Coverage and Cynicism: The AFRL Submission to the WMT 2018 Parallel Corpus Filtering Task

Grant Erdmann, Jeremy Gwinnup
Air Force Research Laboratory
grant.erdmann@us.af.mil,jeremy.gwinnup.1@us.af.mil

Abstract

The WMT 2018 Parallel Corpus Filtering Task aims to test various methods of filtering a noisy parallel corpus, to make it useful for training machine translation systems. We describe the AFRL submissions, including their preprocessing methods and quality metrics. Numerical results indicate relative benefits of different options and show where our methods are competitive.

1 Introduction

For this task the participants were provided with a large corpus of parallel data in English and German. The corpus contains approximately 10^8 lines, with approximately 10^9 words in each language. Hunalign scores (Varga et al., 2005) also were provided for each line. The task organizers built statistical machine translation (SMT) and neural machine translation (NMT) systems from the scores produced, based on parallel training sets of 10^6 and 10^7 words.

Subset selection techniques often strive to reduce a set to the most useful. In this circumstance, this entails:

- Avoiding selecting a line with undue repetition of content of other selected lines. This can extend training times and/or skew the translation system to favor this type of line.
- Avoid selecting long lines, which will be ignored in training an NMT system.

In addition to adapting the corpus to the building of a general-purpose machine translation system, we must also deal with its significant noise. The main types of noise present in the given data are:

- Not natural language
- One or both languages are incorrect
- Correct languages and natural language, but not translations of each other

2 Preprocessing

As a first step, a rough preprocessing filter is applied to the data. This entails removing:

- Lines where either language contains more than 80 words
- Lines where either language contains less than 4 words
- Lines containing "www", as lines with web addresses tend to provide less useful information
- Lines where the ratio of the number of English words to the number of German words is greater than three or less than one third
- Lines containing characters with the Unicode general category of "other"
- Lines where the English text is identical to the German text, after removing space, period, and numeric characters.
- Lines where numeric characters are different (or in a different order) in the two languages
- Lines where the hunalign score is less than 0.5 or greater than 1.5

The first of these criteria is based on limitations of NMT training, where long lines are discarded or truncated. The other criteria are highly empirical, based on indicators of apparent qualitative problems.

The remaining lines are put through further processing prior to scoring:

- Punctuation is normalized

Proceedings of the Third Conference on Machine Translation (WMT), Volume 2: Shared Task Papers, pages 885–889

https://doi.org/10.18653/v1/W18-6476

- Words are truncated to 72 characters. The tokenizer attempts to separate German compound words, and long words cause it to hang.

- Language-specific tokenization is performed, using SYSTRAN's Linguistic Development Kit. Subword units are generated via byte-pair-encoding (BPE) (Gage, 1994). The BPE models are learned on a per-language basis, trained with 2000 byte-pair encoding merges, over all WMT 2018 news translation task parallel German–English data[1] without the Paracrawl[2] corpus. This small vocabulary was chosen to reduce the number of out-of-vocabulary tokens resulting from morphology and compounding.

- The BPE form is transformed into the format used for character-based processing, with denoted spaces and no subword continuation markers (e.g., `stand@@ ard prac@@ tice` becomes `stand ard _ prac tice`)

- Case features are removed, essentially allowing BPE formation using case but scoring lowercased.

This preprocessed text is used to generate the scores that determine a line's usefulness.

3 Coverage Metric

We use two metrics to estimate the relative appropriateness of a selected set to a reference. The first is our own coverage metric (Gwinnup et al., 2016), which we reproduce here. Let us select a subset S from a larger set C to maximize its similarity to a representative set T. Let our preferred subselected set size be τ times the size of T. Let $\mathcal{V}$ be a set of vocabulary elements of interest. Define $c_v(X)$ to be the count of the occurrence of feature $v \in \mathcal{V}$ in a given corpus X and $c_v^\tau(T) = c_v(T)/\tau$ to be the scaled count that accounts for the preferred size of the selected set. The coverage g is then given by

$$g(S,T,\tau) = \frac{\sum_{v\in\mathcal{V}} f(\min(c_v(S), c_v^\tau(T)))}{\sum_{v\in\mathcal{V}} f(c_v^\tau(T)) + p_v(S,T,\tau)} \quad (1)$$

where the oversaturation penalty $p_v(S,T,\tau)$ is

$$\max(0, c_v(S) - c_v^\tau(T))\,[f(c_v^\tau(T)+1) - f(c_v^\tau(T))]\,.$$

Here f can be any submodular function, and we choose exclusively $f(x) = \log(1+x)$.

