# Fine-tuning a multilingual machine translation model for medical diagnostics technical content

## Final project for CAS NLP 2023-2024

*Chiara Baffelli*
*chiara.baffelli@gmail.com*

### Abstract

*This report presents the findings of a project focused on fine-tuning a transformer-based multilingual translation model for medical diagnostics technical content. The project involved data collection, preprocessing, model selection, and the fine-tuning process. 3 transformer-based models, T5, FlanT5, and Helsinki-NLP/opus-mt-en-roa, were evaluated and compared based on their performance in translating English into 5 languages: French, Italian, Spanish, Portuguese, Romanian. The results showed that the Helsinki-NLP/opus-mt-en-roa model performed the best, achieving an impressive BLEU score of 68 after fine-tuning. The evaluation and benchmarking of the fine-tuned model demonstrated improvements in translation quality compared to the baseline model.*

## Table of contents

# 1. Introduction

Accurate and efficient translation of medical diagnostics technical content across multiple languages is crucial. Machine translation models have shown great potential in meeting this demand in an efficient way. In this project, I explore the process of fine-tuning a multilingual machine translation model tailored for medical diagnostics technical content. By fine-tuning existing language models, I hope to leverage their capabilities and refine their performance for specific languages and domains.
In this report, I present the methodology, description of the dataset used in this project, model selection and fine-tuning process, evaluation and benchmarking as well as the final result, a discussion of limitations, possible improvements, next steps, and conclusion.

## 1.1. Background and motivation

The motivation for this project stems mainly from my interest in the field of translation, and my own experience of working with machine translation models at my current place of employment, Roche Diagnostics. Currently, the translation setup at Roche Diagnostics relies on individual machine translation models for each language. Documents are translated into up to 40 languages, meaning that 40 different models must be managed and updated when needed[1]. Having a model that can efficiently translate in several languages could simplify the current setup.

## 1.2. Project objectives and scope

The main objective of this project is to fine-tune a transformer-based multilingual translation model for the following 5 language pairs: English - French, English - Italian, English - Spanish, English - Portuguese, and English - Romanian.

The language pairs have been chosen based on the following criteria:

- English is always the source language in which technical documentation is written at Roche Diagnostics;

[1] As a reference, currently training 1 model for 1 language pair takes around 12 hours. Only 2 models can be trained concurrently.

- The 5 target languages are the main languages in which most technical documentation is translated into at Roche;
- The 5 language models all belong to the group of Romance languages, and they all share similar linguistic structures, vocabulary, and grammar rules.

Multilingual models are "presumed to be helpful because they leverage syntactic or semantic similarities between languages" (Ting-Rui Chiang, Yi-Pei Chen, Yi-Ting Yeh, Graham Neubig, 2022), therefore choosing languages in the same family can be seen as an advantage. The resulting model is a so-called *one-to-many* model, meaning a model that can translate from one source language (English) into various target languages. By focusing on fine-tuning instead of training from scratch, I aim to leverage the existing capabilities of available models and refine their performance for specific languages and domains. Fine-tuning all languages together can be computationally efficient and requires fewer resources compared to fine-tuning separate models.

## 2. Methodology

Figure 1 represents all the steps that have been completed as part of this project. First of all, I collected and prepared the data used in this project for further analysis and processing. This step is described in detail in Chapter 3, together with an explorative data analysis. Then, I explored different available models to find the best suitable for this task, and experimented with different hyperparameters and fine-tuning setups, as described in Chapter 4.
I then evaluated the resulting fine-tuned model against the baseline model and against the existing machine translation models used by Roche Diagnostics (Chapter 5). Finally, I created a frontend through which the model can be deployed and used.

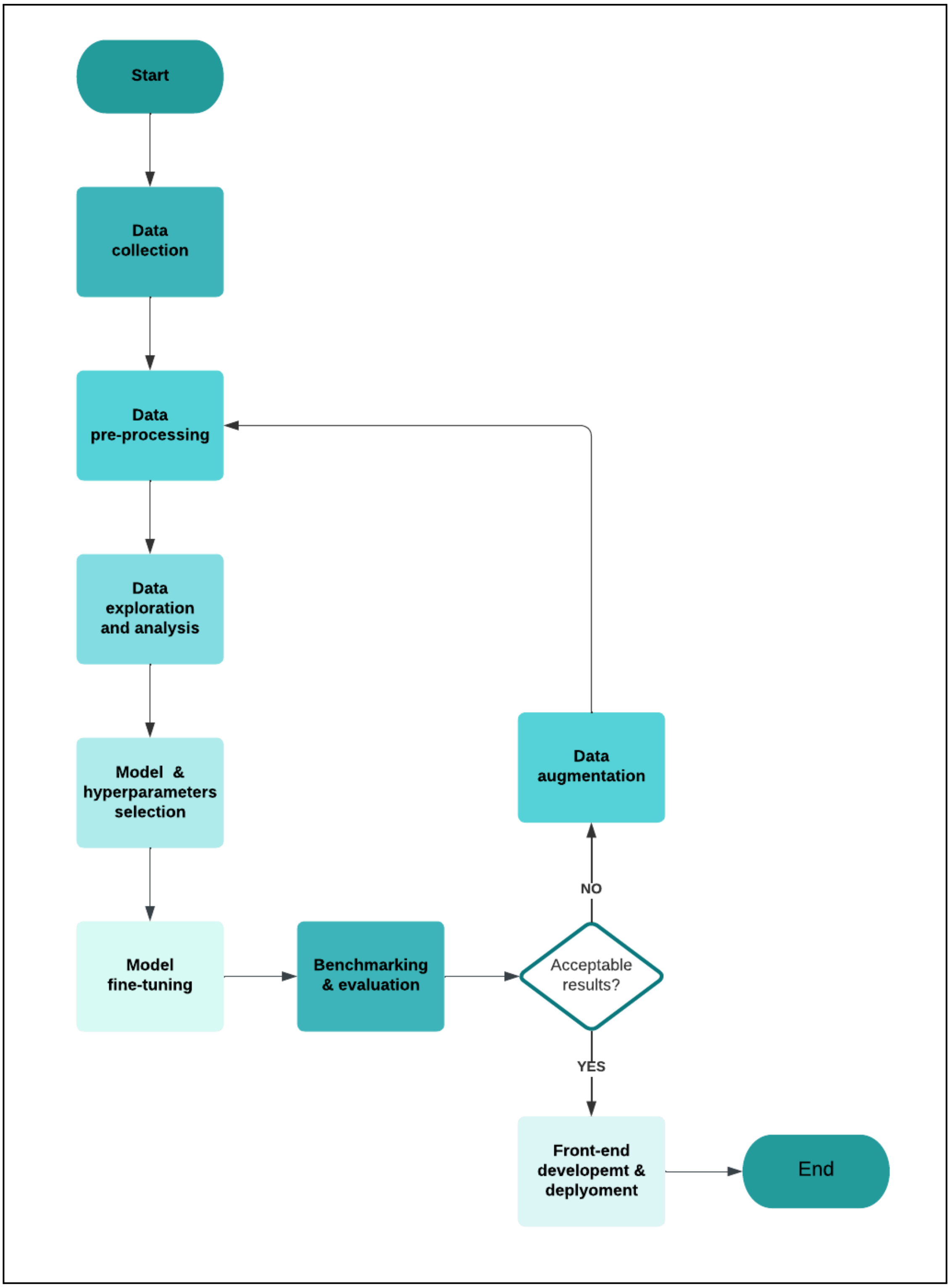

Start
Data collection
Data pre-processing
Data exploration and analysis
Model & hyperparameters selection
Model fine-tuning
Benchmarking & evaluation
Acceptable results?
NO
Data augmentation
YES
Front-end developemt & deplyoment
End

