Selecting the Best Data Filtering Method for NMT Training

Frederick Bane fbane@transperfect.com
Anna Zaretskaya azaretskaya@transperfect.com
TransPerfect, Barcelona

Abstract

Performance of NMT systems has been proven to depend on the quality of the training data. In this paper we explore different open-source tools that can be used to score the quality of translation pairs, with the goal of obtaining clean corpora for training NMT models. We measure the performance of these tools by correlating their scores with human scores, as well as rank models trained on the resulting filtered datasets in terms of their performance on different test sets and MT performance metrics.

1 Introduction

More and more parallel corpora are available today for MT training (Tiedemann, 2012; Smith et al., 2013). However, when using data from public sources we can never be certain of the data quality, which is extremely important for an MT system's performance (Khayrallah and Koehn, 2018). In a commercial setting like ours, we typically face several data-related challenges. First, we want to be able to use publicly available parallel corpora which are already aligned, such as the OPUS corpus (Tiedemann, 2012). Second, we want to align our customers' translated documents on a sentence level and reliably filter out misaligned or poor quality sentence pairs. And finally, we want to use our customers' translation memories (TMs) and be able to automatically select only the sentences that are relevant for NMT training.

A large part of the data we use for MT training comes from TMs where human translations are stored and are already aligned on a sentence level, which means that our data are generally better in terms of alignment and translation quality than the typical data collected from the web. However, there are other challenges that this type of corpora present for MT engine training. One example of this is that TMs can contain expanded acronyms (the source segment contains an acronym and the target segment contains this acronym together with its expanded version), which can cause hallucinations. That is why in this experiment we focus on the task of cleaning specifically TM data.

We explored different open-source tools that can be used for bilingual data cleaning. Our goal was to choose the one that yields the best results when it comes to MT performance in order to incorporate it into our MT engine training pipeline. As a first step, we randomly selected 5 million sentence pairs from a corpus that contains all our potential training data in five different language directions:

- English-Chinese;
- English-German;
- English-Japanese;

- English-Russian;
- English-Spanish.

These sentences were then scored by four tools:

- Marian Scorer[1] - part of the MarianNMT toolkit, computes negative log likelihood;
- LASER[2] - creates sentence representations in an aligned multilingual vector space;
- MUSE[3] - creates sentence representations in an aligned multilingual vector space;
- XLM-R[4] - creates sentence representations in an aligned multilingual vector space.

As a next step, we selected approximately 100 sentence pairs from each language direction to be scored by professional linguists according to their translation quality. We then correlated the scores produced by each of the tools with the human scores. In addition, we used the human scores to establish a threshold for filtering the data for the MT training, and proceeded to create separate corpora for each language direction using only the sentences with scores above the threshold for that tool. Next, we trained an NMT model with each data set for each language and compared the model performance. Based on these results we make conclusions on whether they are in line with the results we achieved based on the correlation with human scores and which of the tools will be our preferred option for data cleaning.

The remainder of the paper is structured as follows. Section 2 includes an overview of previous related research, Section 3 describes the experimental setup, and in Sections 4 and 5 we discuss the results and the conclusions respectively.

2 Related Research

Collecting and filtering parallel data has been a major topic in MT research. Now it is more relevant than ever since neural MT performance is highly dependent on the size of the training data (Koehn and Knowles, 2017) as well as its quality (Khayrallah and Koehn, 2018).

Most works in this area focus on filtering noisy data collected from the web. One of the earlier methods used an outlier detection algorithm to filter a parallel corpus (Taghipour et al., 2011). The method proposed by Xu and Koehn (2017) is based on generating synthetic noisy data (inadequate and non-fluent translations) and using these data to train a classifier to identify good sentence pairs from a noisy corpus. Cui et al. (2013) propose an unsupervised method to clean bilingual data, which uses a graph-based random walk algorithm and extracts phrase-pair scores to weight the phrase translation probabilities to bias towards more trustworthy ones. The method is based on the observation that better sentence pairs often lead to better phrase extraction and vice versa. Another method proposed by Carpuat et al. (2017) aims to identify semantic differences in translation pairs using cross-lingual textual entailment and additional length-based features.

