MACHINE LEARNING I - PROJECT

Contradictory, My Dear Watson

Centrale Supélec

MSc in Big Data Management and Analytics, 2022/23

Niccolò Morabito

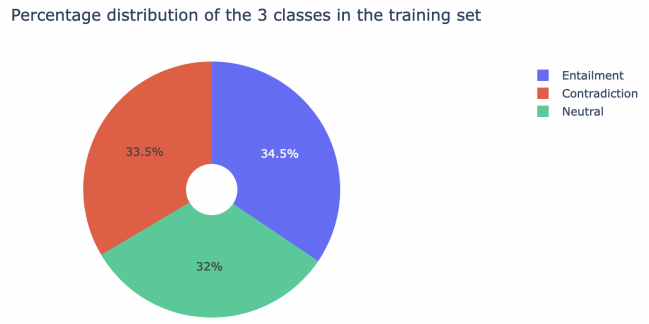
Yanjian Zhang

Introduction about the problem

Natural language processing (NLP) has increasingly grown over the past few years. Machine learning models tackle question-answering, text extraction, sentence generation, and many other complex tasks. The task of this project is to detect contradiction and entailment in multilingual text. To be specific, it is to create an NLI model that assigns labels of 0, 1, or 2 (corresponding respectively to entailment, neutral, and contradiction) to pairs of premises and hypotheses. The dataset and the task are taken from the competition “Contradictory, My Dear Watson” on Kaggle.

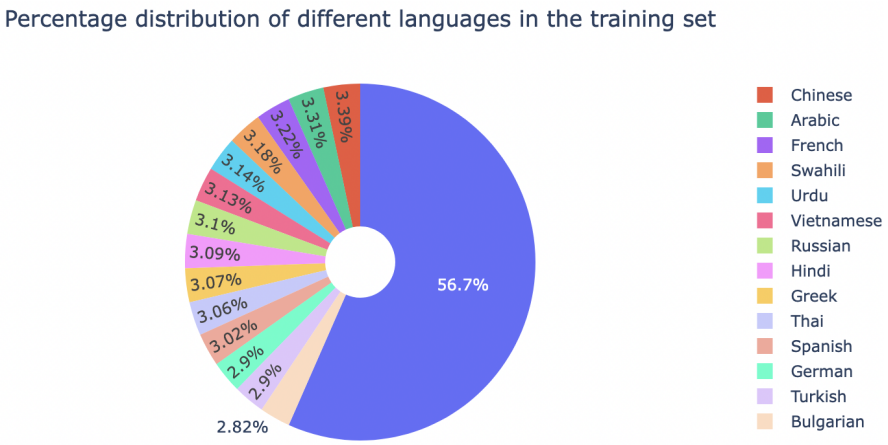
Exploratory Data Analysis

The competition provides a training set and a test set with 12120 and 5195 instances respectively, each containing a negligible `id`, a `premise` and a `hypothesis`, the `language` of the sentence and the `label` for the training instances. The different classes in the training set (Entailment, Contradiction and Neutral) are pretty balanced, as the following picture shows.

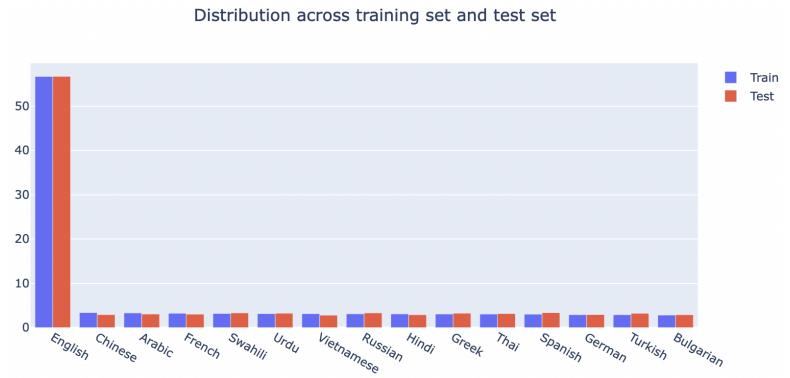


Languages

Looking at the distribution of the languages in the training set, we can notice a big dominance of English, covering more than half (56.7%) of the instances. In comparison, the remaining 14 languages are uniformly represented (about 3% of the instance for each of them).