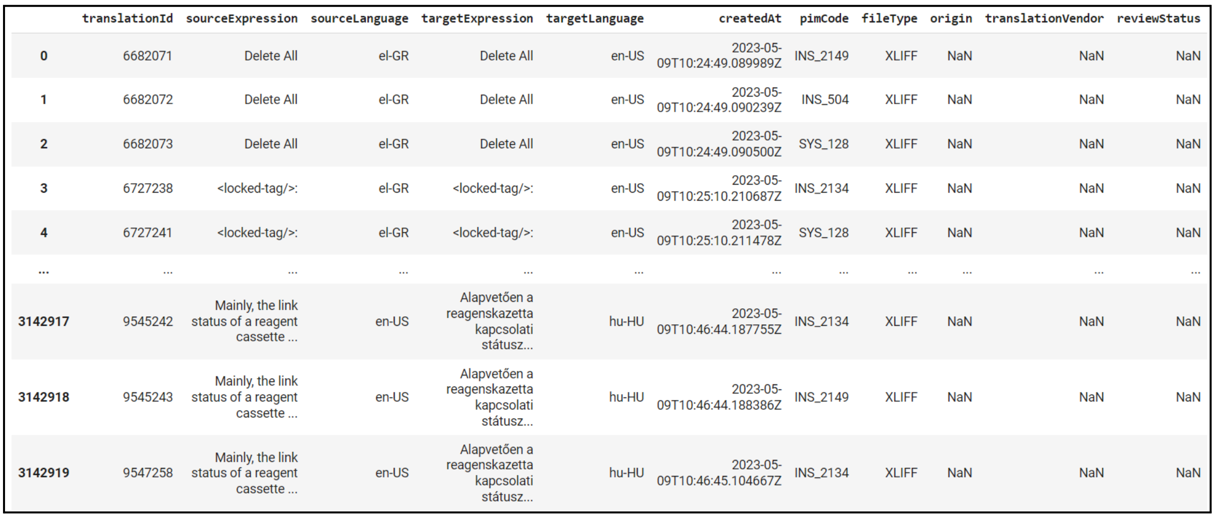
Figure 1 - Flowchart of methodology used in this project

# 3. Dataset description & analysis

In this section, I provide a brief description of the dataset used for the fine-tuning and outline the preprocessing steps that were applied to the dataset. Finally, I perform a data exploration and illustrate it via different figures.

## 3.1. Dataset description

The dataset used in this project belongs to Roche Diagnostics[2] and is a parallel dataset containing translation from English into 40 languages. The dataset has been collected from 2010 through 2024 and is used to pre-translate technical documentation produced at Roche Diagnostics.
The original dataset is in JSON format and has a size of 18 GB. Figure 2 shows a subset of the original dataset converted in a dataframe.



| | translationId | sourceExpression | sourceLanguage | targetExpression | targetLanguage | createdAt | pimCode | fileType | origin | translationVendor | reviewStatus |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 6682071 | Delete All | el-GR | Delete All | en-US | 2023-05-09T10:24:49.089989Z | INS_2149 | XLIFF | NaN | NaN | NaN |
| 1 | 6682072 | Delete All | el-GR | Delete All | en-US | 2023-05-09T10:24:49.090239Z | INS_504 | XLIFF | NaN | NaN | NaN |
| 2 | 6682073 | Delete All | el-GR | Delete All | en-US | 2023-05-09T10:24:49.090500Z | SYS_128 | XLIFF | NaN | NaN | NaN |
| 3 | 6727238 | <locked-tag/>: | el-GR | <locked-tag/>: | en-US | 2023-05-09T10:25:10.210687Z | INS_2134 | XLIFF | NaN | NaN | NaN |
| 4 | 6727241 | <locked-tag/>: | el-GR | <locked-tag/>: | en-US | 2023-05-09T10:25:10.211478Z | SYS_128 | XLIFF | NaN | NaN | NaN |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 3142917 | 9545242 | Mainly, the link status of a reagent cassette ... | en-US | Alapvetően a reagenskazetta kapcsolati státusz... | hu-HU | 2023-05-09T10:46:44.187755Z | INS_2134 | XLIFF | NaN | NaN | NaN |
| 3142918 | 9545243 | Mainly, the link status of a reagent cassette ... | en-US | Alapvetően a reagenskazetta kapcsolati státusz... | hu-HU | 2023-05-09T10:46:44.188386Z | INS_2149 | XLIFF | NaN | NaN | NaN |
| 3142919 | 9547258 | Mainly, the link status of a reagent cassette ... | en-US | Alapvetően a reagenskazetta kapcsolati státusz... | hu-HU | 2023-05-09T10:46:45.104667Z | INS_2134 | XLIFF | NaN | NaN | NaN |

Figure 2 - The original dataset represented as a dataframe

## 3.2. Dataset preprocessing

To bring the original dataset in a format useful for fine-tuning, the following steps have been taken:

- Convert the original JSON dataset into a dataframe;
- Drop columns that are not needed;
- Harmonize the language codes to a standard version;
- Replace special characters with the correct encoding;

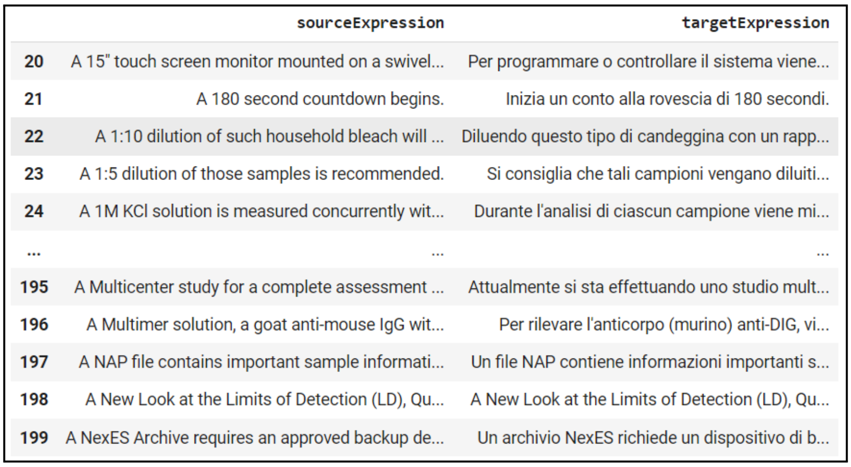
[2] As the data belongs to Roche Diagnostics and contains sensible data, the full dataset cannot be shared outside of Roche Diagnostics.

- Remove markup tags;
- Strip unneeded spaces;
- Remove duplicate sentence pairs;
- Drop empty rows;
- Remove sentences that consist of only numbers or special characters;
- Remove sentences that consist of more than 50% non-alphanumeric characters;
- Remove sentences that are longer than 80 tokens and shorter than 5 tokens. Setting a maximum sentence length helps control the resources required for training. Longer sentences can be more complex and harder for the model to learn accurately. At Roche Diagnostics, we follow a minimalistic writing approach, so longer sentences are infrequent.

After these preprocessing steps have been applied, a subdataset has been created for each language pair. The rest of the report focuses on the data used in the project, namely the subdatasets for Italian, French, Romanian, Spanish, and Portuguese.

