.

For the extended-PESA phrase alignment algorithm described in Section 3.2, we used both inside and outside alignments as explained earlier. For each source phrase pair, we select the best scoring target phrase using Algorithm 3.

As our goal is to use these methods to extract parallel data for low resource situations, we tested each method with several lexica, trained on different amounts of initial parallel data. Starting from the full Arabic–English parallel corpus with 127 million English tokens, we generated three additional parallel corpora, namely *Large*, *Medium* and *Small*, with 1/3, 1/9 and 1/27 of the original size, respectively. The three new corpora contain roughly 40 million, 13 million and 4 million English tokens. *Medium* and *Small* corpora are a better match to the resource levels for many languages. We also used data from the BTEC (Kikui *et al.* 2003) corpus. This corpus contains conversational data from the travel domain, which is from a different genre than the document collections. Compared to other corpora, it is much smaller (about 190 thousand English tokens).

Table 2 gives the results for all three alignment approaches. Results are presented as percentages of: exact matches found (Exact), precision (P), recall (R) and F. The Viterbi alignment gives the lowest performance. This shows that the standard phrase extraction process, which works well for parallel sentences, is ill-suited for comparable sentences. Despite the fact that the classifier incorporates several additional features including the lexical features, the performance of the extended-PESA alignment shows consistently higher precision than the classifier approach. This demonstrates that computing both inside and outside probabilities for the sentence pair helps the phrase extraction. The classifier lacks this ability because the phrase pair is evaluated in isolation, without the context of the sentence.

https://doi.org/10.1017/S1351324916000139 Published online by Cambridge University Press

Except for the BTEC corpus, the performance degradation is smaller as the lexicon size is reduced, indicating that the approaches are robust for smaller amounts of parallel data.

Instead of using token precision, an alternative method of evaluating partial matches, is to give credit based on the length of the overlap between the extracted phrase and the reference. Precision and recall can then be defined based on the longest common contiguous subsequence, similar to Bourdaillet *et al.* (2010). Results we obtained using these methods were similar to the results in Table 2.

In the translation experiments below, we compare the performance of phrases extracted from the standard Viterbi alignment and the extended-PESA alignment.

4.2 Translation results

4.2.1 Arabic–English

For Arabic–English experiments, we used a large baseline parallel corpus of about ≈6 million sentence pairs (120 million English words). We extract comparable sentence pairs from the Gigaword Arabic and Gigaword English corpora, using a maximum entropy classifier similar to Tillmann and Hewavitharana (2013). Extracted sentence pairs with a similarity score above an empirically determined threshold were selected as parallel sentences. The remaining sentences that were below the cut-off threshold, were used for the phrase extraction experiments (see Figure 1 for the data extraction pipeline).

From the comparable sentences, two sets of phrase pairs were extracted: one using the Viterbi alignment and the other using the extended-PESA method as in Algorithm 3. The extraction process followed the same steps explained in Section 4.1, i.e. using different sizes of initial parallel data to generate translation lexica, as a simulation of various resource availability scenarios. The phrases extracted from the Viterbi alignment were sampled to include only those needed to cover the source side of the test sets.

The Algorithm 3 extracts target translations for a given set of source phrases. We used two test sets from previous MT evaluations by NIST: MT05 (≈1 K sentences) and MT06 (≈1.7 K sentences). For each test set, we generated all source n-grams up to seven words, and used them to extract phrases using Algorithm 3. The model parameters were optimized using MIRA as described in Section 3.2.6 using manually aligned data. For each source phrase, we extract up to ten target translation alternatives. From the comparable sentences, we extracted ≈800 K phrase pairs, out of which ≈100 K were new source phrases. The extracted phrases had an average length of 2.7 words (compared to 2.0 words for the phrases extracted using the Viterbi alignment).

