# Evaluating Corpus Cleanup Methods in the WMT'22 General Translation Task

**Marilena Malli**
Department of Informatics and Telecommunications,
University of Athens
Ilissia, 15784, Greece
mallimariaeleni@gmail.com

**George Tambouratzis**
ILSP, Athena R.C.
6 Artemidos Str.,
Maroussi, 15125, Greece
giorg_t@athenarc.gr

## Abstract

This paper describes the LT'22 team's constrained submission to the WMT General Machine Translation task. NMT transformer-based systems have been implemented using only the WMT'22 released parallel corpora, without using any pre-trained models. Two language pairs have been tackled, namely German to English and German to French. Emphasis was placed on removing the noisy sections of parallel corpora where the degree of parallelism is very limited, for which a publicly-available tool was-used. Comparative results are reported with baseline systems.

## 1 Introduction

This submission presents the contribution of the LT'22 team to the WMT22: General MT Task. It focuses on studying the effectiveness of cleaning tools when these are applied to real-world parallel corpora, to eliminate noisy sections and improve the resulting NMT systems.

Traditionally, parallel corpora are used as the primary data source for machine translation (MT) models. The development of MT has been aided by the availability of extensive parallel corpora. The majority of these data have several areas of reduced parallelism and are usually characterized as imperfect or noisy. The use of noisy data may result in a neural machine translation model being inadequately prepared. Researchers (e.g. (Koehn and Knowles, 2017) (Khayrallah and Koehn, 2018)) have reported that neural machine translation models are much more affected by noisy data than statistical machine translation models.

A number of software packages to implement noise-removal from parallel corpora have been implemented and released to the community. These include publicly available tools such as qe-clean (Denkowski)[1], as well as Zipporah (Xu and Koehn, 2017). (Zariņa et al., 2015) have used a combination of alignment-indicating features to clean corpora. For cleaning large-scale corpora in multilingual setups, a cosine-distance metric has been proposed (Schwenk and Li, 2018). Finally, the suite of the paired Bifixer and Bicleaner software tools (Ramírez-Sánchez et al., 2020) has been proposed for parallel corpora cleaning purposes, with Bifixer implementing restorative cleaning and Bicleaner providing the ability to remove sentences with very low parallelism in the parallel corpus.

For the experiments reported here, two language pairs have been chosen, namely German-to-English (denoted as De-to-En) and German-to-French (denoted as De-to-Fr). Compared with other systems reported in WMT, our NMTs have a couple of identifying features: (1) the use of a fully-constrained setup with respect to WMT'22 rules and (2) the setting of a relatively low threshold to the allowed training epochs, in an effort to comply to a setup with limited computational resources. Whilst our translation systems are not as accurate as they could be if more epochs were allowed, it was decided to adopt an approach that is more realistic when training resources are not unlimited.

To implement the LT'22 participation to the WMT'22 shared task work, we used the following three software packages: (i) the Marian NMT Toolkit (Version: v1.11.5), which was used for the training of the neural machine translation models and (ii) Bifixer and (iii) Bicleaner, which were used in order to correct and clean our data.

Regarding the structure of the paper, in the second section the selection of data on which to train the translation systems is reported. In the third section, the method used to carry out all essential experiments is detailed. In the fourth section, the corpus-cleaning tools are analyzed. In the fifth section the translation systems and their parameters are reported. The sixth section is devoted to details related to experiments. Finally, we review the

[1] https://github.com/mjdenkowski/qe-clean

*Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 335–341
Abu Dhabi, December 7–8, 2022. ©2022 Association for Computational Linguistics

findings of this series of experiments and examine potential future research directions.

## 2 Training Data

Our experiments involve comparing the translation outputs for a series of NMT models for two language pairs: German-to-English (denoted as De-to-En) and German-to-French (denoted as De-to-Fr). It should be noted that for these two language pairs no pretrained models for either Bifixer or Bicleaner are available at the respective repository. All the NMT models reported here are trained using only the parallel training data specified by WMT'22, and no monolingual training data are used. In-training validation has been performed using the development data recommended in WMT'22, whilst for evaluating the trained NMT systems (developed prior to the release of WMT'22 test data), the relevant test data from WMT'20 were used. Moreover, the translations submitted at the WMT22 shared task have been produced using the test data released by WMT'22.

