Three-Layer Denoiser: Denoising Parallel Corpora for NMT Systems

Felipe de A. Florencio [1], Matheus S. de Lacerda[1], Anderson P. Cavalcanti[1], Vitor Rolim[2]

[1]SiDi – Recife – PE – Brazil

[2]Centro de Informática
Universidade Federal de Pernambuco (UFPE) – Recife – PE – Brazil

{fa.florencio, m.lacerda, a.cavalcanti}@sidi.org.br, vbr@cin.ufpe.br

Abstract. *In recent years, the field of Machine Translation has witnessed the emergence and growing popularity of Neural Machine Translation (NMT) systems, especially those constructed using transformer architectures. A critical factor in developing an effective NMT model is not just the volume, but also the quality of data. However, removing noise from parallel corpora, which involves the intricacies of two distinct languages, presents a significant challenge. In this paper, we introduce and assess a method for eliminating such noise, known as the Three-layer Denoiser. The first layer of this process, termed textual normalization, involves data cleaning using predetermined rules. The second layer incorporates a text feature extractor and a binary classifier, while the third layer evaluates the quality of sentence pairs using a pre-trained transformer model. Experimental results, obtained from training various NMT models with both clean and raw data, indicate a rise of up to 2.64 BLEU points in the models trained with sentence pairs that were filtered by the Denoiser.*

1. Introduction

In recent years, Neural Machine Translation (NMT) systems have been established as the state of the art in the field of Machine Translation. Architectures based on Recurrent Neural Networks (RNNs), such as Sequence to Sequence [Sutskever et al. 2014], already have quality gains in relation to Statistical Machine Translation (SMT) systems. In 2017, the proposal for the *Transformers* architecture [Vaswani et al. 2017], based only on attention mechanisms and which eliminates the need for mechanisms of recurrence and convolution, motivated the adoption of NMT systems. In general, transformer models are more accurate, more parallelizable, and require less time to train than RNN-based models.

Although the quality gains presented by current NMT systems, [Khayrallah and Koehn 2018] state that neural models are generally more impaired by noise than statistical models. In their work, the authors explain that the quality of data should be considered in addition to the amount of data. A large amount of noisy data can generate an inaccurate model. [Khayrallah and Koehn 2018] point out some common types of noise, such as the presence of sentences from a third language, pairs of sentences whose target sentence does not match the translation of the source sentence, disordered words, and untranslated sentences. While the approach should be able to detect and eliminate different types of noise, it should also consider the different linguistic characteristics of the language pairs of the corpus.

In this article, we present and evaluate an approach to remove parallel corpus noise for NMT: the Three-layer *Denoiser*. The first layer is named textual normalization and consists of the cleaning and elimination of noisy data through predefined rules. In the second layer, there is a text feature extractor and a binary classifier. The instances are classified into pairs of aligned sentences and pairs of misaligned sentences through a Random Forest estimator that receives the extracted features as input. The third layer scores the quality of sentence pairs through a pre-trained transformer model, and instances with low scores are removed.

The evaluation of the Denoiser consisted of the evaluation of the performance of three different NMT models: (i) one with 5 million unfiltered instances; (ii) one with 5 million instances filtered by *Denoiser*; (iii) and one model with approximately 3.8 million instances filtered by *Denoiser*. The NMT models were evaluated and compared using the *BLEU* [Papineni et al. 2002], *TER* [Snover et al. 2006] and *chrF* [Popović 2015] metrics. The objective was to verify the efficiency of the Three-Layer Denoiser through the gains in translation quality of the models trained with filtered data. The results show an increase of up to 2.64 *BLEU* points in the models trained with sentence pairs filtered by the Denoiser.

To achieve a better understanding of the obtained results, this work was divided into six sections. Section 2 presents the Background, with state-of-the-art and the theoretical basis of the field of noise detection in parallel corpora, as well as the research questions. Section 3 presents the methodology used for the creation and evaluation of the Three-Layer Denoiser, this includes the description of the pipeline and the description of the experiments. Section 4 illustrates the results of the experiments with the evaluation of the trained models. In section 5, the results are discussed with regard to the obtained metrics and the comparison of performance between the models. Section 6 presents the conclusion.

2. Background

2.1. Approaches to Noise Reduction in Parallel Corpus

In the scientific literature, there are different methods for noise detection, elimination, and data selection in Parallel Corpora. One type of approach used for selecting data for Machine Translation (MT) systems is the calculation of entropy. [Axelrod et al. 2011] present the bilingual cross-entropy difference (*bML*) method and explore two other methods: source-side cross-entropy (*Cross-Ent*) and source-side cross-entropy difference (*Moore-Lewis* method) [Moore and Lewis 2010]. The results show that the *Cross-ent* and *bLM* methods provide significant gains in *BLEU* even with the reduction of the training corpus. The *Moore-Lewis* method was able to work almost as well as the reference model with only 35k sentences. The model trained with the *bML* method obtained the highest increase in *BLEU* in relation to the reference model, 1.8 *BLEU* points with a corpus of 35,000 instances. In more recent work, [Junczys-Dowmunt 2018] introduced dual conditional cross-entropy filtering for noisy parallel data. The authors evaluated the method with the WMT2018 (shared task on parallel corpus filtering) and achieved the highest overall scores for the task.

