Online Word Alignment for Online Adaptive Machine Translation

M. Amin Farajian
FBK-irst,
University of Trento
Trento, Italy
farajian@fbk.eu

Nicola Bertoldi
FBK-irst
Trento, Italy
bertoldi@fbk.eu

Marcello Federico
FBK-irst
Trento, Italy
federico@fbk.eu

Abstract

A hot task in the Computer Assisted Translation scenario is the integration of Machine Translation (MT) systems that adapt sentence after sentence to the post-edits made by the translators. A main role in the MT online adaptation process is played by the information extracted from source and post-edited sentences, which in turn depends on the quality of the word alignment between them. In fact, this step is particularly crucial when the user corrects the MT output with words for which the system has no prior information. In this paper, we first discuss the application of popular state-of-the-art word aligners to this scenario and reveal their poor performance in aligning unknown words. Then, we propose a fast procedure to refine their outputs and to get more reliable and accurate alignments for unknown words. We evaluate our enhanced word-aligner on three language pairs, namely English-Italian, English-French, and English-Spanish, showing a consistent improvement in aligning unknown words up to 10% absolute F-measure.

1 Introduction

In the adaptive MT the goal is to let the MT system take as soon and as much as possible advantage of user feedback, in order to learn from corrections and to hence avoid repeating the same mistakes in future sentences.

A typical application scenario is the usage by a professional translator of a Computer Assisted Translation (CAT) tool enhanced with a SMT system. For each input sentence, first the translator receives one or more translation suggestions from either a Translation Memory or a SMT system, then (s)he chooses which suggestion is more useful, and finally (s)he creates an approved translation by post-editing. The pair of input sentence and post-edit is a valuable feedback to improve the quality of next suggestions. While the sentence pair is trivially added to the Translation Memory, how to exploit it for improving the SMT system is far to be a solved problem, but rather is a hot and quite recent topic in the MT community.

In online MT adaptation specific issues have to be addressed, which distinguish it from the more standard and investigated task of domain adaptation. First of all, the SMT system should adapt very quickly, because the time between two consecutive requests are usually short, and very precisely, because the translator is annoyed by correcting the same error several time. Then, a crucial point is which and how information is extracted from the feedback, and how it is exploited to update the SMT system. Finally, model updating relies on a little feedback consisting of just one sentence pair.

In this work we focus on the word alignment task which is the first and most important step in extracting information from the given source and its corresponding post-edit. In particular, we are interested in the cases where the given sentence pairs contain new words, for which no prior information is available. This is an important and challenging problem in the online scenario, in which the user interacts with the system and expects that it learns from the previous corrections and does not repeat the same errors again and again.

Unfortunately, state-of-the-art word-aligners show poor generalization capability and are prone to errors when infrequent or new words occur in the sentence pair. Word alignment errors at this stage could cause the extraction of wrong phrase pairs, i.e. wrong translation alternatives, which can lead in producing wrong translations for those

Workshop on Humans and Computer-assisted Translation, pages 84–92,
Gothenburg, Sweden, 26 April 2014.

words, if they appear in the following sentences.

Our investigation focuses on how to quickly build a highly precise word alignment from a source sentence and its translation. Moreover, we are interested in improving the word alignment of unknown terms, i.e. not present in the training data, because they are one of the most important source of errors in model updating.

Although we are working in the online MT adaptation framework, our proposal is worthwhile per se; indeed, having an improved and fast word aligner can be useful for other interesting tasks, like for instance terminology extraction, translation error detection, and pivot translation.

In Section 2 we report on some recent approaches aiming at improving word alignment. In Section 3, we describe three widely used toolkits, highlight their pros and cons in the online MT adaptation scenario, and compare their performance in aligning unknown terms. In Section 4 we propose a standalone module which refines the word alignment of unknown words; moreover, we present an enhanced faster implementation of the best performing word aligner, to make it usable in the online scenario. In Section 5 we show experimental results of this module on three different languages. Finally, we draw some final comments in Section 6.

2 Related works

Hardt et al. (2010) presented an incremental retraining method which simulates the procedure of learning from post-edited MT outputs (references), in a real time fashion. By dividing the learning task into word alignment and phrase extraction tasks, and replacing the standard word-alignment module, which is a variation of EM algorithm (Och and Ney, 2003), with a greedy search algorithm, they attempt to find a quick approximation of the word alignments of the newly translated sentence. They also use some heuristics to improve the obtained alignments, without supporting it with some proofs or even providing some experimental results. Furthermore, the running time of this approach is not discussed, and it is not clear how effective this approach is in online scenarios.

