Effect of Noise in Neural Machine Translation September 2022

Does Noise Matter when Training Neural Machine Translation Engines?

Author: John Handley Derecho

Advisors: Nora Aranberri

hap/lap

Hizkuntzaren Azterketa eta Prozesamendua
Language Analysis and Processing

Final Thesis

September 2022

Departments: Computer Systems and Languages, Computational Architectures and Technologies, Computational Science and Artificial Intelligence, Basque Language and Communication, Communications Engineer.

Laburpena

Ikertzaileek aurrerapen nabarmenak egin dituzte itzultzaile automatiko neuronalen alorrean lehen aldiz aurkeztu zirenetik. Izan ere, lan hau idazterakoan, Google Scholarren itzultzaile horien gaineko bilaketa bat eginez gero, milaka artikulu bil zitezkeen. Hala ere, ez dirudi ikertzaile askok garrantzi handia ematen diotenik ikerketa horiek egiteko erabiltzen dituzten datuen kalitateari. Lan honetan gai hori azpimarratzea espero dugu, eta honako hau frogatzen duten emaitzak lortzea: itzultzaile neuronaletan erabiltzen den testu kantitatea bezain garrantzitsua da horren kalitatea. Horretarako, hainbat garbiketa-metodo aztertu ditugu, aukeratutako corpusean aplikatu, eta emaitzak automatikoki zein eskuz alderatu ditugu. Emaitzek adierazten dute itzultzaileen kalitatea hobetu egiten dela esaldi-luzera, eta sorburuko eta helburuko segmentuen arteko karaktere-aldea mugatuta; berdin gertatzen da atzerriko hizkuntzetako hitzak eta itzuli gabekoak dituzten segmentuak corpusetik kenduta. Hala ere, bai automatikoki egindako ebaluazioak bai eskuz egindakoak frogatu dute garbiketa-teknika guztiak aplikatuz gero, itzultzailearen emaitza nahasgarria dela eta, inoiz, ez duela jatorrizko testuan ageri den informazio osoa jasotzen.

Abstract

Researchers have reached tremendous advances in the field of Neural Machine Translation (NMT) since it was first introduced. Up to the date of writing this thesis, searching for *Neural Machine Translation* in Google Scholar triggers hundreds of thousands of papers. However, not a lot of researchers seem to attribute much importance to the quality of the data they are using to carry out these studies. In this paper, we bring more awareness to this topic and provide tangible results that prove the following statement: quality is just as important as quantity in terms of training data when referring to NMT engines. To this end, we explore different cleaning methods for our chosen corpus and compare the results of these methods, both manually and automatically. The results prove that limiting segment length and the character difference between source and target segments, and removing untranslated and foreign language segments enhance the translations of the trained engines. Nonetheless, automatic evaluation metrics and a manual assessment agree that when all cleaning methods are applied, the outcome is confusing and, at times, lacks information from the source segment.

Keywords: neural machine translation, machine translation, noise, text quality, Spanish, English

Contents

List of Tables

1 Introduction

From the earliest days of civilisation, there has been a notable need for translation. This necessity has but grown widely; nowadays, multilingual encounters are quite common, rooted in the ever growing globalisation, a widespread multiculturalism and the internationalisation of information as well as communication. With the help of the continuously increasing advances in Natural Language Processing (NLP), a new methodology joined the translation process: Machine Translation (MT).

Experts are not yet certain of how far back dates the idea of MT, though strong references can be found from the seventeenth century onwards (Hutchins, 1995; Schwartz, 2018). Back then, the thought of a communication mechanism that solved the translation of natural languages was unfathomable. However, as some MT systems prove nowadays, this field of investigation has reached very far and we can now proclaim successful and, not only understandable, but also creative results.

Nevertheless, NLP is a challenging field all in itself: using computers to process and automatise natural language is no easy task, and even less so given all its nuances, subtleties and overtones as well as the consideration of context. But, computers are not alone in the hardships of language learning; the U.S. Foreign Service Institute separates a variety of languages into four different categories increasing in difficulty, measured by the average amount of time a native English speaker would need to reach a "Professional Working Proficiency" level of the given language (FSI, 2022). Category I Languages, for instance, include Spanish, Italian, Norwegian and French, which are estimated to take between 600 and 750 class hours to reach that proficiency level. Parallel to time, we find quality; has anyone ever corrected your pronunciation on a certain word, or let you know that the way you positioned words in a sentence was wrong? Most of us find unlearning wrong language habits somehow complicated, just like eradicating an old unhealthy habit, but it is very well doable. As is with us, we hypothesised that our fellow computers also need the equivalent of long hours of work (which is translated to the large amounts of text) and corpora of pristine quality. We wondered, then, precisely how influential training data really is in NLP, specifically in Neural Machine Translation (NMT).

1.1 Motivation and goals

Neural Machine Translation (NMT) is a task that requires large amounts of high quality parallel corpora. Low resourced NMT engines tend to yield a much lower BLEU score; the more data that an NMT is fed, the steeper its learning curve and the higher the BLEU score (Koehn and Knowles, 2017). As such, NMT generally proves a better ability at exploiting increasing amounts of data.

Moreover, NMT has been proven to be sensitive to noise (Khayrallah and Koehn, 2018), notably when content from either side of parallel corpora is untranslated, but also due to misaligned sentences, misordered words, foreign languages instead of the expected source or target languages, and segments that are too short.

As widespread as these facts are in the world of NLP, we still witness a slight disregard

for quality regarding the training material and a huge focus on quantity. To make matters worse, more often than not we find that NMT papers barely highlight any information regarding the corpora preprocessing, with no more than a paragraph explaining what steps were taken. We find that the *normalization* and *tokenization* are often mentioned (Bei et al., 2019; Bandyopadhyay et al., 2021; Koszowski et al., 2021), and, at times, we can observe the use of language detection to remove foreign language segments (Hendy et al., 2021). Some researchers even go down the route of increasing the amount of noise by adding artificial spelling mistakes or grammatical errors (Popel, 2020) in order to provide the model with a more realistic dataset.

We started our research with one main goal: figuring out a way to prove that the status of the data fed to a Neural Machine Translation (NMT) engine affects the outcome quality, whether for better or for worse. As we indicated in Section 1, our hunch leads towards the belief that if the data can be made of an objectively higher quality, we can achieve more optimal results; in this paper, we set out to resolve this hypothesis ourselves. In our search for the truth, we will use a corpus with a sensible amount of noise and develop a few cleaning and filtering methods to yield several training sets (in-depth explanation of the details can be found in Section 3.3.1), which will be used as training data for the same amount of NMT engines as training sets. Afterwards, we will carry out a thorough comparison of the results via manually reviewing the outcome as well as with the help of MT-focused evaluation metrics.

Here is an overview of the questions to which we aspire to find an answer (specifically in our English-Spanish case):

- Does the quality of the data affect the NMT output?
- How can we best measure the quality of different NMT engines' output and the extent to which it affects it?
- What types of noise are the least beneficial and most harmful for the NMT engines?

2 Related work

Quantity versus quantity, which one is more relevant? This is a question widely discussed in Machine Translation (MT). Firstly, we discuss the importance of quantity in MT, specially in Neural MT, moving on to quality-related papers that deal with noise and its impact, and end mentioning works that researched the measure of text quality.

Novel approaches to NMT maintain a grand focus on training their engines with a large quantity of data. This idea first came to light with Statistical MT (SMT), where doubling the training set has proven to increase BLEU scores (Turchi et al., 2008; Irvine and Callison-Burch, 2013). In the aspect of quantity, when NMT and SMT are compared, however, SMT shows a greater resistance to low-quantity data sets (Koehn and Knowles, 2017). This was tested by building an array of English-Spanish systems, starting at 385.7 million words, and decreasing by $\frac{1}{2}$, ..., $\frac{1}{512}$, $\frac{1}{1024}$. A learning curve was obtained from the results of the trained engines in terms of BLEU scores. NMT was observed to require a training set of at least a few million words to get off the ground.

There is an array of previous studies that have focused on the influence of noise in training data within the Machine Translation (MT) field. Notably, Belinkov and Bisk (2018) explore a way to increase the immunity of character-based NMT systems against natural noise via structure-invariant word representation and by carrying out training on adversarial examples of different kinds. Findings proved that a character-based convolutional neural network (CNN) learnt to address multiple types of errors seen in training, yet rich characteristics typical of natural human error went undetected. Khayrallah and Koehn (2018), a great inspiration of our research, study the impact of a variety of real-world noise types on Neural Machine Translation (NMT) and Statistical Machine Translation (SMT). They introduce artificial noise to the training sets and analyse the degraded performance results compared to the original clean corpus. Their findings showed that untranslated segments proved to be the most egregious type of noise, which resulted in copied source segments in the translation. The work we carry out in this paper is heavily influenced on their project and findings, though in a contrary manner, as we explore the findings of removing noise, whereas they focused on the results of adding it.

Briakou and Carpuat (2021) shed some light in a different approach that attempts to maintain the training samples that are mostly equivalent but contain a small number of semantically divergent tokens. Their findings show that, unlike noisy samples, fine-grained divergences provide a useful training signal for NMT when modeled via given factors. More so, they were able to prove that understanding the properties of training data can help build better NMT systems, as we wish to do in this paper. In a similar manner, Wang et al. (2018) trained an NMT model with noisy training data to later fine-tune (or *denoise*) it with a trusted dataset made of human translations or other high quality sources of translations. NMT systems that were denoised achieved higher BLEU scores than those who were not fine-tuned with the trusted dataset.

On the side of detecting and measuring text quality, we find a broad selection of work out there. As innovators in their field, Higgins et al. (2004) developed a system that evaluates multiple aspects of coherence in essays, identifying characteristics based on se-

mantic similarity measures and discourse structure. With this goal in mind, they worked on creating a labelled data set with the help of writing experts and human annotators. Their findings proved promise, with the capability of ranking global and local aspects of coherence, such as relationship to essay topic, technical quality and the connection between discourse elements. On a similar note, and more focused on evaluating reading difficulty, Miltsakaki and Troutt (2008) created two tools called Read-X and Toreador, whose goal is to automatically analyse and categorise web text for thematic content and assess reading difficulty with the profile of the reader in mind. Read-X focuses on web search and text classification, whereas Toreador analyses the input and predicts reading difficulty. Differentially, focusing on whether a text is well-written and not readability, Louis (2013) explores new insights and techniques to automatise the quality detection of a text. Their work introduced four methods for text quality prediction:

- **organisation quality**, based on syntax and with the idea that intentional structure is predictive of coherent organisation,
- **text specificity**, which uses sentences' proportions and sequences for predicting quality,
- **interesting nature of articles**, dictated by a supervised system specialised on science news article, trained to identify visual nature of writing, narrative structure, beautiful writing and research-related descriptions, and lastly,
- **verbosity**, focused on prediction based on the compatibility of the type of details and article length. Findings showed promising results with a margin for improvement.

3 Data: a closer look

Data is the very own factor that allows the vast majority of NLP tasks to be developed and studied, as well as a sine qua non element in improving and advancing within the field. As we have pointed out in previous sections, and as other researchers have figured out (as mentioned in Section 2), there are several aspects and characteristics pertaining to data that can, and will, make a notable difference in the outcome.

Previous to acquainting ourselves with related research, we point towards the following hypothesis: one of the characteristics that can yield a better outcome in NMT is the objective quality of the data used in training. Thus, here we wish to introduce our concept of objective quality so as to provide a defined meaning to such a subjective topic as *quality* can be. Moving on, we will narrow our focus on the noise aspect of data and how relevant it is to the data's quality.

As a continuation, we will delve deeper into the details of the corpus we have chosen as our subject test. This will provide us with a valuable analysis to assess the kinds of noise found in the chosen corpus and neutralise it via a variety of cleaning methods.

3.1 Defining text quality

For the sake of remaining factual, we introduce this section with the definition of quality. As proposed by the Cambridge Dictionary (2022), quality is:

> How good or bad something is.

In other words, quality will help us determine if our subject is optimal or unsatisfactory to a given set of standards. Applied to NMT, this principle seeks to determine to which degree the outcome of a given trained engine is optimal or unsatisfactory. In Table 1, we gather and write down a few standards of our own expectations (based on our previous experience with typical NMT outcome results), which we have separated in four different categories of increasing quality.

Nonetheless, to have a more complete assessment of the outcome, we will also rely on the renowned Six Traits model (Spandel, 2005), well-known by educators and a standard model to follow. Let us retrieve this brief yet very well explained summary from Louis (2013, p. 13):

> The development of the Six Traits rubric was influenced by an early study by Diederich (1974). He used around fifty people in various capacities such as writers, editors, business executives and teachers and asked them to group 300 student essays into good, mediocre and poor quality categories. Specifically, the raters placed each essay in one of nine rating levels and wrote comments on the problems with each essay. There were two main findings from this study:
>
> - People varied greatly about which class they assigned for each individual essay. [...]

- When the comments reported by people within each group were analyzed, it was found that each group focused greatly on a different aspect of quality. [...]

Later studies (Murray, 1982; Purves, 1992) were also able to replicate these findings and all studies came up with a small (around five or six) definable set of traits for judging quality. These findings led a team of teachers to develop a rubric for grading writing covering the traits highlighted in prior work. This rubric is called the Six Traits (Spandel, 2005) and is immensely popular and standard in the education field.

I. Poor quality

- Output is complex to understand
- Structure is sluggish and lacks fluidity
- Used terminology is wrong; differs from original meaning

II. Okay quality

- The gist of the source text can be inferred from the translated version
- Some, yet slight structure-related mistakes (grammar and stylistics)
- Terminology is sufficient, yet lacks diversity

III. Good quality

- Translation can be understood well, barring certain fluidity
- Simple structures used (grammar and stylistics)
- There is certain variety in terminology

IV. Excellent quality

- The outcome is easy to read; as if written by a native speaker
- Varied, complex and well used structures throughout single segments
- Broad and precise selection of terminology

Table 1: Quality standards of our own expectations, based on our previous experience.

The rubric provided by the Six Traits model are the six qualities that aid us to identify a **good** piece of written text. Find them listed and briefly explained in Tables 2 and 3 (an excerpt from Spandel (2005)).

