Fixing Translation Divergences in Parallel Corpora for Neural MT

MinhQuang Pham[†‡], **Josep Crego**[†], **Jean Senellart**[†] , **François Yvon**[‡]
[†]SYSTRAN / 5 rue Feydeau, 75002 Paris, France
firstname.lastname@systrangroup.com
[‡]LIMSI, CNRS, Université Paris-Saclay 91405 Orsay, France
firstname.lastname@limsi.fr

Abstract

Corpus-based approaches to machine translation rely on the availability of clean parallel corpora. Such resources are scarce, and because of the automatic processes involved in their preparation, they are often noisy. This paper describes an unsupervised method for detecting translation divergences in parallel sentences. We rely on a neural network that computes cross-lingual sentence similarity scores, which are then used to effectively filter out divergent translations. Furthermore, similarity scores predicted by the network are used to identify and fix some partial divergences, yielding additional parallel segments. We evaluate these methods for English-French and English-German machine translation tasks, and show that using filtered/corrected corpora actually improves MT performance.

1 Introduction

Parallel sentence pairs are the only necessary resource to build Machine Translation (MT) systems. In the case of neural MT, a large neural network is trained through maximising a proxy of translation performance on a parallel corpus. Therefore, the quality of MT engines is heavily dependent on the amount but also the quality of available parallel sentences.[1]

Parallel texts are unfortunately, scarce resources: There are relatively few language pairs for which parallel corpora of large sizes exist, and even for those pairs, available corpora only concern few restricted domains. To alleviate the lack of parallel data, several approaches have been developed over the years. They range from methods using non-parallel, or comparable data (Zhao and Vogel, 2002; Fung and Cheung, 2004; Munteanu and Marcu, 2005; Grégoire and Langlais, 2018; Grover and Mitra, 2017; Schwenk, 2018) to techniques that produce synthetic parallel data from monolingual corpora (Sennrich et al., 2016a; Chinea-Rios et al., 2017), using automated alignment/translation engines that are prone to the introduction of noise in the resulting parallel sentences. Mismatches in parallel sentences extracted from translated texts are also reported (Tiedemann, 2011; Xu and Yvon, 2016). This problem is mostly ignored in MT, where parallel sentences are considered to convey the exact same meaning; yet it seems particularly important for neural MT engines (Chen et al., 2016).

en	*What do you feel*, **Spock***?*
fr	*Que ressentez-vous?*
gl	*What do you feel?*
en	*How much do you get paid?*
fr	*T'es payé combien* **de l'heure***?*
gl	*How much do you get paid per hour?*
en	**That seems a lot.**
fr	**40 livres?**
gl	*40 pounds?*
en	*I brought you* **french fries***!*
fr	*Je t'ai rapporté des* **saucisses***!*
gl	*I brought you sausage!*

Table 1: Examples of semantically divergent parallel sentences. English (en), French (fr) and gloss of French (gl). Divergences are in bold letters.

Table 1 gives some examples of English-French parallel sentences that are not completely semantically equivalent, extracted from the OpenSubtitles corpus (Lison and Tiedemann, 2016).

Multiples types of translation divergences are found in parallel corpora: Additional segments are included on either side of the parallel sentences (first and second rows) most likely due to errors in sentence segmentation; Some translations may be completely uncorrelated (third row); Inaccurate translations also exist (fourth row). Note that divergent translations can be due various reasons (Li et al., 2014), the study of which is beyond the

[1] Recent work on neural MT (Lample et al., 2018; Artetxe et al., 2018) completely dispenses with parallel data, using unsupervised methods to obtain performance improvements over word-by-word statistical MT. These systems however lag far behind supervised systems, as considered in this work.

Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 2967–2973
Brussels, Belgium, October 31 - November 4, 2018. ©2018 Association for Computational Linguistics

scope of this paper.