### 3.3. Data exploration and analysis

The application of the preprocessing steps described above resulted in subdatasets, 1 per language pair. Figure 3 shows what the dataset for the English - Italian pair looks like. All other datasets have the same structure. The column 'sourceExpression' contains a sentence in English, and the column 'targetExpression' contains the translated sentence in the target language.



| | sourceExpression | targetExpression |
|---|---|---|
| 20 | A 15" touch screen monitor mounted on a swivel... | Per programmare o controllare il sistema viene... |
| 21 | A 180 second countdown begins. | Inizia un conto alla rovescia di 180 secondi. |
| 22 | A 1:10 dilution of such household bleach will ... | Diluendo questo tipo di candeggina con un rapp... |
| 23 | A 1:5 dilution of those samples is recommended. | Si consiglia che tali campioni vengano diluiti... |
| 24 | A 1M KCl solution is measured concurrently wit... | Durante l'analisi di ciascun campione viene mi... |
| ... | ... | ... |
| 195 | A Multicenter study for a complete assessment ... | Attualmente si sta effettuando uno studio mult... |
| 196 | A Multimer solution, a goat anti-mouse IgG wit... | Per rilevare l'anticorpo (murino) anti-DIG, vi... |
| 197 | A NAP file contains important sample informati... | Un file NAP contiene informazioni importanti s... |
| 198 | A New Look at the Limits of Detection (LD), Qu... | A New Look at the Limits of Detection (LD), Qu... |
| 199 | A NexES Archive requires an approved backup de... | Un archivio NexES richiede un dispositivo di b... |

Figure 3 - Example of the Italian dataset

Figure 4 shows the size of each language-specific dataset in sentence pairs. Not all language pairs have the same amount of sentence pairs, with Romanian being the language with the least amount of data.

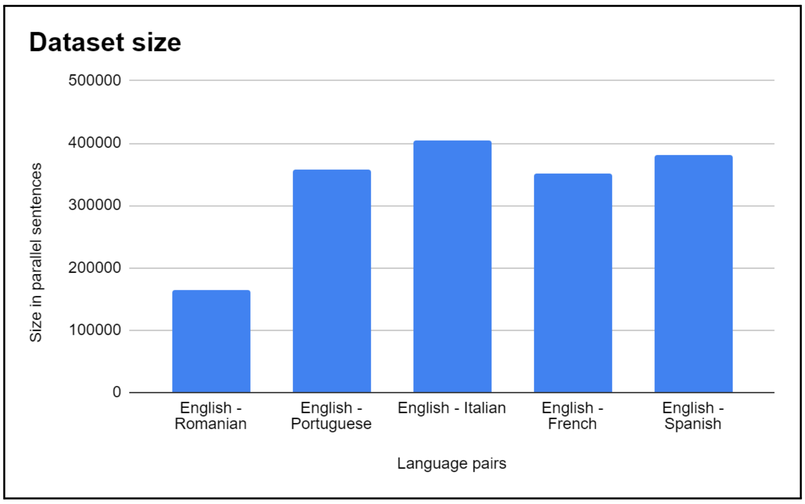


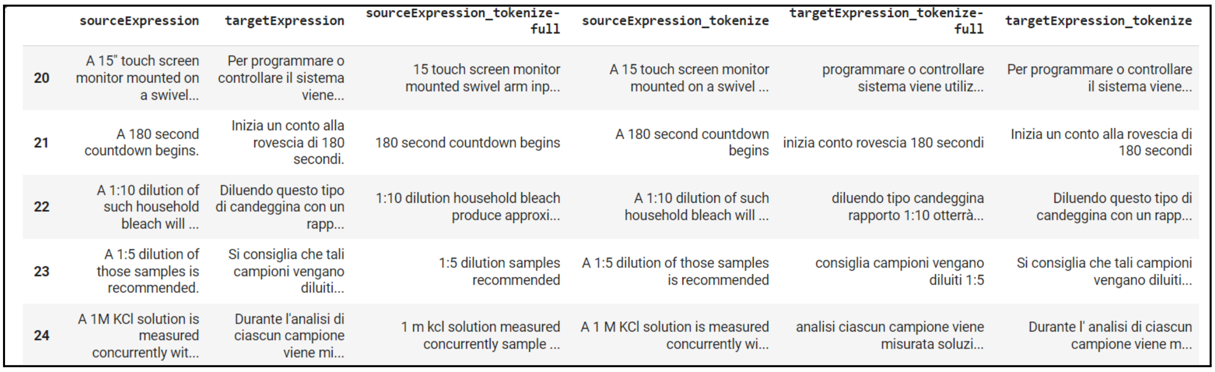
Figure 4 - Dataset size in parallel sentences

In preparation for fine-tuning and to explore the impact different types of data preprocessing have on the final model, I applied an additional preprocessing step to the 5 subdatasets: tokenization.

For the tokenization step, I decided to use the spaCy python library ([*spaCy · Industrial-strength Natural Language Processing in Python*]), as it offers support for all the needed languages. During the tokenization step, I applied 2 variations of tokenization on my data:

- What I call a 'full' tokenization, in which the sentences are tokenized, stopwords and punctuation are removed, and tokens are converted into lower case;
- What I call a 'light' tokenization, in which sentences are tokenized and punctuation is removed.

The tokenization step resulted in a dataframe with additional columns to contain the 'light' and 'full' tokenized sentences, as illustrated in Figure 5.



| | sourceExpression | targetExpression | sourceExpression_tokenize-full | sourceExpression_tokenize | targetExpression_tokenize-full | targetExpression_tokenize |
|---|---|---|---|---|---|---|
| 20 | A 15" touch screen monitor mounted on a swivel... | Per programmare o controllare il sistema viene... | 15 touch screen monitor mounted swivel arm inp... | A 15 touch screen monitor mounted on a swivel ... | programmare o controllare sistema viene utiliz... | Per programmare o controllare il sistema viene... |
| 21 | A 180 second countdown begins. | Inizia un conto alla rovescia di 180 secondi. | 180 second countdown begins | A 180 second countdown begins | inizia conto rovescia 180 secondi | Inizia un conto alla rovescia di 180 secondi |
| 22 | A 1:10 dilution of such household bleach will ... | Diluendo questo tipo di candeggina con un rapp... | 1:10 dilution household bleach produce approxi... | A 1:10 dilution of such household bleach will ... | diluendo tipo candeggina rapporto 1:10 otterrà... | Diluendo questo tipo di candeggina con un rapp... |
| 23 | A 1:5 dilution of those samples is recommended. | Si consiglia che tali campioni vengano diluiti... | 1:5 dilution samples recommended | A 1:5 dilution of those samples is recommended | consiglia campioni vengano diluiti 1:5 | Si consiglia che tali campioni vengano diluiti... |
| 24 | A 1M KCl solution is measured concurrently wit... | Durante l'analisi di ciascun campione viene mi... | 1 m kcl solution measured concurrently sample ... | A 1 M KCl solution is measured concurrently wi... | analisi ciascun campione viene misurata soluzi... | Durante l' analisi di ciascun campione viene m... |

Figure 5 - Italian dataset after the tokenization step

Figure 6 shows the number of tokens and unique tokens in source and target language per each subdataset. The amount of tokens is higher in the target language of each dataset. Romance languages often have more inflections, conjugations, and grammatical complexities compared to English and are therefore usually longer than English.