I. Ideas and development: the writing is clear, focused, and well-developed, with many important, intriguing details.

- The writer is selective, avoiding trivia, and choosing details that keep readers reading.
- Details work together to clarify and expand the main.
- The writer's knowledge, experience, insight or perspective lend the piece authenticity.
- The amount of detail is just right–not skimpy, not overwhelming.

II. Organization: the order, presentation, and structure of the piece are compelling and guide the reader purposefully through the text.

- The entire piece has a strong sense of direction and balance. Key ideas stand out.
- The structure effectively showcases ideas without dominating them.
- An inviting lead pulls the reader in, a satisfying conclusion provides a sense of closure.
- Details fit just where they are placed.
- Transitions are smooth, helpful and natural.
- Pacing is effective; the writer knows when to linger and when to move along.

III. Voice: the writer's passion for the topic drives the writing, making the text lively, expressive and engaging.

- The tone and flavor of the piece are well-suited to topic, purpose, and audience.
- The writing bears the clear imprint of this writer.
- The writer seems to know the audience and to care about their interests and informational needs.
- Narrative text is moving and honest; informational text is lively and engaging.
- This is a piece readers want to share aloud

Table 2: Six Traits model's rubric for a good piece of written text, part I; continued in Table 3.

IV. Word choice: precise, vivid, natural language enhances the message and paints a clear picture in the reader's mind.

- The writer's meaning is clear thoughout the piece.
- Phrasing is original–even memorable–yet the language is never overdone.
- Lively verbs lend the writing energy and power.
- Modifiers are effective and not overworked. Clichés, tired words, and jargon are avoided.
- The writer repeats words only for effect and does not overdo it.
- Striking words or phrases linger in the reader's memory.

V. Sentence Fluency: easy flow and sentence sense make text a delight to read aloud.

- Sentences are well-crafted, with a strong, varied structure that invites expressive oral reading.
- Striking variety in structure and length gives writing texture and interest.
- Purposeful sentence beginnings show how ideas connect.
- The writing has cadence as if the writer hears the beat in his/her head.
- Fragments, if used, add style and punch; dialogue, if used, is natural and effective.

VI. Conventions: the writer shows excellent control over a wide range of age-appropriate conventions and uses them accurately (sometimes creatively) to enhance meaning.

- Errors are so few and minor a reader could skip right over them unless searching for them.
- The text appears clean, edited, polished. It's easy to process.
- Only light touch-ups are needed before publication.
- Conventions enhance the message and voice.
- As appropriate, the writer uses layout to showcase the message.

Table 3: Six Traits model's rubric for a good piece of written text, part II; continuation of Table 2.

By joining Tables 1, 2 and 3, we gather a wide selection of traits and characteristics that will aid us in identifying the quality of the outcome of our NMT engines from an objective perspective. However, we should differentiate the goal of Table 1 and the Six Traits model (Tables 2 and 3); the former has been written with MT outcome in mind, whereas the Six Traits model was created for pieces of written text made from scratch. Thus, we will rely on the former to assess accuracy and conveyed meaning, whereas the latter will act as a guide and reference to assess aspects closer to word choice and positioning, fluency, and similar. Moreover, when referring to the Six Traits model for evaluation, we will only make use of points I, II, IV, and V, as points III and VI are arguably exclusive to tasks that pertain more to text generation from scratch than translation, given that the text to work with is already written and cannot be altered, only conveyed its meaning to the best of our ability.

3.2 Data and noise

Data can be defined as a collection of entries that bestow information. Brodie (1980) relates data quality to the perseverance of the meaning of the data, which roots from designers and users of a given database application. That is to say, the quality of a given dataset will depend on how well it represents the properties of the intended application of use. More often than not, however, data includes and is affected by noise; as Wang et al. (1995) put it, databases are corrupted by mistakes or by entries that do not meet the needs of the users. This stance still holds true today.

This noise perturbance is to be expected, precisely in the retrieval of datasets for MT. As a general rule, and as we have previously mentioned, MT is a task that requires a large amount of data. In specific, NMT has proven this need to feed on a lot of data, even more so than other MT variants (i.e., Statistical MT (SMT), which showed a higher resistance to datasets with a smaller amount of training data). Koehn and Knowles (2017) showed in their paper a learning curve of a series of English-Spanish systems trained with as low as 0.4 million words and as high as 2 billion. With the former, NMT's BLEU scores were quite low. With the latter, NMT even surpassed Phrase-Based (SMT) systems.

One might wonder: where does all the data come from? The answer in most cases is text crawling, the usual go-to technique for large text retrieval, performed on electronically available text in the World Wide Web. Given NMT's need for high-resourced training data evokes the issue of maintaining high quality entries throughout large databases. Ensuring this high quality is quite easy and efficient for small datasets via manual evaluation, however, this would be an impossibility for the large datasets that are required in NMT; as noted by Eckart et al. (2012), it would be a major challenge to manually evaluate millions of sentences. As far as automatic cleaning methods go, there is yet to arise an efficient, thorough and complete technique that guarantees entirely successful results. Nonetheless, there is an array of techniques that can help tremendously in this task.

We would like to propose two examples of corpora collections obtained via text crawling and the proposed cleaning methods by the researchers who developed said collections:

- The Leipzig Corpora Collection (Goldhahn et al., 2012) is a very good example of a well-kept online corpus-based dictionary collection which relies heavily on text crawling. This online resource processes documents available in the web, collects them and offers an in-depth dictionary in their website, in more than 250 languages. They offer different categories for each language, and the corpora are then listed based on the year on which they were crawled. Any corpus can be selected to search for a given term, obtaining feedback such as occurrence, examples, similar words, words that typically accompany the given term and a word graph. The cleaning methods mentioned in Goldhahn et al. (2012) are non-sentence identification via patterns and foreign language detection.

- The Open Parallel Corpus (OPUS) (Tiedemann, 2012), a growing collection of translated texts from the web first brought to life in Tiedemann and Nygaard (2004). In their website, they claim the use of several tools to compile the available collection (crawling being one of them (Skadiņš et al., 2014)). They also note that all preprocessing is done automatically and that no manual corrections are carried out. We have not been able to spot the specific methods for preprocessing carried out by the researchers behind the OPUS collection. However, they do offer their custom parallel corpus filtering toolbox: OpusFilter (Aulamo et al., 2020). This tool is available for free and offers many great options: removing duplicate segments and applying a wide variety of filters such as length, special characters, language identification, language modelling and similarity (amongst others!).

So far, we have reviewed the type of data that is used for training NMT engines, we know it contains a certain amount of noise and we have mentioned a few examples of preprocessing and cleaning methods. Now, our initial queries prevail: does the noise in the training set affect the NMT system? If it does, to what extent does it affect it and how can we measure its influence? Also, what types of noise affect the NMT system the most and what conclusions can we get out of it? Let us figure it out.

3.3 Wikipedia: our case scenario and its noise

Discerning the impact of noise on the training of NMT systems, and spotting the differences of its outcome requires a corpus that includes a significant amount of noise so as to provide a corrupted version and an altered version that highlights and easily identifies the dissimilarities of the raw and processed data. After exploring different possibilities, we choose to proceed with the Wikipedia (Wołk and Marasek, 2014) corpus, available in the Open Parallel Corpus (OPUS) (Tiedemann, 2012) website; as its name reveals, this corpus of parallel sentences was extracted from the popular website Wikipedia (Contributors, 2004), a free online resource made of editable content organised in millions of articles and in more than 300 languages. The Wikipedia corpora covers 20 different languages, but we focus our investigation on English and Spanish, as source and target language, respectively.

First off, we ran a preliminary study of the components of the corpus, which can be observed on Table 4:

Component	**EN**	**ES**
Sentences	1,811,428	1,811,428
Words[1]	34,120,216	34,461,481
Unique words[1]	802,666	875,587
Characters	221,888,062	227,802,200

[1] Computed using the LexicalRichness library (Shen et al., 2021)

Table 4: Preliminary Study of the Wikipedia Corpus for English and Spanish.

As Table 4 reflects, we can see that there are some differences between the English and Spanish versions of the corpus. Immediately, we observe the Spanish variant has over 340,000 more words than the English variant, including over 70,000 more unique words and nearly 6,000,000 more characters. This proves that we will find a wider lexical variety in the Spanish corpus compared to the English one; however, this does not come as a surprise. The Spanish language has a much richer verbal morphology than English (Ahn, 2017; Rodríguez and Carretero, 1996), and more variations for gender and number noun forms, whereas English proves to be a morphologically poorer language that tends to follow a set order of words (subject-verb-object) (Avramidis and Koehn, 2008). Moreover, text expansion and contraction (Aranda, 2007) is also taken into account when translating, which causes a segment to increase or decrease in size and depends on the source and target languages. In our case, we have English to Spanish, where we can observe an estimate of 20% to 25% expansion when translating[1].

In the next sections, we will give a slim introduction to the noise in our corpus by manually analysing a sample subcorpus of 200 extracted segments. Then, we will explain how we went about cleaning the subcorpus, and provide a broader overview of the noise found

[1] https://www.kwintessential.co.uk/resources/expansion-retraction

both in the subcorpus and the entire corpus via an automatic noise detection programme we wrote. Finally, a table will be presented with the five training sets that will be used to train the same amount of NMT systems, all with the same characteristics.

3.3.1 Noise identification and analysis

Before moving forward and applying any preprocessing methods to the corpus, we seek to spot different sorts of noise within our data. This search was mainly inspired by the revelations in Khayrallah and Koehn (2018), where the types of noise as classified as follows: misaligned sentences, misordered words, wrong language, untranslated sentences and short segments. To ensure a more rigorous search, provide further insight and get an initial idea of what we were dealing with, we decide to manually scan the corpus in search for these types of noise.

Thus, we first retrieve a sample subcorpus of 200 segments from the original corpus with the goal of devising a few cleaning modules that allow us to get rid of as much noise as possible. After having reviewed the noise in the subcorpus and applied the cleaning modules, we will apply these changes to the corpus and show the outcome. Results of the preliminary study of the subcorpus can be found in Table 5, where we can observe a similar pattern to that of the corpus in Table 4, presenting a slight spike in the quantity of words and unique words for Spanish.

Component	**EN**	**ES**
Sentences	200	200
Words[1]	3,649	3,708
Unique words[1]	1,649	1,703
Characters	24,299	24,862

[1] Computed using the LexicalRichness library (Shen et al., 2021)

Table 5: Preliminary Study of the Wikipedia Subcorpus for English and Spanish.

Of course, the results of the noise study can vary depending on the extracted segments, but we believe 200 is enough to get a general idea of what the rest of the corpus looks like in regards to types of noise.

To begin the noise study, we start by comparing the source and target files of the subcorpus individually as well as side by side, in a search for flaws that could be classified as noise. In the following sections, we will lay out the findings of our analysis and classify in three main areas the types of noise we found. For each of those areas, we will make the pertaining adjustments to the subcorpus with the necessary method (whether that is editing, adding or removing segments, words or characters). We want to note that, of course, for every source segment that presents an issue that requires removal of said source segment, the pertaining target segment will be removed as well to avoid disparities and misaligned segments in our parallel corpus, and vice versa with target segments that

present said issue. This way, we will maintain the same amount of segments for both source and target files at all times.

Segment length

Reviewing the data, we find that we are dealing with two separate issues regarding length: individual segments with a length that we think to be problematic, and parallel segments whose length difference is too significant. The former refers to those segments with a total word length that is too short or too long. The latter refers to the parallel segments that have a character difference that is too big. But, how could we go about establishing a threshold for both issues? Let us elaborate further in the following paragraphs with insight into the aforementioned issues and how we denoted the thresholds.

Firstly, we want to tackle the significant differences in length between source and target segments. These can produce a variety of issues if included in the training sets for the NMT engines. A dramatic disparity between parallel segments can indicate misaligned sentences, missing information or even differences in sentence order, all of which can arise confusion for the NMT system. To find the appropriate threshold to withhold segments of significant differences, we will perform the following test: removing segments with a character difference in increments of 10, starting at 10 and finishing at 100 or more. Results of our findings are presented in Table 6.

Character difference	**Source**	**%**	**Target**	**%**
10-19 character difference	16	8%	37	18.5%
20-29 character difference	5	2.5%	15	7.5%
30-39 character difference	4	2%	2	1%
40-49 character difference	3	1.5%	1	0.5%
50-59 character difference	0	0%	1	0.5%
60-69 character difference	1	0.5%	0	0%
70-79 character difference	2	1%	1	0.5%
80-89 character difference	0	%	0	0%
90-99 character difference	0	%	0	0%
100+ character difference	0	%	0	0%

Table 6: Analysis of the amount of segments with a character difference in the subcorpus, measured in increments of 10, starting at 10 characters up until 100 or more. Percentage calculated from the total of source and target subcorpus.

Observing Table 6, we first see an affluence of segments with a character difference between 10 and 29. Then, we notice a dramatic decrement from 30 onwards, totalling 10 out of 200 for the source file, and 5 out of 200 for the target file. With this information in mind, we decide to manually check the results to get a more detailed input and to correctly assess if there were any issues that could damage the training of the NMT system.

After reviewing the output, we find that for the character differences ranging from 10 to 49, for the most part, we see valid translations that keep all of the information in both sides of the parallel subcorpus, not a single misaligned sentence and no translation omissions (missing information in the target segment when translating) or additions (added information in the target segment when translating that is not present in the source segment). From 50 onwards, however, we start seeing issues that can incorrectly mislead the NMT engine. Let us provide two examples (with a 46 and 71 character difference) of incorrectly translated segments inferred by their character difference:

- omition in the translation process:

  Source (62 char.): *American Newspaper Publishers Association (then ANPA, now NAA)*

  Target (16 char.): *ANPA (ahora NAA)*

- addition in the translation process:

  Source (125 char.): *María Amparo Rivelles Ladrón de Guevara (11 February 1925 – 7 November 2013) was a Spanish actress, known as Amparo Rivelles.*

  Target (196 char.): *María Amparo Rivelles y Ladrón de Guevara (Madrid, 11 de febrero de 1925 - ibídem, 7 de noviembre de 2013) fue una actriz española de amplia trayectoria teatral y cinematográfica iniciada en 1939.*

Above, we can observe that the first example includes information in the source segment that is missing in the target segment, due to initials that were not explained when translated; then, for the second example, we observe the contrary: the target segment includes far more information than the source segment. Both of these issues can cause problems when training the NMT system, as it will learn to map certain words wrongly. Certainly, then, we find it wise to remove all segments whose character difference rises above 50.