There are approaches that use models based on the transformer architecture for noise detection and elimination. [Lewis et al. 2019] presented a denoising autoencoder called *BART* that combines elements of *BERT* [Devlin et al. 2019] (Bidirectional Transformers) and elements of *GPT* [Radford et al. 2018] (Auto Regressive Transformers). To create the model, the data are corrupted with a function that adds noise, then *BART* is trained for text reconstruction. In a later study, *BART* was used in the context of parallel corpus for NMT. [Liu et al. 2020] used *BART* to create the multilingual approach *mBART*. The authors claim that adding the *mBART* initialization yields performance gains in all configurations except higher resource configurations, including up to 12 *BLEU* points for MT of low-resource languages and more than 5 *BLEU* points for many document-level and unsupervised models.

[Bane and Zaretskaya 2021] evaluated the performance of different data filtering methods for NMT. Benchmarking was performed on the following methods: *LASER* [Chaudhary et al. 2019], *MARIAN Scorer* [Junczys-Dowmunt et al. 2018], *MUSE* [Conneau et al. 2017], and *XLM-R* [Conneau et al. 2019]. The correlation between the human score and the score generated by the methods was calculated and from this, filtering limits were established. The models were trained only with pairs of sentences that had scores above the established limit. In large part of the experiments, *MARIAN Scorer* appeared as the most efficient method of filtering, and the *MUSE* method also presents good results.

This work proposes a noise elimination method that combines three approaches to filter and improve the reliability of the dataset, thus enhancing the quality of the translation model. During the evaluation, we explored the EN-PT language pair, which is little explored in the scientific literature.

2.2. NMT Evaluation

The human evaluation of an MT system can be quite expensive and time-consuming, so some automatic evaluation metrics have been developed. In this work, three metrics were utilized: *BLEU* [Papineni et al. 2002], *TER* [Snover et al. 2006] and *chrF* [Popović 2015].

The *BLEU* (Bilingual Evaluation Understudy) metric was proposed by [Papineni et al. 2002] and it is a widely used metric because of its high correlation with human evaluations. *BLEU* measures how close a sentence generated by an MT is to a reference sentence, the goal being to measure how similar the machine translation is to the human translation. To calculate the score, *BLEU* compares the n-gram of the candidate translation with the n-gram of the reference translation in order to count the number of matches.

Another metric developed to evaluate machine translations is the *TER* (*Translation Error Rate*) proposed by [Snover et al. 2006]. Unlike *BLEU*, *TER* is a translation error rate defined as the minimum number of edits required to change a hypothesis to match exactly one of the references, normalized by the average length of the references.

The *chrF* (character n-gram F-score) is a metric used to assess machine translation quality by measuring the similarity between the reference translation and the machine-generated translation based on character n-grams. It involves tokenizing the translations into n-grams, identifying matching n-grams, calculating precision and recall, and deriving the F-score as the harmonic mean of these values [Popović 2015].

The three metrics compare the sentences generated by an MT system with reference sentences, but each of them calculates different aspects. *BLEU* is an n-gram precision calculation, *TER* is a translation error calculation, and *chrF* is an F-Score calculation with character n-gram.

2.3. Research Questions

As discussed earlier, there are some types of noise commonly found in parallel corpora used to train MT systems. NMT systems are more sensitive to noisy data than SMT systems, so it is important to have noise detection and elimination approaches. We developed the Three-Layer Denoiser, in which each layer uses a different method to detect and eliminate noise. According to [Wang et al. 2018], the quality of an automatic translation model is positively correlated with the quality of the data used for training the model, and filtering noise from the parallel data reduces its negative impact on the translation model. Therefore, in order to verify the validity of this proposal and the effectiveness of noise elimination in creating NMT models, we conducted some experiments to answer the first research question:

> **(RQ1) RESEARCH QUESTION 1:** *Does a Neural Machine Translation model trained with data filtered by the Denoiser have superior quality in relation to a model trained with unfiltered data of the same amount?*

When we apply the Denoiser to a dataset, we are cleaning up noisy data, so the final amount of that dataset will be reduced. Then, the first research question aims to compare two machine translation models trained with Denoiser-filtered data and Denoiser-unfiltered data, keeping the same amount of data for both sets. [Koehn and Knowles 2017] demonstrate in their work the relevance of data volume for training NMT models. However, it is important to evaluate the relationship between data quality and quantity. So we raise a second research question:

> **(RQ2) RESEARCH QUESTION 2**: *Does a Neural Machine Translation model trained with a Denoiser-filtered dataset have superior quality if compared to a model trained with the same dataset without filtering?*

The second research question aims to compare two models trained with a dataset applying the Denoiser and the same dataset without applying the Denoiser. In this case, without taking into account the amount of data. From the answers to the research questions, we can evaluate the quality of the Denoiser by comparing the various trained models with different compositions of the dataset. The answers also allow us to test the hypothesis present in the literature that increasing the size of the training dataset of an MT does not necessarily improve the quality of the model [Wang et al. 2018].

