Filtering of Noisy Parallel Corpora Based on Hypothesis Generation

Zuzanna Parcheta[1] **Germán Sanchis-Trilles**[1] **Francisco Casacuberta**[2]
[1]Sciling S.L.,Carrer del Riu 321, Pinedo, 46012, Spain
{zparcheta, gsanchis}@sciling.com
[2]PRHLT Research Center, Camino de Vera s/n, 46022 Valencia, Spain
fcn@prhlt.upv.es

Abstract

The filtering task of noisy parallel corpora in WMT2019 aims to challenge participants to create filtering methods to be useful for training machine translation systems. In this work, we introduce a noisy parallel corpora filtering system based on generating hypotheses by means of a translation model. We train translation models in both language pairs: Nepali–English and Sinhala–English using provided parallel corpora. To create the best possible translation model, we first join all provided parallel corpora (Nepali, Sinhala and Hindi to English) and after that, we applied bilingual cross-entropy selection for both language pairs (Nepali–English and Sinhala–English). Once the translation models are trained, we translate the noisy corpora and generate a hypothesis for each sentence pair. We compute the smoothed BLEU score between the target sentence and generated hypothesis. In addition, we apply several rules to discard very noisy or inadequate sentences which can lower the translation score. These heuristics are based on sentence length, source and target similarity and source language detection. We compare our results with the baseline published on the shared task website, which uses the Zipporah model, over which we achieve significant improvements in one of the conditions in the shared task. The designed filtering system is domain independent and all experiments are conducted using neural machine translation.

1 Introduction

A large amount of parallel corpora can be extracted using web-crawling. This technique of data acquisition is very useful to increase the training set for low-resourced languages. Unfortunately, the extracted data can include noisy sentence pairs, such as unaligned sentences, partially translated pairs, or sentences containing different languages than those intended. For these reasons the creation of systems for filtering of noisy parallel corpora are needed.

In this paper, we introduce a filtering method for noisy parallel corpora based mainly on generating hypotheses for each sentence pair from noisy data and scoring based on hypothesis and target sentence similarity. This technique consists of building the best possible translation engine for each language pair and generating a translation hypothesis for each sentence of the noisy data. Once the hypotheses are generated, we compute the BLEU (Papineni et al., 2002), smoothed by adding one to both numerator and denominator from (Lin and Och, 2004), between each target and hypothesis. To create a translation engine, which will be used for generating hypothesis for each sentence from noisy corpus, we select sentence pairs using bilingual cross-entropy selection (Axelrod et al., 2011) from all parallel corpora provided (Nepali, Sinhala, Hindi to English) jointly. To apply bilingual cross-entropy, we first train language models using the provided monolingual corpora in Nepali, Sinhala and English. In addition, we use some rules to discard useless sentences by filtering according to sentence length, Nepali and Sinhala characters detection, and BLEU scoring between source and target sentences. The last rule is used to discard highly similar sentence pairs.

The paper is structured as follows: Section 2 describes the shared task, the provided data, the subsampling process and the evaluation system. In Section 3 we describe the developed method for filtering noisy data. We describe the experiments conducted and the results. Conclusions and future work are drawn in Section 4.

Proceedings of the Fourth Conference on Machine Translation (WMT), Volume 3: Shared Task Papers (Day 2) pages 282–288
Florence, Italy, August 1-2, 2019. ©2019 Association for Computational Linguistics

2 WMT 2019 shared task on parallel corpus filtering for low-resource conditions

The task “Parallel Corpus Filtering for Low-Resource Conditions”[1] tackles the problem of cleaning noisy parallel corpora for low-resourced language pairs. Given a noisy parallel corpus, participants are required to develop methods to filter it down to a smaller size with a high quality subset. This year there are two language pairs: Nepali–English and Sinhala–English. Participants are asked to provide score files for each sentence in each of the noisy parallel sets. The scores will be used to subsample sentence pairs into two different training set sizes: 1 million and 5 million English words. For this task, very noisy corpora of 40.6 million English words in Nepali–English and 59.6 million English words in Sinhala–English are provided. The data were crawled from the web as part of the Paracrawl project[2]. The quality of the resulting subsets is determined by the quality of a statistical machine translation (SMT) and neural machine translation (NMT) systems trained on this data. The quality of the machine translation system is measured with the sacreBLEU score (Post, 2018) on a held-out test set of Wikipedia translations for Nepali-English (ne–en) and Sinhala-English (si–en). The organisers provide development and test sets for each pair of languages but due to the fact that the task addresses the challenge of data quality and not domain-relatedness of the data for a particular use case, the test sets may be very different from the final official test set in terms of topics.