Blain et al. (2012) have recently studied the problem of incremental learning from post-editing data, with minimum computational complexity and acceptable quality. They use the MT output (hypothesis) as a pivot to find the word alignments between the source sentence and its corresponding reference. Similarly to (Hardt and Elming, 2010), once the word alignment between the source and post-edit sentence pair is generated, they use the standard phrase extraction method to extract the parallel phrase pairs. This work is based on an implicit assumption that MT output is reliable enough to make a bridge between source and reference. However, in the real world this is not always true. The post-editor sometimes makes a lot of changes in the MT output, or even translates the entire sentence from scratch, which makes the post-edit very different from the automatic translation. Moreover, in the presence of new words in the source sentence, the MT system either does not produce any translation for the new word, or directly copies it in the output. Due to the above two reasons, there will be missing alignments between the automatic translation and post-edit, which ultimately results in incomplete paths from source to post-edit. But, the goal here is to accurately align the known words, as well as learning the alignments of the new words, which is not feasible by this approach.

In order to improve the quality of the word alignments McCarley et al. (2011) proposed a trainable correction model which given a sentence pair and their corresponding automatically produced word alignment, it tries to fix the wrong alignment links. Similar to the hill-climbing approach used in IBM models 3-5 (Brown et al., 1993), this approach iteratively performs small modifications in each step, based on the changes of the previous step. However, the use of additional sources of knowledge, such as POS tags of the words and their neighbours, helps the system to take more accurate decisions. But, requiring manual word alignments for learning the alignment moves makes this approach only applicable for a limited number of language pairs for which manual aligned gold references are available.

Tomeh et al. (2010) introduced a supervised discriminative word alignment model for producing higher quality word alignments, which is trained on a manually aligned training corpus. To reduce the search space of the word aligner, they propose to provide the system with a set of automatic word alignments and consider the union of these alignments as the possible search space. This transforms the word alignment process into

the alignment refinement task in which given a set of automatic word alignments, the system tries to find the best word alignment points. Similar to (McCarley et al., 2011), this approach relies on the manually annotated training corpora which is not available for most of the language pairs.

3 Word Alignment

Word alignment is the task of finding the correspondence among the words of a sentence pair (Figure 1). From a mathematical point of view, it is a relation among the words, because any word in a sentence can be mapped into zero, one or more words of the other, and vice-versa; in other words, any kind of link is allowed, namely one-to-one, many-to-one, many-to-many, as well as leaving words unaligned. So called IBM models 1-5 (Brown et al., 1993) as well as the HMM-based alignment models (Vogel et al., 1996), and their variations are extensively studied and widely used for this task. They are directional alignment models, because permit only many-to-one links; but often the alignments in the two opposite directions are combined in a so-called symmetrized alignment, which is obtained by intersection, union or other smart combination.

Nowadays, word-aligners are mostly employed in an intermediate step of the training procedure of a SMT system; In this step, the training corpus is word aligned as a side effect of the estimation of the alignment models by means of the Expectation-Maximization algorithm. For this task, they perform sufficiently well, because the training data are often very large, and the limited amount of alignment errors do not have strong impact on the estimation of the translation model.

Instead, the already trained word-aligners are rarely applied for aligning new sentence pairs. In this task their performance are often not satisfactory, due to their poor generalization capability; they are especially prone to errors when infrequent or new words occur in the sentence pair.

This is the actual task to be accomplished in the online adaptive scenario: as soon as a new source and post-edited sentence pair is available, it has to be word aligned quickly and precisely. In this scenario, the sentence pair likely does not belong to the training corpus, hence might contain infrequent or new words, for which the aligner has little or no prior information.

3.1 Evaluation Measures

A word aligner is usually evaluated in terms of *Precision*, *Recall*, and *F-measure* (or shortly F), which are defined as follows (Fraser and Marcu, 2007):

$$Precision = \frac{|A \bigcap P|}{|A|} \ , \ Recall = \frac{|A \bigcap S|}{|S|}$$

$$F - measure = \frac{1}{\frac{\alpha}{Precision} + \frac{1-\alpha}{Recall}}$$

where A is the set of automatically computed alignments, and S and P refer to the *sure (unambiguous)* and *possible (ambiguous)* manual alignments; note that $S \subseteq P$. In this paper, α is set to 0.5 for all the experiments, in order to have a balance between Precision and Recall.