3. Method

In this section, we present how the Three-Layer *Denoiser* operates and describe the Denoiser evaluation experiment. In Figure 1, we illustrate the data flow in the *Denoiser*. In each layer, a different filtering approach is used. In all layers, instances are removed for the purpose of eliminating noisy instances.

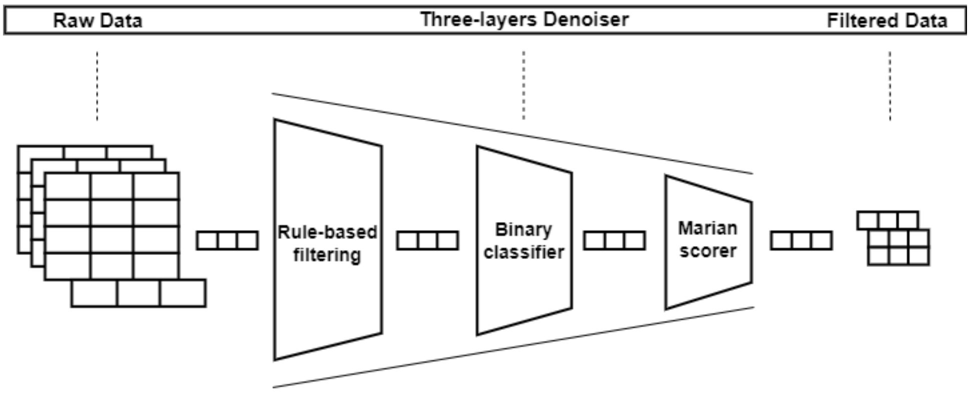

Figure 1. Data stream in Denoiser.

3.1. Rule-Based Filtering

The first layer of data hygiene consists of algorithms and regular expressions that aim at normalizing and removing unwanted sentences or characters from the training dataset. This layer contains 5 steps, where each step aims to remove: (*i*) malformed texts, (*ii*) invalid characters, (*iii*) blank spaces, (*iv*) sentences from other languages and (*v*) European Portuguese.

The first step removes pairs of identical sentences, that is, pairs of sentences where the *source* and *target* are the same. Depending on the number of sentence pairs that have this pattern, it can negatively affect the translation quality of the MT model. The second step contains a series of regular expressions that have been constructed for the purpose of removing unwanted data and patterns. Such patterns are collected from observations and feedback produced by technical translators about random samples of the collected datasets. The third step aims to remove extra spaces at the beginning and end of sentence pairs, as well as double spaces. Generally, such spaces are created due to the extraction of invalid words or characters, or they originate from the dataset itself. The fourth step consists of removing the pairs of sentences that contain characters from the Cyrillic, Chinese, and Arabic writing systems, pictograms, and non-printable or unknown characters.

In order to reduce sentences that may add biases from European Portuguese to the translation model, a fifth pre-processing stage was added, which aims to remove pairs of sentences that contain words exclusive from European Portuguese (pt_EU). The identification of such sentences was made through the intersection between the words present in the sentences and in a dictionary of words exclusive to pt_EU. If the intersection returns a list with a size greater than zero, the sentence is classified as pt_EU. The dictionary contains 197 words of European Portuguese and was prepared and validated by linguists.

3.2. Binary Classification Filtering

The second layer is a binary classifier based on *Random Forest* [Ho 1995] trained using the *Scikit-learn* [Pedregosa et al. 2011] framework. In this layer, the data is processed in two steps: feature extraction and classification of instances based on the features. In the feature extraction step, we selected 88 syntactic, morphological, and semantic metrics to adequately represent the sentences, enabling comparison in terms of translation accuracy.

In the classification step, we used the extracted features to classify the pairs of sentences into two classes, “OK” and “Not OK”. The “OK” class indicates a high similarity between the source sentence and the target sentence, while the “Not OK” class indicates a low similarity between the two sentences. The classifier was trained using

Table 1. Scoring Examples

EN	PT	Score
A beautiful hotel with so much history.	Lindo hotel com tanta história.	-1.032666
This is to the benefit of tourists.	Isso é em benefício dos turistas.	-0.482310
Completely new UI.	totalmente outro eon.	-6.861598

a small set of reliable data where each instance included an English sentence, a Portuguese sentence, and a label. For training the classifier, we randomly selected sentence pairs from the parallel corpora *Opensubtitles* [Lison and Tiedemann 2016] and *CCaligned* [El-Kishky et al. 2020]. These sentence pairs were labeled by two human language experts, resulting in a balanced dataset of 21,176 instances equally distributed across the classes. To ensure data balance, the language experts translated random sentences from the monolingual corpus *BBC* [Sharif 2018], and only the sentence pairs labeled as "OK" were selected.