## 3 Methodology

The aim of our experiments has been to evaluate methods for cleaning-up a parallel corpus and to determine if their use leads to MT systems that generate more accurate translations. For each language pair, baseline NMT models have been trained from raw (i.e. unfiltered) parallel training corpora as specified by WMT'22, while the additional NMT models have been trained with corpora subjected to a special cleaning process via the Bifixer and Bicleaner suite (Ramírez-Sánchez et al., 2020). It should be mentioned that the Bicleaner repository[2] doesn't include pre-trained classifiers for the above language pairs; consequently we trained probabilistic dictionaries in order to produce new models. An added benefit of this choice is that no pre-trained model was used to develop our NMT systems, and thus the submitted systems reviewed here are constrained.

The fundamental differences between the NMT models produced are mainly related to the quality and quantity of the training data, while there are no differences in the training parameters or in the setup of the deep neural network architectures (unless otherwise noted in the experimental section). By doing so, it is possible to safely draw conclusions about the amount of computational resources required while also examining and comparing the translation outputs using automatic assessment methods. The following were the driving factors behind the experiments reported here:

- Using the Bifixer/Bicleaner tool in other language pairs for which they have not been used to date, in order to observe their effectiveness in a different real-world scenario.
- The comparison of the results of cleaned as well as raw parallel corpora, automatically as well as manually.
- The study of the effectiveness of translation models produced with limited computing resources (Arase et al., 2021).

## 4 Cleaning Parallel Corpora

### 4.1 Bifixer

The first tool that was used in the translation pipeline is Bifixer, which undertakes to correct some very specific errors that publicly available parallel corpora usually present. Bifixer implements restorative cleaning of imperfect parallel data, working towards fixing the content and preserving unique parallel sentences before filtering out the noise (Ramírez-Sánchez et al., 2020). The steps followed involve empty side removal, character fixing, orthography fixing, re-splitting, duplicates identification. In order to apply Bifixer, we used the recommended default parameter values, without changes, and noted an improvement in the quality of the parallel corpora.

100 random sentence pairs were examined in order to ascertain the effectiveness of Bifixer. After using of the aforementioned tool, fewer noisy data were observed. Better sentence segmentation, fewer typographical errors and fewer extremely short and big sentences were the most notable modifications.

### 4.2 Bicleaner

Continuing the corpus-cleaning process, we proceeded to the next tool, Bicleaner. This tool filters parallel corpora in order to distinguish the noisiest sentences and then remove them to create a cleaner corpus.

In order to use Bicleaner we need to have an already trained classifier. Hence, we initiated the

[2] https://github.com/bitextor/bicleaner-data/releases/tag/v1.5

Bicleaner training process, following the steps described in the official github page [3].

The assembly of a big corpus consisting of about 10 M sentences was our first concern. In order to avoid bias, the sentences were chosen to be different from those used to train Marian NMT models. The training data went through a simple preprocessing which consists of the following steps; detokenization in case of already tokenized corpora; then tokenization of all sentences. As the same tokenization method will be used during Bicleaner running, and the parallel data needs to be aligned in both directions, we used MGIZA++ (Gao and Vogel, 2008). Another software package we used was Moses[4], which is utilized for tokenization as well as the construction of probabilistic dictionaries in combination with MGIZA++. Following this process, two probabilistic dictionaries are constructed, one for each translation direction.

The next step was to create word frequency files. Two folders are needed, for the source language and the target language. To build these two folders we needed two large monolingual corpora. Besides, ideally a very clean corpus of about 100K sentences is required, though such clean data are not readily available. According to the recommendations in github in this case the data can be cleaned by using Bifixer and the Bicleaner Hardrules, which given a parallel corpus, seek to identify evident noisy sentence pairs (Sánchez-Cartagena et al., 2018).

After gathering the aforementioned material, the final step is the training of the Bicleaner. Furthermore, to create the character language models, we utilize the KenLM software package (Heafield, 2011). Via these steps, a trained classifier ready for use in pre-processing was obtained.