In this work, we present an unsupervised method for building cross-lingual sentence embeddings based on modelling word similarity, relying on a neural architecture (see § 3) that is able to identify several types of common cross-lingual divergences. The resulting embeddings are then used to measure semantic equivalence between sentences. To evaluate our method, we show in § 4 that translation accuracy can be improved after filtering out divergent sentence pairs in an English-to-French and an English-to-German translation tasks. We also show that in some cases, divergent sentences can be fixed by removing divergent segments, further increasing translation quality. All the code used in this paper is freely available.[2]

2 Related Work

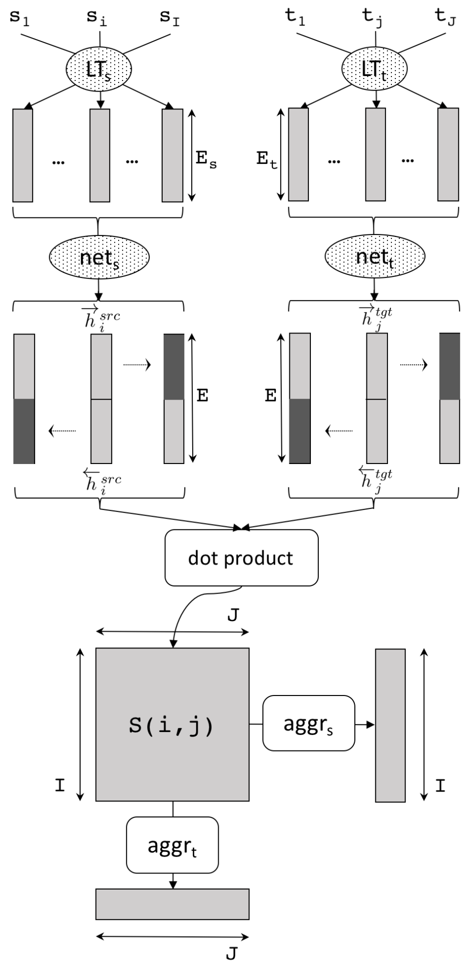
Attempts to measure the impact of translation divergences in MT have focused on the introduction of noise in sentence alignments (Goutte et al., 2012), showing that statistical MT is highly robust to noise, and that performance only degrades seriously at very high noise levels. In contrast, neural MTs seem to be more sensitive to noise (Chen et al., 2016), as they tend to assign high probabilities to rare events (Hassan et al., 2018).

Efforts devoted to characterising the degree of semantic equivalence between two snippets of texts in the same or different languages are presented (Agirre et al., 2016). In (Mueller and Thyagarajan, 2016), a monolingual sentence similarity network is proposed, making use of a simple LSTM layer to compute sentence representations. The authors show that a simple SVM classifier exploiting such sentence representations achieves state-of-the-art results in a textual entailment task. With the same objective, the system of He and Lin (2016) uses multiple convolutional layers and models pairwise word interactions.

Our work is inspired by Carpuat et al. (2017), who train a SVM-based cross-lingual divergence detector using word alignments and sentence length features. Their work shows that an NMT system trained only on non-divergent sentences yields slightly better translation scores, while requiring less training time. A follow-up study by the same authors (Vyas et al., 2018) achieves even better results, using the neural architecture of He and Lin (2016). Our work differs from theirs as we make use of a network with a different, arguably simpler, topology. We model sentence similarity by means of optimising a loss function based on word alignments. Furthermore, the network predicts word similarity scores that we further use to correct divergent sentences.

3 Neural Divergence Classifier

The architecture of our network is inspired by the work on word alignment of Legrand et al. (2016), using however contextual, rather than fixed, word embeddings (see Figure 1).



s_1
s_i
s_I
t_1
t_j
t_J
LT_s
LT_t
...
E_s
E_t
net_s
net_t
$\overrightarrow{h}_i^{src}$
$\overrightarrow{h}_j^{tgt}$
E
E
$\overleftarrow{h}_i^{src}$
$\overleftarrow{h}_j^{tgt}$
dot product
J
S(i,j)
$aggr_s$
I
I
$aggr_t$
J

Figure 1: Illustration of the model.