3.3. Marian Scorer Filtering

The final step of data filtering involves evaluating sentence pairs using the open-source tool *Marian-Scorer* [Junczys-Dowmunt et al. 2018]. This tool is used to assign and quantify scores to the remaining sentence pairs by calculating the Log of Probability per sentence. The evaluation is performed using a pre-trained NMT model, the loss values for each sentence pair are obtained, and the *log* of probabilities are calculated for the entire set of pairs. The result of each sentence pair is a negative value that can vary from 0 to $-\infty$, where values closer to zero mean a low *loss*, that is, proximity to the word estimate of the model. The parameters and weights, along with the previously trained model, are generated using the *Marian NMT* [Junczys-Dowmunt et al. 2018] toolkit.

The data remaining from the previous filtering layer (Binary Classifier) is separated into two plain text files. Each file contains the *source* and *target* sentences, with English and Portuguese used, respectively. The *Marian-Scorer* requires a previously trained NMT model, thus, for the experiments performed in this data filtering layer, a translation model trained with 62 million sentence pairs was used, which presented a *BLEU* of 47.50 in the PT $\rightarrow$ EN direction and 44.30 for the EN $\rightarrow$ PT direction in the test dataset with 19,420 pairs of sentences from different domains.

The *Marian-Scorer* assigns a negative value to each sentence pair found in the *source* and *target* files. This value corresponds to the negative logarithm of probability, indicating the probability of misalignment, as we can observe in Table 1. Filtering is performed by removing sentence pairs with low scores. However, since the score is relative to the NMT model used, a threshold must be defined to determine which sentence pairs will be removed.

3.4. Experiment Description

To evaluate the performance of the Three-Layer Denoiser, three different MT models were trained and evaluated (MT1, MT2, and MT3). MT1 serves as the reference model trained with unfiltered data (raw data), while MT2 and MT3 were trained using data filtered by the *Denoiser*. All models are based on *Transformers* architecture and have been trained bidirectionally with the Portuguese-English language pair using the *Marian NMT* framework [Junczys-Dowmunt et al. 2018].

As mentioned earlier, MT1 was trained with 1 million pairs of sentences from five different corpora, totaling 5 million instances (**Dataset 1**). This training set underwent a single filtering process to remove invalid instances, including rows with *NaN* values, non-printable characters, and empty instances.

For MT2, all data used were processed by the Denoiser, and after processing the datasets, 1 million pairs of sentences from five different corpora were selected, resulting in a total of 5 million instances (**Dataset 2**). As for MT3, Dataset 1 was used for training, but it underwent processing by the Denoiser, reducing the number of instances from 5 million to 3,856,356. The evaluation process for MT2 and MT3 involves training the models with their respective datasets, testing them with two different datasets, and calculating the *BLEU*, *TER*, and *chrF* scores.

The training dataset was created from five parallel corpora available at the *Opus* platform [Tiedemann 2012]. The corpora are: textitCCAligned [El-Kishky et al. 2020], *CCMatrix* [Tiedemann 2012], *Opensubtitles* [Lison and Tiedemann 2016], *Paracrawl* [Bañón et al. 2020] and *SciELO* [Soares et al. 2018]. To evaluate the models, two test datasets were employed: one with unfiltered sentence pairs (Raw Dataset) and another with sentence pairs filtered by the *Denoiser* (Clean Dataset). The pairs of sentences used to create the two datasets were extracted from the same parallel corpora, ensuring the same amount and proportion of data. However, for the creation of the Clean Dataset, the original parallel corpora were filtered prior to the extraction of the sentence pairs. The test dataset contains 19,123 pairs of sentences from different subject domains (biomedical, IT, literature, conversation, and science).

4. Results

This section presents the results of the *Denoiser* Benchmarking in order to answer the two main research questions. In addition, the results of the investigation experiments for the creation of the *Binary Classifier* and *Marian Score* are presented.

4.1. Binary Classifier

To evaluate the effectiveness of the denoising approach in machine translation, we employed the Random Forest and XGBoost classifiers. These classifiers were selected based on their proven success in various natural language processing tasks, including text classification, sentiment analysis, and machine translation evaluation [Chen and Guestrin 2016, Galar et al. 2011]. Both were trained and tested with the same balanced dataset that has 21,176 sentence pairs labeled “OK”and “Not OK”. With respect to the division of the training and test sets, 14,823 instances (70%) were used for training the models, and 6,353 instances (30%) were used for testing. The result of the evaluation of the algorithms used to train the classifier is shown in Table 2.

In addition to the classification into “OK” and “Not OK” classes, the test instances were further classified as “High Score” or “Low Score”. Instances with *Prediction Class Probability* equal to or greater than the set threshold were categorized as “High Score”, while instances with lower probabilities were classified as “Low Score”. The objective is to find the optimal threshold value to confidently identify and remove noisy sentence

Table 2. Algorithms comparison.

