Unsupervised Parallel Sentence Extraction with Parallel Segment Detection Helps Machine Translation

Viktor Hangya and **Alexander Fraser**
Center for Information and Language Processing
LMU Munich, Germany
{hangyav, fraser}@cis.lmu.de

Abstract

Mining parallel sentences from comparable corpora is important. Most previous work relies on supervised systems, which are trained on parallel data, thus their applicability is problematic in low-resource scenarios. Recent developments in building unsupervised bilingual word embeddings made it possible to mine parallel sentences based on cosine similarities of source and target language words. We show that relying only on this information is not enough, since sentences often have similar words but different meanings. We detect continuous parallel segments in sentence pair candidates and rely on them when mining parallel sentences. We show better mining accuracy on three language pairs in a standard shared task on artificial data. We also provide the first experiments showing that parallel sentences mined from real life sources improve unsupervised MT. Our code is available, we hope it will be used to support low-resource MT research.

1 Introduction

The performance of machine translation has improved significantly recently, with some claims of even being close to human parity (Hassan et al., 2018), but a large amount of parallel data is required for high quality systems. For many language pairs the size of the available training data is not adequate. Recently, developments in the field of unsupervised bilingual word embeddings (BWEs) made it possible to build MT systems without any parallel data. Both statistical (Lample et al., 2018b; Artetxe et al., 2018b) and neural (Artetxe et al., 2018c; Lample et al., 2018a) MT approaches were proposed which are promising directions to overcome the data sparsity problem. However, various issues of the approaches still have to be solved, e.g., better word reordering during translation or tuning system parameters. For many interesting low resource language pairs, we do not have enough parallel data, but we do have access to sources of comparable monolingual text. In this paper we propose a strong unsupervised system for parallel sentence mining and show that the mined data improves the performance of unsupervised MT systems.

Previously many approaches tackled the problem of parallel sentence extraction but they were relying on different levels of bilingual signals either to build dictionaries (Grover and Mitra, 2017), parallel sentence classifiers (Bouamor and Sajjad, 2018) or bilingual sentence representations (Schwenk, 2018). An unsupervised system was also proposed which only relied on unsupervised BWEs, thus no additional resources are needed (Hangya et al., 2018). We use this approach as our baseline and show that relying only on word similarity information leads to false positive sentence pairs, such as in this example:

- *The US dollar has a considerable role in the international monetary system.*
- *Die Rolle des US Dollar im internationalen Geldsystem sollte neu überdacht werden.* *(The role of the US dollar in the international monetary system should be reconsidered.)*

Both sentences mention the *role of the US dollar in the international monetary system*, but the overall claim is different. One major disadvantage of the approach of (Hangya et al., 2018) is that, by only relying on word similarities, sentence pairs which have similar meanings but are not exactly parallel are often mined. We overcome this problem by detecting continuous parallel segments in the candidate sentence pairs. We align similar words in the candidate sentence pairs, instead of just averaging their similarity, and use the alignments in order to detect continuous sub-sentential segments on both

Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 1224–1234
Florence, Italy, July 28 - August 2, 2019.

sides that are aligned with each other. In order to increase the precision of our system we only mine similar sentence pairs where the detected parallel segments form a large part of the full sentence pairs thus overcoming the problem of only nearly parallel sentence pairs mentioned above.

We conduct two sets of experiments to show that our system mines more useful parallel sentences and that they are beneficial for MT systems. First, we evaluate the accuracy of the mining approach on the BUCC 2017 shared task data (Zweigenbaum et al., 2017). We show that by looking for continuous parallel segments we can increase the performance significantly compared to (Hangya et al., 2018), especially the precision of the system, on German-, French- and Russian-English language pairs.[1] Second, since the data used in previous work was artificially assembled, we use real life German and English monolingual news crawl data to mine parallel sentences, and use them to improve an unsupervised neural MT system by using the extracted data as silver-standard parallel training data. We show for the first time that exploiting comparable monolingual text sources with an unsupervised parallel sentence mining system helps unsupervised MT. Furthermore, we achieve increased performance compared with the previous unsupervised mining system.

2 Related Work

Most previous systems addressing parallel sentence extraction depend on bilingual resources which makes their applicability problematic in low-resource scenarios. Munteanu et al. (2004) used a bilingual dictionary and a small number of parallel sentences to train a maximum entropy classifier for mining Arabic and English parallel sentences. Similarly, parallel data was used to train IBM Model 1 and a maximum entropy classifier (Smith et al., 2010). Munteanu and Marcu (2006) extracted parallel sub-sentential segments from partly parallel sentences and used them to improve a statistical MT system. We follow this idea in our work and detect continuous parallel segments in order to weight the similarity values of candidate sentence pairs. To further promote the task, the BUCC 2017 shared task – *Identifying parallel sentences in comparable corpora* – was organized, where parallel sentences were automatically inserted into two monolingual corpora to produce gold standard train and test data in order to measure the performance of participating systems (Zweigenbaum et al., 2017). Since then, various neural architectures were proposed. Bilingual word embeddings were used in (Grover and Mitra, 2017), neural sentence pair classifiers were used in (Bouamor and Sajjad, 2018) and bilingual sentence representations were trained in (Schwenk, 2018). The disadvantage of the mentioned methods is that they need a bilingual signal to be trained, in contrast with our approach which only uses monolingual data. A fully unsupervised system was proposed in (Hangya et al., 2018) but the system introduced too much noise by mining sentence pairs with similar words but different meaning. Also, the usefulness of the system in downstream tasks was not tested.