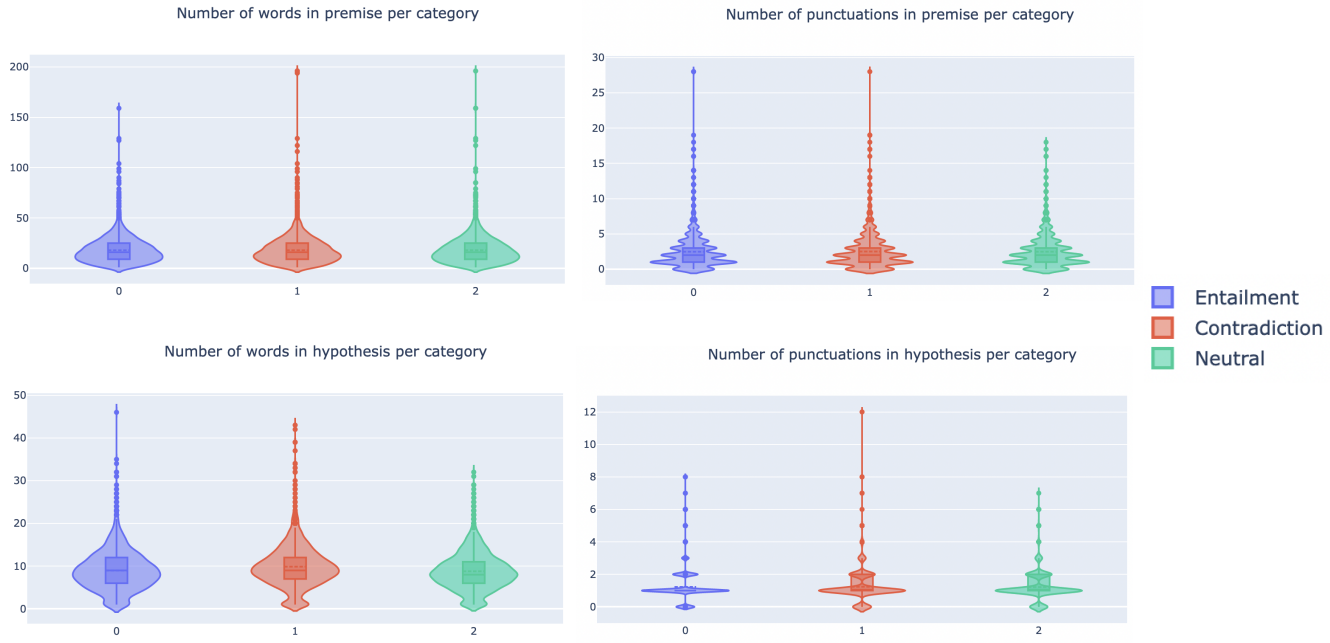
Comparing these results with the ones obtained in the test set, we can realise that they are pretty similar, as the following bar chart shows.



This imbalance in the languages already underlines the biggest weakness of the training set, which we face in the following sections to improve the model.

Words and punctuation in the classes

Analyzing the premises, the distribution of words across the classes is almost the same for contradiction and neutral, whereas, it is a little bit less in entailment. On the other hand, the distribution of punctuations across the classes is almost the same for contradiction and entailment, whereas, it is a little bit less in Neutral. As the top images show below.



Bigger differences among the classes can be noticed by looking, instead, at the hypothesises, in the below images. The distributions are still similar but with a decreasing number of words between entailment, contradiction and neutral and a significantly bigger number of punctuations for contradiction.