It computes the similarity of any source-target sentence pair (s, t), where $s = (s_1, ..., s_I)$ and $t = (t_1, ..., t_J)$. The model is composed of 2 bi-directional LSTM subnetworks, net_s and net_t, which respectively encode source and target sentences. Since both net_s and net_t take the same form, we only describe the former network: it outputs forward and backward hidden states, $\overrightarrow{h}_i^{src}$ and $\overleftarrow{h}_i^{src}$, which are then concate-

[2] https://github.com/jmcrego/similarity

nated into a vector encoding the i^{th} source word as $h_i^{src} = [\overrightarrow{h}_i^{src}; \overleftarrow{h}_i^{src}]$. In addition, the last forward/backward hidden states (in dark grey on Figure 1) are also concatenated to represent whole sentences $h_{src} = [\overrightarrow{h}_I^{src}; \overleftarrow{h}_1^{src}]$. The similarity between sentence pairs can then be obtained using eg. the cosine similarity:

$$sim(h_{src}, h_{tgt}) = \frac{h_{src} \cdot h_{tgt}}{||h_{src}|| * ||h_{tgt}||} \tag{1}$$

Our model is trained to maximize word alignment scores between words in both sentences, using aggregation functions that summarise the alignment scores for each source/target word. Similar to (Legrand et al., 2016), alignment scores $S(i,j)$ are given by the dot-product $S(i,j) = h_i^{src} \cdot h_j^{tgt}$, further aggregated as follows:

$$\begin{aligned} aggr_s(i,S) &= \frac{1}{r}\, log\left(\sum_{j=1}^{J} e^{r*S(i,j)}\right) \\ aggr_t(j,S) &= \frac{1}{r}\, log\left(\sum_{i=1}^{I} e^{r*S(i,j)}\right) \end{aligned} \tag{2}$$

The training loss function is then defined as:

$$\begin{aligned} &\mathcal{L}(src, tgt) = \\ &\sum_{i=1}^{I} log(1 + e^{aggr_s(i,S)*sign_i}) \; + \\ &+ \sum_{j=1}^{J} log(1 + e^{aggr_t(j,S)*sign_j}) \end{aligned} \tag{3}$$

3.1 Training with Negative Examples

Training is performed by minimizing Eq. (3), for which annotated examples of source (sign_i) and target (sign_j) words are needed. As positive examples, we use paired sentences of a parallel corpus; all words in such sentences are labelled as parallel ($\forall i, j, \text{sign}_i = \text{sign}_j = -1$). We consider three types of negative instances: the basic case uses random unpaired sentences; in this case, all words are labelled as divergent ($\forall i, j, \text{sign}_i = \text{sign}_j = +1.$). Since negative pairs may be very easy to classify and we want our network to detect less obvious divergences, we further create more difficult negative examples as follows.

We first replace random sequences of words in source or target by a sequence of words with the same part-of-speeches.[3] Words that are not replaced are deemed parallel ($\text{sign}_i = -1$) while those replaced are annotated as $\text{sign}_i = +1$. Words aligned to some replaced words are also assigned the divergent label ($\text{sign}_i = +1$). For instance, given the original sentence pair:

src:	`What do you feel ?`
tgt:	`Que ressentez-vous ?` ,

we may replace `'you feel'`, with part-of-speech tags `'PRP VB'`, by another sequence with same tags (i.e. `'we want'`), yielding a new negative instance (divergent words are in bold):

src:	`What`	`do`	**we**	**want**	`?`
$\mathcal{Y}^{src}$:	`-1`	`-1`	**+1**	**+1**	`-1`
tgt:	`Que`	**ressentez-vous**			`?`
$\mathcal{Y}^{tgt}$:	`-1`	**+1**			`-1`

Note that we need word alignments to identify as divergent the sequence `'ressentez-vous'`, which was aligned to `'you feel'` in the original sentence. Finally, motivated by sentence segmentation errors observed in many corpora, we also build negative examples by inserting a sentence at the beginning (or end) of the source (or target) sentence. Words in the original sentence pair are annotated $\text{sign}_i = -1$, while the new words inserted are considered divergent ($\text{sign}_i = +1$). Given the same sentence pair as above, a negative example is created by inserting the sentence `'Not .'` at the end of the original source:

src:	`What`	`do`	`you`	`want`	`?`	**Not**	**.**
$\mathcal{Y}^{src}$:	`-1`	`-1`	`-1`	`-1`	`-1`	**+1**	**+1**
tgt:	`Que`	`ressentez-vous`			`?`		
$\mathcal{Y}^{tgt}$:	`-1`	`-1`			`-1`		

To finally avoid the generation of easy negative sentence pairs having a large difference in sentence length, we restrict negative examples to have a length ratio < 2.0 (3.0 for shortest sentences).