Our approach is based on BWEs where representations of source and target language words are in the same bilingual space. Previous approaches building BWEs were using bilingual signals of various granularity. Following Mikolov et al. (2013), many authors map monolingual word embeddings into the same bilingual space (Faruqui and Dyer, 2014; Xing et al., 2015), others leverage parallel texts (Gouws et al., 2015) or create artificial cross-lingual corpora using seed lexicons or document alignments (Vulić and Moens, 2015; Duong et al., 2016) to train BWEs. Several authors have shown that good quality BWEs can be trained by mapping monolingual spaces without any bilingual signal. Conneau et al. (2018) used adversarial training to rotate the source space to match the target and extracted an initial lexicon to fine tune the mapping. Others used word neighborhood information to create an initial mapping (Artetxe et al., 2018a; Alvarez-Melis and Jaakkola, 2018). We use the work of Conneau et al. (2018) to build BWEs for parallel sentence extraction.

The development of unsupervised BWEs opened the door to creating machine translation systems without any parallel data. Unsupervised BWEs are used to make initial word-by-word translating systems which are then improved by iterative back-translation (Sennrich et al., 2016) using neural systems (Lample et al., 2018a; Artetxe et al., 2018c; Yang et al., 2018). It is also possible to initialize phrase tables for statistical MT systems and increase their performance with the

[1] Chinese-English is left for future work, as a study of unsupervised Chinese word segmentation approaches is needed.

same back-translation techniques (Lample et al., 2018b; Artetxe et al., 2018b). Although the initial results are promising, there are many issues still to be solved. In our experiments we use the NMT system of (Artetxe et al., 2018c). We show that the addition of our mined parallel data improves performance over baseline results.

3 Approach

Our approach for mining parallel sentences is based on calculating the similarity of sentence pair candidates. To avoid mining pairs having similar words but different meaning we look for continuous parallel segments in the candidates based on word alignments. We use the length of the segments to either filter the candidate out or to weight the averaged similarity scores of words to get the final score of a given candidate.

3.1 Word Similarity

The first step of our method is to define the similarity of words. For this we use BWEs, where source and target language words are embedded in the same vector space. First, we build monolingual word embeddings and map the source words into the target space. Initially, a seed lexicon of source and target language words was needed to learn a mapping between the two spaces (Mikolov et al., 2013). Conneau et al. (2018) showed that good quality BWEs can be produced without any bilingual signal, by using an adversarial system to learn an initial mapping of the two spaces and mine frequent source words and their most similar pairs from the target language to form an initial seed lexicon. Using this initial lexicon the mapping can be further tuned using orthogonal mapping (Xing et al., 2015). We use the system of Conneau et al. (2018) to build unsupervised BWEs.

To measure similarity of words we use the cosine similarity based *Cross-Domain Similarity Local Scaling* (CSLS) metric (Conneau et al., 2018) which aims to overcome the hubness problem of high dimensional spaces (Dinu et al., 2015). In short, this metric adjusts the similarity values of a word based on the density of the area where it lies, i.e., it increases similarity values for a word lying in a sparse area and decreases values for a word in a dense area. We create a dictionary of the 100 nearest target words for each source language word with their similarities using CSLS.

Even though good quality dictionaries can be built based on BWEs, the translations of some words, such as named entities and rare words, can be improved using orthographic information (Braune et al., 2018; Riley and Gildea, 2018). We follow the approach of Braune et al. (2018) and create a dictionary similar to the dictionary in the previous paragraph but using orthographic similarity of words, i.e., one minus normalized Levenshtein distance, instead of CSLS. We then merge the two dictionaries to get the final set of similar word pairs by taking all target words from both dictionaries for each source language word[2].

To build monolingual embeddings we use *fastText*'s skipgram model (Bojanowski et al., 2017) with dimension size 300 and keeping all other parameters default[3]. We use *MUSE* as the implementation of (Conneau et al., 2018) with default parameters[4] for building unsupervised BWEs.