2.1 Data provided

Organisers provide noisy corpora for the Nepali–English and Sinhala–English language pairs. The main figures of both corpora are shown in Table 1.

In addition, organisers provide links to the permissible third-party sources of bilingual data to be used in the competition. Parallel corpora for the Nepali–English language pair comes from the Bible, Global Voices, Penn Tree Bank, GNOME/KDE/Ubuntu and Nepali Dictionary corpora. For the Sinhala–English language pair, the Open Subtitles and GNOME/KDE/Ubuntu parallel corpora are provided. The main figures of the parallel corpora are shown in Table 2.

Table 1: Main figures of the noisy corpora for the Nepali–English and Sinhala–English language pairs. k denotes thousands of elements and M denotes millions of elements. $|S|$ stands for number of sentences, $|W|$ for number of running words, and $|V|$ for vocabulary size. Figures computed on tokenised and lowercased corpora.

corpus	language	$\|S\|$	$\|W\|$	$\|V\|$
ne–en	Nepali	2.2M	52.3M	925.3k
	English		56.0M	782.9k
si–en	Sinhala	3.6M	61.2M	822.6k
	English		62.6M	803.0k

Table 2: Allowed parallel corpora for Nepali–English and Sinhala–English main figures. k denotes thousands of elements and M denotes millions of elements. $|S|$ stands for number of sentences, $|W|$ for number of running words, and $|V|$ for vocabulary size. Figures computed on tokenised and lowercased corpora.

corpus	language	$\|S\|$	$\|W\|$	$\|V\|$
ne–en	Nepali	573k	4.2M	141.3k
	English		4.5M	64.5k
si–en	Sinhala	692k	4.5M	178.5k
	English		5.0M	69.9k

In addition to the parallel data above, monolingual corpora are also provided. The main figures of the monolingual corpora for Nepali, Sinhala and English are shown in Table 3.

Table 3: Main figures of the monolingual data for Nepali, Sinhala and English languages. k denotes thousands of elements and M denotes millions of elements. $|S|$ stands for number of sentences, $|W|$ for number of running words, and $|V|$ for vocabulary size. Figures computed on tokenised and lowercased corpora.

language	$\|S\|$	$\|W\|$	$\|V\|$
Nepali	3.7M	116.1M	1.4M
Sinhala	5.3M	43.2M	766.7k
English	448.2M	760.2M	9.6M

Additional resources provided in the shared task were a Hindi–English (hi–en) parallel corpus and Hindi monolingual data. The main figures of these two corpora are shown in Table 4.

Finally, development and development test sets

[1] http://www.statmt.org/wmt19/parallel-corpus-filtering.html

[2] https://paracrawl.eu/

Table 4: Main figures of the monolingual (mono.) data for Hindi and bilingual data for Hindi–English (hi–en). k denotes thousands of elements and M denotes millions of elements. $|S|$ stands for number of sentences, $|W|$ for number of running words, and $|V|$ for vocabulary size. Figures computed on tokenised and lowercased corpora.

corpus	lang.	$\|S\|$	$\|W\|$	$\|V\|$
mono.	Hindi	45.1M	838.8k	4.0M
hi–en	Hindi	1.6M	22.4M	333.3k
	English		20.7M	192.5k

are provided in the shared task. Both sets are drawn from Wikipedia articles. These may be very different from the final official test set in terms of topics due to the fact that the task addresses the challenge of data quality and not domain-relatedness of the data. Main figures of development sets are shown in Table 5.

Table 5: Development sets main figures. k denotes thousands of elements. $|S|$ stands for number of sentences, $|W|$ for number of running words, and $|V|$ for vocabulary size. Figures computed on tokenised and lowercased corpora.