Secondly, besides the character difference, we also notice that some of the segments are too short or too long compared to the rest of the subcorpus. We decide to discard those segments that are excessively short (less than 3 words), for both languages. Why three? Because with three words we can build significant and meaningful sentences (per instance: "*The white car*", or "*Voy al parque*"), but with less than 3 words, it is complicated. Then, to deal with the sentences that were far too long, we think the average word count per sentence is a good starting point. We compute it using the entire corpus instead of the subcorpus (for more accurate results), yielding a mean of 20 words per sentence for English and Spanish alike. Now, 20 words would be too short, and filtering out all sentences with

Sentence length	Source	%	Target	%
1 to 2 words long	6	3%	6	3%
3 to 20 words long	121	60.5%	118	59%
21 to 40 words long	63	31.5%	64	32%
41 to 60 words long	6	3%	9	4.5%
61+ words long	4	2%	3	1.5%

Table 7: Analysis of the amount of words in the segments of the subcorpus. Percentage calculated from the total of source and target subcorpus.

less than 20 words would leave us with a very small corpus. Thus, as before, we do a small test, this time in increments of 20. Results can be found in Table 7.

Seeing the results in Table 7, we can observe that most of the segments have a word count between 1 to 40 words long, totalling 378 sentences out of the 400 total of the subcorpus, with 190 sentences belonging to the source file and 188 to the target file. The remaining segments contain a word count of 41 onwards, with the remaining 22 segments. As done previously with the character difference issue, we will now take a look at the results for further insight.

Reviewing the outcome of the analysis, we observe some short sentences with fewer than 3 words that do not make much sense and do not provide significant meaning, as we argued earlier. Then, we can see quite a lot of segments between 3 to 40 words that, all in all, contain barely no apparent issues that could negatively impact the NMT engine. When the length of the sentences goes over 40, however, we start to witness a constant increment in translation problems; let us showcase a few examples to better understand the matter in question:

- a segment that is too short (less than 3 words long):

  Source (2 words): *Lima: CEP.*

  Target (2 words): *Lima: CEP.*

- a segment that is of ideal word length (3 to 40 words long):

  Source (24 words): *XMP allows each software program or device along the workflow to add its own information to a digital resource, which carries its metadata along.*

  Target (24 words): *XMP permite a cada programa o dispositivo agregar su propia información al recurso digital, con lo que puede quedar incorporada en el archivo final.*

- a segment that is too long (41 words or more):

  Source (68 words): *==Points system==The 2001 IAAF points tables use the following formulae:* Points = for track events (faster time produces a better score)* Points = for field events (greater distance or height produces a better score)A, B and C are parameters that vary by discipline, as shown in the table on the right, while P is the performance by the athlete, measured in seconds (running), metres (throwing), or centimetres (jumping).*

  Target (65 words): *== Sistema de puntos ==La tabla de puntos IAAF 2001 usa la siguiente fórmula:* Puntos = INT(A*(B-P)C) para eventos de pista* Puntos = INT(A*(P-B)C) para eventos de campoA, B y C son parámetros que varían con la disciplina, como se muestra en la tabla, mientras que P es la marca obtenida por el atleta, medida en segundos (eventos de pista), metros (lanzamientos), o centímetros (saltos).*

Here we have seen three examples: the first one shows that such a short segment cannot provide any meaningful information, as it does not contain enough context as to what it is trying to say; the second example conveys most of the meaning from the source onto the target segment (missing the *along the workflow* part), being an almost-ideal segment for the training of the NMT system; and, finally, we have the the third example, which as we can see, misses information in the target text. These issues were similar for those segments of a related word length. Thus, with this mind, we decide to filter out all segments with a word count longer than 41 words. Nonetheless, we wonder if this will have a limiting effect on the NMT systems, as it will not properly learn to write long texts as it will remain accustomed to segments of less than 41 segments. This would be an interesting research idea for future work.

Foreign language and untranslated segments

When corpora are created automatically, as is the case with text crawling, it is uncertain exactly what segments are included in the set. For this reason, we decide to scan the subcorpus for segments that contain foreign language content, whether that means a third foreign language, or untranslated segments (whether that means both segments in either one of our working languages [English and Spanish], or reversed source and target languages). This has been made possible thanks to CLD2 (Ooms, 2022), a Python (Van Rossum and Drake, 2009) module that is able to identify over 80 Unicode UTF-8 languages present in a sentence. This module returns a percentage for each detected language, representing the amount of text detected for a given language; if a text has 1000 bytes, and CLD2 returns a 75% for French, and a 25% for Norwegian, that would be 750 bytes in French and 250 bytes in Norwegian.

More often than not, foreign language segments are accidentally included. Say, for example, that we are crawling information about the Boshin War (also referred to as the Japanese Revolution or Japanese Civil War), in English and Spanish. In such a case, it is quite likely that we find Japanese phrases here and there. This type of noise has proven before to highly influence the outcome of NMT engines (Khayrallah and Koehn, 2018).

After observing various results, we decide to maintain those segments that return the desired language (English for source, and Spanish for target) as the highest percentage. This criteria makes the most sense, as it maintains the majority of the correct segments and filters out incorrect ones, as can be observed in Table 8 in Section 3.3.2.

Here we leave three examples; two that were detected as source and target language, and a third example that detected foreign language content:

1. First example:

   - Source segment; detected a 98% of English in the given segment.

     Segment: *The damage done by Sherman was almost entirely limited to the destruction of property.*
     CLD2 detection results: *(('ENGLISH', 'en', **98**, 1393.0), ('Unknown', 'un', 0, 0.0), ('Unknown', 'un', 0, 0.0))*

   - Target segment: detected a 99% of Spanish in the given segment.

     Segment: *El daño causado por Sherman estuvo limitado casi exclusivamente a la destrucción de la propiedad.*
     CLD2 detection results: *(('SPANISH', 'es', **99**, 579.0), ('Unknown', 'un', 0, 0.0), ('Unknown', 'un', 0, 0.0))*

2. Second example:

   - Source segment; detected a 98% of English in the given segment.

     Segment: *The damage done by Sherman was almost entirely limited to the destruction of property.*
     CLD2 detection results: *(('ENGLISH', 'en', **98**, 1393.0), ('Unknown', 'un', 0, 0.0), ('Unknown', 'un', 0, 0.0))*

   - Target segment: detected a 99% of Spanish in the given segment.

     Segment: *El daño causado por Sherman estuvo limitado casi exclusivamente a la destrucción de la propiedad.*
     CLD2 detection results: *(('SPANISH', 'es', **99**, 579.0), ('Unknown', 'un',*

0, 0.0), ('Unknown', 'un', 0, 0.0))

3. Third example:

   - Foreign language segment: detected a 47% of Japanese, 38% of Danish and 14% of English in the given segment.

     Source segment: *The work takes its title from its prefatory passage:* つれづれなるまゝに、日暮らし、硯にむかひて、心にうつりゆくよしなし事を、そこはかとなく書きつくれば、あやしうこそものぐるほしけれ。*"Tsurezurenaru mama ni, hikurashi, suzuri ni mukaite, kokoro ni utsuriyuku yoshinashigoto wo, sokowakatonaku kakitsukureba, ayashū koso monoguruoshikere.*
     CLD2 detection results: *(('Japanese', 'ja', **47**, 3760.0), ('DANISH', 'da', **38**, 401.0), ('ENGLISH', 'en', **14**, 1220.0))*

   - Target segment: detected a 99% of Spanish in the given segment.

     Target segment (detected: Spanish, 57%; Danish, 42%): *La obra toma su nombre del siguiente fragmento::"Tsurezurenaru mama ni, hikurashi, suzuri ni mukaite, kokoro ni utsuriyuku yoshinashigoto o, sokowakatonaku kakitsukureba, ayashū koso monoguruoshikere.*
     CLD2 detection results: *(('SPANISH', 'es', **57**, 453.0), ('DANISH', 'da', **42**, 353.0), ('Unknown', 'un', 0, 0.0))*

In the last example (the third example), we can see that the majority of the source segment is clouded by Japanese characters, and some words are in Rōmaji (for source and target segments), which is the Japanese term for the romanisation of the Japanese language (i.e., *Tsurezurenaru*). Unfortunately, as aforementioned, CLD2 only works with Unicode UTF-8 text, which is why it is identifying Rōmaji as Danish. However, in this case it works in our favour by removing the segments with any foreign language that is not English nor Spanish. Furthermore, since we are removing segments in pairs, in a case where it detects the wrong language on either side of the subcorpus, it will also remove its parallel partner, thus, in this case, even if the target segment indicates Spanish as the highest percentage detected language, it will be discarded due to the source segment not having English as the highest percentage detected language.

Repeating characters and certain symbols

More often than not, we have seen corpora blurred by a plethora of symbols that not only make it complicated to train the NMT engine, but also make it hard to read it. When it comes to our chosen corpus, since it was obtained from Wikipedia articles, we have seen

a great amount of symbols that are meant to represent formatting, which we believe can be a negative factor when training. If the words we want our NMT system to learn are always clouded with symbols, it will be hard to distinguish which symbols are a part of a segment and which are plain noise.

To be more precise with what we found, let us dig in deeper in our findings and specify on what we plan on substituting, and what we want to remove:

- **Substitute or add:**
  - punctuation symbols that are repeating continuously will be substituted for a single unit of said symbol. For example:

    ";;;" → ";", "((" → "(", or "„„" → ","
  - we added missing spaces between symbols and word. For example:

    "(2008);Symbols" → "(2008); Symbols"
  - spaces were also added between lowercase and uppercase letters. For example:

    "HiMadam" → "Hi Madam"
- **Remove:**
  - two or more consecutive non-repeating symbols had the last symbol removed (such as ";?"), to avoid confusion (a sentence will be harder to interpret if it ends with two such different symbols, or if the two symbols are found in the middle of the sentence)
  - certain symbols that seem to represent format were erased (such as "=" and "*")
  - we removed symbols found within a word. For example:

    "B!otero"
  - additional spaces are also discarded. For example:

    "Hello my name is Roger" → "Hello my name is Roger"

We provide here two examples, one per language, extracted from the subcorpus which might be more illustrative:

- Source (English): Formatting tags (==), double symbols (:*) and additional spaces were detected.

  Source (before cleaning): *==Variants of Gipuzkoan==Within Gipuzkoan, there are four main sub-dialects:* The Beterri variant (from the area surrounding Tolosa, towards San Sebastián).*

Source (after cleaning): *Variants of Gipuzkoan Within Gipuzkoan, there are four main sub-dialects: The Beterri variant (from the area surrounding Tolosa, towards San Sebastián).*

- Target (Spanish): Formatting tags were detected.

Target (before cleaning): *==Bibliografía==* Frederic, Louis (2002).*
Target (after cleaning): *Bibliografía Frederic, Louis (2002).*

As mentioned before, quantifying results can be observed in Table 8 in Section 3.3.2.

3.3.2 Results of applying the noise filters

In the previous subsection, we study and analyse our 200 sample subcorpus from the corpus and come up with a few tactics to get rid of the most frequent types of noise discovered. We thoroughly explain the noise found, categorise it in three different subsections and elaborate our thoughts as well as our ideas to obtain a corpus void of this noise. After having gained further insight of the corpus and its noise via the adjustments applied on our sample subcorpus, we want to expand our filters to the entire corpus. In order to apply these changes, we create a small, yet efficient Python programme Van Rossum and Drake (2009) which was capable of taking in the given source and target files, and apply the chosen filter to remove a either one of the three types of noise we classified as well as all the filters at once.

Once the filters were applied, we extracted some information pertaining to the noise found, which is represented in Table 8.

Though there may be differences in the quantities between source and target, we want to note once again that for every segment that encountered an issue that required removal of the entire segment (except symbols [as explained in Section 3.3.1], given that it only needed adjustment at word level, not entire sentence removal), the parallel segment was removed too. Thus, though there may be more segments that are longer in the target language, for all the target sentences that were longer than our threshold, all the source sentences were removed as well, and vice versa.

From Table 8, we can observe that there is an overwhelming amount of segments that have been erased from the corpus, leaving us with fewer data to work with. The table shows that nearly 30,000 English segments had a character difference of over 50 when compared to the Spanish parallel segment, and over 11,000 Spanish sentences had the same issue compared to the English parallel segment; nearly 76,000 English sentences were too short according to our fewer-than-3-words threshold, and nearly 75,000 Spanish sentences shared the same problem; close to 88,000 English sentences exceeded our threshold for long sentences, situated at 41 or more, and so did the over 100,000 Spanish sentences; about 5,000 English sentences were found to be in a foreign third language, as well as close to 600 Spanish sentences; similarly, but with greater figures, there were over 54,000 untranslated

	Corpus				Subcorpus			
Type of noise	**Source**	**%**	**Target**	**%**	**Source**	**%**	**Target**	**%**
50+ character difference	29,479	1.63%	11,098	0.61%	3	1.5%	2	1%
Sentence too short (< 3 words)	76,065	4.20%	75,735	4.18%	6	3%	6	3%
Sentence too long (> 40 words)	88,110	4.86%	100,448	5.55%	10	5%	12	6%
Foreign language	5,408	0.3%	634	0.04%	2	1%	0	0%
Untranslated segments	54,682	3.02%	49,536	2.73%	9	4.5%	6	3%
Symbols removed or substituted	4,604,127	2.07%	3,971,317	1.74%	488	2.01%	454	1.83%

Table 8: Analysis of the amount of noise found in the Wikipedia corpus and subcorpus. Percentage calculated from the total of source and target corpus and subcorpus, respectively. Note that the last row is calculated at character level, whereas the rest are at segment level.

segments in the English corpus, and about 49,000 in the Spanish corpus; finally, when scanned for unnecessary symbols we considered noise, the English corpus turned up with roughly 4,600,000 symbols that were removed or substituted, and the Spanish corpus came close to 3,900,000.

