Parallel Corpus Filtering based on Fuzzy String Matching

Sukanta Sen, Asif Ekbal, Pushpak Bhattacharyya
Department of Computer Science and Engineering
Indian Institute of Technology Patna
{sukanta.pcs15,asif,pb}@iitp.ac.in

Abstract

In this paper, we describe the IIT Patna's submission to WMT 2019 shared task on parallel corpus filtering. This shared task asks the participants to develop methods for scoring each parallel sentence from a given noisy parallel corpus. Quality of the scoring method is judged based on the quality of SMT and NMT systems trained on smaller set of high-quality parallel sentences sub-sampled from the original noisy corpus. This task has two language pairs. We submit for both the Nepali-English and Sinhala-English language pairs. We define fuzzy string matching score between English and the translated (into English) source based on Levenshtein distance. Based on the scores, we sub-sample two sets (having 1 million and 5 millions English tokens) of parallel sentences from each parallel corpus, and train SMT systems for development purpose only. The organizers publish the official evaluation using both SMT and NMT on the final official test set. Total 10 teams participated in the shared task and according the official evaluation, our scoring method obtains 2nd position in the team ranking for 1-million Nepali-English NMT and 5-million Sinhala-English NMT categories.

1 Introduction

In this paper, we describe our submission to the WMT 2019[1] parallel corpus filtering task (Koehn et al., 2019). The aim of this shared task is to extract two smaller sets of high-quality parallel sentences from a very noisy parallel corpus. This parallel corpus is crawled from the web as part of the Paracrawl project and contains all kinds of noise (wrong language in source and target, sentence pairs that are not translations of each other, bad language, incomplete or bad translations, etc.). This task provides the participants two sets of such noisy parallel corpora: one is for Nepali-English with English token count of 40.6 million and another is for Sinhala-English with English token count of 59.6 million. The participants are asked to submit score for each sentence in each of these two parallel corpora (Nepali-English and Sinhala-English). Based on the scores, two smaller sets of parallel sentences that amount to 1 million and 5 millions are extracted from each of those two parallel corpora. The quality of the scoring method is judged based on the quality of the neural machine translation (NMT) and statistical machine translation (SMT) systems trained on these smaller corpora. We participated in both language pair: Nepali-English and Sinhala-English.

Building machine translation (MT) systems, specifically NMT (Kalchbrenner and Blunsom, 2013; Cho et al., 2014; Sutskever et al., 2014; Bahdanau et al., 2015) systems, require supervision of huge amount of high-quality parallel training data. Though recently emerged unsupervised NMT (Artetxe et al., 2018; Lample et al., 2018) has shown promising results on related language pairs, it does not work for distant language pairs like Nepali-English and Sinhala-English (Guzmán et al., 2019). Also, a vast majority of languages in the world fall in the category of low-resource languages as they have too little, if any, parallel data. However, getting parallel training data is not easy as it takes time, money and expert translators. Though we can have parallel data compiled from online sources, it is not reliable as it is often very noisy and poor in quality. It has been found that MT systems are sensitive to noise (Khayrallah and Koehn, 2018). This necessitates to filter out noisy sentences from a large pool of parallel parallel sentences.

Parallel corpus filtering task of WMT 2019 focuses on two new low-resource languages pairs:

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

Proceedings of the Fourth Conference on Machine Translation (WMT), Volume 3: Shared Task Papers (Day 2) pages 291–295
Florence, Italy, August 1-2, 2019.

Nepali-English and Sinhala-English for which we have very little amount of publicly available parallel corpora. We use these parallel corpora for building our scoring scheme based on fuzzy string matching. Total 10 teams participated in the shared task. According the official evaluation, our scoring method obtains 2nd position in the team ranking in two categories: 1-million Nepali-English NMT and 5-million Sinhala-English NMT.

2 Our Approach

The raw parallel corpus is very noisy and main contributing to that is the wrong language. We study both the parallel corpora (Nepali-English and Sinhala-English) and find that there are many parallel sentences which have wrong language at source, target, or both sides. We use language identifier to remove these sentences. The block diagrammatic representation of our approach has been shown in figure 1.