The final score reported for a line is the change it makes to the coverage metric on its inclusion. Lines which are not selected are given scores of zero.

4 Cynical Metric

As another approach we defined a metric based on the cynical selection method (Axelrod, 2017), which seeks to minimize the cross-entropy H. In our terms, this is

$$H(S,T) = -\sum_{v\in\mathcal{V}} \frac{c_v(T)}{\sum_{v'\in\mathcal{V}} c_{v'}(T)} \log \frac{c_v(S)}{\sum_{v'\in\mathcal{V}} c_{v'}(S)}. \quad (2)$$

We prefer to maximize metrics, so we define $h(S,T) = -H(S,T)$ as the cynical metric to maximize. Including the scaling factor τ would have no effect on the cross-entropy value.

Note that Axelrod (2017) defines the cross-entropy purely in terms of unigrams, motivated by an unsmoothed unigram language model. We include unigrams through 4-grams in our feature set $\mathcal{V}$. This extension to n-grams was not recommended by Axelrod (2017). However, we found it useful for this task.

The final score reported for a line is the change it makes to the cynical metric on its inclusion, with a maximum score of 1. Lines which are not selected are given scores of zero.

5 Set-building Algorithm

Whether the metric is our coverage metric or our cynical metric, the method of building the set is the same. We iterate the following two steps until the selected set is large enough:

1. Add the line that has the best effect on the metric.

2. Check if removing a line from the selected corpus would improve the metric. If so, remove the line with greatest such improvement, unless it was the most-recently selected or would lead to infinite cycling.

This is a greedy algorithm with review after each selection.

[1] http://www.statmt.org/wmt18/translation-task.html#download

[2] https://paracrawl.eu

6 Translation Score

The preceding processes and metrics were designed to remove many sources of error mentioned in the introduction of this paper. However, we have not yet dealt with the case of having both English and German lines being natural and useful, but the lines not being translations of one another. To help mitigate this phenomenon, we created a German–English NMT system using OpenNMT(Klein et al., 2017). It was trained on all WMT 2018 news translation task parallel German–English data, excluding the Paracrawl corpus. This system was a 4-layer bidirectional RNN, with 600-dimensional word embeddings and an RNN dimension of 1024, incorporating case features and a vocabulary from 2000 byte-pair encoding merges. The small vocabulary was chosen to reduce the number of out-of-vocabulary tokens resulting from morphology and compounding.

We translated all German the lines that survived the preprocessing step using this MT system. We computed the sentence-level Meteor scores (Denkowski and Lavie, 2011) of the English from the MT system, with the given data as the reference. We simply multiplied positive coverage or cynical scores by their Meteor scores.

7 Application

This section outlines the particulars of the method applied to the given data for this task. First, the Paracrawl data are preprocessed according to the method in §2. This reduces the set of potential lines from 10^8 to 10^7. This reduced set is divided into 100 parts of 10^5 lines for scoring via batch processing.

Five different scoring methods will be considered. The baseline is `cvg-mix`, which uses our coverage metric and sums the coverage score for a small set (τ corresponding to 10^6 total lines) and a large set (τ corresponding to 10^7 total lines). Other scores are variants of this. The treatment `cvg-large` considers only the large set, and `cvg-small` considers only the small set. Meteor scores of translated lines are considered in `cvg-mix-meteor`. Finally, cynical scores are considered in `cyn-mix`.

8 Numerical Results

The results of the WMT 2018 Parallel Filtering Task are given by Bojar et al. (2018). BLEU scores for MT systems built from sets selected via our scoring methods are given in Tables 1-4. We do not consider the development set (newstest2017) in any analysis below, but we include it in the tables for completeness.

Several trends are apparent within our five submissions. First, including the Meteor score is always beneficial for the MT systems trained on smaller sets and rarely detrimental for the systems trained on larger sets. The filtering that includes a translation score, `cvg-mix-meteor`, is our top submission by mean BLEU score for all four MT systems. Second, the filter `cvg-small`, designed for producing a small training set, is poor at producing a large training set. Third, for the small training set there is almost always (test set EMEA in SMT excepted) a benefit from averaging the small training set method and the large training set method. Fourth, the coverage and cynical measures produce very similar results for SMT, but the cynical score is much better for the NMT system that used a small training set. The fact that selection methods differ in performance for SMT and NMT is known (van der Wees et al., 2017), but it is interesting that it is true for our two scoring methods.