		Validation sets		
corpus	lang.	$\|S\|$	$\|W\|$	$\|V\|$
ne–en	Nepali	2.6k	10.2k	37.1k
	English		37.1k	10.2k
si–en	Sinhala	2.9k	48.7k	103.3k
	English		53.5k	6.2k
		Test sets		
corpus	lang.	$\|S\|$	$\|W\|$	$\|V\|$
ne–en	Nepali	2.8k	43.2k	10.9k
	English		51.5k	6.4k
si–en	Sinhala	2.8k	46.4k	9.6k
	English		51.0k	6.1k

2.2 Sub-sampling of noisy data

Participants submit files with numerical scores, giving one score per line for the original unfiltered parallel corpus. A tool provided by the organisers takes as input the scores and the noisy parallel corpus. The tool then selects sentences with higher scores to complete the desired 1M and 5M words in target. Systems trained on these data sets are used for evaluation by the organisers.

2.3 Translation evaluation

As specified in the shared task, the evaluation of a selected subset of sentences is done using SMT and NMT. The SMT system is implemented using Moses (Koehn et al., 2007) and the NMT system is built using the FAIRseq (Ott et al., 2019) toolkit. Organisers provided scripts which allow for implementing the same translation system which will be used in the final evaluation. However, we only conducted experiments using NMT. The FAIRseq system tokenises source and target sentences and applies BPE (Sennrich et al., 2016). The tokenisation of Nepali, Sinhala and Hindi sentences is done using the Indic NLP Library[3]. The system (Guzmán et al., 2019) uses a Transformer architecture with 5 encoder and 5 decoder layers, where the number of attention heads, embedding dimension and inner-layer dimension are 2, 512 and 2048, respectively. The model is regularised with dropout, label smoothing and weight decay. The model is optimised with Adam (Kingma and Ba, 2014) using $\beta_1 = 0.9$, $\beta_2 = 0.98$, and $\epsilon = 1e - 8$. The learning rate is fixed to $lr = 1e3$, as described in (Ott et al., 2019). The NMT system from the shared task is trained for 100 epochs and models are saved every 10 epochs. The best model is chosen according to validation set loss function value. The script which allowed us to reproduce the network used in the shared task can be found at https://github.com/facebookresearch/flores. All experiments were performed using NVidia Titan Xp GPUs.

3 System description

In this section, the entire process of sentence scoring is detailed.

Our process for scoring noisy corpora is as follows:

1. We apply bilingual cross-entropy selection (described in 3.1.1) to select the best set of sentences from Nepali, Sinhala and Hindi to English jointly for each language pair: Nepali–English and Sinhala–English.

2. We train an NMT engine using the above selected data for each language pair.

3. Once the NMT engine is trained, we generate a hypothesis for each sentence in the noisy corpus.

[3] https://anoopkunchukuttan.github.io/indic_nlp_library/

4. We then compute smoothed BLEU for each target sentence in the noisy corpus, along with its corresponding hypothesis. These computed BLEU scores will be used for the selection of the required subsets of 1M and 5M words of English tokens for the final evaluation.

5. Additionally, we apply a few rules (described in 3.3) to discard some sentences which are considered useless, by replacing their smoothed BLEU score to zero, effectively avoiding that the selection algorithm includes such sentences into the selected subsets.

3.1 Translation engine

The main core of the scoring process is hypothesis generation using a well-trained translation model. To create the translation model we used the NMT system from the shared task and we selected sentences from all provided bilingual corpora in all three language pairs jointly: Nepali, Sinhala and Hindi to English. To select the subset of sentences to train the translation model we used the bilingual cross-entropy selection method (Moore and Lewis, 2010) described in the next subsection.

3.1.1 Bilingual Cross-Entropy selection

We ranked sentences from all bilingual corpora by their perplexity score according to a language model trained on the monolingual corpora in Nepali, Sinhala and English. The perplexity ppl of a string s with empirical ngram distribution p given a language model q is:

$$ppl(s) = 2^{-\sum_{x \in s} p(x) log\, q(x)} = 2^{H(p,q)} \qquad (1)$$

where $H(p, q)$ is the cross-entropy between p and q. Selecting the sentences with the lowest perplexity is therefore equivalent to choosing the sentences with the lowest cross-entropy according to the language model trained on monolingual data. To compute bilingual cross-entropy score $\mathcal{X}(s)$ of a sentence s, we sum the cross-entropy difference over each side of the corpus, both source and target:

$$\begin{aligned}\mathcal{X}(s) = [H_{M-src}(s) - H_{N-src}(s)] + \\ [H_{M-tgt}(s) - H_{N-tgt}(s)]\end{aligned} \qquad (2)$$

where $H_{M-src}(s)$ and $H_{M-trg}(s)$ are the cross-entropy of a source/target sentence, respectively, according to a language model trained on the monolingual data provided, and $H_{N-src}(s)$ and $H_{N-trg}(s)$ are the cross-entropy of a source/target sentence, respectively, according to a language model trained on the noisy corpora. Lower scores are presumed to be better.