In our scoring scheme, 0 is the lowest score of a parallel sentence. We set score 0 in the following scenarios:

- Wrong source or target: we detect the language of a sentence pair using *langid*[2] and if any of the source or target has wrong language id, we set 0 score to that sentence pair. This helps in filtering out many wrong parallel sentences.

- As official evaluation is done using MT systems trained on sub-sampled sentences having maximum 80 tokens, we set score 0 to all the sentence pairs that have a source or target length more than 80 tokens.

For further scoring, we translate the Nepali (or Sinhala) sentences from remaining parallel sentences into English and find the lexical matching between a English sentence E and translated English $E^{'}$. To score each pair XX-English (XX is Nepali or Sinhala), we consider four fuzzy string matching scores based on Levenshtein distance (Levenshtein, 1966) between target (English) and source (translated into English). These score are implemented in fuzzywuzzy[3], a python-based string matching package, as:

- *Ratio (R_1)*: ratio between E and $E^{'}$ defined as:

$$\frac{|E| + |E^{'}| - L}{|E| + |E^{'}|} \quad (1)$$

  where $|E|$ and $|E^{'}|$ are the lengths of E and $E^{'}$, and L is the Levenshtein distance between E and $E^{'}$.

- *Partial ratio (R_2)*: same as R_1 but based on sub-string matching. It first finds the best matching sub-string between the two input strings E and $E^{'}$. Then it finds R_1 between the sub-string and shorter string among the two input strings.

- *Token sort ratio (R_3)*: E and $E^{'}$ are sorted and then R_1 is calculated between the sorted E and $E^{'}$.

- *Token set ratio (R_4)*: It first removes the duplicate tokens in E and $E^{'}$ and then calculates R_1.

We combine these four scores (R_1, R_2, R_3, R_4) in two different ways (taking arithmetic mean or geometric mean):

$$Score_{AM} = \frac{1}{4}\sum_{i=1}^{4} R_i \quad (2)$$

$$Score_{GM} = \left(\prod_{i=1}^{4} R_i\right)^{\frac{1}{4}} \quad (3)$$

3 Datasets

Source	#Sents	#Tokens
Nepali-English		
Bible	61,645	1,507,905
Global Voices	2,892	75,197
Penn Tree Bank	4,199	88,758
GNOME/KDE/Ubuntu	494,994	2,018,631
Total	563,640	
Sinhala-English		
Open Subtitles	601,164	3,594,769
GNOME/KDE/Ubuntu	45,617	150,513
Total	646,781	

Table 1: Training data sources and number of sentences. These corpora are used to train SMT systems used for fuzzy string matching. #**Sents**: Sentence counts; #**Tokens**: English token counts.

[2] https://github.com/saffsd/langid.py

[3] https://github.com/seatgeek/fuzzywuzzy

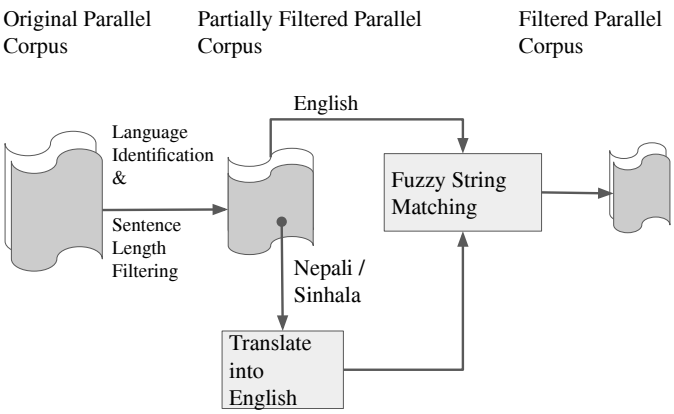

Figure 1: Block diagrammatic representation of our approach. We first apply language identification and set maximum sentence length of up to 80 to get partially filtered corpus from original corpus. Then translate non-English (Nepali / Hindi) sentence into English. Finally, we apply fuzzy string matching between original English and translated English to get filtered corpus.

Set	Nepali-English	Sinhala-English
dev	2,559	2,898
devtest	2,835	2,766

Table 2: Number of sentences in dev and devtest.