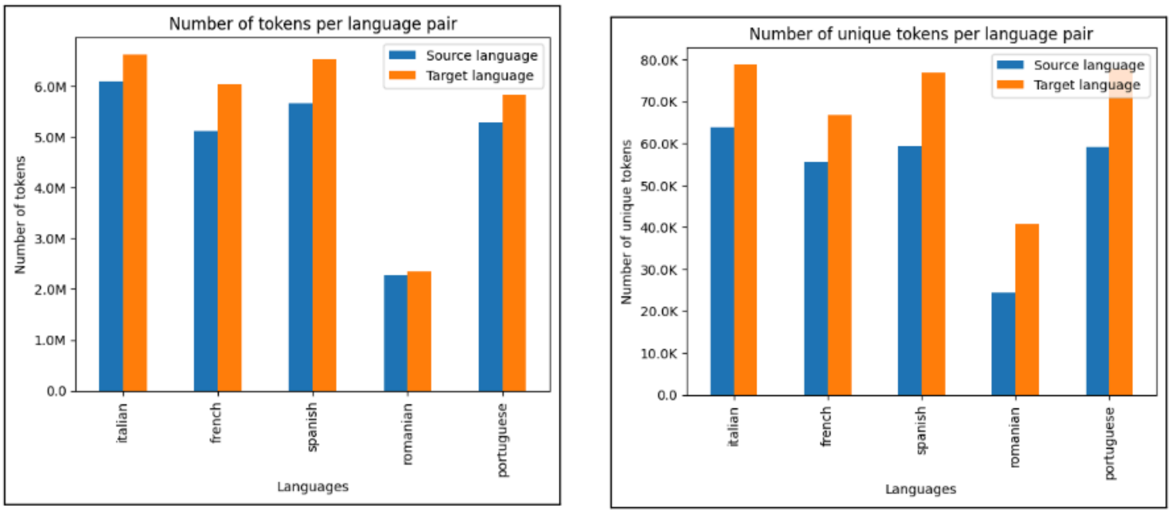


Figure 6 - Token count for each dataset

In Figure 7 below, the distribution of sentence length between source and target can be seen. Most of the sentences have a length of 20 tokens or less - as mentioned above, this is in line with the minimalistic writing approach, in which shorter sentences are preferred.

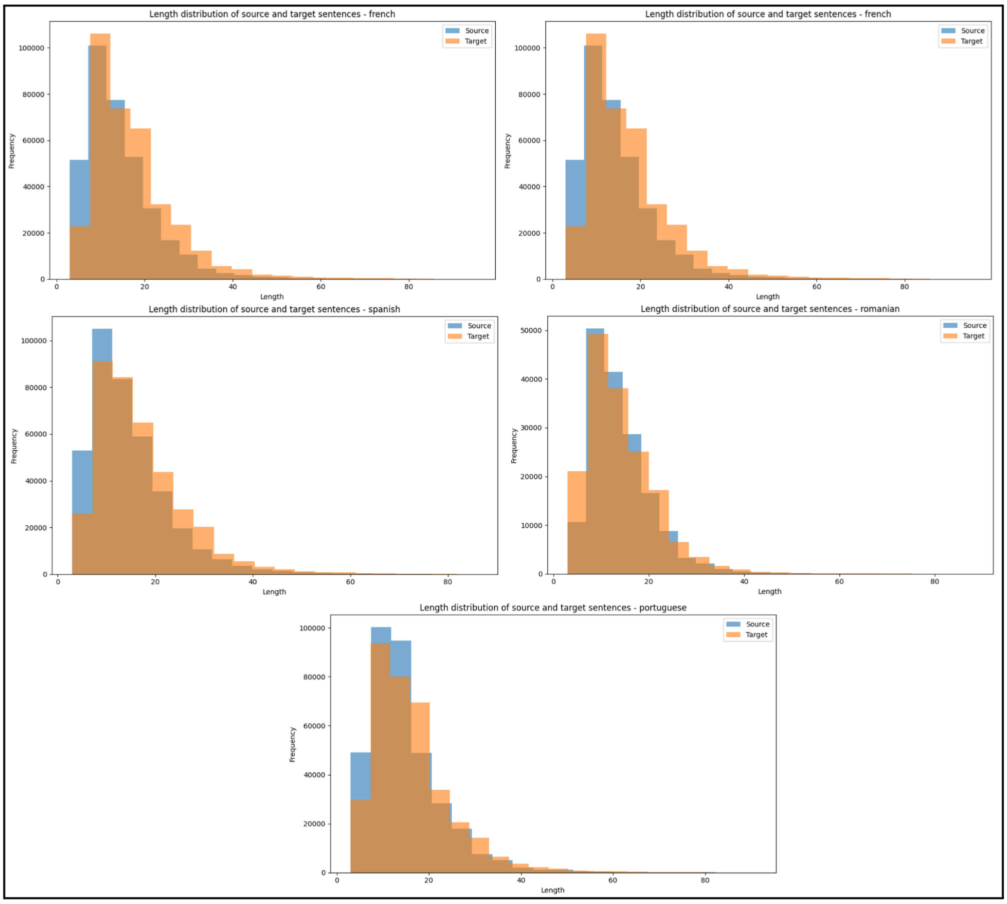


Figure 7 - Sentence length distribution for each dataset

# 4. Model selection & fine-tuning process

Once the data is preprocessed and ready, the next steps are the selection of the models and the fine-tuning process and experiments.

## 4.1. Model selection

Given the nature of the task, I decided to use transformer-based models due to their ability to capture long-range dependencies, handle sequential data effectively, and leverage the attention mechanism. The attention mechanism plays a vital role in machine translation by allowing the model to focus on relevant parts of the source sentence while generating the target sentence. It also enables the model to align words in the source and target languages, facilitating accurate translation.

Based on the criteria highlighted above, I have selected 3 transformer-bases models (see Table 1) on which to focus during this project. I aim to test different models and pick the best performing model at the end.

| Model name | Task | Languages |
|---|---|---|
| T5[3] (*T5*) | Multi-task, include translation | English, French, Romanian, German |
| FlanT5 (*FLAN-T5*) | Multi-task, include translation | Several languages including the 6 languages needed for this project |
| Helsinki-NLP/opus-mt-en-roa (*Helsinki-NLP/opus-mt-en-roa · Hugging Face*) | Translation | English and all Romance languages |

Table 1 - Models used in this project

### 4.2. Fine-tuning process[4]

Figure 8 represents the process I followed during the fine-tuning process. I will not describe every step in detail in this report but only focus on the most important ones: data preparation with the prefix (Section 4.2.1.), evaluation metrics (Section 4.2.2.), model setup (Section 4.2.3.), data variations and augmentation (Section 4.2.4.).
As I experimented with different setups (model setup, additional data, different hyperparameters), this is also represented in Figure 8. However, it does not represent the full combination of experiments.

[3] Although the T5 model does not include all the needed languages, I wanted to experiment with including a model for which some of the languages are still unknown.

[4] The basis of the process and part of the code is based on the 'Translation' task of HuggingFace (*Translation task*), which I then adapted further for my needs.