We also include in Table 8 the same information regarding the subcorpus, which allows us to compare it to the corpus. When contrasted, we see that the total percentages are very close for the corpus and the subcorpus in each category, with not very differential results; this helps us quantify a if the changes done to the corpus are relevant all throughout given our sample subcorpus. Given that the percentages are quite close, we can objectively argue that the changes applied are relevant.

Now that all cleaning methods are applied, we will be able to tell how exactly they affect the performance of the NMT engines: will they show an increase in translation quality, or will they be too strict and decrease their performance?

3.3.3 Data set division

As a final step in our data analysis and processing, we need to create the final data sets to be used for our NMT training and evaluation. Before proceeding further into the data set separation, we want to begin by giving a brief explanation on how each filter affects and alters a given dataset with the following list:

1. Raw data set: contains the parallel corpus in its raw format, with no preprocessing

whatsoever.

2. Constrained length data set: the sentences that were too short or too long according to our thresholds, or the segments that showed a character difference of over 50 were removed from the raw data and saved in this data set.

3. Foreign language or untranslated segments removed data set: the segments that were detected as written in a foreign language or untranslated were removed from the raw data and saved in this data set.

4. Symbols removed or substituted data set: the entire raw data set is included, though it was scanned for unwanted symbols and these were either removed or substituted accordingly.

5. All filters applied data set: the filters mentioned in points 2, 3 and 4 were applied to the raw data set in the following order: 4; 2; 3. This order allowed to first clean the corpus of unnecessary or unwanted symbols, which may have triggered the removal sentences that were too short to show as longer, or sentences of just the right size to appear as longer, then remove the segments that were rightfully too short or too long once cleaned from symbols and unwanted characters, and, finally, detect the segments' language once they were free of confusing characters.

For more detailed information regarding the filters themselves, please refer to Section 3.3.1.

Now, in the upcoming sections we are going to present our training, validation and test sets and detail their characteristics.

Training sets

In total, we yield 5 different training sets, which add up to 10 different files as every training set is made up of both source and target languages. In Table 9 we present every training set (see the list in the beginning of Section 3.3.3 for more details) and its correspondent language, along with details regarding that training set's amount of sentences, words and unique words.

Table 9 offers us valuable information: most notably, as is expected, the more filters that apply, the fewer data we are left with. As was observed in Table 4, the target training files (Spanish), include more words and unique words than the source training files (English), proving Spanish as a richer language.

Validation sets

For our validation sets, we do not carry out any preprocessing methods; we leave the text as it was in the original dataset. We yield two different files, one for the source language and another for the target language.

Training set	Language	Sentences	Words[1]	Unique words[1]
Raw	Source (EN)	1,800,000	33,906,241	800,592
	Target (ES)	1,800,000	34,243,749	873,244
Repeating symbols	Source (EN)	1,800,000	34,199,233	722,099
	Target (ES)	1,800,000	34,417,292	802,630
Constrained length	Source (EN)	1,591,489	26,884,914	612,346
	Target (ES)	1,591,489	27,399,959	683,953
Foreign and untranslated	Source (EN)	1,454,179	28,150,530	578,476
	Target (ES)	1,454,179	28,663,278	633,400
All filters	Source (EN)	1,308,949	23,451,385	445,256
	Target (ES)	1,308,949	23,889,637	507,538

[1] Computed using the LexicalRichness library (Shen et al., 2021)

Table 9: Training sets that will be used to train the NMT systems, categorised by language and amount of sentences, words and unique words.

Validation set	Language	Sentences	Words[1]	Unique words[1]
Raw	Source (EN)	1,428	29,351	7,414
	Target (ES)	1,428	30,032	8,477

[1] Computed using the LexicalRichness library (Shen et al., 2021)

Table 10: Validation sets that will be used to validate the NMT systems whilst training, categorised by language and amount of sentences, words and unique words.

As a quick observation, we see in Table 10 that, similarly to the training sets, the target file (Spanish) offers a slightly wider variety in unique words and close to a thousand more words than the source file (English).

Testing sets

When it came to the idea of testing our models, we wanted to ensure that all scenarios were covered. For this reason, we create a preprocessed test set for every filter that was used on the training set, but also plan to use the raw test set (unprocessed) to test every NMT engine trained with preprocessed data sets. In short, for every training set (and thus, for every NMT engine), we will evaluate the outcome of two test sets (the raw one, and the preprocessed or filtered one). The only exception is the NMT engine trained with the raw data set, as it never underwent any preprocessing, so it only needs testing with the correspondent test set that is unprocessed. An example may be more illustrative: for the *Foreign and untranslated* segments filter, we have the source and target training set with said filter applied. To test the NMT engine that was trained with this training set, we have the raw test file (unprocessed, just like the original data set, with 10,000 segments),

and the preprocessed test file that was cleaned with the same filter as the training set (the *Foreign and untranslated* segment filter), which amounts to 8,465 segments after filtering out those 1,535 segments that were detected as being in a foreign language. Thus, we get a total of 2 testing sets per NMT engine. The reasoning behind these two test sets is that we want to acknowledge the possibility of increasing an NMT engine performance and output quality by preprocessing the input, as it may be more recognisable for the system and, thus, easier to translate. This idea would be of interest for future work.

In Table 11 we dig in deeper into the details of the test sets as we have done for the validation and training sets. Correspondingly to these sets, the testing sets show a continuous rising trend for Spanish's vocabulary, with more words and unique words to offer than English.

Testing set	**Language**	**Sentences**	**Words**[1]	**Unique words**[1]
Raw	Source (EN)	10,000	184,624	23,212
	Target (ES)		187,700	26,753
Repeating symbols	Source (EN)	10,000	185,398	22,536
	Target (ES)		188,410	26,348
Constrained length	Source (EN)	8,941	151,527	20,609
	Target (ES)		154,516	23,851
Foreign and untranslated	Source (EN)	8,465	169,083	20,818
	Target (ES)		172,403	23,736
All filters	Source (EN)	7,549	138,133	17,764
	Target (ES)		141,244	20,662

[1] Computed using the LexicalRichness library (Shen et al., 2021)

Table 11: Testing sets that will be used to test the NMT systems, categorised by language and amount of sentences, words and unique words.

Now that we have gathered our training, validation and test sets, we can start with the NMT engine development: in the following section, we explained how we trained the NMT systems, which characteristics we used and how we preprocessed our training sets.

4 Neural Machine Translation: developed systems

Training NMT engines is a delicate and time-consuming task that requires of preparation prior to actually putting together an NMT system and training it. We spent some time researching which way benefited us most when it came to training our NMT engines. In the end, we decide to do it with the OpenNMT project (Klein et al., 2017).

4.1 OpenNMT: an NMT ecosystem

For the arduous task of training our NMT systems, we rely on the OpenNMT project[2] (Klein et al., 2017). OpenNMT is an open source ecosystem that focuses on Neural Machine Translation and Neural Sequence Learning. The project begun back in December 2016 by the Harvard NLP group and SYSTRAN and has been used in a plethora of research and industry applications.

We decide to rely on this tool mainly due to its straightforward documentation and its friendly community forum that offers reactive support. Furthermore, this project has been deployed in two different implementations: PyTorch and TensorFlow. We use the PyTorch variation as it is the variant we were most familiar with. We shall note that each implementation has their own set of unique features, and though some are interchangeable, not all are.

4.2 Tokenization and BPE

On the one hand, we have tokenization, known as the process of separating punctuation from words and splitting tokens into words, is a key step commonly executed in Machine Translation tasks. It has been studied thoroughly by researchers (Nguyen et al., 2010; Domingo et al., 2018; Park et al., 2021) to be helpful in improving the outcome of NMT.

On the other hand, we have Byte-Pair Encoding (BPE) (Gage, 1994; Sennrich et al., 2016), an algorithm that aids in preventing out-of-vocabulary issues when translating with MT systems. BPE works in such a way that it breaks down words into smaller pieces, such as *supermarket* → *super_ mark_ et*, which helps the MT system learn words in smaller chunks and tie certain chunks with others that may have initially belonged to a different word. We present a more illustrative example in Table 12. Notice that when BPE is processed, a space will be separating the characters of certain words, such as *archae ology*.

Raw	Sir Barry Cunliffe, the emeritus professor of European archaeology at Oxford.
BPE	_Sir _Barry _Cun_ li_ ffe , _the _emeritus _professor _of _European _archea_ ology _at _Oxford.

Table 12: Aggressive tokenization example with BPE subword encoding and a spacer annotator (underscore).

[2] https://github.com/OpenNMT/OpenNMT-py

Lucky for us, the OpenNMT project not only offers an NMT ecosystem for training and testing NMT engines, but also a Tokenizer[3] with BPE support that they call Pyonmttok. As an overview, we would like to display some further information regarding Pyonmttok (extracted from their GitHub page).

First of all, we would like to showcase the several tokenization modes that can be applied:

- **Conservative:** standard OpenNMT tokenization.
  - *It costs £2,000.* → *It costs £ 2,000 .*
- **Aggressive:** standard OpenNMT tokenization but only keep sequences of the same character type (e.g. "2,000" is tokenized to "2 , 000", "soft-landing" to "soft - landing", etc.)
  - *It costs £2,000.* → *It costs £ 2 , 000 .*
- **Char:** character tokenization.
  - *It costs £2,000.* → *I t c o s t s £ 2 , 0 0 0 .*
- **Space:** space tokenization; extra spaces are trimmed.
  - *It costs £2,000.* → *It costs £2,000.*
- **None:** no tokenization is applied.
  - *It costs £2,000.* → *It costs £2,000.*

Out of all the tokenization options, we select the aggressive one. We think it best fits our goals and can help us achieve a more focused subword tokenization. The same tokenizer is applied for all training sets.

Now, we also want to highlight the customisation options available in Pyonmttok:

- **Reversible tokenization:** marking joints or spaces by annotating tokens.
- **Subword tokenization:** support for training and using BPE and SentencePiece models.
- **Advanced text segmentation:** split digits, segment on case or alphabet change, segment each character of selected alphabets, etc.
- **Case management:** lowercase text and return case information as a separate feature.

[3] https://github.com/OpenNMT/Tokenizer

- **Protected sequences:** sequences can be protected against tokenization with the special characters ((and)).

For our training sets, we choose to apply a reversible tokenization technique called spacer annotator, which allows to mark spaces. Moreover, as we mentioned before, we choose to apply BPE as the subword encoding tokenization. We train a BPE model for each training set (a total of 10 models for each training set and each language), with a vocabulary size of 32,000 on both sides of the parallel corpus.

On Table 12 we leave an illustrative example of the tokenization and subword encoding results.

One great feature of Pyonmttok is that it can be applied on-the-fly when executing the training commands for the NMT systems, as tokenization and subword regularisation are naturally embedded in the data configuration of OpenNMT-py. We applied it this way for an easier and faster implementation.

4.3 Developed systems' characteristics

OpenNMT offers a large array of customisation options for the NMT systems and a few configuration examples[4] from which we have gotten inspiration for our systems. On Table 13 we list the chosen components and options for our systems. As we have mentioned in previous sections, we have trained a total of 5 NMT systems, one per data set as per Section 3.3.3.

We have opted to use the renowned Transformer model (Vaswani et al., 2017), a network architecture based solely on attention mechanisms which revolutionised Natural Language Processing tasks and is still one of the top preferred architectures. The Transformer model also contributed to a decreased training time on GPUs and proved to work well when generalised to other tasks.

Each system takes an approximate 36 to 48 hours to train on a multi-GPU environment (NVIDIA Titan Xp) via the IXA group server[5], a research group from the University of the Basque Country.

200,000 epochs per model seems appropriate enough to put our hypothesis to the test and be able to analyse the effect of noise in our corpus. With the validation steps per 10,000 epochs, we could identify if and how the model improved with perplexity and average score measures.

Whilst training, all NMT engines show a constant increase in accuracy and perplexity, finally reaching the 200,000 epochs with the highest scores, which is why we decide to keep this checkpoint for all models to test on the test sets.

We will now move on to the following section, where we will explain how we translated the test sets using the NMT engines and we go into detail on the findings of the translated segments. To quantify the results, we will make use of automatic evaluation metrics, and

[4]https://github.com/OpenNMT/OpenNMT-py/tree/master/examples
[5]https://www.ixa.eus/

Component	**Quantity/Option**
Epochs	200,000
Validation steps	10,000
RNN size	512
Layers	6
Optimisation	Adam
Learning rate	2.0
Batch size	128
Dropout	0.1

Table 13: NMT engines components.

to qualify them, we will manually review them and provide a subjective opinion based on Tables 1, 2 and 3 as explained in Section 3.1.

5 Findings

Following the training of the NMT systems, we proceed to evaluate their performance. To do so, we want to approach two methods: automatic evaluation metrics and manual evaluation in terms of qualitative results. We will first start with the former, for which we have chosen BLEU (Papineni et al., 2002) and chrF++ (Popović, 2017) as non-trained metrics, and BLEURT (Sellam et al., 2020) and Comet (Rei et al., 2020) as trained metrics. For the latter, we will take a look at some translation samples ourselves and categorise the results accordingly following the rubric explained in Section 3.1.

We will start by introducing both the automatic and the manual evaluation methods in the pertaining sections. We will continue by presenting a table with the results for the BLEU, chrF++, BLEURT and Comet automatic metrics, and then, comment an extract of translations from the NMT engines' output to analyse and review the results of their performance.

Firstly, though, before using either of the evaluation methods, we ought to obtain the results of the NMT engines by passing the respective test sets through the competent NMT model. This process involves tokenizing all of the test sets with the BPE subword encoding models created from the training sets. Then, we use the latest checkpoint of every model to translate both the raw file and the preprocessed file with the same filter as the given NMT system, as explained in Section 3.3.3. Thus, for example, we use the model of the NMT system trained on the data set with the foreign language filter applied to test both the raw, unprocessed test set file and the filtered, preprocessed test set file.