Threshold	Algorithm	Accuracy	Precision	Recall	F1	AUC
0.5	Random Forest	0.706	0.707	0.700	0.703	0.788
	XGBoost	0.718	0.712	0.728	0.720	0.791
0.725	Random Forest	**0.915**	**0.948**	**0.878**	**0.912**	**0.948**
	XGBoost	0.841	0.870	0.790	0.828	0.890

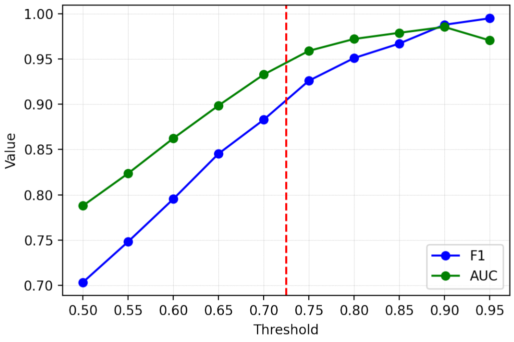

Figure 2. Model Evaluation - Random forest.

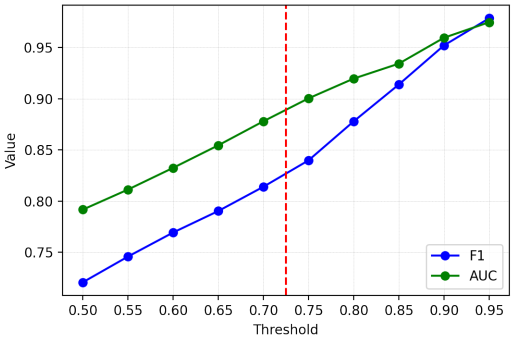

Figure 3. Model Evaluation - XGBoost.

pairs. Specifically, instances are removed when classified as “Not OK” and having a *Prediction Class Probability* equal to or greater than the threshold value.

Both algorithms exhibit similar overall performance, disregarding their reliability in classification. However, when considering instances with high *Prediction Class Probability* for the “OK” or “Not OK” classes, the *Random Forest* model outperforms the *XGBoost* model. Notably, the evaluation emphasizes *F1* and *AUC* metrics.

In Figures 2, 3, 4 and 5, it is possible to analyze the performance according to the *Prediction Class Probability* threshold. The red dashed line represents the selected threshold for the template used in the *Denoiser*. In Figure 2 it is possible to observe that the *Random Forest* model has greater performance variations in the first threshold ranges. With threshold values above 0.7, the performance of the model starts to have considerably smaller variations. The *XGBoost* model, whose performance is illustrated in Figure 3, has minor variations in performance across all threshold ranges.

The *Random Forest* model achieves higher *AUC* and *F1* scores for higher *Prediction Class Probability* values. Based on this observation, a threshold value of 0.725 was selected as it corresponds to *AUC* and *F1* values greater than 0.90. Furthermore, this threshold range includes a substantial number of “High Score” instances that can be effectively removed with a minimal error rate.

4.2. Marian Scorer

As demonstrated in Section 3.3, Marian-Scorer utilizes a pre-trained NMT model to calculate the logarithm of probabilities for each sentence pair. However, due to the nature of the probability scores generated by Marian-Scorer, a decision threshold needed to be

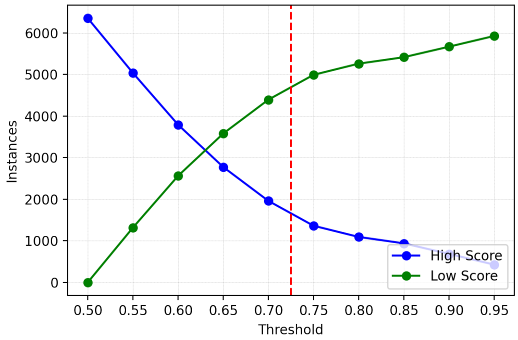

Figure 4. Number of Instances per Threshold - Random Forest.

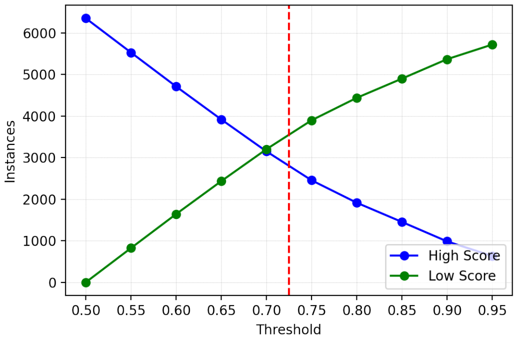

Figure 5. Number of Instances per Threshold - XGBoost.

Table 3. Labeled Dataset

EN	PT	Label
forbanhold.com and 1=1	cbdoil50.com	Incorrect
English to Croatian Translation	Inglês to Croata Tradução	Incorrect
For how long you have that problem ?	Por quanto tempo você tem esse problema ?	Correct

defined to exclude potentially poor-quality sentences. To determine the threshold, scores were computed for a dataset consisting of 33,119 sentence pairs in the English-Portuguese (EN-PT) language pair. These pairs were labeled by language experts to indicate whether the translation was correct or not, as shown in Table 3.

To calculate the best decision threshold that maximizes the number of correct sentence pairs and reduces the number of incorrect pairs, it was necessary to define a reduced sample space, due to the variance of the *scores* being between $-\infty$ and 0. To this end, the metrics of Recall, Precision, Accuracy, and F1 for the predictions were calculated using the thresholds defined in the minimum value range of (-9.59) and the 1st quartile (-1.56) of the *scores* of the correct sentences. This range covers the intersection between the *scores* of the data labeled as incorrect and the data labeled as correct.