In this paper we are mainly interested how the word-aligner performs on the unknown words; hence, we define a version of Precision, Recall, and F metrics focused on the *oov-alignment* only, i.e. the alignments for which either the source or the target word is not included in the training corpus. The subscript all identifies the standard metrics; the subscript oov identifies their oov-based versions.

In Figure 1 we show manual and automatic word alignments between an English-Italian sentence pair. A sure alignment, like *are-sono*, is represented by a solid line, and a possible alignment, like *than-ai*, by a dash line. An oov-alignment, like that linking the unknown English word *deployable* to the Italian word *attivabili*, is identified by a dotted line. According to this example, Precision and Recall will be about 0.85 (=11/13) and 0.91 (=10/11), respectively, and the corresponding F is hence about 0.88. Focusing on the oov-alignment only, Precision_{oov} is 1.00 (=1/1), Recall_{oov} is 0.50 (=1/2), and F_{oov} is 0.67.

3.2 Evaluation Benchmark

In this paper, we compare word-alignment performance of three word-aligners introduced in Section 3.3 on three distinct tasks, namely English-Italian, English-French, and English-Spanish; the training corpora, common to all word-aligners, are subset of the JRC-legal corpus[1] (Steinberger et al.,), of the Europarl corpus V7.0 (Koehn, 2005), and of the Hansard parallel corpus[2], respectively.

[1] langtech.jrc.it/JRC-Acquis.html

[2] www.isi.edu/natural-language/download/hansard/index.html

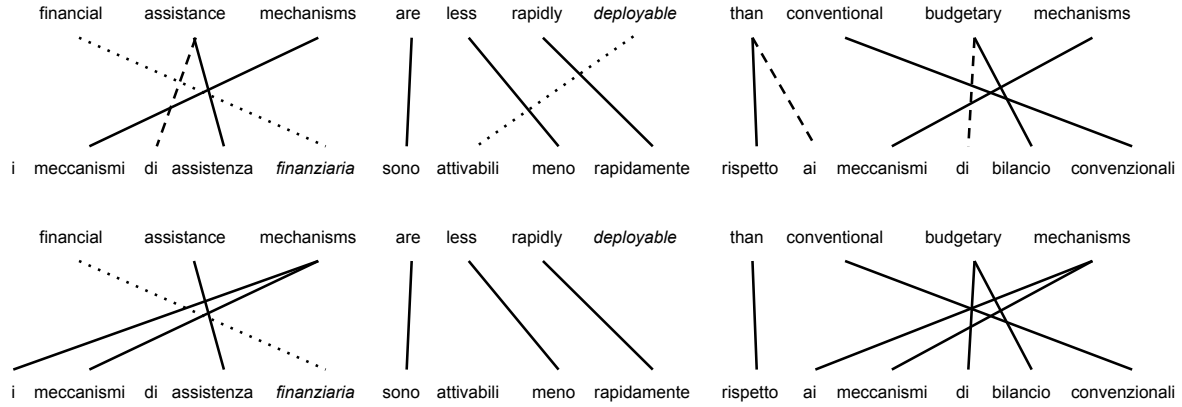

Figure 1: Example of manual (above) and automatic (below) word alignments between an English-Italian sentence pair. Sure and possible alignments are identified by solid and dash lines, respectively, and the oov-alignments by a dotted line. The OOV words, like *deployable* (English) and *finanziaria* (Italian), are printed in italics.

Statistics of the three training corpora are reported in Table 1.

	En-It	**En-Fr**	**En-Es**
Segments	940K	1.1M	713K
Tokens$_{src}$	19.8M	19.8M	19.8M
Tokens$_{trg}$	20.3M	23.3M	20.4M

Table 1: Statistics of the training corpora for English-Italian, English-French, and English-Spanish tasks.

Three evaluation data sets are also available, which belong to the same domains of the corresponding training corpora. The English-Italian test set was built by two professional translators by correcting an automatically produced word-alignment. The English-French test set is the manually aligned parallel corpus introduced in (Och and Ney, 2000)[3]. The English-Spanish test set was provided by (Lambert et al., 2005)[4]. Statistics of the three test sets are reported in Table 2.