## 5 Training the NMT Systems

For training neural machine translation models, we chose the Marian NMT toolkit (Junczys-Dowmunt et al., 2018). Marian was developed to allow rapid training and translation speed, to facilitate the standardization of research work. All the models we trained adopted the architecture of a sequence-to-sequence transformer with 8 attention heads and 6 layers in both the encoder and decoder, thus largely adhering to the standard transformer configuration from (Vaswani et al., 2017). We also decided to set a specific limit to the number of training epochs to avoid lengthy training sessions, aiming to economize as far as possible on valuable computing resources, as per the recent ACL recommendation for efficient computing (Arase et al., 2021).

The transformer is characterized as innovative and uncomplicated (Vaswani et al., 2017). In our experiments, we activated the dropout mechanism, which is a widely adopted regularisation technique in NMT.

When training our NMT systems, we opted to use the SentencePiece tokenizer, which has the ability to train subword models straight from unprocessed data (Kudo and Richardson, 2018). The vocabulary size was set to 32000 and the range for the batch size was from 64 to 100. For the workspace size we used a variable value across our experiments, as the size of the training corpora varied due to the Bicleaner filtering. As suggested by the Marian developers, the workspace was adapted via a number of trial runs at the start of the training process, to maximise the throughput of training sentences per time unit. The other main parameter choices for the transformer models are shown in Table 1. Moreover, the full command used for training is presented in Table 3.

| **Translation Systems** | |
|---|---|
| encoder/decoder depth | 6 |
| beam size | 6 |
| layer normalization | yes |
| exponential smoothing | yes |
| mormalize factor | 0.6 |
| early stopping | 5 |
| transformer dropout | 1 |
| transformer dropout attention | 1 |
| dropout-rnn | 0.2 |
| dropout-src | 0.1 |
| dropout-trg | 0.1 |

Table 1: Main parameters of the transformer architecture used.

## 6 Experiments

### 6.1 Experimental setup

As discussed above, the training data used to implement all the reported experiments were limited to the parallel corpora released for WMT22 for the two language pairs German-French and German-English. For the baseline systems the text

[3] https://github.com/bitextor/bicleaner/wiki/How-to-train-your-Bicleaner

[4] http://www2.statmt.org/moses/

corpora of the respective language pair were used as released, without any pre-processing or noise-removal. Contrariwise, the remaining experiments were carried out using the aforementioned cleaning tools. After applying Bicleaner, the content of the parallel corpus remains the same, however an extra column is added where the parallelism ratings that the classifier assigned to each pair of parallel sentences are stored. Based on this column, sentence pairs rated below a threshold are discarded. Although 0.5 is suggested as a desirable threshold in relevant literature, we chose to examine other thresholds. For this reason, we tested different threshold values within the range from 0.4 to 0.7 to to discover whether changes in this parameter affect the translation accuracy of neural machine translation models. Table 2 provides details regarding the number of sentences that are retained in the parallel corpus following each application of Bicleaner.

| **Corpora(de-en)** | **Sentences** |
|---|---|
| `baseline_corpus.de_en` | ∼2.800.000 |
| `0.7_corpus.de_en` | ∼1.100.000 |
| `0.6_corpus.de_en` | ∼1.500.000 |
| `0.5_corpus.de_en` | ∼1.600.000 |
| `0.4_corpus.de_en` | ∼1.700.000 |
| **Corpora(de-fr)** | **Sentences** |
| `baseline_corpus.de_fr` | ∼18.000.000 |
| `0.7_corpus.de_fr` | ∼7.800.000 |

Table 2: Volume of data before and after the cleaning process.

### 6.2 Computer resources

For the experiments presented here a workstation was used, equipped with a single Nvidia GeForce RTX-3090 GPU, and an Intel i9-11900 CPU with 32 GB of memory. The first two tools were run on the CPU whilst the NMT models training via Marian involved predominantly the GPU. For all experiments where execution times are reported, these times are obtained with the workstation running exclusively the reported process.

### 6.3 Experimental results

At this point, we will review the Marian NMT training results. In Table 4 the BLEU scores during experimental process are presented. Additionally in Table 5, the WMT22 results of the automatic evaluation metrics can be found. Regarding the German-English language pair, we can observe that the baseline system has the highest score. Implementing the cleaning steps and increasing the threshold, the size of training data gets smaller and smaller, as can be seen in Table 1. Since the size of the initial data was not very big, the decrease of the data may well affect the efficiency of the models.