Figure 6 illustrates the chosen threshold range (represented by red dashed lines)

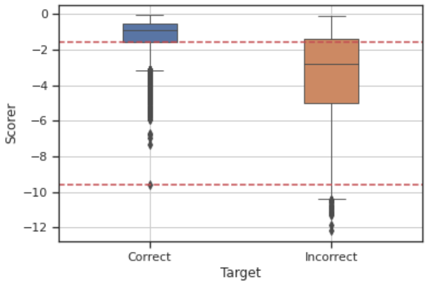

Figure 6. Threshold by labels.

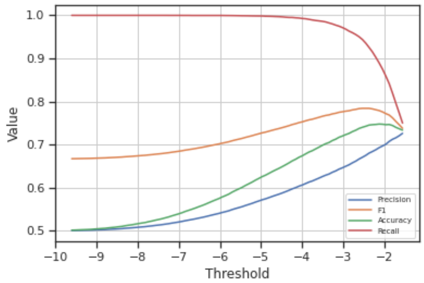

Figure 7. Evaluation Metrics.

for investigation. Once the limits of the sample space were determined, it was necessary to divide this space into *N* parts, where *N* is a heuristic value dependent on the interval size and the sample type. For this experiment, *1* was defined as 120. The behavior of the resulting metrics for each of the 120 thresholds can be observed in Figure 7. It can be seen that Accuracy, F1, and Recall metrics tend to reach their highest peaks between -3 and -2. The three best threshold values and their respective metrics are presented in descending order in Table 4 with the first threshold (-2.45) being the one used in the current pipeline.

Table 4. Metrics of each threshold

Threshold	Precision	F1-Score	Accuracy	Recall
-2.455905	68%	78%	74%	94%
-2.537068	67%	78%	74%	94%
-2.374742	68%	78%	75%	93%

4.3. Pipeline Benchmarking

The five parallel corpora used as the source of sentence pairs for training were completely filtered. Table 5 presents information about the original size of the parallel corpora, the size after filtering, and the percentage of the ratio between the size of the corpus after filtering and the original size of the corpus.

Table 6 presents the results of the training and evaluation of the three models (MT1, MT2 and MT3). Each model was subjected to four different tests: two different datasets (Raw Dataset and Clean Dataset) varying the source language and the target language (EN→PT and PT→EN). The MT 2 model was trained with 5 million sentences in each direction filtered by the *Denoiser*, presented the highest *BLEU* in all tests. The MT3 model was trained with a reduced number of sentences filtered by the *Denoiser*, presenting the second best performance. The MT1 reference model underperformed the other two models in all tests.

MT2 outperformed MT1 in all tests. The test performed with the Clean Dataset in the PT→EN direction had the greatest difference in performance between the models. For the EN→PT direction in the Clean Dataset, there was a difference of +2.64 points and for the PT→EN direction, there was a difference of +1.85 points between MT2 and MT1. The test performed with the Raw Dataset in the PT→EN direction had the smallest difference in performance between MT2 and MT1, with +1.52. The test which shows the smallest difference of *chrF* between the models is the one with the Raw Dataset in the EN→PT direction, the difference being +1.21 points.

Table 5. Cleaning of Training Datasets

Dataset	Original Size	Filtered Data Size	Percentage
CCMatrix	173,743,166	108,377,096	62.38%
Opensubtitles	33,222,606	17,264,469	51.97%
CCAligned	13,650,321	9,823,117	71,96%
Paracrawl	102,633,504	5,670,961	5,52%
SciELO	3,084,830	2,498,613	80,99%

Table 6. Performance Evaluation

Test Dataset	Model	EN → PT			PT → EN		
		BLEU	**TER**	**chrF**	**BLEU**	**TER**	**chrF**
Raw	**MT1**	38.50	52.00	62.13	45.77	45.49	65.98
	MT2	**40.58**	**50.49**	**63.34**	**47.29**	**44.11**	**67.30**
	MT3	39.74	51.43	62.55	46.22	45.03	66.68
Clean	**MT1**	41.92	48.06	65.38	47.26	43.57	67.98
	MT2	**44.56**	**45.88**	**67.02**	**49.11**	**41.78**	**69.61**
	MT3	43.74	46.76	66.28	48.43	42.32	69.04

MT3 outperformed MT1 in all tests. The test performed with the Clean Dataset in the EN→PT direction had the greatest difference in performance between the models in relation to the *BLEU* and *TER* scores. The difference in *BLEU* between MT3 and MT1 is +1.82, and the difference in *TER* is -1.3. The test that showed the greatest difference of *chrF* between the models is the one with the Clean Dataset in the PT→EN direction, and the difference is +0.9 points. The test performed with the Raw Dataset in the PT→EN direction had the smallest difference in performance between the models in relation to the *BLEU* and *TER* scores. The difference in *BLEU* between MT3 and MT1 is +0.45, and the difference in *TER* is -0.46. The test with the smallest difference of *chrF* between the models is the one with the Raw Dataset in the EN→PT direction, the difference being +0.42 points.