Data preparation

All the instances must be properly prepared before being inputted into the model: the preprocessing, common to most of the NLP tasks, includes the following steps:

- tokenize each of the two sentences (the premise and the hypothesis);
- add a token (`<CLS>` for BERT and `<s>` for Roberta) at the beginning of each input to contain the final classification;
- add a token (`<SEP>` for BERT and `</s>` for Roberta) between the two concatenated sentences;
- truncate or pad the input to a fixed number of tokens;
- construct attention masks.

Models comparison

In order to train the model, in addition to the training and the test sets, a validation set is needed to check the evolution of the accuracy on unseen data during the training. Such a set is obtained by sampling 10% out of the training set, even though additional experiments have been carried out using cross-validation to train different models (the most relevant results are summarized in the table below).

In order to maximize the accuracy of the classification, many models, including BERT [1], XLM-Roberta [2] have been trained and tested to choose the best one, but always exploiting pre-trained models in order to optimize the process and minimize the training time. The best results, predictably, are obtained with RoBERTa, and that is why we choose such a model for most of the attempts.

The following table summarizes the main characteristics of each attempt and the accuracy obtained on the test set (submitting on the Kaggle competition, i.e. using unknown and unavailable data).

Pre-trained model name	Number of epochs	Accuracy on test set	Characteristics
bert-base-multilingual-uncased	5	0.647	1 model
bert-base-multilingual-cased	5	0.639	1 model
xlm-roberta-base	5	0.701	1 model with more data
xlm-roberta-base	3	0.706	3 models
xlm-roberta-base	5	0.708	1 model

xlm-roberta-base	**5**	**0.711**	**1 model adding 2000 translated instances**

The number of epochs has been set according to the validation accuracy during training through an early-stop strategy, saving the model with the greatest result on the validation set.

It is clear that the state-of-the-art conclusions are valid also in this use-case since the models based on BERT obtain significantly lower accuracy than those based on RoBERTa (5/6% less).

As mentioned in the EDA section, the main problem of the data was the class imbalance and, in particular, the under-representation of most of the languages. That is why most of our effort focused on dealing with this problem through data augmentation, translating the sentences from English to the less represented languages.

First of all, we used a dataset provided by a Kaggle user, containing the translations of all the examples in English to all the other languages in the original competition data using the M2M100 (a multilingual encoder-decoder model trained for Many-to-Many multilingual translation) from Hugging Face. The model that has been trained with this data did not reach better test accuracy than the model trained with the standard data even though during the training the validation accuracy was very close to 100% even right after the first epoch.

This can be explained by the overfitting of the model to the newly generated instances, which apparently are not representative of the original data and that ended up being the big majority of the original dataset: in fact, the number of instances of the augmented dataset is 181800 (90% of which is used for training), i.e. 15 times the size of the original dataset. This means that what the model learns from most of the data does not help to generalize. Therefore, we focused on finding a better translation of the sentences in order to make the generated data more representative.

Since another experiment on data augmentation using Google Translator has already been carried out (reaching a very satisfying accuracy of 0.774), we wanted to test another translator. We ended up using UlionTse's translators but, due to the limitation of API requests, our test translates only 2000 instances of the original dataset.

Specifically, we picked 1000 (premise, hypothesis) pairs in English, and randomly chose a different language to translate into. Also, we picked 1000 pairs NOT in English and translated them in English. Training with this small amount of additional data caused a little increase in the test accuracy, showing again that more effort in this regard could improve the performance of the model.

Conclusions

In this report, we have carried out the exploratory data analysis of the original dataset, in several dimensions, including language proportion and word length. We have conducted a number of comparisons over large-scale pre-trained models and their hyperparameters selection, finally choosing XLM-Roberta as our model. To further improve the performance,

we utilized translation methods to augment the dataset, obtaining small improvements that, together with other experiments from the community, show that further work could be carried out in this regard to improve the accuracy.

References

[1] BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding
[2] Unsupervised Cross-lingual Representation Learning at Scale