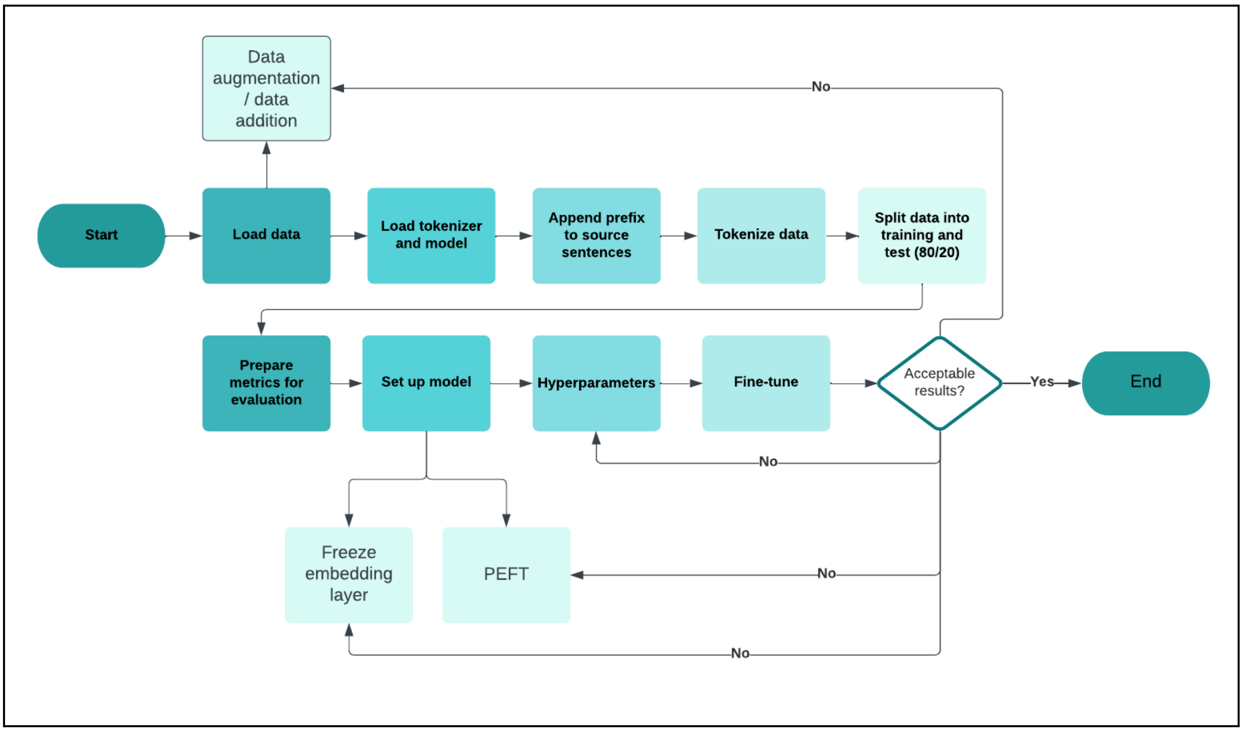


Figure 8 - Flowchart of the fine-tuning process I followed

### 4.2.1. Prefix

The models I selected for this project all use a prefix:

- The T5 and FlanT5 group of models are pre-trained for various tasks and are not specific for translation: the prefix is used to specify the task ('translate') as well as the language to translate into.
- For the OPUS model, the prefix is used to indicate into which language the translation shall be performed.

Therefore, the model-specific prefix must be added before the source sentence before fine-tuning. Figure 9 shows the prefixes used for T5/FlanT5 and for the OPUS model.

```python
#Creates a mapping for the prefixes
prefix_mapping_T5 = {
    'Italian' : 'translate English to Italian: ',
    'French' : 'translate English to French: ',
    'Spanish' : 'translate English to Spanish: ',
    'Romanian' : 'translate English to Romanian: ',
    'Portuguese' : 'translate English to Portuguese: '
}

prefix_mapping_OPUS =  {
    'Italian' : '>>ita<< ',
    'French' : '>>fra<< ',
    'Spanish' : '>>spa<< ',
    'Romanian' : '>>ron<< ',
    'Portuguese' : '>>por<< '
}
```

Figure 9 - Example of the prefixes used for T5 models and for the OPUS model

### 4.2.2. Evaluation metric[5]

During the fine-tuning process, I used the BLEU metric to assess the performance and effectiveness of the model. The BLEU (Bilingual Evaluation Understudy) score compares the n-gram precision of the generated translations against one or more reference translations. It is known for its robustness and correlation with human judgments and is easy to compute. In the rest of this section, when results are presented, the best accuracy is always expressed as the BLEU score.

### 4.2.3. Model setup

During the fine-tuning process, I also experimented with a few different things:

- Hyperparameters
- Freezing the embedding layer;
- Using a PEFT (Parameter-Efficient Fine-Tuning) configuration;

#### 4.2.3.1. Hyperparameters

For the model fine-tuning, I experimented with several hyperparameters, as shown in Table 2.

[5] Additional metrics have been used during the benchmarking and evaluation, see Chapter 5.

| Hyperparameter | Range | Optimal |
|---|---|---|
| Weight decay | 0.0 to 0.001 | 0.01 |
| Batch size | 8, 16, 32, 64 | 32 |
| Learning rate | 1E-02 - 2E-05 | 1E-03 |
| Epochs | 1 - 10 | 4 |

Table 2 - Hyperparameters

Table 2 also shows the hyperparameters I settled on after the various experiments and that performed the best for all 3 models.

The hyperparameter that had the biggest impact overall on the performance was the learning rate (see Figure 10).

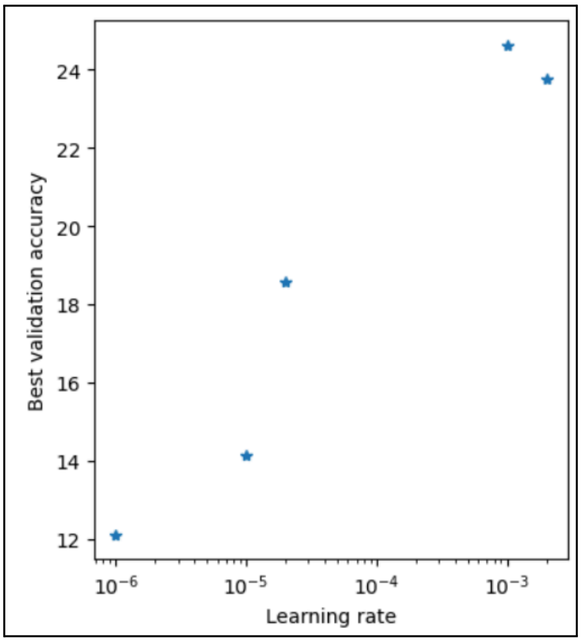


Figure 10 - Effect of different learning rates on accuracy for the T5 model

The number of epochs (4) is relatively low, but my experiments showed that, after 4 epochs, the improvement in the model was minimal and negligible (see, for example, Figure 11), while the computing time increased with each epoch. Therefore 4 epochs was found to be a good parameter to reach a satisfactory result.

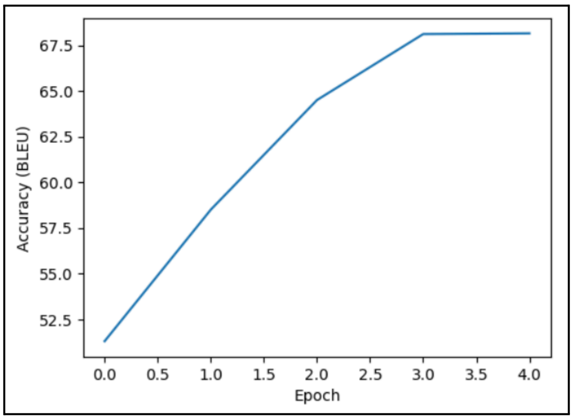


Figure 11 - Correlation between accuracy and epochs for the Helsinki-NLP/opus-mt-en-roa model

#### 4.2.3.2. Embedding layer freezing

Freezing or making certain parameters static, like the shared embedding parameters, can be useful when fine-tuning a model, as shown by Jaejun Lee, Raphael Tang, and Jimmy Lin in their paper 'What Would Elsa Do? Freezing Layers During Transformer Fine-Tuning' (Jaejun Lee, Raphael Tang, and Jimmy Lin, 2019).

Freezing the sharing embeddings means that the gradients for these embeddings are not computed or updated during the fine-tuning process. This ensures consistency and preserves pre-trained representations. In addition, embedding parameters are often large in size, especially when dealing with large vocabularies. By freezing these parameters, we can save memory and reduce the computational burden during training or fine-tuning.
Overall, freezing shared embedding parameters is a technique that can improve model performance, efficiency, and stability in certain scenarios, therefore I decided to experiment with it in my project.