Our best filtering method, `cvg-mix-meteor`, scores better than the mean performance of all non-AFRL methods in the task, for every test set and every MT system type. This method exhibits relatively better quality on the smaller (10^6-word) training sets, where it also bests the median. It is especially competitive with the top two systems using the 10^6-word training sets on the test sets Acquis and KDE.

9 Conclusions

We have described a total of five different methods for filtering parallel data, as submitted to the WMT 2018 Parallel Filtering Task. We present numerical results, showing that our methods are especially competitive on certain test sets in the small training set condition.

Our coverage and cynical metrics yield approximately equivalent results in SMT, but the cynical metric is much better for the NMT system built on a small training set. Cynical scoring requires roughly half the computational time burden, so it is sometimes a good choice for NMT.

The ability to specify the size of the selected set is beneficial for our coverage scoring method in

Table 1: BLEU scores of created systems, 10^6-word SMT. Filter mean excludes the development set (newstest2017). The two additional systems listed are the best performing in the task, by mean test set BLEU score. Set score statistics are over the 43 task submissions from other participants.

Filter name	newstest2017	newstest2018	iwslt2017	Acquis	EMEA	Global Vcs	KDE	mean
cvg-mix	20.61	25.22	18.39	17.65	23.64	19.35	21.12	20.89
cvg-small	20.38	25.03	18.04	15.82	24.31	18.99	20.46	20.44
cvg-large	20.39	25.00	17.97	15.81	24.30	18.98	20.43	20.42
cyn-mix	20.52	25.45	18.44	17.22	23.72	19.16	21.10	20.85
cvg-mix-meteor	**21.46**	**26.41**	**19.01**	**17.98**	**24.55**	**19.90**	**22.06**	**21.65**
microsoft	24.04	28.18	20.39	17.13	26.95	21.20	22.76	22.77
rwth-nn-redundant	24.36	28.40	20.60	18.58	26.12	21.37	21.48	22.76
median	21.77	24.91	18.50	16.11	23.99	18.98	21.48	20.66
mean	20.57	23.70	17.30	14.70	22.71	18.14	20.65	19.53
std dev	3.73	4.81	3.93	3.31	3.79	3.40	2.91	3.69

Table 2: BLEU scores of created systems, 10^7-word SMT. Filter mean excludes the development set (newstest2017). The two additional systems listed are the best performing in the task, by mean test set BLEU score. Set score statistics are over the 43 task submissions from other participants.

Filter name	newstest2017	newstest2018	iwslt2017	Acquis	EMEA	Global Vcs	KDE	mean
cvg-mix	23.36	28.61	**21.24**	**18.67**	28.05	21.49	23.68	23.62
cvg-small	21.10	25.88	18.98	18.19	24.06	20.06	20.97	21.36
cvg-large	**23.40**	**28.76**	21.11	18.61	28.04	21.55	23.75	23.64
cyn-mix	23.19	28.28	21.06	18.49	27.94	21.26	23.59	23.44
cvg-mix-meteor	23.33	28.68	21.12	18.66	**28.22**	**21.66**	**23.85**	**23.70**
microsoft	24.48	29.99	21.98	19.43	29.81	22.63	24.67	24.75
prompsit-al	24.50	29.83	21.67	19.71	29.48	22.54	24.72	24.66
median	23.96	29.26	21.52	19.19	28.89	22.15	24.33	24.22
mean	22.85	27.91	20.46	18.18	27.67	21.27	23.65	23.19
std dev	2.91	3.61	2.70	2.55	3.21	2.46	1.99	2.75

Table 3: BLEU scores of created systems, 10^6-word NMT. Filter mean excludes the development set (newstest2017). The two additional systems listed are the best performing in the task, by mean test set BLEU score. Set score statistics are over the 43 task submissions from other participants.