A standard phrase-based MT decoder (Vogel 2003), which is similar to Moses (Koehn *et al.* 2007) was used in the translation experiments. A 4-gram language model was trained using the English side of the corpus. As the baseline MT system, we not only use parallel data, but also the sentences extracted from comparable data (i.e. step 2 in Figure 1). Two tests from previous NIST MT evaluations were used: MT05 as the development set to optimize the decoder parameters (using MERT [Och 2003])", and MT06 as an unseen test set.

https://doi.org/10.1017/S1351324916000139 Published online by Cambridge University Press

Table 3. *Translation results for Arabic–English extracted phrases when combined with the extracted sentences plus different sizes of initial parallel data. Each row indicates the size of the initial parallel corpus used as the baseline system. Best results for each metric are marked in boldface*

System	Corpus	MT05 (Dev)			MT06 (Test)		
		BLEU	METEOR	TER	BLEU	METEOR	TER
Baseline+Sentences	Small	49.81	**38.74**	**40.89**	35.53	31.04	51.37
	Medium	50.68	39.07	**40.21**	**37.07**	**31.94**	50.24
	Large	**51.96**	**39.61**	39.59	**38.00**	**32.46**	50.03
	Full	**52.90**	**40.09**	38.79	39.24	**33.05**	49.22
+Phrases(Viterbi)	Small	49.19	37.74	41.21	35.02	30.81	51.73
	Medium	50.26	38.26	40.44	36.45	31.41	**50.20**
	Large	51.45	38.92	39.87	37.57	32.19	49.55
	Full	52.87	40.09	38.85	38.13	32.38	49.35
+Phrases(Ext-PESA)	Small	**50.04**	38.15	41.36	**36.00**	**31.38**	**50.83**
	Medium	**51.92**	**39.52**	40.46	36.79	31.85	50.27
	Large	51.74	39.40	**39.55**	37.98	32.16	**49.58**
	Full	52.89	40.05	**38.64**	38.67	32.61	**48.64**

The extracted phrases were combined with phrases extracted from different sizes of initial parallel data, (i.e. 4.5 M(*Small*), 13 M(*Medium*), 40 M(*Large*) and 120 M(*Full*) word corpora). Translation results for Arabic–English are given in Table 3, in BLEU (Papineni *et al.* 2002), METEOR (Banerjee and Lavie 2005) and TER (Snover *et al.* 2006) metrics. The results are shown for three systems:

***Baseline+Sentences*:** Phrases extracted from the initial parallel data plus the extracted parallel sentences from comparable data. This serves as the baseline for the phrase extraction step, which we try to surpass with the extracted phrases.

***+Phrases(Viterbi)*:** Phrases extracted using the Viterbi alignment when combined with phrases from *Baseline+Sentences*.

***+Phrases(Ext-PESA)*:** Phrases extracted using the extended-PESA approach when combined with phrases from *Baseline+Sentences*.

The results show that the extended-PESA phrase extraction method is consistently better than extracting phrases with the standard Viterbi algorithm. When the new phrases are added, there is an improvement in performance for the *Small* system. This difference is statistically significant (at ninety-five per cent level) for MT06. However, for other systems, the extracted phrases fail to improve the performance over the combined Baseline+Sentence system. Although the phrase extraction brings in additional phrases, the improvement in source side coverage compared to the phrases in the baseline system range between two per cent and five per cent for different systems. As the phrases are extracted from comparable data, the additional phrase pairs may introduce noise to the combined system. For the smallest systems with the lowest lexical coverage, the additional phrases help improve the performance.

https://doi.org/10.1017/S1351324916000139 Published online by Cambridge University Press

When the size of the system increases, we notice that the additional phrases starts having less effect, and leads to degradation of results.

The Arabic–English parallel corpus that we used in the experiments is larger in size compared to the parallel resources available for many other language pairs. Next, we apply the same phrase extraction pipeline to Urdu–English, which is a true low-resource language pair.