More recently, a number of new methods were proposed within the shared task on parallel corpus filtering and alignment, which has existed since 2016, although initially it aimed only at collecting parallel document pairs and did not cover the task of sentence alignment (Buck and Koehn, 2016a). In the 2018 edition, the winning system proposed to use neural MT in both directions to score sentence pairs with dual cross-entropy (Junczys-Dowmunt, 2018). One of the winning systems of the 2020 task (Koehn et al., 2020) also used dual cross entropy from

[1] https://marian-nmt.github.io/docs/cmd/marian-scorer/
[2] https://github.com/facebookresearch/LASER
[3] https://github.com/facebookresearch/MUSE
[4] https://arxiv.org/abs/1911.02116

neural MT models trained in both directions but combined it with a number of other features: a bilingual GPT-2 model trained on source-target language pairs as well as monolingual GPT-2 model for each of the languages, and statistical word translation model scores Lu et al. (2020). Another winner of the 2020 task uses an end-to-end classifier that learns to distinguish clean parallel data from misaligned sentence pairs. The model first uses a Transformer model to obtain sentence representations, followed either by a classifier (Siamese network) or additional layers that are fine-tuned (Açarçiçek et al., 2020). Several other recent works use multilingual language models similarly to Lu et al. (2020), such as the 2019 shared task winner LASER (Chaudhary et al., 2019), as well as Lo and Joanis (2020).

Our task of cleaning TM data is, however, different in nature from the task of cleaning noisy data collected from the web. The specific task of cleaning TMs was addressed in the Automatic Translation Memory Cleaning Shared Task organized in 2016 (Barbu et al., 2016). The methods used at the time mostly treated the task as a machine learning classification problem and differ mainly in the sets of features used by the classifier (Ataman et al., 2016; Buck and Koehn, 2016b; Mandorino, 2016; Nahata et al., 2016; Wolff, 2016; Zwahlen et al., 2016).

Our goal is to find out if using multilingual models, which are the basis of many tools used for cleaning noisy corpora, can successfully be applied to our use case of filtering corpora consisting mostly of TM content.

3 Experimental Setup

3.1 Phase 1

In the first phase, we selected five million sentence pairs at random from a large corpus of parallel sentences covering a range of domains for each of five language pairs. The resulting corpora were then scored using the various tools. For LASER, MUSE, and XLM-R, the publicly available models were used. For Marian-scorer, we used our company's existing marian models for the various language directions.

Due to the impracticality of employing human reviewers to score millions of sentence pairs, a smaller corpus of approximately 100 sentence pairs was created for each language, which contained a mix of sentences selected based on different properties (the longest and shortest sentences, the sentences with the most unusual source:target length ratios, and the best and worst scoring sentences as scored by each tool, etc.) and randomly selected sentence pairs.

Professional linguists then reviewed these corpora and assigned a quality score on a scale from 1 to 100 to each translation pair. As translation quality is a subjective concept, special instructions were provided to the linguists that were tailored to our purpose of MT training. For example, linguists were instructed not to penalize spelling mistakes in the source, but to penalize spelling mistakes in the target. Finally, the scores obtained from each tool were compared with the human-assigned scores for each language pair.

The scores obtained from each tool were evaluated in comparison to the "ground truth" human evaluations. For each tool and language pair we calculated the Pearson correlation and root mean squared error (RMSE) between the scores obtained through that tool and the human-assigned scores. We also performed linear regression using the two sequences and calculated the goodness of fit.

3.2 Phase 2

As the relative performance of the tools was mostly consistent across each of the languages (described in greater detail in the Results section), in the second phase we compared only two language pairs, English to German and English to Japanese. We obtained filtered data sets for each tool by removing all sentences with scores below a threshold, which was the equivalent for that tool of a score of 72.5 from the human reviewer, calculated by linear regression. These

Filtering Method	EN→DE	EN→JA
LASER	0.86	0.81
Marian	-1.12	-1.20
MUSE	0.75	0.69
XLM-R	0.86	0.85

Table 1: Score thresholds equivalent to a human-assigned score of 72.5.

Filtering Method	EN→DE	EN→JA
LASER	2707000	2424216
Marian	3425803	3300907
MUSE	3666427	1641008
XLM-R	3168430	2907271
Random	3666427	3300907

Table 2: Number of sentence pairs in each dataset after score-based filtering.

threshold values are shown in Table 1. The value of 72.5 was determined empirically as representing a fair trade-off between the quality of the data and the size of the resulting training set. We also trained models using the full dataset of five million sentence pairs (no filtration), as well as a randomly selected dataset with the same number of segments as the maximum number selected by any of the tools. The number of segments in each dataset is provided in Table 2.