The translate function from OpenNMT also offers the option to include the $<UNK>$ token for unknown words, or force the model to use the most appropriate word that could be computed. We evaluate both scenarios to compare the results, as we want to see how the NMT engine will respond to unknown words and how much does it affect the outcome of the automatic evaluation metrics.

Now, onto the evaluation methods.

5.1 Automatic evaluation metrics

Beginning with our automatic evaluation metrics, we will separate them in two different categories: trained and non-trained metrics. The trained metrics include BLEURT (Sellam et al., 2020) and Comet (Rei et al., 2020); the non-trained metrics are BLEU (Papineni et al., 2002) and chrF++ (Popović, 2017). In the following sections, we recall their history and provide a short explanation of how they work.

5.1.1 Trained metrics

BLEURT

BLEURT (BiLingual Evaluation Understudy with Representations from Transformers) (Sellam et al., 2020) has gained its place in the MT community in the last few years

due to its reliability. BLEURT was created out of necessity for an advanced metric, due to the inability of other evaluation metrics to correlate with human evaluation.

The difference of BLEURT and other automatic evaluation metrics is its novel pre-training scheme that uses millions of synthetic examples to help it generalise. Like so, it can provide more accurate and reliable results, though it takes longer to compute. BLEURT is a metric that is based on BERT (Kenton and Toutanova, 2019), a language model designed to pre-train deep bidirectional representations from unlabelled text, and RemBERT (Chung et al., 2020), a rebalanced multilingual BERT.

This evaluation metric offers a Python programme[6] with a thorough documentation on its utility. In order to compute the results, we used the *BLEURT-20* checkpoint, a more accurate multilingual model.

The results for BLEURT range from 0 to 1, and where 0 indicates a random output and 1 a perfect one.

Comet

Similarly to BLEURT, the Comet metric (Rei et al., 2020) is one of the top-ranking MT evaluation choices for researchers. Comet is capable of reaching state-of-the-art levels of correlation with human judgements.

Comet mainly stands out due to its possibility to also take the source segment as an argument, and not only the target hypothesis and reference. What is more, it is not solely an automatic evaluation metric, but also a neural framework for training custom evaluation models. Currently, it offers up to 12 pre-trained evaluation models, some of which do not even require a reference. It also covers over 100 languages, including English and Spanish.

Via GitHub, Comet offers a Python programme[7] with all of aforementioned tools available. Most importantly, and something worthwhile noting, it is possible to calculate the score of multiple systems as well as running a command to obtain statistical significance with Paired T-test and bootstrap resampling as per Koehn (2004). We will make use of this option as well to compare our NMT engines and declare whether their results are statistically relevant (see Section 5.1.4 for more details).

Results were calculated via the default model *wmt-20-comet-da*, a reference-based regression model built on top of the large instance of XLM-R (Conneau et al., 2020), a transformer-based multilingual masked language model.

Just as with BLEURT, Comet's results are quantified with a score from 0 to 1, and the higher it is, the better the results.

5.1.2 Non-trained metrics

BLEU

The method for automatic evaluation BLEU (BiLingual Evaluation Understudy) (Papineni et al., 2002) is one of the most, if not the most, popular and well-known metrics in Machine

[6] https://github.com/google-research/bleurt
[7] https://github.com/Unbabel/COMET

Translation (MT). BLEU was created with the idea in mind of substituting (to a certain extent) human evaluations due to their expensive and time-consuming nature.

BLEU works by comparing a given candidate sentence to one or more reference sentences. The comparison is done in both word and phrase-level via n-gram precision and recall. It is highly recommended to include more than one reference to get a proper, more accurate BLEU result. In our case, however, we only counted with one reference.

For the BLEU metric calculation, we referred to an NLTK (Bird et al., 2009) package that implements the BLEU score calculation[8]. Scores for BLEU range from 0 to 1, with higher results meaning better outcome.

chrF++

chrF++ (Character *n*-gram F-score ++) was developed by Popović (2017) and is another automatic evaluation tool designed for MT output. It works especially well for morphologically rich target languages, and such is the case with Spanish, our chosen target language. chrF++ is based off of chrF (Popović, 2015), the base algorithm which is focused on character *n*-gram F-score solely. Later on, chrF+ was proposed by adding word unigrams to the standard chrF, and then, chrF++, which adds both word unigrams and bigrams to the standard chrF.

chrF++ works similarly to BLEU, and as we have mentioned previously, the main difference is it places its focus on character n-gram precision and recall, whereas BLEU's focus is at word level.

The code for computing chrF++ can be found at GitHub[9]; it is quite straightforward and easy to use. The programme outputs the starting time, the overall document-level F-score, the overall macro-averaged document-level F-score (which is the arithmetic average of the sentence-level scores) and the ending time. In our results, we add the overall macro-averaged F-score. The qualitative results work the same as with BLEU: from 0 to 1, with results skirting 1 having a better outcome.

5.1.3 Results

Gathered in Table 5.1 are the BLEU, chrF++, BLEURT and Comet results for our test sets as explained in Section 3.3.3.

Highlighted in bold letters are the best results for BLEU (40.78), chrF+ (67.54), BLEURT (71.71) and Comet (66.62). The BLEU best result corresponds to the NMT engine trained with the data set with all the filters applied, tested on the filtered test set and including the <UNK> token; the BLEURT highest result belongs to NMT engine trained with the dataset with the *Foreign and untranslated* filter applied, tested on the filtered test set which does not include the <UNK> token; and, lastly, chrF++ and Comet agree on the best result being the NMT engine trained with the *Constrained length*,

[8]https://www.nltk.org/_modules/nltk/translate/bleu_score.html
[9]https://github.com/m-popovic/chrF

NMT model	Test set[1]	UNK/no	BLEU	BLEURT	chrF++	Comet
Raw	Raw	UNK	38.32	68.98	63.95	45.46
		no UNK	36.63	66.29	66.04	61.19
Repeating symbols	Raw	UNK	37.02	66.26	62.30	37.27
		no UNK	37.89	66.98	63.37	53.23
	Filtered	UNK	40.37	69.06	66.26	52.40
		no UNK	*40.77*	70.25	66.93	62.83
Constrained length	Raw	UNK	38.98	69.41	66.26	53.33
		no UNK	39.55	70.73	67.00	64.79
	Filtered	UNK	40.14	70.26	66.83	55.44
		no UNK	40.36	*71.60*	**67.54**	**66.62**
Foreign and untranslated	Raw	UNK	40.01	69.86	66.34	54.61
		no UNK	39.97	71.09	66.76	65.56
	Filtered	UNK	40.45	70.42	66.77	55.96
		no UNK	40.44	**71.71**	*67.19*	*66.13*
All filters	Raw	UNK	36.34	65.32	61.85	34.92
		no UNK	36.40	65.00	62.11	47.46
	Filtered	UNK	**40.78**	69.77	66.21	53.50
		no UNK	40.73	70.92	66.62	63.75

[1] Whether the test set file was preprocessed with the same filter as the NMT engine or left raw, as explained in Section 3.3.3.

Table 14: Results of the automatic metrics evaluation with BLEU, BLEURT, chrF++ and Comet. Results have been amplified to a scale from 0 to 100 for easier understanding and better visualisation.

tested on the filtered test set that does not include the <UNK> token. We also decided to highlight in cursive letters the second best result according to each metric, as the second best result for BLEURT agrees with the first best result for chrF++ and Comet, and the second best result for chrF++ and Comet agree with the first best result for BLEURT. This trend shows a consistency in the results for chrF++, BLEURT and Comet.

Interestingly enough, the NMT engine trained with the data set that had all of the filters applied and was tested on the raw test sets has yielded the worst BLEU, chrF++, BLEURT and Comet results out of all of the engines. However, when it was tested with the filtered test set including the <UNK> token, it yielded the highest BLEU result of all of the systems, but it still ranked very low for the other metrics: 12th for chrF++, 10th for BLEURT and 11th for Comet. When tested against the same test set but without the <UNK> token, it placed 3rd in the BLEU scale. The same combination resulted in the 4th place for the BLEURT score, 8th for the chrF++ score and 5th for Comet.

It came as a bit of a surprise that the NMT system trained with the data set which filtered the length of the segments ranked so high for three of the metrics: as we mentioned previously, this NMT engine when tested against the filtered data set that does not include

the <UNK> token, it ranks in first place for chrF++ and Comet, and second place for BLEURT. A deeper study on the effect of segment length in NMT might be of interest as future work.

The final interesting engine that we would like to remark is the NMT engine trained on the data set with the foreign language and untranslated filter, which ranks first in the BLEURT scores, and second in the chrF++ and Comet scores, when tested against the filtered test set with no <UNK> token. These results correlate with those of Khayrallah and Koehn (2018), where they studied the impact of noise in NMT doing the reverse of our work: adding various types of synthetic noise to a data set and training different NMT systems to test the results and the damage of each type of noise. They found, as did we, that untranslated segments severely damaged the NMT engine and caused it to copy the source data onto the translation on many occasions.

All in all, Table 14 shows a slight yet informative success in noise filtering. On the one hand, we have an increase of 4.44 BLEU score and of 5.69 chrF++ score between the lowest ranking NMT engine and the highest ranking engine for each metric. On the other hand, we have an increase of 6.71 BLEURT score and an impressive 31.7 Comet score increase between the NMT system with the highest score and the one with the lowest.

5.1.4 T-test: are the results statistically relevant?

We have visualised the results of our automatic metrics and given a thorough explanation of the outcome. We have been able to notice relevant score increases between the results of the lowest and highest ranking NMT systems. However, how truly relevant are these results? For the answer to that question, we rely on the t-test, a statistical response to determine whether the results are enough to support our initial hypothesis that supports that by applying these filters, we will get a relevant increase in the quality of the NMT engines.

In order to do this, we use Comet's statistical significance Paired T-test based on Koehn (2004), which focused on finding bootstrap resampling methods to compute statistical significance tests to validate MT output.

In order to work properly, the input of the t-test needs a source file (the source segment), a reference file (the target segment), and at least two translation files from which the t-test will be computed. All of the provided files require the same amount of segment length. We previously explained in Section 3.3.3 that we have two different test sets for every NMT engine: one raw, and one filtered with the same cleaning method that was used for the training set with which the given NMT engine was trained. For instance, to test the *Foreign and untranslated* NMT engine, we have the raw test set, with 10,000 segments (see Table 11 for more information regarding the test sets' characteristics), and the filtered test set, with 8,465 segments.

Given this requirement, we can only test those NMT engines that were used to translate the raw test sets, as they were never filtered and they did not have any given amount of segments removed. Thus, we compare each NMT engine against the others, and we do this for both the test set which includes the <UNK> token and the one who does not include

it. Results of the t-test can be found in Table 15, computed with a p value of 0.05, where going over the value would prove no statistically significant difference, and being under the value proves a statistically significant difference.

Model 1	**Model 2**	**UNK/no**	**Null hypothesis**	**p value**	**Best model**
Raw	All	UNK	Rejected	0	Raw
		no UNK	Rejected	0	Raw
	Repeating symbols	UNK	Rejected	0	Raw
		no UNK	Rejected	0	Raw
	Constrained length	UNK	Rejected	0	Constrained length
		no UNK	Rejected	0	Constrained length
	Foreign and untranslated	UNK	Rejected	0	Foreign and untranslated
		no UNK	Rejected	0	Foreign and untranslated
All	Repeating symbols	UNK	Rejected	0	Repeating symbols
		no UNK	Rejected	0	Repeating symbols
	Constrained length	UNK	Rejected	0	Constrained length
		no UNK	Rejected	0	Constrained length
	Foreign and untranslated	UNK	Rejected	0	Foreign and untranslated
		no UNK	Rejected	0	Foreign and untranslated
Foreign and untranslated	Repeating symbols	UNK	Rejected	0	Foreign and untranslated
		no UNK	Rejected	0	Foreign and untranslated
	Constrained length	UNK	Rejected	0.0037	Foreign and untranslated
		no UNK	Rejected	0.0397	Foreign and untranslated
Repeating symbols	Constrained length	UNK	Rejected	0	Constrained length
		no UNK	Rejected	0	Constrained length

Table 15: Results of the statistical significance t-test performed on the NMT engines with a p value of 0.05.

Observing the results on Table 15, we can now objectively, according to statistical significance, analyse the performance of the NMT engines. At first glance, we easily see that all null hypothesis have been rejected, meaning that all comparisons yield a statistically significant difference that allows us to safely say that certain models outperforming others well enough to be significant. For most of the comparisons, the p value stays at 0, minimally increasing for the comparison of the *Foreign and untranslated* engine and the *Constrained length* one. This increase in the p value matches the results of the automatic metrics, which highlight these two as the top 2 NMT engines according to BLEURT, chrF++ and Comet.

Lastly, we can also discern that the *Foreign and untranslated* NMT engine outperforms the rest of the engines at all times (8 out of 8), the *Constrained length* engine outperforms all but the *Foreign and untranslated* engine (6 out of 8), the *Repeating symbols* only outperforms the *All* filters engine (2 out of 8), whereas the *Raw* engine outperforms the *All* filters and the *Repeating symbols* engine (4 out of 8) and, finally, the *All* filters is outperformed by its rival model at every comparison. As commented on the previous paragraph, these results correctly align with those of the BLEURT, chrF++ and Comet results, but not entirely with the BLEU results which place the *All* filters NMT engine as the highest ranking of all.

5.2 Manual evaluation

Admittedly, automatic evaluation metrics are extremely helpful to get a rough quality estimate of long lists of sentences that need evaluation. Nonetheless, human assessment allows for increased reliability, accuracy and more nitpicking of the outcome, being easier to determine errors that can go undetected due to linguistic properties that an automated process is yet unable to excel at detecting, such as context, metaphors, idioms and popular references. With this in mind, we decided to do a short review in terms of the qualitative results of the NMT engines.

We have selected 10 random segments from the raw test files and supplied the correspondent segment for each NMT model that was trained. The segments are selected by using Python (Van Rossum and Drake, 2009) to print a random number between 1 and 10,000 (the total number of segments in the test sets). In this case, in contrast with the automatic evaluation metrics and for the sake of simplicity, we will only include the test set version with that includes the <UNK> tokens and that is raw, not filtered, given that the filtered version has been cleaned and can exclude the randomly selected segment, as some filters remove segments if need be, for instance, those that are in a foreign language.