The *Denoiser* has shown significant improvements in translation quality, considering three different metrics that calculate different aspects of NMT translation quality. The increase in *BLEU* indicates an improvement in n-gram accuracy, the reduction in *TER* indicates a reduction in translation errors, and the increase in *chrF* indicates an increase in n-gram *F-Score* at the character level.

5. Discussion

5.1. Research Question 1

In this subsection, we aim to address the research question **RQ1**. This question focuses on the quality of the NMT training data and seeks to validate [Wang et al. 2018] while evaluating the efficiency of our noise elimination method.

To evaluate the performance of the three models, we developed a methodology encompassing four different contexts. Two datasets were used to mitigate possible threats to the experiment's validity. The sentence pairs used in the test datasets were taken from parallel corpora susceptible to noise; therefore, there is a test dataset with sentence pairs filtered by the *Denoiser*. However, it is possible that the *Denoiser* biases MT2 and, consequently, favors MT2 in comparison with the MT1 model using a dataset filtered by *Denoiser*, so there is a test dataset with pairs of sentences not filtered by *Denoiser*.

MT2 consistently outperformed MT1 in all tests, demonstrating the effectiveness of the Denoiser's noise elimination across various data domains and supporting [Wang et al. 2018]. MT2's gains in *BLEU* in relation to MT1 ranged from +1.52 to +2.64. In addition to the *BLEU* metric, used in most works related to *NMT*, we used the *TER* and *chrF* metrics. The MT2's gains in *chrF* relative to MT1 ranged from +1.21 to +1.64 and the *TER* reductions ranged from -1.38 to -2.18. The MT2 model has considerably higher quality than MT1 in different aspects.

While subsection 2.1 highlights diverse *BLEU* gains reported in related works, it is challenging to compare these gains with our method directly. Nevertheless, the evaluation experiments involving the MT1 model (reference model) and the MT2 model (trained with filtered data) clearly demonstrate that reducing noise in the training dataset significantly enhances the quality of the NMT model. Hence, the Neural Machine Translation model trained with data filtered by the Denoiser outperforms a model trained with the same amount of unfiltered data in terms of quality.

5.2. Research Question 2

[Koehn and Knowles 2017] emphasizes the significance of having a large volume of training data for NMT and demonstrates that the quality of the NMT system tends to improve as the amount of data increases. Through the answer to **RQ2**, we aim to verify the validity of the statements presented in the work of [Koehn and Knowles 2017] and, in parallel, evaluate the performance of *Denoiser* in a scenario in which the model is trained with a smaller amount of data and less noise.

To address RQ2, we examined the performance of the MT3 model compared to the MT1 and MT2 models. The evaluation experiments involving the MT1 model and the MT3 model (trained with a reduced amount of filtered data) revealed that reducing the number of noisy sentence pairs in the training dataset significantly enhances the quality of the trained NMT model. Consequently, a Neural Machine Translation model trained with a dataset filtered by the Denoiser outperforms a model trained with the same dataset but without any filtering. MT3 consistently underperformed MT2 in all tests, supporting the assertion made by [Koehn and Knowles 2017]. In a scenario where training datasets have similar noise levels, the data volume significantly impacts the MT model's quality.

In conclusion, the *Denoiser* effectively improves the performance of NMT models, even when noise elimination reduces the number of instances available for training. This demonstrates an increase in data quality. However, it is important to note that greater availability of high-quality training data can further enhance the performance of the NMT model.

6. Conclusion

In this work, we introduced the Three-layer *Denoiser*, a filtering pipeline designed to eliminate noise in parallel corpora. The pipeline comprises three filtering stages: a rule-based noise eliminator, a binary classifier based on *Random Forest*, and a scorer based on Transformers using the *Marian Scorer* tool.

Three bidirectional EN-PT MT models were trained and evaluated in different contexts. The evaluation shows that an MT model trained with data filtered by the *Denoiser* has better translation quality than an MT trained by data that was not filtered by the *Denoiser*, even when there is a smaller amount of data. During the experiments, the *Denoiser* provided increases in *BLEU* between 0.45 and 2.64.The experiments also show that the quality and quantity of data impact the performance of the trained MT model, and there needs to be a balance between the two factors.

The combination of the three layers allows the elimination of noise even when there is little reliable data available due to the first two layers: the Rule-based method, which does not use Machine Learning models and the Binary Classifier. When there is a large number of sentence pairs available, the third layer of the Denoiser can be used, the *Marian Scorer*.

There are some aspects that should be highlighted in the evaluation method of the *Denoiser* developed in this work: the inclusion of sentence pairs from different domains in the same test dataset, the use of the Portuguese-English language pair, and the investigation of the relation between quantity and quality. There are also limitations that must be considered: the volume of data used for training, the lack of an evaluation with other language pairs, and the lack of an evaluation of the impact of each *Denoiser* layer in training the *NMT*. Future work on the *Denoiser* should involve testing it with different language pairs and larger datasets. Assessing each layer independently is crucial to understand their impact on noise elimination and NMT quality.