Instead of setting a score threshold, we also considered using the top n sentence pairs as scored by each tool. While this would provide a better direct comparison between the performance of the different models (by removing doubt that performance differences may be attributed to differences in the sizes of the training sets), for our purposes as a translation company, a score threshold made more sense, as this is what would be used in our training process. In future work we plan to experiment with a fixed data set size.

The engines trained on each different dataset were used to translate two test sets of withheld sentence pairs, one in-domain and the other out-of-domain. The in-domain test sets were comprised of 2000 sentences in each language pair drawn from the same distribution as the original five million sentence corpus. The out-of-domain test sets were the 2020 WMT News test sets. The translations were evaluated using the sacreBLEU python package,[5] with default tokenization for the English-German language pair and the mecab tokenizer for the English-Japanese language pair.

These data sets were then used to train a base transformer model for each tool. A baseline engine was also trained for each language pair using all five million sentence pairs (i.e. no data filtering was performed). To isolate the effects of data selection on the performance of the resulting engine, all configurations and hyperparameters were held fixed across all training runs.

4 Results

4.1 Phase 1

The results of of the Pearson correlation and the RMSE calculation are shown in Tables 3 and 4, respectively. Due to differences in the scoring methods, the scores were normalized in the following way prior to calculating the RMSE: 1 – (x/min(x)) for tools using negative log likelihood (where all scores are negative and a score closer to zero is better) and x/max(x) for

[5]https://pypi.org/project/sacrebleu/

Method	ENDE	ENES	ENJA	ENRU	ENZH	Combined
LASER	0.43	0.50	0.52	0.45	0.58	0.52
Marian	0.53	**0.71**	**0.56**	**0.58**	**0.61**	**0.63**
MUSE	**0.61**	0.60	0.48	0.53	0.60	**0.63**
XLM-R	0.47	0.60	0.52	0.50	0.56	0.60

Table 3: Pearson correlation of each method.

Method	ENDE	ENES	ENJA	ENRU	ENZH	Combined
LASER	0.39	0.36	0.34	0.32	0.36	0.35
Marian	0.37	**0.29**	**0.34**	0.33	0.36	0.35
MUSE	**0.32**	**0.29**	0.35	**0.28**	**0.33**	**0.31**
XLM-R	0.38	0.33	0.35	0.32	0.36	0.34

Table 4: RMSE of each method.

others (where all scores are positive and a higher score is better). We also performed linear regression using the two sequences and calculated the goodness of fit. The results of these calculations are shown in Table 5.

The results of the first phase of our experiment show that Marian-scorer and MUSE were the best predictors of the human-assigned scores. In terms of Pearson correlation with human-assigned scores, Marian-scorer was the best in all but the English-German language pair. When examined in terms of the root mean squared error, MUSE was the the best in all but the English-Japanese language pair. After performing linear regression and calculating the goodness of fit for each tool and the human-assigned scores, Marian-scorer was the best in the English-Spanish, English-Japanese, and English-Russian language pairs, and MUSE was best in the English-German and English-Chinese language pairs.

4.2 Phase 2

Of the models trained with a filtered dataset, the Marian-scorer tool showed the best validation scores and best performance on the in-domain test set. In the English-Japanese language pair, this model even out-performed the model trained on all 5 million sentence pairs, despite seeing only around two-thirds as much training data. In the English-German language pair, the model trained with the full dataset achieved the highest score. The validation BLEU and perplexity of each model during the training process are shown in Figures 1 and 2, respectively. The BLEU scores obtained by each model for the in-domain test set are provided in Table 6.

For the out-of-domain (WMT news) test set, the MUSE model performed best on the English-German language pair, while the model trained on the full dataset achieved the highest marks for the English-Japanese language pair. The BLEU scores obtained by each model for the out-of-domain test set are provided in Table 7.

Method	ENDE	ENES	ENJA	ENRU	ENZH	Combined
LASER	0.19	0.30	0.32	0.20	0.35	0.27
Marian	0.32	**0.53**	**0.44**	**0.40**	0.38	0.40
MUSE	**0.42**	0.49	0.31	0.35	**0.44**	**0.40**
XLM-R	0.25	0.43	0.43	0.30	0.37	0.36

Table 5: Goodness of fit of linear regression calculated with each method and human evaluation scores.