3.2 Parallel Segment Detection

The next step of our approach is to calculate the similarities of sentence pair candidates using the dictionaries created above. Various algorithms were proposed to measure sentence similarities, such as the Hungarian alignment (Kuhn, 1955; Varga et al., 2007) and the Word Mover's Distance (Kusner et al., 2015). On the other hand, these methods are computationally expensive for parallel sentence extraction where the number of sentence pair candidates is huge. Due to performance considerations Hangya et al. (2018) proposed a fast word similarity based method to calculate sentence similarity by averaging the scores of the most similar words. The disadvantage of relying only on similar words is that non-parallel candidates having similar words are often wrongly mined, as already discussed. To overcome this problem, we align words in the candidate sentence pairs in order to detect parallel segments similarly to Munteanu and Marcu (2006). Our hypothesis is that such continuous segments are more related, thus candidates having long enough segments are parallel.

Our algorithm is illustrated in Figure 1. We iterate over the source sentences from left to right and greedily align each source word to the most similar target word that was not already aligned. We note that source words can be left unaligned if

[2]If a translation is in both dictionaries we take the max of the values.

[3]See the Facebook Research fastText GitHub page.

[4]See the Facebook Research MUSE GitHub page.

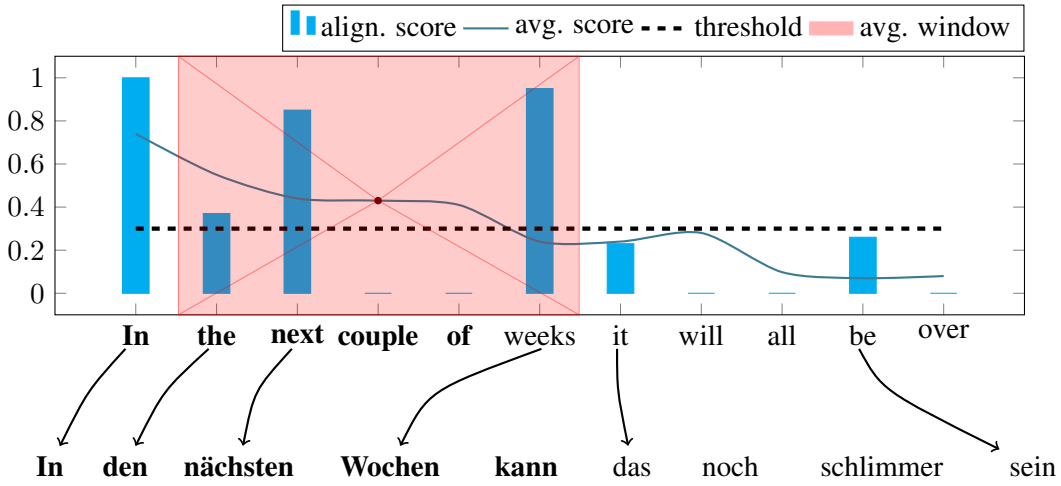

Figure 1: The figure depicts our algorithm for parallel segment detection on a non-parallel sentence pair. The aligned words and their scores are shown together with the smoothed values using average filtering of window size 5. Detected segments with respect to 0.3 threshold value are bolded on both source (En) and target (De) sides. Averaged scores on the target side are calculated based on target sentence word order which is used for target segment detection (we omit this part of the diagram). To decide if the pair is parallel we average word alignment scores of the full source sentence, weight it using the length of the detected segment and check if it reaches a given threshold. Translation of the target sentence: *In the next weeks this can be even worse*.

none of the possible target words are in the used dictionary entry for that word. Similarly, target words could be unaligned as well. We assign an alignment score for each position of the source and target sentences respectively. The alignment score for a word at position i is its similarity score to its aligned word (taken from the dictionary used) or 0 if the word is unaligned. We then look for continuous segments on both source and target sides by looking for sequences of indices where the alignment scores are higher then a given threshold value. Since the use of mostly function words could vary across languages, e.g., En: *in the international* vs. De: *im (in+dem) internationalen*, these words often remain unaligned resulting in gaps in the sequences, and so fragmented parallel segments are formed. To allow a small number of unaligned words in the extracted segments we apply an average filter on the alignment score sequences with a predefined window size at each position giving a smoothed alignment value. After extracting segments from both sides of a candidate pair, we align source and target side segments by matching those which have the most word alignments between each other. The number of segments could be unbalanced on the two sides thus we ignore segments which are not aligned with segments on the other side. Furthermore, we filter segments by dropping all segment pairs if i) either side is shorter than a given threshold and if ii) the length difference of the pair is larger than 5 tokens. We note that our algorithm at this point can be used to mine parallel segments from sentence pairs. However, our focus in this paper is to mine complete sentence pairs which we describe in the following.

3.3 Parallel Sentence Mining

To acquire the final similarity score for a candidate sentence pair we use both word alignment scores and the detected segments. If no parallel segment is detected or remains after the filtering steps we consider the candidate as non-parallel, i.e., set its similarity score to 0. Otherwise, we average word alignment scores of the full sentence and weight it with the ratio between the length of the longest source segment and that of the full sentence. This way if a candidate pair has highly similar words but has unparallel parts we decrease its overall similarity. We consider a candidate pair as parallel if its score is larger than a given threshold value. We note, that we only use the longest segment in order to reach high precision. It is possible that the segments are fragmented in parallel sentence pairs separated by short non-parallel phrases, resulting in false negatives. On the other hand, using the sum of the length of all segments could lead to false positives. Thus, we only rely on the longest segment and use the size parameter of the average filter to balance the fragmentation. We detail the

used parameters for each experiment in the following sections.