However the results (Table 3), show that freezing the embedding layer is beneficial only for the T5 model, where the BLEU score shows a minimal improvement of 0.04 points. For the other models, freezing the embedding layer did not have a positive impact on the final accuracy; the only beneficial impact was on fine-tuning time, which was reduced by a few minutes only.

| | FlanT5 (base) | T5 (base) | Helsinki-NLP/opus-mt-en-roa |
|---|---|---|---|
| With embedding freezing | 9.85 | 24.50 | 67.45 |
| Without embedding freezing | 9.85 | 24.46 | 68.11 |

Table 3 - Result of fine-tuning with and without embedding layer freezing

#### 4.2.3.3. PEFT

Parameter-Efficient Fine-Tuning (PEFT) is a powerful technique that allows for the efficient adaptation of large pretrained models to various downstream applications without the need to fine-tune all of the model's parameters. Developed by Hugging Face (*PEFT*), PEFT addresses the computational and resource challenges associated with traditional fine-tuning approaches.

During fine-tuning, updating all parameters can be computationally expensive. PEFT selectively fine-tunes important parameters, reducing computational burden while balancing model adaptability. PEFT goes beyond embedding freezing, updating a subset of parameters relevant to the task, while freezing the rest.

Similarly to embedding freezing, PEFT can also help improve model performance, efficiency, and stability. The results of my experiments with the PEFT configuration are shown below in Table 4. The use of PEFT did not show an improvement or benefit for any of the models used in this project in terms of BLEU score.

| | FlanT5 (base) | T5 (base) | Helsinki-NLP/opus-mt-en-roa |
|---|---|---|---|
| With PEFT | 9.85 | 16.91 | 55.10 |
| Without PEFT | 9.85 | 24.46 | 68.11 |

Table 4 - Result of fine-tuning with and without PEFT

### 4.2.4. Data variation and augmentation

For most of the experiments, I used the data as-is, without tokenization, stopwords removal, or lowercasing (as shown in Figure 3). I however also wanted to experiment with the tokenized data (‘full’ and ‘light’ shown previously in Figure 5), and experiment with data augmentation.

Table 5 shows the result of fine-tuning the 3 models using the ‘full’, ‘light’, and not tokenized data. Interestingly, the 3 models are all affected in different ways. The OPUS model and the FlanT5 model fare the best when the data is left as-is while we see an improvement of more than 3 points in the BLEU score when using the ‘full’ tokenized data with the T5 model.

|  | FlanT5 (base) | T5 (base) | Helsinki-NLP/opus-mt-en-roa |
|---|---|---|---|
| ‘Full’ tokenized data | 7.09 | 27.61 | 0.62 |
| ‘Light’ tokenized data | 8.17 | 25.86 | 64.98 |
| Baseline | 9.85 | 24.46 | 68.11 |

Table 5 - Result of fine-tuning with different types of tokenization applied to the data

I also experimented with 2 data augmentation techniques (Li, Hou and Che, 2021): random swapping and synonym replacement.

- In random swapping, I randomly choose 2 words in the sentences and swap their positions; with random swapping, the hope is to improve the model's ability to generalize and handle variations in word order.
- For the synonym replacement, I randomly choose 2 words in the source sentences and replace them with a synonym using NLTK’s Wordnet (*NLTK :: Sample usage for wordnet*). I specifically applied this technique to the source sentences only because I want the resulting model to use Roche Diagnostics specific terminology when translating.

Table 6 shows the result of the experiments. The 2 data augmentation techniques do not show any improvement on the BLEU score for any of the models; the random swapping seems to do the most damage on the BLEU score.

| | FlanT5 (base) | T5 (base) | Helsinki-NLP/opus-mt-en-roa |
|---|---|---|---|
| Swap | 5.57 | 18.05 | 62.40 |
| Synonyms | 6.43 | 19.62 | 67.27 |
| Baseline | 9.85 | 24.46 | 68.11 |

Table 6 - Result of fine-tuning with data augmentation techniques

# 5. Results & evaluation

In this section I will briefly present the final result of my experiments, and the result of the evaluation I carried out on the final fine-tuned model.

## 5.1. Result & discussion

In section 4.2. Fine-tuning process I presented the process I used to fine-tune the model and the different experiments I carried out to find the best performing model.

| | Best accuracy (BLEU) |
|---|---|
| FlanT5 (base) | 9.85 |
| T5 (base) | 24.46 |
| Helsinki-NLP/opus-mt-en-roa | 68.11 |

Table 7 - Final results of fine-tuning

All the experiments and the final results in Table 7 clearly show something: the model performing the best for this project and specific dataset is the Helsinki-NLP/opus-mt-en-roa model. This is not entirely surprising, as the model is meant for the same exact task I used in this project (machine translation from English into Romance languages).

The T5 model is performing less well, with a BLEU score of only 24. Surprisingly the T5 model was still able to learn to translate in additional languages it was not pre-trained on, even when using a dataset of limited size. The results with the FlanT5 model are more surprising: FlanT5 is supposed to be an improvement over T5 and to perform better when fine-tuned on a single task compared to T5 (*The Flan Collection: Advancing open source methods for instruction tuning*); this is not the case for this project. Possibly, the dataset is too small. Further experiments with bigger datasets could be considered.

The final setup to fine-tune the Helsinki-NLP/opus-mt-en-roa was the following:

- Data used as-is (no tokenization);
- No embedding layer freezing;
- No PEFT;
- Hyperparameters as shown in Table 2.

The training loss and validation loss of the best-performing model (Figure 12) show that the model is not overfitting or underfitting, rather the training loss and validation loss both decrease and stabilize at a specific point, indicating a good fit.

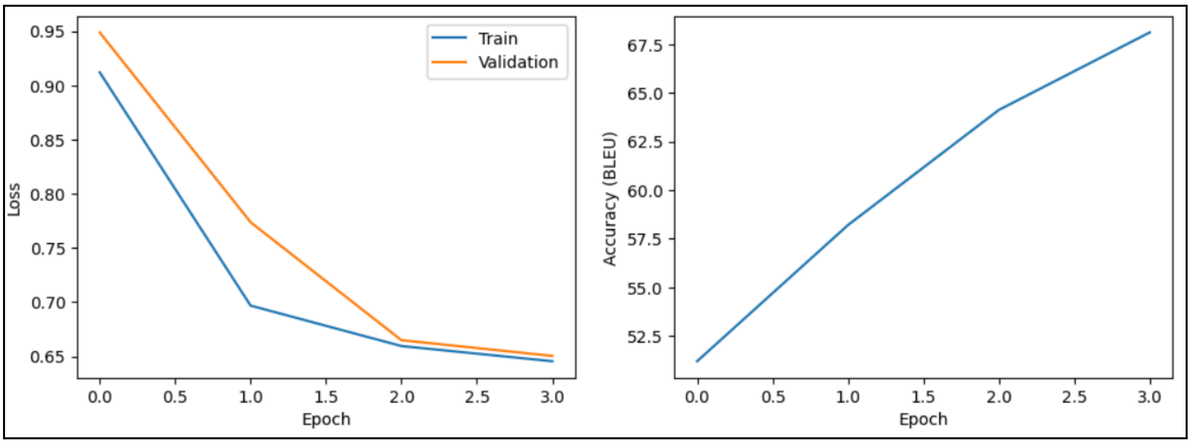


Figure 12 - Training and validation loss of best performing fine-tuned model

The final fine-tuned model has 73 '678' 336 million parameters and took around 12 hours to fine-tune[6].