This filtering task is focused on two language pairs: Nepali-English with a 40.6 million-word (English token count) and Sinhala-English with a 59.6 million-word for which we develop our method to score each pair of sentences. These parallel corpora are compiled from the web. Apart these two parallel corpora, some other publicly available data are provided for development purpose. Nepali and Sinhala have very little publicly available parallel data. Most of the parallel data for Nepali-English originate from GNOME and Ubuntu handbooks, and rest of the parallel sentences are compiled from Bible corpus (Christodouloupoulos and Steedman, 2015), Global Voices, Penn Tree Bank. For Sinhala-English, we have only two sources of parallel data: OpenSubtitles (Lison et al., 2018), and GNOME and Ubuntu handbooks.

We use only above mentioned, shown in Table 1, parallel data for training phrase-based SMT (Koehn et al., 2003) systems to translate non-English (Nepali and Sinhala) into English for calculating fuzzy string matching scores. Apart from those parallel data, participants are provided with development (*dev*) and development test (*devtest*) sets having parallel sentence counts 2559 and 2835 for Nepali-English, and 2898 and 2766 for Sinhala-English, respectively. The details of the data are shown in the Table 1 and 2. We tokenize the training, development and test sets in preprocessing stage. For tokenizing Nepali and Sinhala, we use Indic NLP library[4], and for tokenizing English sentences, we use the Moses tokenizer[5].

4 Experiments

For our fuzzy string matching as well as evaluating the quality of the sub-sampled sets, we build XX-English (XX is Nepali or Sinhala) phrase-based SMT (Koehn et al., 2003) system using the Moses tool (Koehn et al., 2007). For training the SMT system we keep the following settings: grow-diag-final-and heuristics for word alignment, msd-bidirectional-fe for reordering model, and 5-gram language model with modified Kneser-Ney smoothing (Kneser and Ney, 1995) using KenLM (Heafield, 2011). The BLEU[6] (Papineni et al., 2002) scores for these SMT systems are 3.7 and 4.6 for Nepali-English and Sinhala-English, respectively.

5 Results

Crude filtering based on language identification and sentence length filtered out almost 77% and 70% parallel sentences from Nepali-English and Sinhala-English corpora, respectively. However,

[4] https://bitbucket.org/anoopk/indic_nlp_library

[5] https://github.com/moses-smt/mosesdecoder/blob/RELEASE-3.0/scripts/tokenizer/tokenizer.perl

[6] We use sacreBLEU (Post, 2018).

Scoring Scheme	1 million				5 million			
	SMT		NMT		SMT		NMT	
	test	*devtest*	*test*	*devtest*	*test*	*devtest*	*test*	*devtest*
Nepali-English								
Arithmetic Mean	3.84	3.64	5.48	5.94	4.34	4.03	1.29	1.25
Geometric Mean	3.89	3.57	5.28	5.57	4.27	4.01	1.32	1.25
Sinhala-English								
Arithmetic Mean	3.07	3.63	3.16	3.70	4.44	5.12	3.87	4.54
Geometric Mean	3.03	3.52	3.01	3.36	4.42	5.17	4.28	5.08

Table 3: Official BLEU scores for 1-million and 5-million sub-sampled sets.

we observe that the language identifier is not efficient in identifying Nepali or Sinhala sentences and misclassifies many sentences. For example, many Nepali sentences are classified as Hindi or Marathi.

Corpus	Before	After
Nepali-English	2,235,512	509,750
Sinhala-English	3,357,018	1,015,504

Table 4: Number of parallel sentences in the raw parallel corpora before and after applying language identification and sentence length based filtering.

Then using the SMT systems as described in Section 4, we translate the Nepali (or Sinhala) sentences from partially filtered parallel corpora into English, and apply fuzzy string matching to score each pair of sentences. We sub-sample sets with 1 million and 5 million English tokens. The size of the sub-sampled sets are shown in the Table 5. To judge the quality of the sub-sampled sets, we train SMT systems following the settings described in 4. We measure the quality of theses sub-samples using BLEU scores shown in Table 6.

Official Evaluation Total 10 teams participated in the shared task. The organizers (Koehn et al., 2019) publish the BLEU scores of the 1-million and 5-million sub-sampled sets on the final official test sets. Official BLEU scores for our systems are shown in the Table 3.