Acknowledgements

The results presented in this paper Florencio et al. 2023 have been developed as part of a project between SiDi, financed by Samsung Eletrônica da Amazonia Ltda., under the auspices of the Brazilian Federal Law of Informatics no. 8248/91.

References

Axelrod, A., He, X., and Gao, J. (2011). Domain adaptation via pseudo in-domain data selection. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 355–362, Edinburgh, Scotland, UK. Association for Computational Linguistics.

Bane, F. and Zaretskaya, A. (2021). Selecting the best data filtering method for NMT training. In *Proceedings of Machine Translation Summit XVIII: Users and Providers Track*, pages 89–97, Virtual. Association for Machine Translation in the Americas.

Bañón, M., Chen, P., Haddow, B., Heafield, K., Hoang, H., Esplà-Gomis, M., Forcada, M. L., Kamran, A., Kirefu, F., Koehn, P., Ortiz Rojas, S., Pla Sempere, L., Ramírez-Sánchez, G., Sarrías, E., Strelec, M., Thompson, B., Waites, W., Wiggins, D., and Zaragoza, J. (2020). ParaCrawl: Web-scale acquisition of parallel corpora. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4555–4567, Online. Association for Computational Linguistics.

Chaudhary, V., Tang, Y., Guzmán, F., Schwenk, H., and Koehn, P. (2019). Low-resource corpus filtering using multilingual sentence embeddings. In *Proceedings of the Fourth Conference on Machine Translation (Volume 3: Shared Task Papers, Day 2)*, pages 261–266, Florence, Italy. Association for Computational Linguistics.

Chen, T. and Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, pages 785–794, New York, NY, USA. ACM.

Conneau, A., Khandelwal, K., Goyal, N., Chaudhary, V., Wenzek, G., Guzmán, F., Grave, E., Ott, M., Zettlemoyer, L., and Stoyanov, V. (2019). Unsupervised cross-lingual representation learning at scale. *CoRR*, abs/1911.02116.

Conneau, A., Lample, G., Ranzato, M., Denoyer, L., and Jégou, H. (2017). Word translation without parallel data. *arXiv preprint arXiv:1710.04087*.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

El-Kishky, A., Chaudhary, V., Guzmán, F., and Koehn, P. (2020). CCAligned: A massive collection of cross-lingual web-document pairs. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP 2020)*.

Galar, M., Fernandez, A., Barrenechea, E., Bustince, H., and Herrera, F. (2011). A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(4):463–484.

Ho, T. K. (1995). Random decision forests. In *Proceedings of 3rd International Conference on Document Analysis and Recognition*, volume 1, pages 278–282 vol.1.

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.

Junczys-Dowmunt, M., Grundkiewicz, R., Dwojak, T., Hoang, H., Heafield, K., Neckermann, T., Seide, F., Germann, U., Fikri Aji, A., Bogoychev, N., Martins, A. F. T., and Birch, A. (2018). Marian: Fast neural machine translation in C++. In *Proceedings of ACL 2018, System Demonstrations*, pages 116–121, Melbourne, Australia. 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. 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.

Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., and Zettlemoyer, L. (2019). Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Lison, P. and Tiedemann, J. (2016). OpenSubtitles2016: Extracting large parallel corpora from movie and TV subtitles. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 923–929, Portorož, Slovenia. European Language Resources Association (ELRA).

Liu, Y., Gu, J., Goyal, N., Li, X., Edunov, S., Ghazvininejad, M., Lewis, M., and Zettlemoyer, L. (2020). Multilingual Denoising Pre-training for Neural Machine Translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Moore, R. C. and Lewis, W. (2010). Intelligent selection of language model training data. In *Proceedings of the ACL 2010 Conference Short Papers*, pages 220–224, Uppsala, Sweden. Association for Computational Linguistics.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, page 311–318, USA. Association for Computational Linguistics.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Popović, M. (2015). chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395, Lisbon, Portugal. Association for Computational Linguistics.

Radford, A., Narasimhan, K., Salimans, T., and Sutskever, I. (2018). Improving language understanding by generative pre-training.

Sharif, P. (2018). Bbc news summary. Last accessed 14 Jan 2022.

Snover, M., Dorr, B., Schwartz, R., Micciulla, L., and Makhoul, J. (2006). A study of translation edit rate with targeted human annotation. In *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas: Technical Papers*, pages 223–231, Cambridge, Massachusetts, USA. Association for Machine Translation in the Americas.

Soares, F., Moreira, V., and Becker, K. (2018). A large parallel corpus of full-text scientific articles. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018).*

Sutskever, I., Vinyals, O., and Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, page 3104–3112, Cambridge, MA, USA. MIT Press.

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).

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., and Polosukhin, I. (2017). Attention is all you need. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Wang, W., Watanabe, T., Hughes, M., Nakagawa, T., and Chelba, C. (2018). Denoising neural machine translation training with trusted data and online data selection. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 133–143, Brussels, Belgium. Association for Computational Linguistics.