The selected segments can be in Section 5.2.1, where we will comment them.To qualify and comment the segments, we will make a brief assessment inspired in Tables 1, 2 and 3 as we explained in Section 3.1. As a quick reminder (more thoroughly explained in said section), we will only refer to points I, II, IV and V for the Six Traits model as they relate more to the task of translation, whereas the others relate more to tasks such as text generation. Results of our qualitative evaluation can be found in Table 16.

Note that the parallel target segment extracted from the corpus may not always be as

precise, accurate and well-composed as could be; after all, the corpus was extracted from Wikipedia, where anyone is free to write and translate articles, with no assurance of its quality.

5.2.1 Qualitative results

Segment 1

Source (EN, test subject): *During the war between the US-backed Contras and the government of the Sandinistas in the 1980s, much of the country's infrastructure was damaged or destroyed.*

Target (ES, reference): *Durante la guerra entre los Contras y Sandinistas en los años 80, mucha de la infraestructura del país fue dañada o destruida.*

NMT engines organised by filters, and their output listed

- **Raw:** Durante la guerra entre los Contras respaldados por Estados Unidos y el gobierno de los sandinistas en la década de 1980, gran parte de la infraestructura del país fue dañada o destruida.
- **Repeating symbols:** Durante la guerra entre los Estados Unidos respaldó a Contras y el gobierno de los sandinistas en la década de 1980, gran parte de la infraestructura del país fue dañada o destruida.
- **Constrained length:** Durante la guerra entre las estupas respaldadas y el gobierno de los sandinistas en la década de 1980, gran parte de la infraestructura del país fue dañada o destruida.
- **Foreign and untranslated:** Durante la guerra entre Estados Unidos y el gobierno de los sandinistas en la década de 1980, gran parte de la infraestructura del país fue dañada o destruida.
- **All filters:** Durante la guerra entre los Estados Unidos y el gobierno de los sandinistas en la década de 1980, gran parte de la infraestructura del país fue dañada o destruida.

Starting with the first segment, we have a mid-length sentence presenting some challenges (like *US-backed*; *Contras* and *Sandinistas* (terminology). First off, we see that the target sentence is missing some pieces of information (that the Contras were US-backed, and that the Sandinistas were a government). Moving on to the translations of the NMT engines, the **Raw** one is without a doubt the most accurate of all. It includes all the information present in the source segment and, contrary to the rest of the engines, it correctly translated *US-backed Contras*; the others alleged that the US was one of the sides of the war, and some even skipped the *Contras*. The writing in general, however, is very high

quality in regards to organisation, how the word placement is chosen, there is fluency in the segments and most of the word choices (except for **Constrained length**, given that *estupas* here would not make any sense) are accurate. The only exception is **Repeating symbols**, given that the first part of the sentence makes no sense and throws the reader off from the beginning.

Segment 2

Source (EN, test subject): *Regression is also similar, but it takes into account that the data is imprecise.*

Target (ES, reference): *La regresión es también similar, pero tiene en cuenta que los datos son imprecisos.*

NMT engines organised by filters, and their output listed

- **Raw:** La regresión también es similar, pero tiene en cuenta que los datos son impredecibles.
- **Repeating symbols:** La regresión también es similar, pero tiene en cuenta que los datos son incontables.
- **Constrained length:** La regresión es también similar, pero tiene en cuenta que los datos son implantados.
- **Foreign and untranslated:** La regresión es también similar, pero toma en cuenta que los datos son imposibles.
- **All filters:** La regresión es también similar, pero tiene en cuenta que los datos son impecables.

For the second segment, we have a very short testing agent that allows for quick assessment. The target segment presents no issues; and there are no straight-forward challenges to be aware of. We can observe that, unfortunately, all engines except the **Raw** one used the wrong adjective for *imprecise*: uncountable, implanted, impossible and impeccable. This little detail changes the entire meaning of the sentence, deeming it unusable. However, the writing is overall impeccable, and does not lack fluidity and structure organisation.

Segment 3

Source (EN, test subject): *=== Worship of Achilles in modern times: The Achilleion in Corfu ===In the region of Gastouri (Γαστούρι) to the south of the city of Corfu Greece, Empress of Austria Elisabeth of Bavaria also known as Sissi built in 1890 a summer palace*

with Achilles as its central theme and it is a monument to platonic romanticism.

Target (ES, reference): *=== El culto de Aquiles en la época moderna: El "Achilleion" de Corfú ===En la región de Gastouri (Γαστούρι) al sur de la ciudad de Corfú (Grecia), la emperatriz austriaca Sissi construyó en 1890 un palacio de verano con Aquiles como tema principal, en lo que supone un monumento al romanticismo platónico.*

NMT engines organised by filters, and their output listed

- **Raw:** === Culto de Aquiles en tiempos modernos ===En la región de Gastal (en alemán: "ἔστασίύρι") al sur de la ciudad de Corfú Grecia, la emperatriz Isabel de Baviera también conocida como Sissi construyó en 1890 un palacio de verano con Aquiles como su tema central y es un monumento al romanticismo platónico.

- **Repeating symbols:** Fiestas de Aquiles en tiempos modernos: La Achilleión en Corfú En la región de Gastovia (en griego: Κασ Κύρι) al sur de la ciudad de Corfú, emperatriz de Austria Isabel de Baviera también conocida como Sissi construida en 1890 un palacio de verano con Aquiles como tema central y es un monumento al romanticismo platónico.

- **Constrained length:** === Culto de Aquiles en tiempos modernos: La Achilión en Corfú ===En la región de Gastrana (en griego: Παστούρι) al sur de la ciudad de Corfú Grecia, la emperatriz de Austria Isabel de Baviera también conocida como Sissi construida en 1890 un palacio de verano con Aquiles como su tema central y un romanticismo.

- **Foreign and untranslated:** === Culto de Aquiles en tiempos modernos: La Achilleion en Corfu ===En la región de Gastonov (en griego Καστούρι) al sur de la ciudad de Corfu Grecia, la emperatriz de Austria Elisabeth de Baviera también conocida como Sissi construyó en 1890 un palacio de verano con Aquiles como tema central y es un monumento al romanticismo platónico.

- **All filters:** En la región de Gasteup (en griego: Καστούρι) al sur de la ciudad de Corfuilles, la emperatriz de Austria, también conocida como Sissi, es un monumento en el centro de Grecia.

We now encounter the third segment, which is quite a long one. The last test subject present a few challenges: there are symbols that seem to be formatting tags (===); there is a word written with the Greek alphabet (*Γαστούρι*); there are locations (*Corfu*) and names (*Elisabeth of Bavaria*) that require translation and terminology (*platonic romanticism*). The target segment is well translated overall, but we believe it was a mistake to use Elisabeth of Bavaria's nickname *Sissi* alone, without including her full name. Right off the bat, we observe that all translations used the wrong Greek word and wrongly translate *Gastouri* to a variety of options, none correct.

More specifically, the **Raw** segment is missing the translation for the phrase *The Achilleion in Corfu* and it refers to the Greek word as German. Otherwise, the translation is understandable and the meaning is maintained. For the **Repeating symbols**, it removes the symbols from the original (which could be right or wrong depending on the goal; but if it is in the source, generally it should maintain it); it wrongly adapts *Achilleion* to Spanish with the tilde in the name, though in Spanish it should stay as the original or be adapted to *Aquileón*; it is missing some separation (i.e., a dot) after the first *Corfú*; misses to indicate that *Corfu* is in *Greece*; does not make sense entirely and suggested that the empress was the one who was built and not the palace. For the **Constrained length**, *Achilleion* is translated wrong; produces confusion as with the previous NMT system, as it suggests that the empress was the one built; and, lastly, does not specify the type of romanticism mentioned in the source segment. Regarding the **Foreign and untranslated** is the most accurate translation of all, barring the errors mentioned in the last phrase of the previous paragraph; otherwise, it is perfect! Finally, the **All** translation lacks the first segment; uses the wrong term for *Corfu*; and does not include the last part of the source segment, cutting off abruptly and stating that the empress was a monument in the centre of Greece.

Segment 4

Source (EN, test subject): *Similarly, men of African ancestry have significantly higher levels of testosterone than men of European ancestry, and have a correspondingly much higher level of prostate cancer.*

Target (ES, reference): *Del mismo modo, los hombres de ascendencia africana tienen niveles significativamente más altos de testosterona que los hombres de ascendencia europea, con un nivel correspondientemente mucho más alto de cáncer de próstata.*

NMT engines organised by filters, and their output listed

- **Raw:** Similarmente, los hombres de ascendencia africana tienen niveles significativamente mayores de testosterona que los hombres de ascendencia europea, y tienen un nivel mucho más alto de cáncer de próstata.

- **Repeating symbols:** Similarmente, los hombres de ascendencia africana tienen niveles significativamente mayores de testosterona que los hombres de ascendencia europea, y tienen un nivel de cáncer de próstata.

- **Constrained length:** Del mismo modo, los hombres de ascendencia africana tienen niveles significativamente mayores de testosterona que los hombres de ascendencia europea, y tienen un nivel correspondientemente mayor de cáncer de próstata.

- **Foreign and untranslated:** Similarmente, los hombres de ascendencia africana tienen niveles significativamente más altos de testosterona que los hombres de ascendencia europea, y tienen un nivel mucho más alto de cáncer de próstata.

- **All filters:** Similarmente, los hombres de ascendencia africana tienen significativamente mayores niveles de testosterona que los hombres de ascendencia europea, y tienen un nivel de cáncer de próstata.

The fourth segment proves to be a challenge for the *Repeating symbols* and *All filters* when translating the last part. These two NMT engines wrongfully omitted that the prostate cancer corresponds to the higher levels of testosterone, and simply stated that men of African ancestry have *a level* of prostate cancer, which does not make a lot of sense (as the level can be measured in lower or higher, but not in numbers such as one level, two levels, etc.). This issue also lowers the writing comprehension due to its lack of sense.

The other engines performed very well, notably the *Constrained length* engine which used a more native-like connector at the beginning (*Del mismo modo* instead of *Similarmente*) and correctly pointed out that the higher level of prostate cancer is corresponding to that of the testosterone, whereas the NMT engines *Raw* and *Foreign and untranslated* translated that they just have *a much higher prostate cancer level*, not linking it to the testosterone levels with a word such as *corresponding*.

Segment 5

Source (EN, test subject): *===TLS 1.2===TLS 1.2 was defined in RFC 5246 in August 2008.*

Target (ES, reference): *=== TLS 1.2 ===TLS 1.2 fue definido en el RFC 5246 en agosto del 2008.*

NMT engines organised by filters, and their output listed

- **Raw:** === TLS 1.2 ===TLS 1.2 fue definido en el RFC 5246 en agosto de 2008.

- **Repeating symbols:** El RFC 5246 fue definido en el RFC 5246 en agosto de 2008.

- **Constrained length:** === TLS 1.2 ===TLS 1.2 fue definido en RFC 5246 en agosto de 2008.

- **Foreign and untranslated:** TLS 1.2 fue definido en el RFC 5246 en agosto de 2008.

- **All filters:** El RFC 5246 fue definido en el RFC 5246 en agosto de 2008.

Firstly, we can observe that for this segment, the *Repeating symbols*, *Foreign and untranslated* and the *All filters* engines have skipped the first part of the segment (*TLS 1.2*), possibly due to symbols being removed from the training. This can cause important issues in translation, as the goal is to void the translation of unnecessary elements, but translate and keep the text. Moreover, this part is also skipped in the second part of the segment for the *Repeating symbols* and the *All filters* engines, when it is present in the source segment, and substituted by an element that appears later on in the segment (*RFC 5246*); as a result, this element is duplicated, appearing twice in the translations when it originally only appears once.

Moving on, we notice that for the rest of the *Foreign and untranslated* engine, the translation is spot on, and that the *Constrained length* results is almost perfect, though it is missing the article *el* for the *RFC 5246*, which adds a vagueness to it. Finally, we note that the *Raw* translation presents no issues.

Segment 6

Source (EN, test subject): *One example of enzyme deficiency is the most common type of phenylketonuria.*

Target (ES, reference): *Un ejemplo de esto es el tipo más común de fenilcetonuria.*

NMT engines organised by filters, and their output listed

- **Raw:** Un ejemplo de deficiencia de la enzima es el tipo más común de fenilketonuria.
- **Repeating symbols:** Un ejemplo de deficiencia de enzimas es el tipo más común de fenilketonuria.
- **Constrained length:** Un ejemplo de deficiencia enzimática es el tipo más común de fenilketonuria.
- **Foreign and untranslated:** Un ejemplo de deficiencia de enzimas es el tipo más común de fenilketonuria.
- **All filters:** Un ejemplo de deficiencia de enzimas es el tipo más común de fenilketonuria.

Our fifth segment presents an example where the target reference is lacking information that, by looking at it, we infer is present within the context where it was extracted from. Regardless, the NMT engines do a really good job at not omitting this information. In fact, we only note here that the *Raw* translation uses the singular to refer to enzyme, which in English makes a lot of sense, but in Spanish we would use the plural for generalisation, and the singular if we refer to a specific enzyme. Also, the *Constrained length* translation very accurately uses *deficiencia enzimática*, which would approximate more to the structure that a native would use to refer to *enzyme deficiency*.