3.2 Divergence Correction

Our training corpora contains many divergent sentences that follow a common pattern, consisting of adding some extra leading/trailing words. Accordingly, we implemented a simple algorithm that discards sequences of leading/trailing words on both

[3] The rationale is to try to keep the generated sentences as grammatical as possible; Otherwise, the network could learn to flag non-grammatical sentences as non-parallel.

sides. To find optimal source (u, v) and target (x, y) indices that enclose parallel segments within the original sentence, we compute:

$$\arg\max_{u,v,x,y}\Big\{\sum_{u\leq I\leq v}\max_{x\leq j\leq y}\{S(i,j)\}\Big\}$$

The $\mathcal{N}$-best sequences (s_u^v, t_x^y) are considered as likely corrections, in which we use the one having the highest similarity score to replace the original (s_1^I, t_1^J). Note that short sentences are not considered and we enforce $v - u > \tau$ and $y - x > \tau$. Figure 2 (left) displays an example of an alignment matrix $S(i, j)$. An acceptable correction is: *Que ressentez-vous ? ⇔ What do you feel ?*. corresponding to $u = 1$, $v = 5$, $x = 1$ and $y = 3$.

4 Experiments

4.1 Corpora

We filter out divergences from the English-French OpenSubtitles corpus (Lison and Tiedemann, 2016), which consists of a collection of movie and TV subtitles. We also use the very noisy English-German Paracrawl[4] corpus. Both corpora present many potential divergences. To evaluate English-French translation performance, we use the En-Fr Microsoft Spoken Language Translation corpus, created from actual Skype conversations (Federmann and Lewis, 2016). English-German performance is evaluated on the publicly available Newstest-2017 (Bojar et al., 2017), corresponding to news stories selected from online sources.

In order to better assess the quality of our classifier when facing different word divergences, we also collected from the original OpenSubtitles corpus 500 sentences containing different types of examples: 200 paired sentences; 100 unpaired sentences; 100 sentences with replace examples; and 100 sentences with insert examples (see § 3.1). All data is preprocessed with `OpenNMT`[5], performing minimal tokenisation. After tokenisation, each out-of-vocabulary word is mapped to a special UNK token, assuming a vocabulary containing the $50,000$ more frequent words.

4.2 Neural Divergence

Word embeddings of $E_s = E_t = 256$ cells are initialised using `fastText`,[6] further aligned by means of `MUSE`[7] following the unsupervised method of Lample et al. (2018). Both bi-LSTMs use 256-dimensional hidden representations ($E = 512$). Network optimization is done using SGD with gradient clipping (Pascanu et al., 2013). For each epoch, we randomly select 1 million sentence pairs that we place in batches of 32 examples. We run 10 epochs and start decaying at each epoch by 0.8 when the loss on validation set increases. Divergence is computed as in equation (1) and setting $r = 1.0$; For divergence correction, we use $\mathcal{N} = 20$ and $\tau = 3$. The same number of examples are always generated for each type of example (`P`aired, `U`npaired, `R`eplace and `I`nsert). Alignments needed for `R`eplace and `I`nsert methods are performed using `fast_align`[8].

4.3 Neural Translation

In addition to the basic tokenisation detailed above, we perform Byte-Pair Encoding (Sennrich et al., 2016b) with 30000 merge operations learned by joining both language sides. Neural systems are based on the open-source project `OpenNMT`; using a Transformer model similar to the model of Vaswani et al. (2017): both encoder and decoder have 6 layers; Multi-head attention is performed over 8 head; the hidden layer size is 512; and the inner layer of feed forward network is of size 2048. Word embeddings have 512 cells. We set the dropout probability to 0.1 and the batch size to 3072. The optimiser is Lazy Adam with $\beta_1 = 0.9$, $\beta_2 = 0.98$, $\epsilon = 10^{-9}$, $warmup\_steps = 4000$. Training stops after 30 epochs.