4.2.2 Urdu–English

For Urdu–English experiments, we used a parallel corpus with ≈55 K sentence pairs (≈1 million English words. Comparable sentence pairs were extracted using Urdu and English news articles from Voice of America, and BBC News, which resulted in a collection of ≈250 K sentence pairs. Out of this, we selected the top twenty five per cent scoring sentences as parallel (≈100 K sentence pairs), and combined with the baseline to produce the 'Baseline+Extracted Top twenty five per cent' system that serves as our baseline for the phrase extraction. We used the remaining sentences for the phrase extraction task, and extracted two sets of phrases, one each using the Viterbi alignment and the extended-PESA approach as in Algorithm 3. Two test sets from previous NIST MT evaluations (Newswire portions): MT08-NW and MT09-NW, were used as evaluation sets.

Similar to Arabic–English experiments in the previous section, we generated all source n-grams up to length 7 from the two test sets , and extracted up to ten target translations for each source n-gram. From the comparable sentences, we extracted ≈425 K phrase pairs, out of which ≈180 K were new phrase pairs. These phrases covered ≈20 K additional source n-grams, which is a thirty per cent increase in source n-gram coverage compared to the 'Baseline+Extracted Top twenty five per cent' system. The extracted phrases had an average length of 3.24 words (compared to 2.1 words for the phrases extracted using the Viterbi alignment). The Viterbi alignment added ≈700 K new phrase pairs which covers ≈50 K new source n-grams.

We combined the extracted phrases with the phrases extracted from the 'Baseline+ Extracted Top twenty five per cent' system and used in the translation experiments. The translation setup including the decoder and language model remained the same as in Arabic–English. We used the MT08 development set to optimize the decoder parameters using MER training. Two translation systems were built, one each from the phrases extracted using the Viterbi alignment and the extended-PESA alignment.

Translation results for Urdu–English are given in Table 4. Results are given in BLEU, METEOR and TER metrics. For each test set, the first line gives the results for the baseline. The second and third lines give the results for the phrase extraction using Viterbi approach and the proposed approach, respectively. The fourth line 'Baseline+Extracted All' indicates combining the baseline parallel corpus with all extracted comparable sentence pairs. The results show that the extended-PESA phrase extraction method is significantly better than the Viterbi phrase extraction. This is despite the fact that the Viterbi extraction approach adding

https://doi.org/10.1017/S1351324916000139 Published online by Cambridge University Press

Table 4. *Translation results for Urdu–English extracted phrases when combined with the baseline corpus. Best results for each metric are marked in boldface*

System	MT08-NW (Dev)			MT09-NW (Test)		
	BLEU	METEOR	TER	BLEU	METEOR	TER
Baseline+Extracted Top 25 per cent	23.93	29.63	63.17	26.33	30.71	60.10
+Phrases(Viterbi)	23.78	29.82	63.25	26.52	30.89	59.92
+Phrases(Ext-PESA)	**25.03**	**30.45**	**62.44**	**28.64**	**31.05**	**58.16**
Baseline+Extracted All	22.89	28.96	63.68	25.24	30.01	61.86

much more phrases than the proposed approach. It is clear that much of these phrases have not helped in improving the translation performance due to the noisy nature of the translations. When the extracted phrases are combined with the baseline system, it shows statistically significant improvement (at ninety five per cent level) for both test sets, compared to all the extracted sentences (last line in Table 4). This shows that selectively extracting phrase level parallel data leads to better translation performance than generating new models with the full extracted corpus.

From the experiments with Arabic–English and Urdu–English, we can see that extracting parallel data at phrase level helps translation performance when the lexical coverage is limited. When the amount of available parallel data increases, the effect of the extracted data diminishes.