Segment 7

Source (EN, test subject): *Other more radical left-wing groups that broke away were Partido Comunista de los Trabajadores (formed by the Left Opposition of PCE in 1977) and PCE (VIII-IX Congresos) (formed in 1971).*

Target (ES, reference): *Ese mismo año un sector denominado Oposición de Izquierda (OPI), que había surgido tras el VIII Congreso, abandona el PCE y adopta el nombre de Partido Comunista de los Trabajadores (PCT).*

NMT engines organised by filters, and their output listed

- **Raw:** Otros grupos de izquierda más radicales que se separaron fueron el Partido Comunista de los Trabajadores (formado por la oposición izquierda del PCE en 1977) y el PCE (VIII-IX Congresos) (formado en 1971).
- **Repeating symbols:** Otros grupos de dirección izquierda más radicales que se separaron fueron el Partido Comunista de los Trabajadores (formado por la Oposición de Izquierda del PCE en 1977) y el PCE (VIII GUIX Congreso) (formado en 1971)
- **Constrained length:** Otros grupos de izquierda más radicales que se separaron fueron el Partido Comunista de los Trabajadores (formado por la Oposición de Izquierda del PCE en 1977) y el PCE (VIII-IX Congresos) (formado en 1971).
- **Foreign and untranslated:** Otros grupos de izquierdas más radicales que se separaron fueron el Partido Comunista de los Trabajadores (formado por la Oposición de Izquierda de PCE en 1977) y el PCE (VIII-IX Congresos) (formado en 1971).
- **All filters:** Otros grupos de izquierda más radicales que se separaron fueron el Partido Comunista de los Trabajadores (formado por la Oposición de Izquierda de PCE en 1977) y PCEVIII Congresos (formado en 1971)

In this case, segment 7 is missing quite a bit of information on the target reference segment compared to the source segment, and information is not organised correctly. Similarly to segment 6, the NMT engines translated correctly following the source segment and did not follow the chaos of the target reference.

From the results, we observe that the *Repeating symbols* and *All* engines make the most mistakes: the *Raw* engine incorrectly translates *left-wing* as *left direction*, and adds an unwanted character to *VIII-IX Congress* which can be confusing; the *All* engine makes a similar mistakes in for the *VIII-IX Congress* part, but otherwise performs a good translation.

The rest of the engines do a perfect job in the translation, with only needing to note that the *Raw* engine does not capitalise *Oposición Izquierda*, though it should.

Segment 8

Source (EN, test subject): *In the Haber process for the production of ammonia, hydrogen is generated from natural gas.*

Target (ES, reference): *En el proceso de Haber para la producción de amoníaco, se genera hidrógeno a partir de gas natural.*

NMT engines organised by filters, and their output listed

- **Raw:** En el proceso Haber para la producción de amoníaco, el hidrógeno se genera a partir de gas natural.
- **Repeating symbols:** En el proceso de Haber para la producción de amoníaco, el hidrógeno se genera a partir de gas natural.
- **Constrained length:** En el proceso Haber para la producción de amoniaco, el hidrógeno es generado de gas natural.
- **Foreign and untranslated:** En el proceso de Haber para la producción de amoníaco, el hidrógeno se genera a partir del gas natural.
- **All filters:** En el proceso de Haber para la producción de amoníaco, el hidrógeno es generado por gas natural.

Reviewing this segment, we want to remark that the translations are very well done, and though some vary, the meaning is conveyed for all except the *All filters* engine, because at the end it uses the wrong preposition, which does not assimilate to the way a native speaker would segment it and makes the segment sound odd, though the original meaning is understandable.

Segment 9

Source (EN, test subject): *The fourth section for version 2 of the license and the seventh section of version 3 require that programs distributed as pre-compiled binaries are accompanied by a copy of the source code, a written offer to distribute the source code via the same mechanism as the pre-compiled binary, or the written offer to obtain the source code that the user got when they received the pre-compiled binary under the GPL.*

Target (ES, reference): *La cuarta sección de la versión 2 de la licencia y la decimoséptima sección de la versión 3 requieren que que los programas distribuidos como binarios precompilados estén acompañados de una copia del código fuente, una oferta por escrito para distribuir el código fuente a través del mismo mecanismo que el binario precompilado, o una oferta por escrito para obtener el código fuente del binario recibido bajo*

la GPL.

NMT engines organised by filters, and their output listed

- **Raw:** La cuarta sección para la versión 2 de la licencia y la séptima sección de la versión 3 requieren que los programas distribuidos como binarios pre compilado están acompañados por una copia del código fuente, una oferta escrita para distribuir el código fuente a través del mismo mecanismo que el pre- compilado binario, o la oferta escrita para obtener el código fuente que el usuario tiene cuando recibieron el pre-compilado binario bajo la GPL.
- **Repeating symbols:** La cuarta sección para la versión 2 de la licencia y la séptima sección de la versión 3 requieren que los programas distribuidos como binarios preógenos compactos están acompañados por una copia del código fuente, una oferta escrita para distribuir el código fuente a través del mismo mecanismo que la binaria compactada, o la oferta escrita para obtener el código fuente que el usuario obtuvo cuando recibieron el binario compilado antes de la GPL.
- **Constrained length:** La cuarta sección para la versión 2 de la licencia y la séptima sección de la versión 3 requieren que los programas distribuidos como binarios precompilados sean acompañados por una copia del código fuente, una oferta escrita para distribuir el código fuente a través del mismo mecanismo que el binario pre-compilado, o la oferta escrita para obtener el código fuente que el usuario recibiera cuando recibieron la pre-compiladora bajo la GPL.
- **Foreign and untranslated:** La cuarta sección de la versión 2 de la licencia y la séptima sección de la versión 3 requieren que los programas distribuidos como binarios precopilados están acompañados por una copia del código fuente, una oferta escrita para distribuir el código fuente a través del mismo mecanismo que el pre- compilado binario, o la oferta escrita para obtener el código fuente que el usuario obtuvo cuando recibían el pre- compilado bajo la GPL.
- **All filters:** La cuarta sección para la versión 2 de la licencia y la séptima sección de la versión 3 requieren que los programas distribuidos como binarios preconformes son acompañados por una copia del código fuente, una oferta escrita para distribuir el código fuente a través del mismo mecanismo que el servidor binario predeterminado.

At first glance, we notice that the target segment wrongfully indicates the *seventeenth section* instead of *seventh*, but is otherwise greatly translated. Looking at our NMT systems' translations, we see several errors. The most notable one, in our opinion, is the third plural person used for *user*, which in English is used when the gender of the named person is unknown, and has been wrongly translated in the same form to Spanish, but should have been kept to singular masculine as is the norm in Spanish. Then, there are a variety of translations for *pre-compiled binary*, with a space between *pre* and *compiled*, or between

either word and the slash -; it translates better as *precompilado*. It was also wrongfully translated as *binaria compactada* in one occasion, and as *binarios preógenos*, a term that does not exist in Spanish (for the *Repeating symbols engine*). We also see some confusion with the tenses used, using both conditional and past tense to refer to the same thing (i.e., for *Constrained length*: [...] *que el usuario* ***recibiera*** *cuando* ***recibieron*** *la* [...]).

All in all, we witness a generally good translation but some important mistakes overall for most of the engines, notably from a grammatical point of view; that the meaning is conveyed does not mean that a segment is well-written. However, we see that once again the *All filters* engine has skipped relevant information when translating.

Segment 10

Source (EN, test subject): *Alhama Valley.*

Target (ES, reference): *Valle del Alhama.*

NMT engines organised by filters, and their output listed

- **Raw:** Alhama Valley.
- **Repeating symbols:** Alhama Valley.
- **Constrained length:** Alhama Valley.
- **Foreign and untranslated:** Alhama Valley.
- **All filters:** Alhama Valley.

Lastly, we have the 10th segment, for which all of our NMT engines have failed to properly translate the source segment, yielding an untranslated segment.

Apart from commenting the results of the segments following Tables 1, 2 and 3, we have reflected on Table 16 a summary of our findings, with a quantitative measure based off of the aforementioned tables; since they include 4 different points, we have estimated a mark for each segment given how well translated and written is was, guided by the points in the tables. Some segments may not convey the source meaning in the translated segment, but be very well-written, and some others may maintain all the meaning but have grammatical errors that difficult the reading and understanding of a given sentence.

As a last note, we would like to note that the results of Table 16 are not as reliable as the automatic evaluation metrics due to its subjectivity; however, it is quite surprising to see these compared to the automated metrics. The results here indicate that the *Raw* engine is one of the best translators, though all automatic metrics ranked it in the last 5 positions out of 18, with lower scores than their counterparts. We also observe that the **All filters** system was ranked first for BLEU, but with our samples it is performing poorly,

Segment 1					
	Raw	Repeating	Length	Foreign lang	All
Own standards (Table 1)	4/4	2/4	2.5/4	2.5/4	2.5/4
Six Traits (Tables 2 and 3)	4/4	1.5/4	3/4	4/4	4/4
Segment 2					
	Raw	Repeating	Length	Foreign lang	All
Own standards (Table 1)	4/4	1.75/4	1.75/4	1.75/4	1.75/4
Six Traits (Tables 2 and 3)	4/4	4/4	3.25/4	4/4	3/4
Segment 3					
	Raw	Repeating	Length	Foreign lang	All
Own standards (Table 1)	2/4	2.5/4	2.25/4	4/4	0/4
Six Traits (Tables 2 and 3)	2/4	2/4	2/4	3.75/4	0/4
Segment 4					
	Raw	Repeating	Length	Foreign lang	All
Own standards (Table 1)	3.75/4	2.75/4	4/4	3.75/4	2.75/4
Six Traits (Tables 2 and 3)	4/4	2/4	4/4	4/4	2/4
Segment 5					
	Raw	Repeating	Length	Foreign lang	All
Own standards (Table 1)	4/4	1/4	3.5/4	3/4	1/4
Six Traits (Tables 2 and 3)	4/4	2/4	4/4	4/4	2/4
Segment 6					
	Raw	Repeating	Length	Foreign lang	All
Own standards (Table 1)	3/4	3.75/4	4/4	3.75/4	3.75/4
Six Traits (Tables 2 and 3)	2.75/4	3.75/4	4/4	3.75/4	3.75/4
Segment 7					
	Raw	Repeating	Length	Foreign lang	All
Own standards (Table 1)	3.75/4	2.5/4	4/4	4/4	3.5/4
Six Traits (Tables 2 and 3)	4/4	3/4	4/4	4/4	3.5/4
Segment 8					
	Raw	Repeating	Length	Foreign lang	All
Own standards (Table 1)	4/4	4/4	4/4	4/4	3/4
Six Traits (Tables 2 and 3)	4/4	4/4	4/4	4/4	3/4
Segment 9					
	Raw	Repeating	Length	Foreign lang	All
Own standards (Table 1)	3.25/4	2.75/4	3.75/4	3.75/4	0.5/4
Six Traits (Tables 2 and 3)	1.75/4	1.5/4	4/4	3.5/4	3.25/4
Segment 10					
	Raw	Repeating	Length	Foreign lang	All
Own standards (Table 1)	0/4	0/4	0/4	0/4	0/4
Six Traits (Tables 2 and 3)	0/4	0/4	0/4	0/4	0/4
Total					
Own standards	**31.75**/40	23/40	25.75/40	30.5/40	18.75/40
Six Traits (Tables 2 and 3)	30.5/40	23.75/40	32.25/40	**35**/40	24.5/40

Table 16: Qualitative results of ten random segments, following Tables 1, 2 and 3.

occasionally leaving out information that is omitted in the translation and using the wrong terms, as we saw in our overview of the extracted segments; it also ranks, according to Table 16, as the 4th in terms of the Six Traits model, ranking low for its poor writing skills. On the other hand, the *Foreign and untranslated* engine ranks the first for the Six Traits model scoring, with a very well written style in general. This engine also ranked second in terms of translation quality according to our own standards, reflected in Table 1.

All in all, we agree that the overall results concur with those of the statistical significance t-test in Table 15 and automatic evaluation metrics in Table 14, notably those for BLEURT, chrF++ and Comet and more so for the *Constrained Length*, *Repeating symbols*, *All filters* and *Foreign and untranslated* engines. The most surprising result of all is that of the *Raw* engine, which shows good translation and writing skills despite ranking relatively low in the automatic evaluation metrics table.

5.3 Correlation with initial hypothesis

Now that we have given a rigorous explanation of the findings of the automatic evaluation metrics and a manual evaluation, we now seek to link the results with our initial hypothesis via the questions laid out in Section 1.1:

- Does the quality of the data affect the NMT output?

  The results in Sections 5.1 and 5.2.1 have shown that certain filters can appreciably improve the quality output of NMT systems. Most notably, when removing sentences that are too short, too long or with a significant character difference (with justified thresholds as explained in Section 3.3.1), or sentences that have not been translated or are in a foreign third language to those of the source and target segments.

- How can we best measure the quality of different NMT engines' output and the extent to which it affects it?

  Our best attempt at quantifying the quality of the NMT engines trained here is done via the state-of-the-art and popular, well-known automatic evaluation metrics and a brief manual evaluation that correlates with the automatic metrics' findings. Both these methods have given us a certain margin to be able to illustrate, quantify and qualify the effect that the filters have had in the quality of the NMT systems.

- What types of noise are the least beneficial and most harmful for the NMT engines?

  n We have observed that the best resulting systems are those who have had untranslated and foreign segments removed, as well as segments with a compromising length (as explained in the reply to the first question). Now, when it came to applying both these filters alongside removing repeating characters and uncertain symbols, it proved to result in worse translation quality, even more so than that of the NMT engine trained with the raw data.

6 Conclusion

Our thesis project has revisited, reviewed and reconsidered the topic of data quality in Neural Machine Translation (NMT), bringing more attention and awareness to its importance and impact on the data set meant as training material. We hoped to reveal a hidden improvement in NMT engines that pertained to certain cleaning methods to rid their corpora of the most common types of noise.

We made use of a noisy corpus (extracted from Wikipedia and obtained via the Tiedemann (2012) project) to verify if our initial hypothesis would prove true: clearing a corpus of its noise will yield a higher quality output than if it is left with the raw data. Having a noisy corpus allowed us to have a clear distinction of cleaned, preprocessed training sets and raw, unprocessed and noisy training sets. Inspired by Khayrallah and Koehn (2018), we set out to analyse our corpus and categorise it in different categories according to noise type (segment length, foreign language and untranslated segments and repeating characters and symbols), which was done via a sample subcorpus of 200 segments. Results were then applied to the entire corpus, of nearly 1,800,000 sentences.

Thanks to the OpenNMT (Klein et al., 2017) project, we were able to train 5 different NMT systems, with a different training set each, which allowed us to compare the results of different cleaning methods and their effect on the translations' output. For reproducibility and accuracy, all NMT engines were trained with the same characteristics.