3.2 Filtering by hypothesis

Here, the purpose is to filter the noisy data according to the potential smoothed BLEU score of the sentence pair and the generated hypothesis. With the purpose of building a translation system for obtaining this probability, we trained an NMT system with different training set sizes selected using the bilingual cross-entropy technique above. The system was trained for 200 epochs, which was enough to achieve convergence. As development set, and for selecting the best model for computing the BLEU score of the hypothesis associated to a sentence pair, we used the same development set as provided in the shared task. We selected the best epoch according to validation set loss function value. In Table 6 we show sacreBLEU scores for models trained with different number of sentences.

Table 6: Validation sacreBLEU scores for bilingual cross-entropy selection results depending on the number of training sentences for Nepali–English and Sinhala–English. M denotes millions of elements. Best system marked in bold.

Nepali–English	
Training size	Validation
1.0M	11.7
1.5M	12.3
2.0M	12.2
2.5M	12.2
3.0M	**14.9**
3.5M	13.5
Sinhala–English	
Training size	Validation
1.0M	8.3
1.5M	8.8
2.0M	9.8
2.5M	9.5
3.0M	**9.9**
3.5M	9.5

In both language pairs, Nepali–English and Sinhala–English, the best model was achieved using 3M sentences. Once the best models were selected, we translated the noisy corpora and we obtained the hypothesis for each sentence, which allowed us to compute the corresponding smoothed BLEU score. This is the final score provided as competition result. However, and before providing the score, we also applied other filtering strategies, as described in the following subsections.

3.3 Rule based Filtering

After obtaining the hypothesis for each sentence from the noisy corpora, we applied a few rules to filter the sentence pairs. These rules are the following:

1. Remove sentence pairs where the source or target sentence contains more than 250 BPE segments.

2. Remove sentences where the lower-cased source sentence is equal to the lower-cased target sentence.

3. Remove sentence pairs which do not contain any Nepali/Sinhala characters in the source sentence.

4. Remove sentences where the smoothed BLEU score between the source and the target sentence is higher than a fixed threshold μ. We explored different values for this threshold $\mu = \{0.20, 0.25, 0.30, 0.35, 1.0\}$. Note that the space between 0.35 and 1.0 was not explored because values of μ only slightly above 0.35 already implied that no sentences were filtered.

The order in which the rules are applied is important, since sentences that are filtered out with zero score assigned by one rule will not be a candidate for selection in subsequent rules. After applying different threshold values we used the provided script to subsample sentence pairs to amount to 1 million and 5 million English words. The results of training the final NMT system by applying different thresholds μ are shown in Tables 7 and 8.

Finally, we selected thresholds $\mu = 0.35$ for the Nepali–English corpus, and $\mu = 1.00$ (no threshold, all BLEU values between source and target sentences accepted) for the Sinhala–English language pair. In Table 9, the number of removed sentences by each rule are shown.

In total, we discarded 1.2M from Nepali noisy corpus and 1.9M sentences from Sinhala noisy corpus. The rest of sentences from noisy corpus were scored using target-hypothesis smoothed BLEU described previously.

Table 7: SacreBLEU scores for final NMT system trained using sentences selected with different values of threshold μ for Nepali–English.

Nepali–English			
Eng. words	μ	Valid	Test
1M	0.20	0.1	0.2
	0.25	3.3	4.1
	0.30	3.4	4.2
	0.35	**3.4**	**4.3**
	1.00	2.4	3.0
5M	0.20	0.2	0.2
	0.25	2.6	3.0
	0.30	2.8	3.2
	0.35	**3.0**	**3.4**
	1.00	3.0	3.3

Table 8: SacreBLEU scores for final NMT system trained using sentences selected with different values of threshold μ for Sinhala–English.