Regarding the German-French language pair, the best score is observed in the model trained on cleaned data. As is mentioned in a related study (Ramírez-Sánchez et al., 2020), it has been observed that the Bifixer/Bicleaner tools work better on big data. In this case the number of the sentences continues to be adequate even after the cleaning process.

## 7 Conclusions and Future Work

In this paper, we have presented our submission to the WMT22: General MT Task. In order to rectify and filter noisy sentences from the corpora recommended by WMT'22, we have applied two cleaning approaches for the parallel corpus. After experimenting with various categorization criteria, we created seven distinct parallel corpora. We discovered that as expected, thoroughly cleaned corpora require fewer computer resources, as a large number of sentences are removed. Additionally, we noticed that differences in the BLEU score across cleaned corpora are relatively small.

Our main submissions to the shared task were two, one for each language pair. Regarding the language pair German to English, the highest quality translation result was obtained by training a transformer model using the raw baseline corpus, and thus the use of Bifixer/Bicleaner did not lead to an improvement. The best result was obtained for the language pair German to French by training a transformer model using the bifixed and bicleaned parallel corpus with a threshold of 0.5.

In upcoming research, the Back Translation technique is planned to be utilized in order to expand the size of the training data, since the size of the sentence pairs is reduced after the cleaning procedure. The translations that emerged from the aforementioned experimental process could be filtered and reused so as to train the NMT system with bigger and cleaner parallel corpora.

In the future, it would be highly interesting to develop probabilistic dictionaries with more than 10 M parallel sentences as well as to train the Bicleaner in more than 100 K parallel sentences. Additionally, we want to use these methods on even more information about the language pairs we previously stated. In order to achieve an even cleaner corpus, it would also be quite fascinating to investigate comparatively other cleaning techniques such as those reported in the introduction.

A final direction for future work would be to use larger models such as the Big Transformer (Vaswani et al., 2017) to see if for this architecture the effect of pre-filtering with Bifixer/Bicleaner will be more marked, and what the trade-off between the improvement in translation quality and the increased training time would be.

**Limitations** One potential limitation of the present work is the relatively limited range and number of Bicleaner thresholds tested, though the values include both the recommended and default values. Another limitation concerns the use of a single architecture, whilst ideally a second architecture (such as the Transformer-Big configuration of (Vaswani et al., 2017)) could be used. Finally, a comparison with other corpus-cleaning methods would be desirable, though such work is beyond the scope of the present work.

**Ethics Statement** The present work is not expected to have any effect on ethical issues and to the authors' best knowledge complies with the ACL Ethics Policy.

**Acknowledgements** The current research was carried out in the context of a master's thesis implemented by the first author in the course "Language Technology" organised by the National and Kapodistrian University of Athens and ILSP/Athena R.C.. We would like to acknowledge the assistance of the WMT'22 organisers and especially Tom Kocmi during the implementation of the experiments and the submission of our results.

## References

Yuki Arase, Phil Blunsom, Mona Diab, Jesse Dodge, Iryna Gurevych, Percy Liang, Colin Raffel, Andreas Rücklé, Roy Schwartz, Noah A. Smith, and Emma Strubell. 2021. Efficient nlp policy document. In *Efficient NLP Policy Document, Association of Computational Linguists, November*.

Qin Gao and Stephan Vogel. 2008. Parallel implementations of word alignment tool. In *Software Engineering, Testing, and Quality Assurance for Natural Language Processing*, pages 49–57, Columbus, Ohio. Association for Computational Linguistics.

Kenneth Heafield. 2011. KenLM: Faster and smaller language model queries. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 187–197, Edinburgh, Scotland. Association for Computational Linguistics.

Marcin Junczys-Dowmunt, Roman Grundkiewicz, Tomasz Dwojak, Hieu Hoang, Kenneth Heafield, Tom Neckermann, Frank Seide, Ulrich Germann, Alham Fikri Aji, Nikolay Bogoychev, André F. T. Martins, and Alexandra Birch. 2018. Marian: Fast neural machine translation in C++. In *Proceedings of ACL 2018, System Demonstrations*, pages 116–121, Melbourne, Australia. Association for Computational Linguistics.

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*, pages 74–83, Melbourne, Australia. Association for Computational Linguistics.

Philipp Koehn and Rebecca Knowles. 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.