We applied pre-filtering of candidates due to the large number of possible sentence pairs. Following Grégoire and Langlais (2017), we only consider the 100 most similar target sentences for each source sentence as candidates. We calculate sentence similarity by embedding them using averaged word vectors and measuring their cosine distance which can be run efficiently using GPUs even on large datasets (Johnson et al., 2017).

4 Evaluation on BUCC 2017

We conduct our first set of experiments on the BUCC 2017 shared task data (Zweigenbaum et al., 2017). The aim of this shared task is to quantitatively evaluate methods for extracting parallel sentences from comparable monolingual corpora. Train, development and test datasets were built for 4 language pairs German-, French-, Russian- and Chinese-English language pairs. The data was built automatically by inserting parallel news commentary sentences into monolingual wikipedia dumps. To make sure that the insertions are not easy to detect parallel sentences were only inserted if other strongly related sentences in terms of their topic are present in the monolingual corpus. We use the system of (Hangya et al., 2018) as our baseline and run experiments on the first three language pairs (as we already mentioned, we would need to study Chinese unsupervised word segmentation to run Zh-En experiments). We consider English as the target language in all cases.

4.1 Evaluation Setup

Following the data selection and preprocessing steps of the baseline we use monolingual news crawls, downloaded between 2011 and 2014 taken from the WMT 2014 shared task (Bojar et al., 2014), for building the initial monolingual word embeddings. We tuned our system parameters using the development data on all language pairs. We performed tuning in the following intervals: threshold value for segment detection $0.2 - 0.4$; window size of average filter $5 - 20$; threshold value for deciding parallelism $0.1 - 0.6$; minimum segment length $20\% - 50\%$ of the original sentence. We note that for the experiments in this section we kept the minimum segment length low in order not to filter out candidates aggressively but to decrease their scores instead. This way candidates with short segments could still be mined. In Section 5 we will use a higher value to favor precision over recall. Besides using a static value for deciding parallelism we also used the dynamic thresholding proposed in (Hangya et al., 2018):

$$th = \bar{S} + \lambda * std(S) \quad (1)$$

where S is a set containing the similarity values of each source sentence in the test set and its most similar target candidate, $\bar{S}$ and $std(S)$ are its mean and standard deviation. We performed a less intensive tuning of λ as suggested. As in previous work, we evaluate our system on the training set of the shared task since the official test set is undisclosed. We do not use the train set to either train or tune our system.

		P (%)	R (%)	F_1 (%)
De-En	avg	23.71	44.57	30.96
	align-static	44.63	41.13	42.81
	align-dyn	48.53	39.18	43.35
Fr-En	avg	39.02	52.61	44.81
	align-static	43.20	41.27	42.21
	align-dyn	50.51	38.11	43.44
Ru-En	avg	16.75	24.20	19.80
	align-static	25.85	23.33	24.53
	align-dyn	37.44	18.73	24.97

Table 1: Precision, recall and F_1 scores for our proposed system and the baseline (avg) on the BUCC 2017 dataset.

4.2 Results

We show precision, recall and F_1 scores in Table 1 for the three language pairs. In addition to the baseline (avg) system, which only relies on averaged word similarity scores, we show the performance of our proposed system with static and dynamic thresholding. Our system achieved a significant increase of F_1 for German- and Russian-English language pairs. For both pairs we achieved a large increase of precision, especially in the case of German-English where the improvement is over 20%. On the other hand, we experienced a slight drop of recall due to our stricter approach for the mining process. For the French-English language pair the F_1 score has decreased slightly. It can be seen that the precision of the system was significantly increased for this language pair as well, proving that we extract less pairs which are similar but not parallel. In contrast,

1.	Benchmarking-Ergebnisse werden u.a. im Global Competitiveness Report des World Economic Forum veröffentlicht. *Benchmarking results are published among others in the World Economic Forum's Global Competitiveness Report.* These ratios are compiled and published by the World Economic Forum.
2.	Ende 1994 gelang es dem afghanischen Verteidigungsminister Ahmad Schah Massoud, Hekmatyr und die verschiedenen Milizen militärisch in Kabul zu besiegen. *At the end of 1994, Afghan defense minister Ahmad Shah Massoud succeeded in defeating Hekmatyr and the various militias in Kabul.* In late 1994, Rabbani's defense minister, Ahmad Shah Massoud defeated Hekmatyr in Kabul and ended ongoing bombardment of the capital.
3.	Die 20 größten Städte der Welt sind, bis auf drei Ausnahmen, in Schwellenländern zu finden. *The 20 largest cities in the world, with three exceptions, can be found in emerging markets.* Indeed, all but three of the worlds 20 largest cities are in emerging markets.