Relying on automatic evaluation metrics, we first computed the results using BLEU (Papineni et al., 2002), chrF++ (Popović, 2017), BLEURT (Sellam et al., 2020) and Comet (Rei et al., 2020). The results for BLEU showed a different trend to those of the rest of the metrics, placing the system trained with the training set that had all of the filters applied in the first position, whereas the other metrics placed the engines with the *Constrained length* and the *Foreign and untranslated* in first and second place, respectively. Overall, however, we saw a significant increase in the results. These results were also backed by the statistical significance t-test, which successfully rejected all the null hypotheses and confirmed that the increase in performance and output quality, when comparing the NMT engines between each other using as reference the target file, was statistically relevant.

To provide more assurance and reliability, however, we set out to manually inspect a total of ten sentences from the test set and their translations. We were slightly surprised to find that the BLEU results did not correlate with what we observed in the sentences: the NMT engine trained with the data set that had all the filters applied performed significantly worse than the rest in the majority of occasions. However, the rest of the automatic evaluation metrics and the t-test already pointed towards this idea. These were more accurate in their results when we reviewed the sentences, ranking in the first two places the NMT engine trained with the data set preprocessed with the foreign language and untranslated segments filter, and the NMT engine trained with the training set that had the constrained length filter applied, both which overall excelled in the qualitative results and correlated well with our manual evaluation.

Nonetheless, what neither the automatic metrics were able to measure in their results and the t-test spotted is that the NMT engine that was trained with the raw, original

files, without undergoing any preprocessing, also excelled in the manual evaluation. It is true that the t-test noted that the this engine outperformed the NMT engine trained with the data set that had all the filters applied and the NMT engine that had all of the symbols removed, but according to our manual evaluation, the NMT engine trained with the raw data obtained the best translation results. Though these results are more subjective and can vary between evaluators, we believe our results are quite justified following the assessments indicated in Section 3.1. Note, nonetheless, that our manual evaluation may not reflect the results of the entire dataset, as its size is not enough to correctly reflect the entire test set and is a mere representation.

7 Future work

While researching, working and writing this thesis, we have been able to explore different aspects of noise in NMT that have raised our curiosity in how we could explore this topic differently, with the privilege of having additional insight into the issue at hand.

We would be interested in researching a bit deeper into other noise cleaning methods, specifically from a linguistic point of view. We also raise the following questions: how much can the difference of unique words, synonyms, or similar, affect translation?; or, can we detect and remove or fix sentences that are positioned incorrectly, that have misordered words and missing information?

Another interesting idea is to repeat a similar experiment on corpora from other fields and with a bigger and lower size than our current subject.

We also mentioned in Section 3.3.3 the idea of researching how the preprocessing of the NMT engine input affects its output quality. Our trained systems have showed a tendency towards higher leniency when tested against test sets that were filtered with the same methods as the corresponding NMT engine.

Finally, we believe it would also be an interesting topic of research to consider sentence length when translating, as it has shown certain promise amongst our trained NMT engines: were the short sentences inhibiting the rest of the engines? Or was it the long sentences? Where could the line be drawn to maximise quality output in translation? How challenging would it be for our engine to translate longer or shorter sequences?

References

Foreign Language Training - United States Department of State, Mar 2022. URL https://www.state.gov/foreign-language-training/.

Natalie Ahn. Rule-Based Spanish Morphological Analyzer Built From Spell Checking Lexicon. *CoRR*, abs/1707.07331, 2017. URL http://arxiv.org/abs/1707.07331.

Lucía V Aranda. *Handbook of Spanish-English Translation.* University Press of America, 2007.

Mikko Aulamo, Sami Virpioja, Jørg Tiedemann, et al. OpusFilter: A Configurable Parallel Corpus Filtering Toolbox. In *58TH Annual Meeting of the Association for Computational Linguistics (ACL 2020): System Demonstrations*. The Association for Computational Linguistics, 2020.

Eleftherios Avramidis and Philipp Koehn. Enriching Morphologically Poor Languages for Statistical Machine Translation. In *Proceedings of ACL-08: HLT*, pages 763–770, 2008.

Saptarashmi Bandyopadhyay, Tasnim Kabir, Zizhen Lian, and Marine Carpuat. The University of Maryland, College Park Submission to Large-Scale Multilingual Shared Task at WMT 2021. In *Proceedings of the Sixth Conference on Machine Translation*, pages 383–386, 2021.

Chao Bei, Hao Zong, Conghu Yuan, Qingming Liu, and Baoyong Fan. GTCOM Neural Machine Translation Systems for WMT19. In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 116–121, 2019.

Yonatan Belinkov and Yonatan Bisk. Synthetic and Natural Noise Both Break Neural Machine Translation. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=BJ8vJebC-.

Steven Bird, Ewan Klein, and Edward Loper. *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit.* O'Reilly Media, Inc., 2009.

Eleftheria Briakou and Marine Carpuat. Beyond noise: Mitigating the Impact of Fine-grained Semantic Divergences on Neural Machine Translation. *arXiv preprint arXiv:2105.15087*, 2021.

Michael L Brodie. Data Quality in Information Systems. *Information & Management*, 3 (6):245–258, 1980.

Cambridge Dictionary. *Quality.* Cambridge University Press, 2022. URL https://dictionary.cambridge.org/dictionary/english/quality.

Hyung Won Chung, Thibault Fevry, Henry Tsai, Melvin Johnson, and Sebastian Ruder. Rethinking Embedding Coupling in Pre-trained Language Models. *arXiv preprint arXiv:2010.12821*, 2020.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised Cross-lingual Representation Learning at Scale. In *ACL*, pages 8440–8451, 2020. URL https://doi.org/10.18653/v1/2020.acl-main.747.

Wikipedia Contributors. Wikipedia, The Free Encyclopedia, 2004. URL https://en.wikipedia.org/wiki/Main_Page. Online; accessed 29-August-2022.

Paul B Diederich. Measuring Growth in English. page 107, 1974. URL https://eric.ed.gov/?id=ED097702.

Miguel Domingo, Mercedes García-Martínez, Alexandre Helle, Francisco Casacuberta, and Manuel Herranz. How Much Does Tokenization Affect Neural Machine Translation? *CoRR*, abs/1812.08621, 2018. URL http://arxiv.org/abs/1812.08621.

Thomas Eckart, Uwe Quasthoff, and Dirk Goldhahn. Language statistics-based quality assurance for large corpora. In *Proceedings of Asia pacific corpus linguistics conference*, 2012.

Philip Gage. A New Algorithm for Data Compression. *C Users J.*, 12(2):23–38, feb 1994. ISSN 0898-9788.

Dirk Goldhahn, Thomas Eckart, and Uwe Quasthoff. Building large monolingual dictionaries at the leipzig corpora collection: From 100 to 200 languages. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 759–765, 2012.

Amr Hendy, Esraa A Gad, Mohamed Abdelghaffar, Jailan S ElMosalami, Mohamed Afify, Ahmed Y Tawfik, and Hany Hassan Awadalla. Ensembling of Distilled Models from Multi-task Teachers for Constrained Resource Language Pairs. *arXiv preprint arXiv:2111.13284*, 2021.

Derrick Higgins, Jill Burstein, Daniel Marcu, and Claudia Gentile. Evaluating Multiple Aspects of Coherence in Student Essays. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*, pages 185–192, 2004.

W John Hutchins. Machine Translation: A Brief History. In *Concise history of the language sciences*, pages 431–445. Elsevier, 1995.

Ann Irvine and Chris Callison-Burch. Combining Bilingual and Comparable Corpora for Low Resource Machine Translation. In *Proceedings of the Eighth Workshop on Statistical Machine Translation*, pages 262–270, Sofia, Bulgaria, aug 2013. Association for Computational Linguistics. URL https://aclanthology.org/W13-2233.

Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186, 2019.

Huda Khayrallah and Philipp Koehn. On the Impact of Various Types of Noise on Neural Machine Translation. In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*. Association for Computational Linguistics, 2018. doi: 10.18653/v1/w18-2709. URL `https://doi.org/10.18653\%2Fv1\%2Fw18-2709`.

Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander Rush. OpenNMT: Open-Source Toolkit for Neural Machine Translation. In *Proceedings of ACL 2017, System Demonstrations*, pages 67–72, Vancouver, Canada, jul 2017. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/P17-4012`.

Philipp Koehn. Statistical Significance Tests for Machine Translation Evaluation. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 388–395, Barcelona, Spain, jul 2004. Association for Computational Linguistics. URL `https://aclanthology.org/W04-3250`.

Philipp Koehn and Rebecca Knowles. Six Challenges for Neural Machine Translation. In *Proceedings of the First Workshop on Neural Machine Translation*, pages 28–39, Vancouver, aug 2017. Association for Computational Linguistics. doi: 10.18653/v1/W17-3204. URL `https://aclanthology.org/W17-3204`.

Mikołaj Koszowski, Karol Grzegorczyk, and Tsimur Hadeliya. Allegro. EU Submission to WMT21 News Translation Task. In *Proceedings of the Sixth Conference on Machine Translation*, pages 140–143, 2021.

Annie Priyadarshini Louis. *Predicting Text Quality: Metrics for Content, Organization and Reader Interest*. University of Pennsylvania, 2013.

Eleni Miltsakaki and Audrey Troutt. Real Time Web Text Classification and Analysis of Reading Difficulty. In *Proceedings of the third workshop on innovative use of NLP for building educational applications*, pages 89–97, 2008.

Donald M Murray. *Learning by Teaching: Selected Articles on Writing and Teaching*. ERIC, 1982.

ThuyLinh Nguyen, Stephan Vogel, and Noah A Smith. Nonparametric Word Segmentation for Machine Translation. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 815–823, 2010.

Jeroen Ooms. *cld2: Google's Compact Language Detector 2*, 2022. https://docs.ropensci.org/cld2/ (docs) https://github.com/ropensci/cld2 (devel) https://github.com/cld2owners/cld2 (upstream).

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA, jul 2002. Association for Computational Linguistics. doi: 10.3115/1073083.1073135. URL https://aclanthology.org/P02-1040.

Chanjun Park, Sugyeong Eo, Hyeonseok Moon, and Heui-Seok Lim. Should We Find Another Model?: Improving Neural Machine Translation Performance with One-piece Tokenization Method Without Model Modification. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Industry Papers*, pages 97–104, 2021.

Martin Popel. CUNI English-Czech and English-Polish Systems in WMT20: Robust Document-level Training. In *Proceedings of the Fifth Conference on Machine Translation*, pages 269–273, 2020.

Maja Popović. 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, sep 2015. Association for Computational Linguistics. doi: 10.18653/v1/W15-3049. URL https://aclanthology.org/W15-3049.

Maja Popović. chrF++: Words Helping Character N-grams. In *Proceedings of the second conference on machine translation*, pages 612–618, 2017.

Alan C Purves. Reflections on Research and Assessment in Written Composition. *Research in the Teaching of English*, pages 108–122, 1992.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. COMET: A Neural Framework for MT Evaluation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2685–2702, Online, nov 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.213. URL https://aclanthology.org/2020.emnlp-main.213.

Santiago Rodríguez and Jesús Carretero. A Formal Approach to Spanish Grammar: the COES Tools. *Procesamiento del Lenguaje Natural*, 19, 1996.

Lane Schwartz. The History and Promise of Machine Translation. *Innovation and expansion in translation process research*, page 161, 2018.

Thibault Sellam, Dipanjan Das, and Ankur Parikh. BLEURT: Learning Robust Metrics for Text Generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892, Online, jul 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.704. URL https://aclanthology.org/2020.acl-main.704.

Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural Machine Translation of Rare Words with Subword Units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany, aug 2016. Association for Computational Linguistics. doi: 10.18653/v1/P16-1162. URL https://aclanthology.org/P16-1162.

L. Shen, C. Bedetti, and D. Lesieur. Lexical Richness. https://github.com/LSYS/LexicalRichness, 2021.

Raivis Skadiņš, Jørg Tiedemann, Roberts Rozis, and Daiga Deksne. Billions of parallel words for free: Building and using the eu bookshop corpus. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 1850–1855, 2014.

Vicki Spandel. *Creating writers: Through 6-trait Writing Assessment and Instruction*. Allyn & Bacon, 2005.

Jørg Tiedemann and Lars Nygaard. The OPUS Corpus-Parallel and Free: http://logos. uio. no/opus. In *LREC*. Citeseer, 2004.

Jörg Tiedemann. Parallel Data, Tools and Interfaces in OPUS. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Mehmet Ugur Dogan, Bente Maegaard, Joseph Mariani, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey, may 2012. European Language Resources Association (ELRA). ISBN 978-2-9517408-7-7.

Marco Turchi, Tijl De Bie, and Nello Cristianini. Learning Performance of a Machine Translation System: a Statistical and Computational Analysis. In *Proceedings of the Third Workshop on Statistical Machine Translation*, pages 35–43, Columbus, Ohio, jun 2008. Association for Computational Linguistics. URL https://aclanthology.org/W08-0305.

Guido Van Rossum and Fred L. Drake. *Python 3 Reference Manual*. CreateSpace, Scotts Valley, CA, 2009. ISBN 1441412697.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is All you Need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Richard Y Wang, Veda C Storey, and Christopher P Firth. A framework for analysis of data quality research. *IEEE transactions on knowledge and data engineering*, 7(4): 623–640, 1995.

Wei Wang, Taro Watanabe, Macduff Hughes, Tetsuji Nakagawa, and Ciprian Chelba. Denoising Neural Machine Translation Training with Trusted Data and Online Data Selection, 2018.

Krzysztof Wołk and Krzysztof Marasek. Building Subject-aligned Comparable Corpora and Mining it for Truly Parallel Sentence Pairs. *Procedia Technology*, 18:126–132, 2014. ISSN 2212-0173. doi: https://doi.org/10.1016/j.protcy.2014.11.024. URL `https://www.sciencedirect.com/science/article/pii/S2212017314005453`. International workshop on Innovations in Information and Communication Science and Technology, IICST 2014, 3-5 September 2014, Warsaw, Poland.