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

Gema Ramírez-Sánchez, Jaume Zaragoza-Bernabeu, Marta Bañón, and Sergio Ortiz Rojas. 2020. Bifixer and bicleaner: two open-source tools to clean your parallel data. In *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, pages 291–298, Lisboa, Portugal. European Association for Machine Translation.

Víctor M. Sánchez-Cartagena, Marta Bañón, Sergio Ortiz-Rojas, and Gema Ramírez. 2018. Prompsit's submission to WMT 2018 parallel corpus filtering shared task. In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pages 955–962, Belgium, Brussels. Association for Computational Linguistics.

Holger Schwenk and Xian Li. 2018. A corpus for multilingual document classification in eight languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz

Kaiser, and Illia Polosukhin. 2017. Attention is all you need.

Hainan Xu and Philipp Koehn. 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.

Ieva Zariņa, Pēteris Ņikiforovs, and Raivis Skadiņš. 2015. Word alignment based parallel corpora evaluation and cleaning using machine learning techniques. In *Proceedings of the 18th Annual Conference of the European Association for Machine Translation*, pages 185–192, Antalya, Turkey.

**Translation Systems**

```
∼/marian/build/marian –model modelsname.npz \
- -vocabs modelsname/vocabsname.deen.spm modelsname/vocabsname.spm \
- -type transformer - -transformer-heads 8 - -train-sets ∼/corpus.srl \
∼/corpus.trl - -disp-freq 100 - -mini-batch-fit - -workspace 21000 \
- -layer-normalization - -exponential-smoothing \
- -sentencepiece-alphas 0.2 0 \
- -dim-vocabs 32000 32000 \
- -after-epochs 21 - -dropout-rnn 0.2 - -dropout-src 0.1 - -dropout-trg 0.1 - -valid-metrics cross-entropy \
- -valid-sets ∼/dev.srl ∼/dev.trl - -valid-freq 10000 \
- -beam-size 6 - -normalize=0.6 - -early-stopping 5 \
- -cost-type=ce-mean-words - -max-length 200 - -save-freq 10000 \
- -overwrite - -keep-best - -log ∼/transformer.log \
- -valid-log ∼/transformer_valid.log \
- -enc-depth 6 - -dec-depth 6 - -learn-rate 0.0001 \
- -lr-warmup 8000 - -lr-decay-inv-sqrt 8000 - -lr-report \
- -seed 1 - -label-smoothing 0.1
```

Table 3: An example command used in order to train NMT systems with Marian.

| Data | Cleaning Method | Threshold | BLEU | Training Time |
|---|---|---|---|---|
| System1.de-en | None(raw data) | - | 17.4 | ∼66h |
| System2.de-en | Bifixer/Bicleaner | 0.4 | 22.7 | ∼26h |
| System3.de-en | Bifixer/Bicleaner | 0.5 | 23.2 | ∼26h |
| System4.de-en | Bifixer/Bicleaner | 0.6 | 24.1 | ∼19h |
| System5.de-en | Bifixer/Bicleaner | 0.7 | 23.3 | ∼15h |
| System1.de-fr | None(raw data) | - | 26.3 | ∼92h |
| System2.de-fr | Bifixer/Bicleaner | 0.7 | 27.6 | ∼74h |

Table 4: BLEU scores on WMT20 test during the development process.

| Data | Cleaning Method | Threshold | BLEU | chrF | COMET-A | COMET-B |
|---|---|---|---|---|---|---|
| System1.de-en* | None(raw data) | - | 26.0 | 0.5 | 25.6 | 33.3 |
| System2.de-en | Bifixer/Bicleaner | 0.4 | 24.3 | 0.5 | N/A | N/A |
| System3.de-en | Bifixer/Bicleaner | 0.5 | 25.3 | 0.5 | N/A | N/A |
| System4.de-en | Bifixer/Bicleaner | 0.6 | 24.9 | 0.5 | N/A | N/A |
| System5.de-en | Bifixer/Bicleaner | 0.7 | 24.0 | 0.5 | N/A | N/A |
| System1.de-fr* | None(raw data) | - | 24.4 | 0.5 | N/A | N/A |
| System2.de-fr | Bifixer/Bicleaner | 0.7 | 28.3 | 0.5 | 10.4 | 54.4 |

Table 5: Cleaning method, WMT22 automatic scores and training time for all submitted NMT systems. *Systems defined as primaries.