Sinhala–English			
Eng. words	μ	Valid	Test
1M	0.20	2.0	2.4
	0.25	2.2	2.2
	0.30	2.3	3.1
	0.35	2.3	2.4
	1.00	**2.4**	**2.3**
5M	0.20	2.6	2.8
	0.25	3.1	3.0
	0.30	3.6	3.4
	0.35	3.3	3.4
	1.00	**4.2**	**4.3**

3.4 Baseline comparision

Once we selected the best models, we compared the obtained sacreBLEU scores with the Zipporah model results published on wmt2019 website. The Zipporah model extracts a bag-of-words translation feature, and trains logistic regression models to classify good data and synthetic noisy data in the proposed feature space. The trained model is used to score parallel sentences in the data pool for selection. In Table 10 we show our result compared to the Zipporah model.

Table 9: Statistics of how many sentences of noisy corpus were set their final score as zero after applying different rules. The number in parenthesis indicates the rule described in the enumerated list above. k denotes thousands of elements and M denotes millions of elements.

Nepali–English	
Rule	Removed sentences
(1) BPE >250	89.4k
(2) src=trg	186.8k
(3) No Nepali symbols	722.7k
(4) src-trg BLEU > 0.35	207.2k

Sinhala–English	
Rule	Removed sentences
(1) BPE >250	76.7k
(2) src=trg	78.3k
(3) No Sinhala symbols	1.7M
(4) src-trg BLEU > 1.00	None

Table 10: SacreBLEU scores for NMT system comparison with the Zipporah model.

Nepali–English		
Eng. words	Model	Test
1M	Sciling	4.3
	Zipporah	**5.2**
5M	Sciling	**3.4**
	Zipporah	1.9

Sinhala–English		
Eng. words	Model	Test
1M	Sciling	2.3
	Zipporah	**4.7**
5M	Sciling	**4.3**
	Zipporah	3.7

4 Conclusions and future work

We introduced filtering of noisy parallel corpora based on hypothesis generation and combined this filtering with several filtering rules. We submitted only the best set of scores for each language pair to the shared task. In both language pairs, Nepali–English and Sinhala–English, we achieved results that performed better than the Zipporah baseline with corpora containing 5M English words. Our conclusion is that the designed filtering method is able to reach better performance when confronted with larger amounts of data.

Future work should concentrate on further improving of our filtering method. We would train a logistic model to combine the BLEU score between the generated hypothesis and target with the BLEU score between source and target instead of threshold values. Also, we would apply data selection techniques such as infrequent n-gram selection (Parcheta et al., 2018) or continuous vector-space representation of sentences (Chinea-Rios et al., 2019).

Acknowledgments

Work partially supported by MINECO under grant DI-15-08169 and by Sciling under its R+D programme. The authors would like to thank NVIDIA for their donation of Titan Xp GPU that allowed to conduct this research.

References

Amittai Axelrod, Xiaodong He, and Jianfeng Gao. 2011. Domain adaptation via pseudo in-domain data selection. In *Proc. of EMNLP*, pages 355–362.

Mara Chinea-Rios, Germán Sanchis-Trilles, and Francisco Casacuberta. 2019. Vector sentences representation for data selection in statistical machine translation. *Computer Speech & Language*, 56:1–16.

Francisco Guzmán, Peng-Jen Chen, Myle Ott, Juan Pino, Guillaume Lample, Philipp Koehn, Vishrav Chaudhary, and Marc'Aurelio Ranzato. 2019. Two new evaluation datasets for low-resource machine translation: Nepali-english and sinhala-english. *arXiv preprint arXiv:1902.01382*.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, et al. 2007. Moses: Open source toolkit for statistical machine translation. In *Proc. of ACL*, pages 177–180.

Chin-Yew Lin and Franz Josef Och. 2004. Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics. In *Proc. of ACL*, pages 605–615.

Robert C. Moore and William Lewis. 2010. Intelligent selection of language model training data. In *Proc. of ACL*, pages 220–224.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and

Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. *arXiv preprint arXiv:1904.01038*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proc. of ACL*, pages 311–318.

Zuzanna Parcheta, Germán Sanchis-Trilles, and Francisco Casacuberta. 2018. Data selection for nmt using infrequent n-gram recovery. pages 219–228.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proc. of WMT*, pages 186–191.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proc. of ACL*, volume 1, pages 1715–1725.