To have a better understanding of the behavior of the word aligners on the unknown words, we created new test sets with an increasing ratio of the unknown words (*oov-rate*), for each task. Starting from each of the original test set, we replaced an increasing portion of randomly chosen words by strings which do not exist in the training corpus; the oov-noise artificially introduced ranges from

[3] www.cse.unt.edu/~rada/wpt/data/English-French.test.tar.gz

[4] www.computing.dcu.ie/~plambert/data/epps-alignref.html

	En-It	**En-Fr**	**En-Es**
Segments	200	484	500
Tokens$_{src}$	6,773	7,681	14,652
Tokens$_{trg}$	7,430	8,482	15,516
oov-rate$_{src}$	0.90	0.27	0.35
oov-rate$_{trg}$	0.84	0.34	0.32
#alignment	7,380	19,220	21,442

Table 2: Staticts of the test corpora for English-Italian, English-French, and English-Spanish tasks. oov-rate$_{src}$ and oov-rate$_{trg}$ are the ratio of the new words in the source and target side of the test corpus, respectively.

1% to 50%. For each value of the artificial oov-noise ($m = 1, ..., 50$), we randomly selected m% words in both the source and target side independently, and replaced them by artificially created strings. For selecting the words to be replaced by artificially created strings, we do not differentiate between the known and unknown words; hence the actual oov-rate in the test corpus, used in the plots, might be slightly larger.

To further make sure that the random selection of the words does not affect the systems, for each oov-noise we created 10 different test corpora and reported the averaged results. One might think of other approaches for introducing oov-noise, such as replacing singletons or low-frequency words which have more potential to be unknown, instead of randomly selection of the words. But in this paper we decided to follow the random selection of the words.

3.3 State-of-the-art Word Aligners

We consider three widely-used word aligners, namely *berkeley*, *fast-align*, and *mgiza++*. We analyze their performance in aligning an held-out test corpora; in particular, we compare their capability in handling the unknown words. For a fair comparison, all aligners are trained on the same training corpora described in Section 3.2.

berkeley aligner (Liang et al., 2006) applies the co-training approach for training the IBM model 1 and HMM. We trained *berkeley* aligner using 5 iterations of model 1 followed by 5 iterations of HMM. When applied to new sentence pairs, the system produces bi-directional symmetrized alignment.

fast-align is a recently developed unsupervised word aligner that uses a log-linear reparametrization of IBM model 2 for training the word alignment models (Dyer et al., 2013). We exploited the default configuration with 5 iterations for training. As the system is directional, we trained two systems (source-to-target and target-to-source). When applied to new sentence pairs, we first produced the two directional alignments, and then combined them into a symmetrized alignment by using the *grow-diag-final-and* heuristic (Och and Ney, 2003).

mgiza++ (Gao and Vogel, 2008) and its ancestors, i.e. *giza*, and *giza++*, implement all the IBM models and HMM based alignment models. *mgiza++* is a multithreaded version of giza++, which enables an efficient use of multi-core platforms. We trained the system using the following configuration for model iterations: $1^5h^53^34^3$. *mgiza++* also produces directional alignment; hence, we followed the same protocol to create a symmetrize alignment of sentence pairs as we did for *fast-align*.

Differently from *berkeley* and *fast-align*, *mgiza++* somehow adapts its models when applied to new sentence pairs. According to the so-called "forced alignment", it essentially proceeds with the training procedure on these new data starting from pre-trained and pre-loaded models, and produces the alignment as a by-product. In preliminary experiments, we observed that performing 3 iterations of model 4 is the best configuration for *mgiza++* to align the new sentence pairs.

These word aligners are designed to work in offline mode; they load the models and align the whole set of available input data in one shot. However, in the online scenario where a single sentence pair is provided at a time, they need to reload the models every time which is very expensive in terms of I/O operations. In this paper we first were interested in measuring the quality of the word aligners to select the best one. Therefore, we mimic the online modality by forcing them to align one sentence pair at a time.

	Precision		**Recall**		**F-measure**	
	all	*oov*	*all*	*oov*	*all*	*oov*
English-Italian						
fast-align	82.6	33.3	82.8	19.6	82.7	24.7
berkeley	91.9	–	81.0	–	86.1	–
mgiza++	86.2	84.6	89.4	30.8	87.8	45.2
English-French						
fast-align	81.5	47.2	91.8	19.5	86.3	27.6
berkeley	87.9	–	92.9	–	90.3	–
mgiza++	89.0	88.2	96.0	17.2	92.4	28.8
English-Spanish						
fast-align	81.5	31.3	71.8	12.7	76.3	18.1
berkeley	88.7	–	71.2	–	79.0	–
mgiza++	89.2	95.5	80.6	35.6	84.7	51.9

Table 3: Comparison of different widely-used word aligners in terms of precision, recall, and F-measure on English-Italian, English-French, and English-Spanish language pairs. Columns *all* report the evaluation performed on all alignments, while columns *oov* the evaluation performed on the oov-alignments.