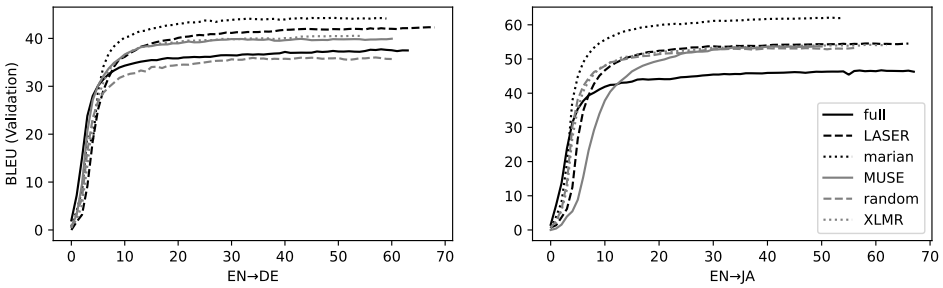

Figure 1: Validation BLEU scores for each model.

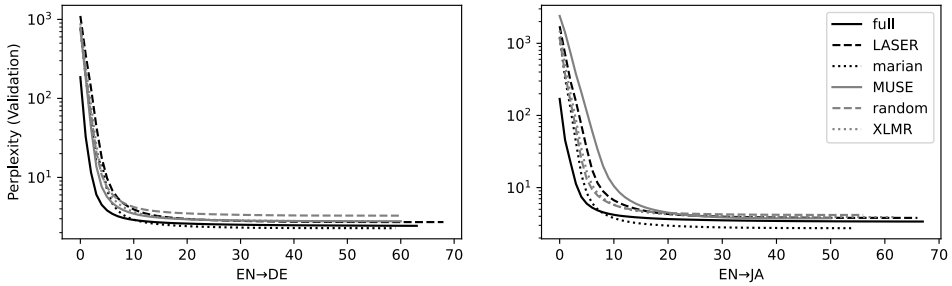

Figure 2: Validation perplexity scores for each model.

5 Conclusions and Future Work

The two phases of this study suggest that using the right method to filter training data can result in similar or improved engine performance despite reducing the total amount of data the engine is exposed to. While training on an unfiltered (larger) dataset typically produced better results in terms of automated metrics, in practice we have observed more hallucinations and unacceptable translations from models trained without any form of data filtering. This is particularly pronounced when there is noise in the target, such as [*sic*] tags or expanded acronyms that do not exist in the source. Among the data filtering methods we tested, our results show that marian-scorer and MUSE produce the best results. However, the limited scope and scale of the study mean that the results are far from generalizable. Future work is still required to confirm or deny the validity of the results on a larger scale.

Filtering Method	**EN→DE**	**EN→JA**
LASER	35.7	36.6
Marian	**36.3**	**37.5***
MUSE	36.0	32.3
XLM-R	35.6	36.3
Random	35.9	36.6
None (Full Dataset)	36.8	37.1

Table 6: SacreBLEU scores for different machine translation models on the in-domain test sets. Note: * indicates a result superior to the model trained on the full dataset.

Filtering Method	EN→DE News	EN→JA News
LASER	18.3	16.9
Marian	17.6	15.9
MUSE	**18.4***	13.6
XLM-R	17.9	**17.1**
Random	17.6	16.6
None (Full Dataset)	18.3	17.7

Table 7: SacreBLEU scores for different machine translation models on the out-of-domain test sets. Note: * indicates a result superior to the model trained on the full dataset.

For example, repeating the second phase of this experiment training three models per tool instead of one and taking the average score would help mitigate potential effects resulting from random weight initializations; human review of the model output would help ensure the automated evaluations in the second stage correspond with human judgment; and obtaining evaluations from more reviewers and calculating inter-rater reliability would help mitigate potential bias resulting from the use of a single reviewer on such a limited sample.

There are also additional practical considerations that call for further investigation. How can an appropriate score threshold be identified in an automated way? Do the appropriate threshold values vary across domains as well as languages? As the models trained on the full data set show some advantages over the models trained on filtered data, could using a two-step training process (training first on all available data, then fine-tuning on a subset of the cleanest data) produce superior models that demonstrate both robustness to input noise and high translation quality?

Beyond the topics enumerated above, our team plans to address several more analytical questions relevant to this line of inquiry in future research. Multiple factors contribute to translation quality, and several different types of errors affecting translation quality exist; are these tools more likely to identify certain error types than others? Do they identify problems with fluency equally as well as adequacy? Are the conclusions drawn in this paper as applicable to the life sciences domain as the leisure and hospitality domain? And what biases are introduced by filtering data in this way? Despite the limitations described here, we hope our work will provide a useful reference for other MT practitioners hoping to identify the best quality sentence pairs for use in their engine training.