5 Results

We first evaluate the ability of our divergence classifier to predict different types of divergences at the level of words. We use the test set manually annotated for that purpose and train our model on the OpenSubtitles corpus. A word is considered divergent when associated to a negative aggregation score (see Equation (2)). Accuracies obtained for various combinations of negative examples, where we see that non-divergent words in parallel and unpaired sentences (columns `P` and `U`) are easy to spot, as long as the model has seen these types of examples in training. However, the accuracy drops dramatically when the model is not trained with unpaired sentences (rows `PR`, `PI` and `PRI`). Regarding columns `R` and `I`, accuracies are lower

[4] http://paracrawl.eu/

[5] http://opennmt.net

[6] https://github.com/facebookresearch/fastText

[7] https://github.com/facebookresearch/MUSE

[8] https://github.com/clab/fast_align

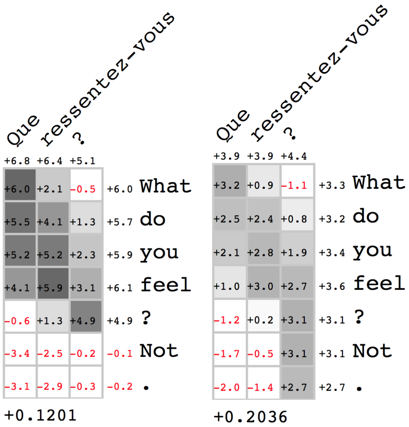
since these sentences contain a mix of divergent and non-divergent words.

Accuracy		Test examples				
		P	U	R	I	PURI
Train examples	PU	**0.996**	**0.994**	0.671	0.673	0.874
	PR	**0.995**	0.033	**0.951**	0.689	0.746
	PI	**0.998**	0.071	0.697	**0.725**	0.705
	PUR	**0.994**	**0.989**	**0.919**	0.710	0.932
	PUI	**0.995**	**0.996**	0.662	**0.769**	0.887
	PRI	**0.991**	0.161	**0.924**	**0.719**	0.768
	PURI	**0.995**	**0.980**	**0.916**	**0.788**	**0.942**

Table 2: Word divergence accuracies according to different type of examples used in train/test.

Models that were trained with the matching examples (R and I) obtain the highest accuracies (in bold letters). Column PURI gives results for the complete test set, mixing all type of examples. As expected, the best accuracy is also obtained when training on all types of examples.

Figure 2 illustrates the output of our network when trained using PU examples (right) and PURI examples (left). The former (right) fails to predict some divergences, most likely because its training set does not contain sentences mixing divergent and non-divergent words. Furthermore, the network trained with PURI examples correctly assigns a lower similarity score to this pair, as both sentences do not convey the exact same meaning.



	Que	ressentez-vous	?		
	+6.8	+6.4	+5.1		
	+6.0	+2.1	-0.5	+6.0	What
	+5.5	+4.1	+1.3	+5.7	do
	+5.2	+5.2	+2.3	+5.9	you
	+4.1	+5.9	+3.1	+6.1	feel
	-0.6	+1.3	+4.9	+4.9	?
	-3.4	-2.5	-0.2	-0.1	Not
	-3.1	-2.9	-0.3	-0.2	.
+0.1201					

	Que	ressentez-vous	?		
	+3.9	+3.9	+4.4		
	+3.2	+0.9	-1.1	+3.3	What
	+2.5	+2.4	+0.8	+3.2	do
	+2.1	+2.8	+1.9	+3.4	you
	+1.0	+3.0	+2.7	+3.6	feel
	-1.2	+0.2	+3.1	+3.1	?
	-1.7	-0.5	+3.1	+3.1	Not
	-2.0	-1.4	+2.7	+2.7	.
+0.2036					

Figure 2: Sentence pair with similarity scores produced by our model when trained with PU examples (right) and over PURI examples (left). Aggregation scores (Eq. (2)) are shown next to words. Matrices contain alignment scores. Sentence similarities (Eq. (1)) are below matrices.