The three word aligners were evaluated on the three tasks introduced in Section 3.2. Table 3 shows their performance on the full set of alignments (*all*) and on the subset of oov-alignments (*oov*) in terms of Precision, Recall, and F-measure. The figures show that all aligners perform well on the whole test corpus. *mgiza++* is definitely superior to *fast-align*; it also outperforms *berkeley* in terms of F-measure, but they are comparable in terms of Precision.

Unfortunately, the quality of the word alignments produced for the new words is quite poor for all systems. *mgiza++* outperforms the other aligners in all the language pairs on oov-alignments, and in particular it achieves a very high precision. On the contrary, *berkeley* aligner always fails to detect out-of-vocabulary words; its precision is hence undefined, and consequently its F-measure. To our knowledge of the system, this behavior is expected because of the joint alignment approach used in *berkeley* which produces an alignment between two terms if both the directional models

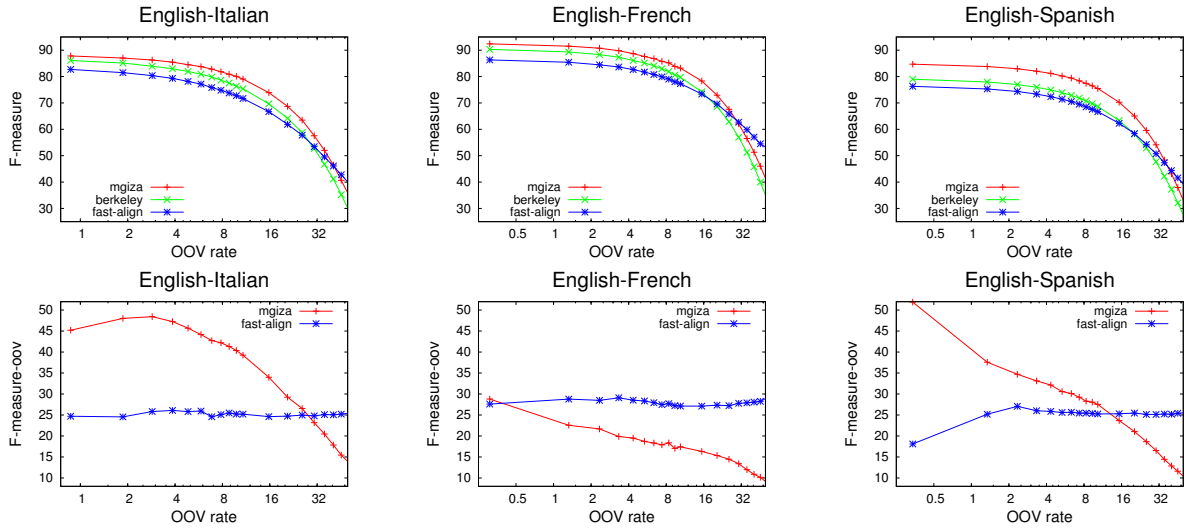

Figure 2: Performance in terms of standard F-measure (above) and oov-based F-measure (below) of the word aligners on test sets with increasing oov-rate, for all language pairs. The oov-based F-measure for *berkeley* is not reported because it is undefined.

agree, and this hardly occurs for unknown words.

To further investigate the behavior of the word aligners on the unknown words, we evaluated their performance on the artificially created test sets, described in Section 3.2. The performance of the word aligners in terms of standard and oov-based F-measure is shown in Figure 2. As expected, the overall F-measure decreases by introducing unknown words. *mgiza++* is more accurate than the other aligners up to oov-rate of 16%.

We observe that *mgiza++* outperforms the others in terms of the oov-based F-measure on the English-Italian and English-Spanish language pairs up to oov-noise of 32% and 16%, respectively. *fast-align* instead performs better in the English-French task. *fast-align* always show a better quality when the oov-rate is very high. oov-based F-measure is not reported for *berkeley* because this aligner is not able to detect oov-alignments as explained above.