References

Açarçiçek, H., Çolakoğlu, T., Aktan Hatipoğlu, P. E., Huang, C. H., and Peng, W. (2020). Filtering noisy parallel corpus using transformers with proxy task learning. In *Proceedings of the Fifth Conference on Machine Translation*, pages 940–946, Online. Association for Computational Linguistics.

Ataman, D., Sabet, M. J., Turchi, M., and Negri, M. (2016). FBK HLT-MT Participation in the 1st Translation Memory Cleaning Shared Task. Online.

Barbu, E., Parra Escartín, C., Bentivogli, L., Negri, M., Turchi, M., Orasan, C., and Federico, M. (2016). The first automatic translation memory cleaning shared task. *Machine Translation*, 30(3):145–166.

Buck, C. and Koehn, P. (2016a). Findings of the WMT 2016 bilingual document alignment shared task. In *Proceedings of the First Conference on Machine Translation: Volume 2,*

Shared Task Papers, pages 554–563, Berlin, Germany. Association for Computational Linguistics.

Buck, C. and Koehn, P. (2016b). UEdin participation in the 1st Translation Memory Cleaning Shared Task. Online.

Carpuat, M., Vyas, Y., and Niu, X. (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.

Chaudhary, V., Tang, Y., Guzmán, F., Schwenk, H., and Koehn, P. (2019). Low-resource corpus filtering using multilingual sentence embeddings. *CoRR*, abs/1906.08885.

Cui, L., Zhang, D., Liu, S., Li, M., and Zhou, M. (2013). Bilingual data cleaning for SMT using graph-based random walk. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 340–345, Sofia, Bulgaria. Association for Computational Linguistics.

Junczys-Dowmunt, M. (2018). Dual conditional cross-entropy filtering of noisy parallel corpora. In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pages 888–895, Belgium, Brussels. Association for Computational Linguistics.

Khayrallah, H. and Koehn, P. (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*, pages 74–83, Melbourne, Australia. Association for Computational Linguistics.

Koehn, P., Chaudhary, V., El-Kishky, A., Goyal, N., Chen, P.-J., and Guzmán, F. (2020). Findings of the WMT 2020 shared task on parallel corpus filtering and alignment. In *Proceedings of the Fifth Conference on Machine Translation*, pages 726–742, Online. Association for Computational Linguistics.

Koehn, P. and Knowles, R. (2017). Six challenges for neural machine translation. In *Proceedings of the First Workshop on Neural Machine Translation*, pages 28–39, Vancouver. Association for Computational Linguistics.

Lo, C.-K. and Joanis, E. (2020). Improving parallel data identification using iteratively refined sentence alignments and bilingual mappings of pre-trained language models. In *Proceedings of the Fifth Conference on Machine Translation*, pages 972–978, Online. Association for Computational Linguistics.

Lu, J., Ge, X., Shi, Y., and Zhang, Y. (2020). Alibaba submission to the WMT20 parallel corpus filtering task. In *Proceedings of the Fifth Conference on Machine Translation*, pages 979–984, Online. Association for Computational Linguistics.

Mandorino, V. (2016). The Lingua Custodia Participation in the NLP4TM2016 TM Cleaning Shared Task. Online.

Nahata, N., Nayak, T., Pal, S., and Kumar Naskar, S. (2016). Rule Based Classifier for Translation Memory Cleaning. Online.

Smith, J. R., Saint-Amand, H., Plamada, M., Koehn, P., Callison-Burch, C., and Lopez, A. (2013). Dirt cheap web-scale parallel text from the Common Crawl. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1374–1383, Sofia, Bulgaria. Association for Computational Linguistics.

Taghipour, K., Khadivi, S., and Xu, J. (2011). Parallel corpus refinement as an outlier detection algorithm. In *MT Summit XIII. Machine Translation Summit (MT Summit-11), 13., September 19-23, Xiamen, China*. NA.

Tiedemann, J. (2012). Parallel data, tools and interfaces in OPUS. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 2214–2218, Istanbul, Turkey. European Language Resources Association (ELRA).

Wolff, F. (2016). Unisa system submission at NLP4TM 2016. Online.

Xu, H. and Koehn, P. (2017). Zipporah: a fast and scalable data cleaning system for noisy web-crawled parallel corpora. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2945–2950, Copenhagen, Denmark. Association for Computational Linguistics.

Zwahlen, A., Carnal, O., and Läubli, S. (2016). Automatic TM Cleaning through MT and POS Tagging: Autodesk's Submission to the NLP4TM 2016 Shared Task. Online.