Finally, BLEU scores obtained with varying training data configurations are in Table 3: The entire[9] data sets (all); The most similar pairs after optimizing Eq. (3) (sim); After applying the correction algorithm of § 3.2 (sim+fix). Columns Ref and Fix indicate the number of original and corrected sentences (in millions) used in training.

Data	Ref (M)	Fix (M)	Test (BLEU)
OpenSubtitles English-French			
all	27.2	-	42.18
sim	15.5	-	43.12 (+0.94)
sim	18.0	-	43.19 (+1.01)
sim+fix	15.5	2.5	**44.19** (+2.01)
Paracrawl English-German			
all	22.2	-	19.27
sim	15.0	-	21.52 (+2.25)
sim	17.5	-	21.97 (+2.70)
sim+fix	15.0	2.5	**22.42** (+3.15)

Table 3: BLEU scores obtained by neural MT using different subsets of the OpenSubtitles and Paracrawl corpora.

Results obtained after filtering sentence pairs (sim) clearly outperform the baseline (all) by +0.94 and +2.25 BLEU respectively. Regarding OpenSubtitles, when fixing 2.5M sentences (4^{th} row) the accuracy is further boosted to +2.01, whereas the same sentence pairs do not show any improvement when added in their original form (3^{rd} row). Similar results are obtained for the Paracrawl corpus. Results after fixing 2.5M sentences ($4^{th}row$) outperform those obtained with their original form ($3^{rd}row$).

6 Conclusions and outlook

We presented an unsupervised method based on deep neural networks for detecting translation divergences in parallel corpora. Our model optimizes word alignments, and computes a fine grained divergence prediction at the level of words. Misaligned/divergent words can then be filtered out, yielding larger and better training sets. Our experiments on two machine translation tasks show significant improvements in comparison to training with the entire data set.

We plan to use our model to predict sentence embeddings over monolingual corpora, allowing to collect parallel pairs through vector similarity measures. In addition, we would like to measure the performance of our model after applying subword tokenisation, as well as using multiple LSTM layers, a technique well known to capture hierarchical structure in the context of MT.

Acknowledgements

We are grateful to J. Legrand for his fruitful comments regarding the neural divergence classifier.

[9] Paracrawl contains more than 100M sentences. We reduced its size to 22.2M using standard filtering techniques.

References

Eneko Agirre, Carmen Banea, Daniel M. Cer, Mona T. Diab, Aitor Gonzalez-Agirre, Rada Mihalcea, German Rigau, and Janyce Wiebe. 2016. Semeval-2016 task 1: Semantic textual similarity, monolingual and cross-lingual evaluation. In *SemEval@NAACL-HLT*, pages 497–511. The Association for Computational Linguistics.

Mikel Artetxe, Gorka Labaka, Eneko Agirre, and Kyunghyun Cho. 2018. Unsupervised neural machine translation. In *Proceedings of the Sixth International Conference on Learning Representations*, Vancouver, Canada.

Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Shujian Huang, Matthias Huck, Philipp Koehn, Qun Liu, Varvara Logacheva, Christof Monz, Matteo Negri, Matt Post, Raphael Rubino, Lucia Specia, and Marco Turchi. 2017. Findings of the 2017 Conference on Machine Translation (WMT17). In *Proceedings of the Second Conference on Machine Translation*, pages 169–214, Copenhagen, Denmark. Association for Computational Linguistics.

Marine Carpuat, Yogarshi Vyas, and Xing Niu. 2017. Detecting cross-lingual semantic divergence for neural machine translation. In *Proceedings of the First Workshop on Neural Machine Translation*, pages 69–79, Vancouver. Association for Computational Linguistics.

Boxing Chen, Roland Kuhn, George Foster, Colin Cherry, and Fei Huang. 2016. Bilingual methods for adaptive training data selection for machine translation. In *Proceedings of the 12th biennial conference of the Association for the Machine Translation in Americas (AMTA2016)*, Austin, TX.

Mara Chinea-Rios, Álvaro Peris, and Francisco Casacuberta. 2017. Adapting neural machine translation with parallel synthetic data. In *Proceedings of the Second Conference on Machine Translation*, pages 138–147, Copenhagen, Denmark. Association for Computational Linguistics.