4 Enhancement to Word Alignment

4.1 Refinement of oov-alignments

To address the problem of unaligned new words, we present a novel approach, in which the word alignments of the source and target segment pair are induced in two-steps. First, a standard word aligner is applied; most of the words in the source and target sentence pair will be aligned, but most of the unknown words will not. It is worth mentioning that aligning unknown words in this step depends on the quality of the employed word aligner. Once the alignments are computed and symmetrized (if required), phrase extraction procedure is applied to extract all valid phrase-pairs. Note that un-aligned words are included in the extracted phrase pairs, if their surrounding words are aligned.

It has been shown that inclusion of un-aligned words in the phrase-pairs, generally, has negative effects on the translation quality and can produce errors in the translation output (Zhang et al., 2009). Nevertheless, the overlap among phrase-pairs, which contain un-aligned unknown words, can be considered as a valuable source of knowledge for inducing the correct alignment of these words. To get their alignments from the extracted phrase-pairs we follow an approach similar to (Esplá-Gomis et al., 2012) in which the word alignment probabilities are determined by the *alignment strength* measure. Given the source and target segments ($S = \{s_1, \ldots, s_l\}$ and $T = \{t_1, \ldots, s_m\}$), and the set of extracted parallel phrase-pairs (Φ), the alignment strength $\mathcal{A}_{i,j}(S, T, \Phi)$ of the s_i and t_j can be calculated as follows:

$$\mathcal{A}_{i,j}(S,T,\Phi) = \sum_{(\sigma,\tau)\in\Phi} \frac{cover(i,j,\sigma,\tau)}{|\sigma|.|\tau|}$$

$$cover(i,j,\sigma,\tau) = \begin{cases} 1 & \text{if } s_i \in \sigma \text{ and } t_j \in \tau \\ 0 & \text{otherwise} \end{cases}$$

where $|\sigma|$ and $|\tau|$ are the source and target lengths (in words) of the phrase pair (σ, τ).

$cover(i, j, \sigma, \tau)$ simply spots whether the word-pair (s_i, t_j) is covered by the phrase pair (σ, τ).

The alignment strengths are then used to produce the a directional source-to-target word alignments; s_i is aligned to t_j if $\mathcal{A}_{i,j} > 0$ and $\mathcal{A}_{i,j} \geq \mathcal{A}_{i,k}, \ \forall k \in [1, |T|]$. One-to-many alignment is allowed in cases that multiple target words have equal probabilities to be aligned to i-th source word ($\mathcal{A}_{i,j} = \mathcal{A}_{i,k}$). The directional word alignments are then symmetrized.

The new set of symmetrized alignments can be used in different ways: (i) as a replacement of the initial word alignments as in (Esplá-Gomis et al., 2012), or (ii) as additional alignment points to be added to the initial set. According to a preliminary investigation, we choose the latter option: only a subset of the new word alignments is used for updating the initial alignments. More specifically, we add only the alignments of the new words which are not already aligned.

Moreover, our approach differs from that proposed by Esplá-Gomis et al. (2012) in the procedure to collect the original set of phrase pairs from the source and target sentence pair. They rely on the external sources of information such as online machine translation systems (e.g. Google Translate, and Microsoft Translator). Communicating with external MT systems imposes some delays to the pipeline, which is not desired for the online scenario. Furthermore, the words that are not known by the machine translation systems are not covered by any phrase-pair, hence the refinement module is not able to align them.

We instead employ the *phrase-extract* software[5] provided by the Moses toolkit, which relies on the alignment information of the given sentence pair, and allows the inclusion of un-aligned unknown words in the extracted phrase pairs; hence, the refinement module has the potential to find the correct alignment for those words.

Note that there is no constraint on the word alignment and phrase extraction modules used in the first step, hence, any word aligner and phrase extractor can be used for computing the initial alignments and extracting the parallel phrase pairs from the given sentence pairs. But, since the outputs of the first aligner make the ground for obtaining the alignments of the second level, they need to be highly accurate and precise.

[5]The "grow-diag-final-and" heuristic was set for the symmetrization.

4.2 onlineMgiza++

The experiments to compare state-of-the-art word aligners, reported and discussed in Section 3, are carried out offline. This is because the aforementioned word aligners are not designed to work online, and need to load the models every time receives a new sentence pair. Loading the models is very time consuming, and depending on the size of the models might take several minutes, which is not desired for the online scenario.