Table 2: German-English examples with translations of German sentences shown in italic. Examples 1 and 2 are false positives of the baseline but not our proposed system while example 3 is a false negative of our approach.

our conservative approach also misses true parallel pairs resulting in a significant drop in recall. However, we argue that precision is more important for downstream tasks, since noise in the data often hurts performance. Based on non-mined parallel examples we found that French segments tend to be more fragmented compared to other languages which leads to a stronger decrease in the sentence pair similarity scores. One solution to the problem could be to use a larger window size when detecting parallel segments.

Using static and dynamically calculated threshold values performs comparably. It can be seen that dynamic thresholding achieved higher precision but lower recall when compared with the static value. Furthermore, the increase of precision is higher than the decrease of recall, resulting in better F_1 scores as well. In the baseline dynamic thresholding was needed due to the system's sensitiveness to the threshold value. In contrast, for our system there is a bigger gap between similarity scores of parallel and non-parallel sentence pairs due to segment length based weighting, so for this reason the tuned static value worked well on the test set.

We manually analyzed German-English examples to highlight the differences of our system and the baseline. We show samples in Table 2 where 1 and 2 are falsely mined by the baseline while 3 is missed by our proposed system. Although example 1 seems parallel, there is some additional information on the source side. Since the words are similar, the baseline system incorrectly mines this pair. On the other hand, our approach ignores it because the detected segment is only *Competitiveness Report des World Economic Forum veröffentlicht*, while the words in the beginning do not form a continuous segment thus decreasing its overall score aggressively. Similarly, example 2 has different content at the end of the sentence pair which makes the detected segment short even though there are similar words in the pair. Example 3 is a parallel sentence pair which was missed by our system but not by the baseline. The reason lies in the wording of a short segment in the sentences. The source side phrase *bis auf drei Ausnahmen (with three exceptions)* is expressed as *all but three* on the target side. This difference results in two shorter segments (*die 20 größten Städte der Welt* and *in Schwellenländern zu finden*) in the sentence which decreases the similarity score below the threshold. Such false negatives occurred when a short non-parallel segment divides a longer parallel segment which could be solved by either using larger window size for the average filter or by merging segments if they are a few tokens away from each other. On the other hand, this could also introduce false positives.

In general, we can conclude that we improved F_1 score significantly, except for French-English where the baseline performed only a couple of percentage points better. Furthermore, our method achieved the highest precision, out-performing the baseline in all three language pairs, which is more important when mining from the web (Xu and Koehn, 2017).

5 Improving Unsupervised MT

Since, parallel sentence mining is mostly important for downstream tasks such as low resource machine translation, we now show that mined sentences improve MT performance, which was not

shown before. In this section we mine parallel data from real life data sources and use the extracted sentences to improve the performance of unsupervised MT. For this we simulate a low-resource setup for the German-English language pair similarly to previous work on unsupervised MT (Artetxe et al., 2018c; Lample et al., 2018b).

5.1 Evaluation Setup

To mine parallel sentence pairs we use comparable monolingual data for both German and English. For this we use the news crawl data between 2007 and 2015 released by the WMT 2016 translation shared task (Bojar et al., 2016) containing about $140M$ and $114M$ German and English sentences respectively after length based filtering (see below).

As a first step, we build unsupervised BWEs on the same data as (Artetxe et al., 2018c), i.e., newscrawl between 2007 and 2013, using the same procedure mentioned earlier. The built BWEs are used to create the dictionary of word similarities for the mining and to initialize the NMT system. We consider German as the source language during the mining process. Before running our system on the full data to extract sentences we batch the data to decrease the number of sentence pair candidates. Assuming that different news portals cover a given event in the same year we only look for parallel sentences within the same year. We note that further use of batching could be possible if more fine grained date information is available. Furthermore, we also batch texts based on their length assuming that sentences with very different number of tokens are not parallel. We use sentences with length between 10 and 50 tokens and make batches with step size 5. We also apply pre-filtering within the batches. This method drastically decreased the runtime of the mining procedure which took around 1 week using 40 threads on a $2.27GHz$ CPU.

Since tuning would have been time consuming, we based our hyperparameters on the experiments in the previous section and on preliminary experiments. In order to increase the precision of mined sentences we chose an aggressive setup for window size and minimum segment length, requiring long continuous segments in the sentences. We made the following choices: threshold value for segment detection 0.3; window size of average filter 5; threshold value for deciding parallelism 0.3; minimum segment length 70%. At the end we extracted around $220K$ parallel sentence pairs from the full dataset.