Christian Federmann and Will Lewis. 2016. Microsoft speech language translation (MSLT) corpus: The IWSLT 2016 release for English, French and German. In *Proceedings of the International Workshop on Spoken Language Translation (IWSLT2016)*, Seattle, WA.

Pascale Fung and Percy Cheung. 2004. Mining very-non-parallel corpora: Parallel sentence and lexicon extraction via bootstrapping and EM. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, EMNLP.

Cyril Goutte, marine carpuat, and Georges Foster. 2012. The impact of sentence alignment errors on phrase-based machine translation performance. In *The Tenth Biennial Conference of the Association for Machine Translation in the Americas*, San Diego, California.

Francis Grégoire and Philippe Langlais. 2018. Extracting parallel sentences with bidirectional recurrent neural networks to improve machine translation. In *Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018, Santa Fe, New Mexico, USA, August 20-26, 2018*, pages 1442–1453.

Jeenu Grover and Pabitra Mitra. 2017. Bilingual word embeddings with bucketed CNN for parallel sentence extraction. In *Proceedings of ACL 2017, Student Research Workshop*, pages 11–16, Vancouver, Canada. Association for Computational Linguistics.

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, Frank Seide, Xu Tan, Fei Tian, Lijun Wu, Shuangzhi Wu, Yingce Xia, Dongdong Zhang, Zhirui Zhang, and Ming Zhou. 2018. Achieving human parity on automatic Chinese to English news translation. *CoRR*, abs/1803.05567.

Hua He and Jimmy Lin. 2016. Pairwise word interaction modeling with deep neural networks for semantic similarity measurement. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 937–948, San Diego, California. Association for Computational Linguistics.

Guillaume Lample, Alexis Conneau, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2018. Word translation without parallel data. In *International Conference on Learning Representations*, Long Beach, CA.

Joël Legrand, Michael Auli, and Ronan Collobert. 2016. Neural network-based word alignment through score aggregation. In *Proceedings of the First Conference on Machine Translation: Volume 1, Research Papers*, pages 66–73, Berlin, Germany. Association for Computational Linguistics.

Junyi Jessy Li, Marine Carpuat, and Ani Nenkova. 2014. Cross-lingual discourse relation analysis: A corpus study and a semi-supervised classification system. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 577–587, Dublin, Ireland. Dublin City University and Association for Computational Linguistics.

Pierre Lison and Jörg Tiedemann. 2016. Opensubtitles2016: Extracting large parallel corpora from movie and tv subtitles. In *10th International Conference on Language Resources and Evaluation (LREC 2016)*, Portoroz, Slovenia. European Language Resources Association (ELRA).

Jonas Mueller and Aditya Thyagarajan. 2016. Siamese recurrent architectures for learning sentence similarity. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, AAAI'16, pages 2786–2792, Phoenix, Arizona. AAAI Press.

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

Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. 2013. On the difficulty of training recurrent neural networks. In *Proceedings of the 30th International Conference on International Conference on Machine Learning - Volume 28*, ICML'13, pages III–1310–III–1318, Atlanta, GA, USA. JMLR.org.

Holger Schwenk. 2018. Filtering and mining parallel data in a joint multilingual space. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 228–234, Melbourne, Australia. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016a. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96, Berlin, Germany. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016b. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Jörg Tiedemann. 2011. *Bitext Alignment*. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 6000–6010, Long Beach, CA.

Yogarshi Vyas, Xing Niu, and Marine Carpuat. 2018. Identifying semantic divergences in parallel text without annotations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1503–1515, New Orleans, Louisiana. Association for Computational Linguistics.

Yong Xu and François Yvon. 2016. Novel elicitation and annotation schemes for sentential and subsentential alignments of bitexts. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Portoroz, Slovenia. European Language Resources Association (ELRA).

Bing Zhao and Stephan Vogel. 2002. Adaptive parallel sentences mining from web bilingual news collection. In *Proceedings of the 2002 IEEE International Conference on Data Mining*, ICDM '02, Washington, DC, USA. IEEE Computer Society.