## 5.2. Evaluation & benchmarking

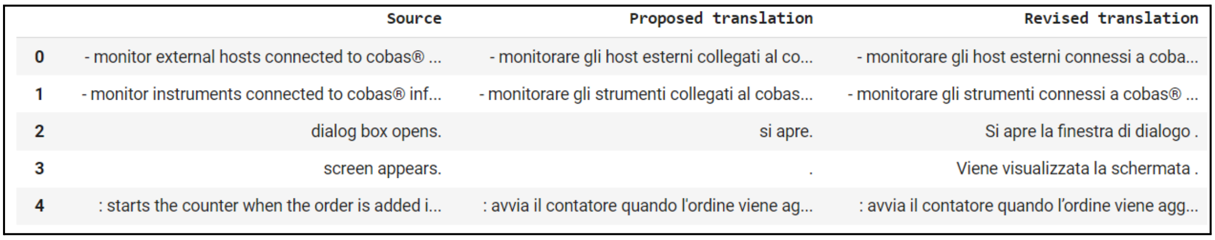
[6] It takes around the same time to train 1 model for 1 language pair for the Roche machine translation system.

For the evaluation and benchmarking, I have decided to compare my fine-tuned model against the existing machine translation models used by Roche Diagnostics for French, Spanish, Italian, Portuguese, and Romanian, as well as to compare it to the baseline Helsinki-NLP/opus-mt-en-roa model.
For the metrics, I have a few additional metrics in addition to BLEU (see Table 8). The evaluation of machine translation presents many challenges, especially because translation can be subjective, i.e. a sentence can be translated in multiple different ways while still keeping the same meaning. Automatic metrics are premised on the machine translation output having an exact match with a reference text, and minor differences can have an impact on the final score. For this reason I decided to employ multiple metrics during the evaluation to have a comprehensive assessment.

| BLEU | ROUGE | METEOR | BERT |
|---|---|---|---|
| BLEU (Bilingual Evaluation Understudy) score compares the n-gram precision of the generated translations against one or more reference translations. It is known for its robustness and correlation with human judgments | ROUGE1 measures the overlap of unigrams, ROUGE2 examines the overlap of bigrams, and ROUGE-L captures the longest common subsequence between the generated and reference translations. These metrics provide a comprehensive view of the model's ability to generate accurate and fluent translations. | METEOR (Metric for Evaluation of Translation with Explicit ORdering) measures the quality of translation outputs by considering several factors such as unigram precision, recall, and alignment error rate. METEOR incorporates linguistic and syntactic information, making it a comprehensive metric to assess the fluency and accuracy of translations. | BERT Score is designed to evaluate the quality of machine translation outputs by leveraging the contextualized embeddings provided by BERT. It measures the similarity between the generated translations and the reference translations by considering the contextual information, capturing the semantic similarity and capturing nuances that other metrics might miss. |

Table 8 - Metrics used during evaluation / benchmarking

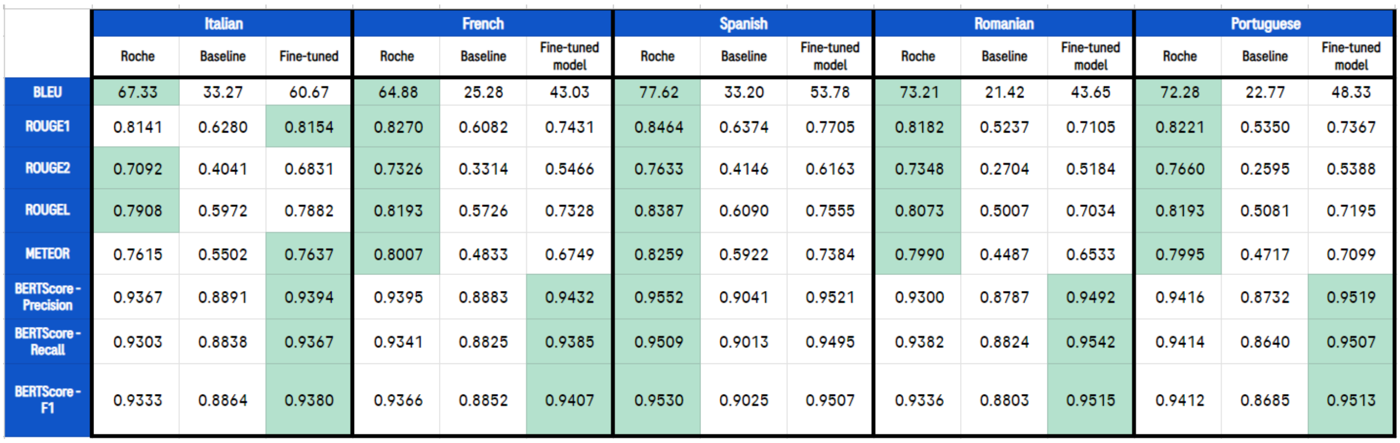
In order to keep the evaluation fair, I have selected a dataset of sentences that are unknown both to the fine-tuned model and to the trained models belonging to Roche Diagnostics. The dataset consists of a set of 150 English sentences translated in all 5 languages, and it includes the raw output obtained by the Roche Diagnostics machine translation system (column ‘Proposed translation’ in the Figure 13 below), as well as the translation after post-editing and review by 2 professional translators. This is used as a reference translation during the evaluation.



| | Source | Proposed translation | Revised translation |
|---|---|---|---|
| **0** | - monitor external hosts connected to cobas® ... | - monitorare gli host esterni collegati al co... | - monitorare gli host esterni connessi a coba... |
| **1** | - monitor instruments connected to cobas® inf... | - monitorare gli strumenti collegati al cobas... | - monitorare gli strumenti connessi a cobas® ... |
| **2** | dialog box opens. | si apre. | Si apre la finestra di dialogo . |
| **3** | screen appears. | . | Viene visualizzata la schermata . |
| **4** | : starts the counter when the order is added i... | : avvia il contatore quando l'ordine viene ag... | : avvia il contatore quando l'ordine viene agg... |

Figure 13 - Data used during the evaluation

Table 9 shows the result of the benchmark and evaluation. In general, the Roche model outperforms the fine-tuned model and the baseline model in most of the metrics. However, the fine-tuned model shows great improvements over the baseline for all languages, with Italian achieving an impressive BLEU score of 60, followed by Spanish with 53. Interestingly, the fine-tuned model consistently shows higher scores in terms of BERTScore. A higher BERTScore could suggest that the fine-tuned model is better at capturing the contextual meaning and similarity between sentences compared to the other models.



| | Italian | | | French | | | Spanish | | | Romanian | | | Portuguese | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Roche | Baseline | Fine-tuned | Roche | Baseline | Fine-tuned model | Roche | Baseline | Fine-tuned model | Roche | Baseline | Fine-tuned model | Roche | Baseline | Fine-tuned model |
| **BLEU** | 67.33 | 33.27 | 60.67 | 64.88 | 25.28 | 43.03 | 77.62 | 33.20 | 53.78 | 73.21 | 21.42 | 43.65 | 72.28 | 22.77 | 48.33 |
| **ROUGE1** | 0.8141 | 0.6280 | 0.8154 | 0.8270 | 0.6082 | 0.7431 | 0.8464 | 0.6374 | 0.7705 | 0.8182 | 0.5237 | 0.7105 | 0.8221 | 0.5350 | 0.7367 |
| **ROUGE2** | 0.7092 | 0.4041 | 0.6831 | 0.7326 | 0.3314 | 0.5466 | 0.7633 | 0.4146 | 0.6163 | 0.7348 | 0.2704 | 0.5184 | 0.7660 | 0.2595 | 0.5388 |
| **ROUGEL** | 0.7908 | 0.5972 | 0.7882 | 0.8193 | 0.5726 | 0.7328 | 0.8387 | 0.6090 | 0.7555 | 0.8073 | 0.5007 | 0.7034 | 0.8193 | 0.5081 | 0.7195 |
| **METEOR** | 0.7615 | 0.5502 | 0.7637 | 0.8007 | 0.4833 | 0.6749 | 0.8259 | 0.5922 | 0.7384 | 0.7990 | 0.4487 | 0.6533 | 0.7995 | 0.4717 | 0.7099 |
| **BERTScore - Precision** | 0.9367 | 0.8891 | 0.9394 | 0.9395 | 0.8883 | 0.9432 | 0.9552 | 0.9041 | 0.9521 | 0.9300 | 0.8787 | 0.9492 | 0.9416 | 0.8732 | 0.9519 |
| **BERTScore - Recall** | 0.9303 | 0.8838 | 0.9367 | 0.9341 | 0.8825 | 0.9385 | 0.9509 | 0.9013 | 0.9495 | 0.9382 | 0.8824 | 0.9542 | 0.9414 | 0.8640 | 0.9507 |
| **BERTScore - F1** | 0.9333 | 0.8864 | 0.9380 | 0.9366 | 0.8852 | 0.9407 | 0.9530 | 0.9025 | 0.9507 | 0.9336 | 0.8803 | 0.9515 | 0.9412 | 0.8685 | 0.9513 |