6 Conclusion

In this paper, we report our submission to WMT 2019 shared task on parallel corpus filtering. The aim of this task is to score each parallel sentence from two very noisy parallel corpora: Nepali-English and Sinhala-English. We develop a fuzzy string matching scoring scheme based on Levenshtein distance between and English and translated English sentences. Quality of the scoring technique is judged by the quality of SMT and NMT systems. For development purpose, we train only SMT systems to check the quality of the scoring method. Total 10 teams participated in the shared task. The organizers publish the official evaluation using both SMT and NMT on the final official test set. In the team ranking, our scoring method obtains 2nd position in 1-million Nepali-English NMT and 5-million Sinhala-English NMT categories.

Scoring Scheme	1 million	5 million
Nepali-English		
Arithmetic Mean	56,868	200,725
Geometric Mean	53,821	185,978
Sinhala-English		
Arithmetic Mean	70,114	264,271
Geometric Mean	67,888	249,275

Table 5: Number of sentences for 1-million and 5-million sub-sampled sets for two scoring schemes.

Scoring Scheme	1 million	5 million
Nepali-English		
Baseline	3.40	4.22
Arithmetic Mean	4.20	3.50
Geometric Mean	4.30	3.80
Sinhala-English		
Baseline	4.16	4.77
Arithmetic Mean	4.20	5.10
Geometric Mean	4.00	5.30

Table 6: BLEU scores on *devtest* for SMT systems trained on two sub-sampled sets. Baseline is the official baseline as reported in shared task page. We use sacreBLEU (Post, 2018).

References

Mikel Artetxe, Gorka Labaka, Eneko Agirre, and Kyunghyun Cho. 2018. Unsupervised Neural Machine Translation. In *Proceedings of the Sixth International Conference on Learning Representations (ICLR)*.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural Machine Translation by Jointly Learning to Align and Translate. In *Proceedings of the 3rd International Conference on Learning Representation (ICLR 2015)*.

Kyunghyun Cho, Bart Van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the Properties of Neural Machine Translation: Encoder-decoder Approaches. In *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*, pages 103–111.

Christos Christodouloupoulos and Mark Steedman. 2015. A massively parallel corpus: the Bible in 100 languages. *Language resources and evaluation*, 49(2):375–395.

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*.

Kenneth Heafield. 2011. KenLM: Faster and Smaller Language Model Queries. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 187–197. Association for Computational Linguistics.

Nal Kalchbrenner and Phil Blunsom. 2013. Recurrent Continuous Translation Models. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP 2013)*, pages 1700–1709.

Huda Khayrallah and Philipp Koehn. 2018. On the Impact of Various Types of Noise on Neural Machine Translation. In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, Melbourne, Australia. Association for Computational Linguistics.

Reinhard Kneser and Hermann Ney. 1995. Improved backing-off for m-gram language modeling. In *Acoustics, Speech, and Signal Processing, 1995. ICASSP-95., 1995 International Conference on*, volume 1, pages 181–184. IEEE.

Philipp Koehn, Francisco Guzmán, Vishrav Chaudhary, and Juan M. Pino. 2019. Findings of the WMT 2019 Shared Task on Parallel Corpus Filtering for Low-Resource Conditions. In *Proceedings of the Fourth Conference on Machine Translation, Volume 2: Shared Task Papers*, Florence, Italy. Association for Computational Linguistics.

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 *Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions*, pages 177–180. Association for Computational Linguistics.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical Phrase-Based Translation. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 48–54. Association for Computational Linguistics.

Guillaume Lample, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018. Unsupervised Machine Translation using Monolingual Corpora Only. In *Proceedings of the Sixth International Conference on Learning Representations (ICLR)*.

Vladimir I Levenshtein. 1966. Binary codes capable of correcting deletions, insertions, and reversals. In *Soviet physics doklady*, volume 10, pages 707–710.

Pierre Lison, Jörg Tiedemann, and Milen Kouylekov. 2018. Opensubtitles2018: Statistical Rescoring of Sentence Alignments in Large, Noisy Parallel Corpora. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318, Philadelphia, Pennsylvania.

Matt Post. 2018. A Call for Clarity in Reporting BLEU Scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Belgium, Brussels. Association for Computational Linguistics.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to Sequence Learning with Neural Networks. In *Proceedings of Advances in neural information processing systems (NIPS 2014)*, pages 3104–3112.