5 Conclusions and future directions

In this paper, we focused on extracting parallel phrase pairs from comparable data. The rationale for this was that by selecting only the parallel phrases from comparable sentences, we prevent the non-parallel parts from adding noise to the clean parallel data. We proposed a novel phrase alignment approach for comparable corpora, which is an extension of the PESA algorithm originally designed for parallel sentences. We proposed several algorithms to extract phrase pairs. We evaluated the proposed alignment approach with two additional methods: the standard Viterbi phrase alignment, and a maximum entropy-based binary classifier that works at phrase-level. Evaluations with Arabic–English phrase alignments showed that the proposed method has a better F-score than the other two approaches.

We evaluated the potential of the extracted phrase pairs in the context of an end-to-end translation system. Translation results with the extracted phrases for Arabic–English showed only marginal improvements in evaluation metrics, when the initial parallel corpus is small. However, for Urdu–English, which is a low-resourced language pairs, we saw 1.21 BLEU per cent improvement over the baseline. Further, the translations results showed that selectively extracting parallel phrases from comparable sentences produce better results.

https://doi.org/10.1017/S1351324916000139 Published online by Cambridge University Press

The proposed models are based primarily on the word alignment probability. We also included additional features based on phrase length and target language model. Other types of features such as part-of-speech, cognates, transliteration models may help the phrase alignment process. Markup data (e.g. anchor text, image titles, urls, etc.), when available, may also be used as features. In our translation experiments, we did not distinguish phrase pairs extracted from comparable corpora from those extracted from parallel data. Alternatively, the two types of phrases could be used with different weights, giving priority to phrase pairs originated from parallel corpora. As the extracted phrases tend to be noisy, this approach may help improve the translation accuracy.

The lack of parallel data for many language pairs is the primary motivation for using comparable corpora to extract translational equivalences. When crawling comparable articles for low resource languages, we observed that these documents are available in multiple languages, giving the possibility of collecting multilingual comparable corpora. It is quite interesting to explore the current framework to extract data from multilingual corpora. Recent research on multilingual SMT has focused on using a language, usually with richer resources (e.g. English), as a pivot language to overcome the resource limitations in certain language pairs. We plan to explore a similar approach to extract data from multilingual comparable corpora to extract data for language pairs that have low resources.

References

Banerjee, S., and Lavie, A. 2005. METEOR: an automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, Ann Arbor, Michigan, USA, June, pp. 65–72.

Bourdaillet, J., Huet, S., Langlais, P., and Lapalme, G. 2010. TransSearch: from a bilingual concordancer to a translation finder. *Machine Translation* **24**(3–4): 241–71, December.

Brown, P. F., Della Pietra, S. A., Della Pietra, V. J., and Mercer, R. L. 1993. The mathematics of statistical machine translation: parameter estimation. *Computational Linguistics* **19**(2): 263–311.

Crammer, K., Dekel, O., Keshet, J., Shalev-Shwartz, S., and Singer, Y. 2006. Online passive-agressive algorithms. *Journal of Machine Learning Research* **7**(March): 551–85.

Fung, P., and Cheung, P. 2004. Mining very non-parallel corpora: parallel sentence and lexicon extraction via bootstrapping and EM. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, Barcelona, Spain, pp. 57–63.

Fung, P., and Yee, L. Y. 1998. An IR approach for translating new words from nonparallel, comparable texts. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics*, Montreal, Canada, pp. 414–20.

Gupta, M., Hewavitharana, S., and Vogel, S. 2011. Extending a probabilistic phrase alignment approach for SMT. In *Proceedings of the International Workshop on Spoken Language Translation (IWSLT)*, San Francisco, CA, December.

Gupta, R., Pal, S., and Bandyopadhyay, S. 2013. Improving MT system using extracted parallel fragments of text from comparable corpora. In *Proceedings of the 6th Workshop on Building and Using Comparable Corpora*, Sofia, Bulgaria, August.