5.2 Machine Translation System

As the unsupervised MT system we use the neural approach proposed by Artetxe et al. (2018c). The system is based on unsupervised BWEs as the initial bilingual signal connecting the source and target languages. The system mostly follows the standard encoder-decoder architecture using RNN layers and attention mechanism (Bahdanau et al., 2014). One difference compared to the standard architecture is its dual structure. In contrast to general NMT systems which are usually built for a specific translation direction, the system is capable of performing both source→target and target→source translation. This is achieved by having a shared encoder for both languages which encodes source and target sentences similarly. The encoders of the system are initialized with the pretrained BWEs which are kept fixed during training. On top of the shared encoders separate decoders generate the translation of the input for each language using the encoder's output.

Training is performed in an iterative manner where each iteration consists of a denoising and an on-the-fly backtranslation step. The goal of the denoising step is to learn good quality representations of both source and target sentences in the encoder and to learn how to decode these representations. Since parallel data is not available, this process is done monolingually, i.e., encoding the input and decoding to the original language, similarly to auto encoding. In order to prevent simple copying of words, a random noise is applied on the input sentences and the task is to denoise the input. To tie source and target representations more strongly backtranslation is also performed at each iteration (Sennrich et al., 2016), and synthetic parallel data is generated, by translating sentences to the other language using the system's current parameters, and then running a training step using the backtranslation as input to predict the original sentence.

To incorporate the mined parallel sentences we used them during the iterative process. At each iteration on top of the denoising and backtranslation steps we also run a training step on the mined parallel sentences in both source→target and target→source directions to train model pa-

		unsup	07-13 all		07-13 long		07-15 all		07-15 long		europarl
		-	avg	align	avg	align	avg	align	avg	align	-
WMT14	de-en	10.35	10.47	11.26	10.77	11.56	10.59	11.79	11.05	11.20	14.14
	en-de	6.30	6.23	6.91	5.14	6.82	6.55	7.26	6.16	6.78	8.96
WMT16	de-en	13.07	13.35	14.35	14.09	14.95	12.99	15.39	14.16	14.29	18.06
	en-de	8.59	8.72	9.69	7.10	10.01	8.92	10.23	8.62	9.79	12.66

Table 3: NMT experiments using mined parallel sentences. We compare results using mined sentence pairs from Hangya et al. (2018) and our approach. Texts before 2014 is used in *07-13* while all data is used in *07-15*. We also restrict the minimum sentence length to 16 tokens in case of *long*. We show a fully unsupervised system using no parallel sentences, and an oracle using europarl parallel sentences.

		all	long
avg	mined from 07-13	3,945,931	2,626,599
	mined from 07-15	10,651,736	6,858,384
align	mined from 07-13	90,707	8,358
	mined from 07-15	218,126	16,677
	europarl	218,126	—

Table 4: Number of parallel sentence pairs in the datasets.

rameters. We use words as tokens in our experiments (but we note that byte-pair encoding was slightly better in (Artetxe et al., 2018c)).

5.3 Results

We evaluate MT experiments on the WMT14 and WMT16 test sets and present BLEU scores with the neural MT system in Table 3. We compare our approach (using dynamic thresholding) with two baseline systems. We rerun[5] the setup presented in (Artetxe et al., 2018c) without any mined parallel data (unsup). In addition, we use the system of (Hangya et al., 2018) with dynamic thresholding to mine parallel sentences (avg). We ran multiple sets of experiments by splitting the mined data along two dimensions. We used sentences before 2014 only in lines *07-13* in order to use data that are from the past when evaluating on the WMT14 test set. All the data was used in *07-15*. Furthermore, looking at the mined data we noticed that shorter sentences tend to be more noisy. For this reason, we only used sentences that are at least 16 tokens long in *long*. As an oracle experiment, we used true parallel sentences from europarl by randomly sampling the same amount as the overall mined pairs to give a theoretic upper bound of the results with the used NMT system. The exact number of sentence pairs in each dataset used is shown in Table 4.

Based on the scores in Table 3 it can be seen that by using mined sentences we achieved a significant performance increase compared to the unsupervised baseline. Our system outperformed the avg baseline as well in all setups. Furthermore, our approach achieved improvements compared to the unsupervised system in all cases while the avg baseline approach achieved negative results as well. Based on Table 4 avg mines significantly more sentence pairs compared to our proposed approach, which contains noise leading to performance degradation. This result supports the claim of our work, i.e., relying only on word similarities can lead to the mining of sentence pairs which have similar meanings but are not exactly parallel.

For all test sets best results were achieved using all mined data by our system. Looking at the effect of length filtering it can be seen that this step helped when mining from *07-13* but not when using data from all years. From this we conclude, that if there are only a smaller number of parallel sentences better quality is important but quantity suppresses a small amount of noise in the *07-15* setup. Comparing scores on WMT14 with and without data from the same year and the future no clear difference can be seen. Furthermore, the BLEU score differences between the time intervals on WMT14 strongly follows that on WMT16 where all of the sentences are from the past. From this we conclude that the unsupervised MT system generalizes well using older data.