Filter name	newstest2017	newstest2018	iwslt2017	Acquis	EMEA	Global Vcs	KDE	mean
cvg-mix	15.16	18.81	10.36	20.97	25.04	14.06	20.84	18.35
cvg-small	8.11	10.40	5.28	13.20	22.18	8.08	15.40	12.42
cvg-large	8.42	10.70	5.80	13.31	22.27	8.43	15.92	12.74
cyn-mix	22.35	28.06	20.35	21.44	27.29	21.49	22.03	23.44
cvg-mix-meteor	**26.43**	**32.03**	**22.01**	**22.50**	**28.01**	**24.10**	**22.89**	**25.26**
microsoft	27.22	34.32	23.86	20.87	30.75	25.46	25.47	26.79
rwth-nn-redundant	28.08	34.65	23.96	22.01	29.23	25.38	21.50	26.12
median	24.04	29.90	20.53	18.46	25.71	22.42	21.50	23.09
mean	21.21	26.25	18.20	16.07	23.42	19.74	19.07	20.46
std dev	6.93	8.80	6.64	5.75	6.82	6.46	6.41	6.81

Table 4: BLEU scores of created systems, 10^7-word NMT. Filter mean excludes the development set (newstest2017). The two additional systems listed are the best performing in the task, by mean test set BLEU score. Set score statistics are over the 40 task submissions from other participants.

Filter name	newstest2017	newstest2018	iwslt2017	Acquis	EMEA	Global Vcs	KDE	mean
cvg-mix	28.76	36.00	24.81	22.94	32.88	**26.89**	26.13	28.27
cvg-small	17.00	22.33	15.95	19.71	24.31	18.17	16.77	19.54
cvg-large	28.81	35.63	**25.10**	**23.20**	33.06	26.75	26.13	28.31
cyn-mix	28.04	34.82	23.85	22.78	32.91	26.21	25.68	27.71
cvg-mix-meteor	**28.98**	**36.07**	24.79	23.19	**33.15**	26.84	**26.29**	**28.39**
microsoft	31.04	38.39	26.06	24.91	34.68	28.04	28.37	30.07
alibaba-div	30.55	38.02	25.71	25.03	34.65	27.90	28.33	29.94
median	29.47	36.84	25.19	24.17	33.46	27.00	27.44	29.02
mean	26.25	32.72	22.17	21.42	30.63	24.40	25.29	26.11
std dev	7.47	9.50	6.67	6.15	6.77	6.36	5.37	6.80

the small training set conditions, where it yields about the same results for an order of magnitude less computation time. Unfortunately, specifying a desired output set size is not as obvious for cynical scoring.

Inclusion of a translation metric score such as Meteor is beneficial, and the simplistic version given here produced our best system. Introducing of a translation metric score directly in the set-building process would help in avoiding redundancy.

Optimizing the heuristic and empirical prefiltering and preprocessing steps given here could yield substantial benefit. We have doubtlessly removed some beneficial lines in the prefiltering, which excluded up to 90% of the data. In fact, the prefiltering could conceivably be replaced by moving the application of the machine translation system to before scoring, rather than after. Unfortunately this change would cause much more of a computational burden, as every line would need to be translated.

References

Amittai Axelrod. 2017. Cynical selection of language model training data. *Computing Research Repository*, arXiv:1709.02279. Version 1.

Ondřej Bojar, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Matthias Huck, Philipp Koehn, and Christof Monz. 2018. Findings of the 2018 conference on machine translation (WMT18). In *Proceedings of the Third Conference on Machine Translation, Volume 2: Shared Task Papers*, Brussels, Belgium. Association for Computational Linguistics.

Michael Denkowski and Alon Lavie. 2011. Meteor 1.3: Automatic metric for reliable optimization and evaluation of machine translation systems. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 85–91, Edinburgh, Scotland. Association for Computational Linguistics.

Philip Gage. 1994. A new algorithm for data compression. *C Users Journal*, 12:23–38.

Jeremy Gwinnup, Tim Anderson, Grant Erdmann, Katherine Young, Michaeel Kazi, Elizabeth Salesky, and Brian Thompson. 2016. The AFRL-MITLL WMT16 news-translation task systems. In *Proceedings of the First Conference on Machine Translation*, pages 296–302, Berlin, Germany. Association for Computational Linguistics.

Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander Rush. 2017. Opennmt: Open-source toolkit for neural machine translation. In *Proceedings of ACL 2017, System Demonstrations*, pages 67–72. Association for Computational Linguistics.

Dániel Varga, László Németh, Péter Halácsy, András Kornai, Viktor Trón, and Viktor Nagy. 2005. Parallel corpora for medium density languages. In *Proceedings of the Recent Advances in Natural Language Processing RANLP 2005*, pages 590–596, Borovets, Bulgaria.

Marlies van der Wees, Arianna Bisazza, and Christof Monz. 2017. Dynamic data selection for neural machine translation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1400–1410, Copenhagen, Denmark. Association for Computational Linguistics.