Table 9 - Results of the evaluation / benchmarking

I was not able to carry out a human evaluation for all languages. However, since Italian is my first language, I manually compared the results of the Roche model and my fine-tuned model. In general I did not notice significant quality differences between the 2 models. Both models are able to follow Roche standard terminology. The fine-tuned model appears to be better than the Roche model on recognizing parts of the sentence that shall remain untranslated (see Table 10 below), while the Roche model is better in staying close to the usual style of Roche Diagnostics technical content.

| Source | Fine-tuned model | Roche model | Reference translation |
|---|---|---|---|
| About the Calibration and QC widget | Informazioni sul widget **Calibration** and QC | Informazioni sul widget **Calibrazione** e QC | Informazioni sul widget **Calibration** and QC |
| About the Out of range results widget | Informazioni sul widget **Out of range results** | Informazioni sul widget **Risultati fuori range** | Informazioni sul widget **Out of range results** |
| cobas® infinity production monitoring | cobas® infinity production monitoring | **monitoraggio** della produzione cobas® infinity | cobas® infinity production monitoring |

Table 10 - Examples from the benchmarking data

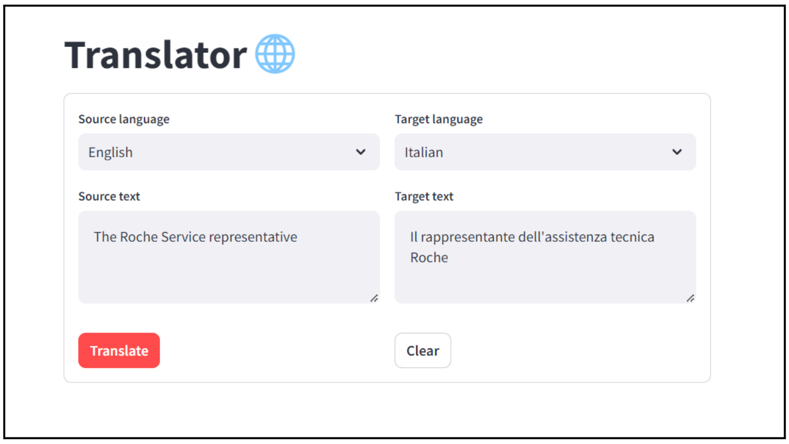
I have noticed some instances in which both models proposed a translation that was perfectly correct, however had some slight differences in word order or style from the reference translation (see Table 11). In order to have a more accurate and fair evaluation, it could be a good idea to have additional reference translations.

| Source | Fine-tuned model | Roche model | Reference translation |
|---|---|---|---|
| The new password is set. | Viene impostata la nuova password. | La nuova password è impostata. | A questo punto la nuova password è impostata. |

Table 11 - Examples of sentences with the same meaning but expressed differently

## 5.3. Front-end

In order to use the final fine-tuned model, I have decided to create a small front-end application using Streamlit (*[Streamlit • A faster way to build and share data apps]()*). The app can be reached via this URL: [https://cas-nlp-machine-translation.streamlit.app/](https://cas-nlp-machine-translation.streamlit.app/)



Source language: English
Target language: Italian
Source text: The Roche Service representative
Target text: Il rappresentante dell'assistenza tecnica Roche

Figure 14 - Front-end created with streamlit

## 6. Conclusion & outlook

In conclusion, this project focused on fine-tuning a transformer-based multilingual translation model for medical diagnostics technical content.
Through a systematic approach, I experimented with 3 transformer-based models: T5, FlanT5, and Helsinki-NLP/opus-mt-en-roa. The results indicate that the Helsinki-NLP/opus-mt-en-roa model performed the best, achieving an impressive BLEU score of 68 after fine-tuning.
The evaluation and benchmarking of the fine-tuned model were conducted using multiple metrics and showed that the fine-tuned model consistently outperformed the baseline for all languages. Although the Roche model generally outperformed the fine-tuned model in most metrics, the fine-tuned model demonstrated improvements in translation quality compared to the baseline and showed a very high BERTScore for all languages.

Looking ahead, there are several opportunities for further improvement and exploration. Increasing the dataset size and incorporating additional domain-specific data such as glossaries could enhance the performance of the models. One could also explore adding out-of-domain data to make the model more robust. Furthermore, conducting a human evaluation would provide a deeper understanding of the model's performance and its alignment with Roche Diagnostics' specific terminology and style. An evaluation on data rather than diagnostic technical content could also be performed, to measure the model's performance on unseen and out-of-domain data.

## 7. Appendix

GitHub repository: https://github.com/CBaffelli/CAS-NLP_Machine-translation

Front-end app: https://cas-nlp-machine-translation.streamlit.app/

## 8. References

FLAN-T5 (no date). Available at: https://huggingface.co/docs/transformers/model_doc/flan-t5 (Accessed: 20 May 2024).

Helsinki-NLP/opus-mt-en-roa · Hugging Face (no date). Available at: https://huggingface.co/Helsinki-NLP/opus-mt-en-roa (Accessed: 20 May 2024).

Jaejun Lee, Raphael Tang, and Jimmy Lin (2019) 'What Would Elsa Do? Freezing Layers During Transformer Fine-Tuning', ArXiv, abs/1911.03090. Available at: http://arxiv.org/abs/1911.03090.

Li, B., Hou, Y. and Che, W. (2021) 'Data Augmentation Approaches in Natural Language Processing: A Survey'. Available at: https://doi.org/10.1016/j.aiopen.2022.03.001.

NLTK :: Sample usage for wordnet (no date). Available at: https://www.nltk.org/howto/wordnet.html (Accessed: 21 May 2024).

PEFT (no date) Hugging Face. Available at: https://huggingface.co/docs/peft/index (Accessed: 19 May 2024).

spaCy · Industrial-strength Natural Language Processing in Python (no date). Available at: https://spacy.io/ (Accessed: 19 May 2024).

Streamlit • A faster way to build and share data apps (no date). Available at: https://streamlit.io/ (Accessed: 25 May 2024).

T5 (no date). Available at: https://huggingface.co/docs/transformers/model_doc/t5 (Accessed: 20 May 2024).

The Flan Collection: Advancing open source methods for instruction tuning (no date). Available at: http://research.google/blog/the-flan-collection-advancing-open-source-methods-for-instruction-tuning/ (Accessed: 23 May 2024).

Ting-Rui Chiang, Yi-Pei Chen, Yi-Ting Yeh, Graham Neubig (2022) 'Breaking Down Multilingual Machine Translation', Findings of the Association for Computational Linguistics: ACL 2022, pp. 2766–2780.

Translation task (no date) Hugging Face. Available at: https://huggingface.co/docs/transformers/tasks/translation (Accessed: 19 May 2024).