Using true parallel data from europarl achieved even higher results. The reason for this is that the majority of the mined sentences are short and more noisy. Based on this, one possible future improvement could be to use more aggressive parameters when mining from short sentences while using more permissive parameters to mine longer sentences.

[5]Original results were shown only on WMT14 which are comparable to our BLEU scores.

source	Wenn Justin Bieber einen Kaffee trinkt, staunt man an der Fensterscheibe.
reference	When Justin Bieber drinks coffee people goggle through the window.
unsup	If Justin Timberlake ate a coffee, you buzzing to the window.
07-15 all	If Justin Bieber drank a coffee, you wonder at the window.
source	Etwa die Hälfte der demokratischen Wähler der Vorwahlen landesweit sagen, dass sie mit Begeisterung Clinton unterstützen würden, wenn sie von der Partei nominiert würde.
reference	About half of Democratic primary voters nationwide say they would enthusiastically support Clinton if she became the party's nominee.
unsup	Roughly half of the pro-election voters nationwide voters say they would support Obama's support with Clinton if they would be nominated by the party.
07-15 all	About half of the Democratic primary voters nationwide say that they would support Clinton with enthusiasm if they would be nominated by the party.
source	und sagte, er habe auf jemand geschossen und jemand getötet
reference	and said he had shot and killed someone
unsup	and he said he had been shot on someone and killed
07-15 all	and he said he had shot and killed someone

Table 5: Example translations comparing the unsupervised baseline with adding mined parallel sentences on WMT16.

We manually analyzed the translations given by the unsupervised baseline system and the setup when we used all the sentence pairs mined by our approach on WMT16. We show examples depicting differences in Table 5. One aspect where the added parallel sentences clearly helped is the handling of named entities. As the first and second examples show, the baseline system often mixes up names which is due to their similar representations in BWE space. By adding parallel data the system could learn to match the source and target side representations of a given entity, i.e., copy the correct word to the translation. We also found that the fluency of translations is also improved which is demonstrated by the second and third examples. The second example shows an important weakness of the baseline, which is that it tends to be redundant, e.g., by mentioning *voters* and *support* twice. In addition, it mentions US presidency related entities twice, once as *Clinton* and once confusing it with *Obama*. On the other hand, by using parallel sentences the results are more fluent and accurate. While the meaning of the third example was correctly translated, the wording used by the baseline is unnatural in contrast to the *07-15 all* setup.

6 Conclusions

Parallel sentence extraction is important for providing an additional bilingual signal for many downstream tasks in low resource setups. Most previous work tackled this problem using supervised techniques which made their applicability problematic. In this work, we proposed a fully unsupervised system for parallel sentence extraction. We showed that a previous unsupervised system, which only relies on word similarity in source and target language sentences, often mines false positives because not all sentences having similar words are parallel. To overcome this problem we introduced the detection of continuous parallel segments based on word alignments. We filter candidates having too short segments and weight the similarity score of the rest based on segment lengths. We showed that using our method better performance could be achieved on the BUCC 2017 parallel sentence extraction task compared to previous work. In contrast to previous unsupervised work, we also extracted sentences from real world comparable corpora and showed better translation performance when using these sentence pairs, opening up new possibilities for using small amounts of parallel data in purely unsupervised MT approaches. Our analysis showed that both handling of named entities and the fluency of sentences improved. We publicly release our system[6] to support MT communities especially for low-resource setups.

Acknowledgments

We would like to thank the anonymous reviewers for their valuable input. This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement № 640550).

[6] https://github.com/hangyav/UnsupPSE

References

David Alvarez-Melis and Tommi Jaakkola. 2018. Gromov-Wasserstein Alignment of Word Embedding Spaces. In *Proc. EMNLP*.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2018a. A robust self-learning method for fully unsupervised cross-lingual mappings of word embeddings. In *Proc. ACL*.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2018b. Unsupervised Statistical Machine Translation. In *Proc. EMNLP*.

Mikel Artetxe, Gorka Labaka, Eneko Agirre, and Kyunghyun Cho. 2018c. Unsupervised Neural Machine Translation. In *Proc. ICLR*.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. In *Proc. ICLR*.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5.

Ondrej Bojar, Christian Buck, Christian Federmann, Barry Haddow, Philipp Koehn, Johannes Leveling, Christof Monz, Pavel Pecina, Matt Post, Herve Saint-Amand, Radu Soricut, Lucia Specia, and Aleš Tamchyna. 2014. Findings of the 2014 Workshop on Statistical Machine Translation. In *Proc. 9th Workshop on Statistical Machine Translation*.

Ondrej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, Varvara Logacheva, Christof Monz, et al. 2016. Findings of the 2016 conference on machine translation. In *Proc. Conference on Machine Translation*.

Houda Bouamor and Hassan Sajjad. 2018. H2@ BUCC18: Parallel Sentence Extraction from Comparable Corpora Using Multilingual Sentence Embeddings. In *Proc. BUCC*.