To overcome this problem, we decided to implement an online version of *mgiza++* which provides the best performance as shown in Section 3.3. This new version, called *onlineMgiza++*, works in client-server mode. It consists of two main modules *mgizaServer* and *mgizaClient*. *mgizaServer* is responsible for computing the alignment of the given sentence pairs. To avoid unnecessary I/O operations, *mgizaServer* loads all the required models once at the beginning of the alignment session, and releases them at the end. *mgizaClient* communicates with the client applications through the standard I/O channel.

In our final experiments we observed some unexpected differences between the results of *mgiza++* and *onlineMgiza++*. Therefore, we do not present the results of *onlineMgiza++* in this paper. However, we expect the two systems produce the same results.

5 Experimental Results

In this section we evaluate the effectiveness of the proposed refinement module. Each considered word aligner was equipped by our refinement module, and compared to its corresponding baseline. Figure 3 shows the oov-based F-measure achieved by the baseline and enhanced word aligners on all test sets and all tasks. We observe that the refinement module consistently improves the F-measure of all aligners on all language pairs;

The improvement for *mgiza++* are big (up to 10%) for very low oov-rates and decreases when the oov-rate increases; the same but smaller behavior is observed for *fast-align*. This is due to the fact that by inserting more oov words into the test sets the systems are able to produce less accurate alignment points, which leads in lower contextual information (i.e. smaller number of overlapping phrase-pairs) for aligning the unknown words. Interestingly, the refinement module applied to the *berkeley* output permits the correct detection of

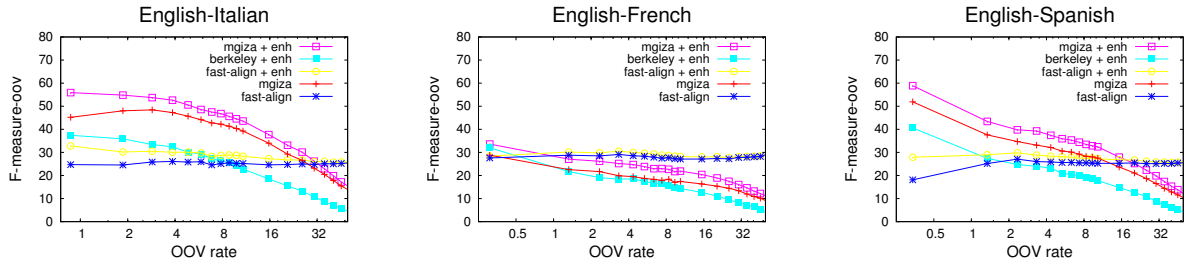

Figure 3: Performance in terms of oov-based F-measure of the baseline and enhanced word aligners on test sets with increasing oov rate, for all language pairs. The oov-based F-measure for *berkeley* is not reported because it is undefined.

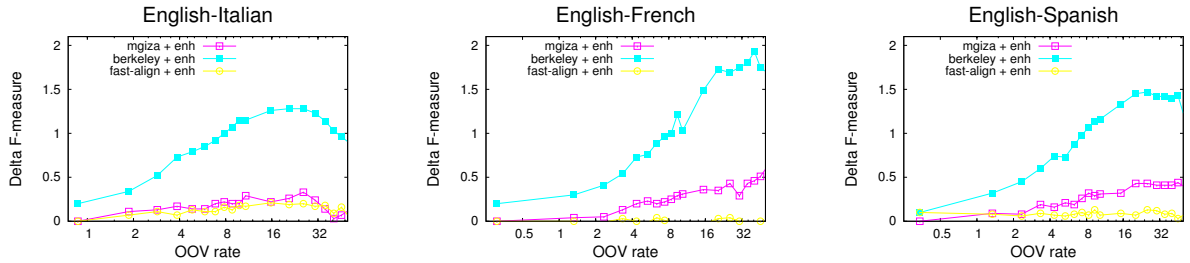

Figure 4: Difference of performance in terms of standard F-measure of the enhanced word aligners from their corresponding baselines on test sets with increasing OOV rate, for all language pairs.

many oov-alignments, which the baseline system can not find most of them.

Furthermore, Figure 4 reports the F-measure differences achieved by the enhanced word-aligners from their corresponding baselines on the full data sets. The refinement module slightly but consistently improves the overall F-measure as well, especially for high oov-rates. The highest improvement is achieved by the enhanced *berkeley* aligner, mainly because its baseline performs worse in this condition.