Hewavitharana, S., and Vogel, S. 2011. Extracting parallel phrases from comparable data. In *Proceedings of the 4th Workshop on Building and Using Comparable Corpora: Comparable Corpora and the Web*, Portland, Oregon, pp. 61–8.

https://doi.org/10.1017/S1351324916000139 Published online by Cambridge University Press

Hewavitharana, S., and Vogel, S. 2013. Extracting parallel phrases from comparable data. In S. Sharoff, R. Reinhard, P. Zweigenbaum, and P. Fung (eds.), *Building and Using Comparable Corpora*. Berlin Heidelberg: Springer, pp. 191–204.

Kikui, G., Sumita, E., Takezawa, T., and Yamamoto, S. 2003. Creating corpora for speech-to-speech translation. In *Proceedings of EUROSPEECH*, Geneva, pp. 381–84.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., and Herbst, E. 2007. Moses: open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*, Prague, Czech Republic, June.

Kumano, T., Tanaka, H., and Tokunaga, T. 2007. Extracting phrasal alignments from comparable corpora by using joint probability SMT model. In *Proceedings of the International Conference on Theoretical and Methodological Issues in Machine Translation*, Skvde, Sweden, September.

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

Munteanu, D. S., and Marcu, D. 2006. Extracting parallel sub-sentential fragments from non-parallel corpora. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics*, Sydney, Australia, pp. 81–8.

Och, F. J. 2003. Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, Sapporo, Japan, pp. 160–67.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, Philadelphia, Pennsylvania, USA, July, pp. 311–18.

Quirk, C., Udupa, R. U., and Menezes, A. 2007. Generative models of noisy translations with applications to parallel fragment extraction. In *Proceedings of the Machine Translation Summit XI*, Copenhagen, Denmark, pp. 377–84.

Rapp, R. 1995. Identifying word translations in non-parallel texts. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics,* Cambridge, Massachusetts, pp. 320–22.

Rapp, R. 1999. Automatic identification of word translations from unrelated English and German corpora. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, College Park, Maryland, USA, pp. 519–26.

Resnik, P., and Smith, N. 2003. The web as a parallel corpus. *Computational Linguistics* **29**(3): 349–80.

Snover, M., Dorr, B., Schwartz, R., Micciulla, L., and Makhoul, J. 2006. A study of translation edit rate with targeted human annotation. In *Proceedings of Association for Machine Translation in the Americas*, Cambridge, MA.

Tillmann, C., and Hewavitharana, S. 2011. An efficient unified alignment algorithm for bilingual data. In *Proceedings of Interspeech 2011*, Florence, Italy, August.

Tillmann, C., and Hewavitharana, S. 2013. A unified alignment algorithm for bilingual data. *Natural Language Engineering* **19**(01): 33–60, Januray.

Tillmann, C., and Xu, J.-M. 2009. A simple sentence-level extraction algorithm for comparable data. In *Companion Vol. of NAACL HLT 09*, Boulder, CA, June.

Utiyama, M., and Isahara, H. 2003. Reliable measures for aligning Japanese-English news articles and sentences. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, Sapporo, Japan, pp. 72–9.

Vogel, S. 2003. SMT decoder dissected: word reordering. In *Proceedings of IEEE International Conference on Natural Language Processing and Knowledge Engineering*, Beijing, China, October, pp. 561–66.

https://doi.org/10.1017/S1351324916000139 Published online by Cambridge University Press

Vogel, S. 2005. PESA: phrase pair extraction as sentence splitting. In *Proceedings of the Machine Translation Summit X*, Phuket, Thailand, September.

Zhao, B., and Vogel, S. 2002a. Adaptive parallel sentence mining from web bilingual news collection. In *Proceedings of the IEEE International Conference on Data Mining*, Maebashi City, Japan, pp. 745–48.

Zhao, B., and Vogel, S. 2002b. Full-text story alignment models for Chinese-English bilingual news corpora. In *Proceedings of the ICSLP '02*, Denver, CO, September.

https://doi.org/10.1017/S1351324916000139 Published online by Cambridge University Press