Fabienne Braune, Viktor Hangya, Tobias Eder, and Alexander Fraser. 2018. Evaluating bilingual word embeddings on the long tail. In *Proc. NAACL-HLT*.

Alexis Conneau, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2018. Word Translation Without Parallel Data. In *Proc. ICLR*.

Georgiana Dinu, Angeliki Lazaridou, and Marco Baroni. 2015. Improving Zero-Shot Learning by Mitigating the Hubness Problem. In *Proc. workshop track at ICLR*.

Long Duong, Hiroshi Kanayama, Tengfei Ma, Steven Bird, and Trevor Cohn. 2016. Learning crosslingual word embeddings without bilingual corpora. In *Proc. EMNLP*.

Manaal Faruqui and Chris Dyer. 2014. Improving vector space word representations using multilingual correlation. In *Proc. EACL*.

Stephan Gouws, Yoshua Bengio, and Greg Corrado. 2015. Bilbowa: Fast bilingual distributed representations without word alignments. In *Proc. ICML*.

Francis Grégoire and Philippe Langlais. 2017. BUCC 2017 Shared Task: a First Attempt Toward a Deep Learning Framework for Identifying Parallel Sentences in Comparable Corpora. In *Proc. BUCC*.

Jeenu Grover and Pabitra Mitra. 2017. Bilingual Word Embeddings with Bucketed CNN for Parallel Sentence Extraction. In *Proc. ACL, Student Research Workshop*.

Viktor Hangya, Fabienne Braune, Yuliya Kalasouskaya, and Alexander Fraser. 2018. Unsupervised Parallel Sentence Extraction from Comparable Corpora. In *Proc. IWSLT*.

Hany Hassan, Anthony Aue, Chang Chen, Vishal Chowdhary, Jonathan Clark, Christian Federmann, Xuedong Huang, Marcin Junczys-dowmunt, William Lewis, Mu Li, Shujie Liu, Tie-yan Liu, Renqian Luo, Arul Menezes, Tao Qin, and Microsoft Ai. 2018. Achieving Human Parity on Automatic Chinese to English News Translation. *arXiv:1803.05567*.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2017. Billion-scale similarity search with GPUs. *CoRR*, abs/1702.08734.

Harold W Kuhn. 1955. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2).

Matt Kusner, Yu Sun, Nicholas Kolkin, and Kilian Weinberger. 2015. From Word Embeddings to Document Distances. In *Proc. ICML*.

Guillaume Lample, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018a. Unsupervised Machine Translation Using Monolingual Corpora Only. In *Proc. ICLR*.

Guillaume Lample, Myle Ott, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018b. Phrase-Based & Neural Unsupervised Machine Translation. In *Proc. EMNLP*.

Tomas Mikolov, Quoc V. Le, and Ilya Sutskever. 2013. Exploiting similarities among languages for machine translation. *CoRR*, abs/1309.4168.

Dragos Stefan Munteanu, Alexander Fraser, and Daniel Marcu. 2004. Improved machine translation performance via parallel sentence extraction from comparable corpora. In *Proc. NAACL-HLT*.

Dragos Stefan Munteanu and Daniel Marcu. 2006. Extracting parallel sub-sentential fragments from non-parallel corpora. In *Proc. ACL*.

Parker Riley and Daniel Gildea. 2018. Orthographic Features for Bilingual Lexicon Induction. In *Proc. ACL*.

Holger Schwenk. 2018. Filtering and Mining Parallel Data in a Joint Multilingual Space. In *Proc. ACL*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Improving neural machine translation models with monolingual data. In *Proc. ACL*.

Jason R. Smith, Chris Quirk, and Kristina Toutanova. 2010. Extracting Parallel Sentences from Comparable Corpora using Document Level Alignment. In *Proc. NAACL-HLT*.

Dániel Varga, Péter Halácsy, András Kornai, Viktor Nagy, László Németh, and Viktor Trón. 2007. Parallel corpora for medium density languages. *Amsterdam Studies In The Theory And History Of Linguistic Science Series 4*.

Ivan Vulić and Marie-Francine Moens. 2015. Bilingual word embeddings from non-parallel document-aligned data applied to bilingual lexicon induction. In *Proc. ACL*.

Chao Xing, Dong Wang, Chao Liu, and Yiye Lin. 2015. Normalized Word Embedding and Orthogonal Transform for Bilingual Word Translation. In *Proc. NAACL-HLT*.

Hainan Xu and Philipp Koehn. 2017. Zipporah: a fast and scalable data cleaning system for noisy web-crawled parallel corpora. In *Proc. EMNLP*.

Zhen Yang, Wei Chen, Feng Wang, and Bo Xu. 2018. Unsupervised Neural Machine Translation with Weight Sharing. In *Proc. ACL*.

Pierre Zweigenbaum, Serge Sharoff, and Reinhard Rapp. 2017. Overview of the Second BUCC Shared Task: Spotting Parallel Sentences in Comparable Corpora. In *Proc. BUCC*.