6 Conclusion

In this paper we discussed the need of having a fast and reliable online word aligner in the online adaptive MT scenario that is able to accurately align the new words. The quality of three state-of-the-art word aligners, namely *berkeley*, *mgiza++*, and *fast-align*, were evaluated on this task in terms of Precision, Recall, and F-measure. For this purpose we created a benchmark in which an increasing amount of the words of the test corpus are randomly replaced by new words in order to augment the oov-rate. The results show that the quality of the aligners on new words is quite low, and suggest that new models are required to effectively address this task. As a first step, we proposed a fast and language independent procedure for aligning the unknown words which refines any given automatic word alignment. The results show that the proposed approach significantly increases the word alignment quality of the new words.

In future we plan to evaluate our approach in an end-to-end evaluation to measure its effect on the final translation. We also plan to investigate the exploitation of additional features such as linguistic and syntactic information in order to further improve the quality of the word alignment models as well as the proposed refinement procedure. However, this requires other policies of introducing new words, rather than just randomly selecting the words and replacing them by artificial strings.

Acknowledgments

This work was supported by the MateCat project, which is funded by the EC under the 7^{th} Framework Programme.

References

F. Blain, H. Schwenk, and J. Senellart. 2012. Incremental adaptation using translation information and post-editing analysis. In *International Workshop on Spoken Language Translation*, pages 234–241, Hong-Kong (China).

Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. 1993. The

mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–312.

Chris Dyer, Victor Chahuneau, and Noah A. Smith. 2013. A simple, fast, and effective reparameterization of ibm model 2. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 644–648, Atlanta, Georgia, June. Association for Computational Linguistics.

Miquel Esplá-Gomis, Felipe Sánchez-Martínez, and Mikel L. Forcada. 2012. A simple approach to use bilingual information sources for word alignment. *Procesamiento del Lenguaje Natural*, (49):93–100.

Alexander Fraser and Daniel Marcu. 2007. Measuring word alignment quality for statistical machine translation. *Comput. Linguist.*, 33(3):293–303.

Qin Gao and Stephan Vogel. 2008. Parallel implementations of word alignment tool. In *Software Engineering, Testing, and Quality Assurance for Natural Language Processing*, SETQA-NLP '08, pages 49–57, Stroudsburg, PA, USA. Association for Computational Linguistics.

Daniel Hardt and Jakob Elming. 2010. Incremental re-training for post-editing smt. In *9th Conference of the Association for Machine Translation in the Americas (AMTA)*, Denver, United States.

Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *Proceedings of the Tenth Machine Translation Summit (MT Summit X)*, pages 79–86, Phuket, Thailand.

Patrik Lambert, Adrià de Gispert, Rafael E. Banchs, and José B. Mariño. 2005. Guidelines for word alignment evaluation and manual alignment. *Language Resources and Evaluation*, 39(4):267–285.

Percy Liang, Ben Taskar, and Dan Klein. 2006. Alignment by agreent. In *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference*, pages 104–111, New York City, USA, June. Association for Computational Linguistics.

J. Scott McCarley, Abraham Ittycheriah, Salim Roukos, Bing Xiang, and Jian-ming Xu. 2011. A correction model for word alignments. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 889–898, Stroudsburg, PA, USA. Association for Computational Linguistics.

Franz Josef Och and Hermann Ney. 2000. Improved statistical alignment models. In *Proceedings of the 38th Annual Meeting of the Association of Computational Linguistics (ACL)*.

F.J. Och and H. Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.

Ralf Steinberger, Bruno Pouliquen, Anna Widiger, Camelia Ignat, Tomaž Erjavec, Dan Tufiş, and Dániel Varga. The jrc-acquis: A multilingual aligned parallel corpus with 20+ languages. In *In Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC)*, pages 2142–2147, Genoa, Italy.

Nadi Tomeh, Alexandre Allauzen, Guillaume Wisniewski, and Franois Yvon. 2010. Refining word alignment with discriminative training. In *Proceedings of the ninth Conference of the Association for Machine Translation in the America (AMTA)*.

S. Vogel, H. Ney, and C. Tillmann. 1996. HMM-based word alignment in statistical translation. In *Proceedings of COLING*, pages 836–841, Copenhagen, Denmark.

Yuqi Zhang, Evgeny Matusov, and Hermann Ney. 2009. Are unaligned words important for machine translation? In *Conference of the European Association for Machine Translation*, pages 226–233, Barcelona, Spain.