Task on Oppositional thinking analysis: Conspiracy theories vs critical thinking narratives

Iñaki Diez Lambies - Juan Alejandro Furió Agustí

Team: CHEEXIST

# Common work

## Scheduler and early stopping mechanism.

To improve model training process, we integrated a learning rate scheduler into the code. Unlike fixed epochs, the scheduler adjusts the learning rate dynamically, starting high to speed up convergence and then reducing it for fine-tuning. This approach often results in better generalization and accuracy by preventing the model from getting trapped in suboptimal minima.

Additionally, we implemented an early stopping mechanism. This technique halts training if the Matthews Correlation Coefficient (MCC) metric shows no improvement over 10 epochs, preventing overfitting and saving computational resources. Early stopping ensures the model is trained just enough to reach optimal performance without unnecessary iterations.

Cross-validation was employed with both the scheduler and early stopping mechanisms in place. This setup allowed us to fine-tune the model parameters effectively, using the epoch count from cross-validation to guide the final training. The final model training used a fixed epoch count based on these insights, applied to the entire dataset.

These strategies collectively enhanced training efficiency and model reliability, ensuring robust performance across our experiments. By dynamically adjusting the learning rate and applying early stopping, we achieved more effective training methods that could lead us to better results.

# Task 1 - Distinguishing between critical and conspiracy texts

## Experiments

### Data augmentation: translation and backtranslation

To enhance the training dataset for this task, we employed data augmentation techniques involving translation and backtranslation. This approach was implemented to create multiple versions of the dataset, thereby enriching the training data and improving model robustness.

Translation and backtranslation involve translating the original text to another language and then back to the original language. This process introduces variations in the text that can help the model generalize better. For instance, an English text might be translated to French and then back to English, resulting in slight changes in wording and structure. These variations help the model learn more diverse linguistic patterns and improve its ability to generalize to unseen data.

We developed a script that modifies the original dataset loading functions to incorporate these augmented datasets. The script allows training with different versions of the dataset: the original language, the original plus its translation, or the backtranslated version from languages like Italian, German, and French. This flexibility enables us to create a richer training corpus, potentially leading to better model performance.

The implementation involves loading the dataset, translating the text using a pretrained MarianMT model from Hugging Face, and then optionally backtranslating it. The translated datasets are cached for efficiency, so repeated translations are avoided. During training, the augmented datasets are seamlessly integrated, providing the model with a diverse set of training examples.

The benefits of this approach are manifold. Firstly, it increases the dataset size, providing more training samples without requiring additional data collection. Secondly, it introduces linguistic variability, helping the model become more robust to different phrasings and expressions. Lastly, it can help mitigate overfitting by exposing the model to a broader range of inputs, thus improving its generalization capabilities.

### Classification layer modification

To enhance the model's capability in extracting relevant information, we replaced the default classification layer with a custom one. This custom classification layer aims to expand the dimensions of the embedding output before reducing it, thereby allowing the model to capture and distill more nuanced information from the text embeddings.

The modification involved creating a custom top module that consists of several fully connected (fc) layers. The process begins with expanding the dimensions through one or more "up" layers, each followed by a ReLU activation function. This expansion allows the model to increase the complexity of the representation, making it more capable of capturing intricate patterns and dependencies in the data.

Following the expansion, the output is then gradually reduced through a series of "down" layers, again interspersed with ReLU activations. This reduction compresses the enriched information into a lower-dimensional space, ideally preserving the critical features while discarding irrelevant noise. The final output layer maps this reduced representation to the desired number of classes for classification.

This custom top module was integrated into the model during initialization, replacing the default classification layer. By doing so, we enabled the model to process and refine the embeddings ideally more effectively, which is expected to improve its classification performance.

### Masking words from the corpus

To enhance the robustness and generalization of our models, we implemented a word masking strategy. This process involved analyzing the word frequency distributions in both English and Spanish datasets for each class (critical and conspiracy). We generated histograms to identify the 100 most frequent terms in each class, then isolated the words unique to each class. These "unique" words, which appeared prominently in one class but not in the other, became the focal point of our masking strategy.

During training, we preprocessed the input texts by masking words with a certain probability. This masking could either target the unique words specific to each class or apply randomly across the text to ensure a subtle yet effective transformation. By masking these unique words, we aim to create a more challenging training environment for the model, forcing it to learn to identify class-specific features without relying too heavily on the presence of particular terms. This approach helps the model to generalize better when encountering new, unseen texts.

The benefits of this masking strategy are manifold. First, it reduces the model's dependence on specific words, encouraging it to capture broader contextual and semantic patterns. Second, it makes the model more resilient to variations in text, as it learns to make predictions based on a more comprehensive understanding of the text rather than specific keywords. Third, this method can help mitigate the risk of overfitting, as the model is continually exposed to slightly altered versions of the input data, enhancing its ability to generalize from the training data to real-world scenarios.

Implementing this masking strategy involves dynamically modifying the input text during training. Each word in the text has a probability of being replaced by a mask token. If the word is part of the unique word list for its class, it has a higher chance of being masked. This dynamic preprocessing step ensures that the model is consistently challenged, promoting robust learning and improving its performance in distinguishing between critical and conspiracy narratives. By applying such a nuanced masking technique, we enhance the model's ability to perform well across varied and unpredictable inputs.

### New models

For our tasks, we opted to use specialized models tailored for detecting fake news: one for English and one for Spanish. Both models are based on the RoBERTa architecture, which offers several advantages over the traditional BERT model.

Specialized models like `jy46604790/Fake-News-Bert-Detect` and `Narrativaai/fake-news-detection-spanish` are trained on large corpora specifically curated for fake news detection. This focus allows them to capture the nuances and linguistic patterns associated with deceptive content, which can be crucial for our tasks of distinguishing between conspiracy and critical texts.

These models have been fine-tuned on datasets containing various types of news articles, making them adept at identifying subtle indicators of misinformation. By leveraging such models, we can benefit from their pre-existing knowledge and expertise in the domain of fake news, potentially leading to better performance in our specific tasks.

RoBERTa (Robustly optimized BERT approach) improves upon BERT by utilizing larger training data and longer training periods, among other optimizations. It discards BERT's Next Sentence Prediction (NSP) objective, which has been shown to be less beneficial, and instead focuses on dynamic masking and training on longer sequences. This results in a model that is often more effective at capturing context and relationships within text.

For the English dataset, we used `jy46604790/Fake-News-Bert-Detect`, a model fine-tuned on over 40,000 news articles from various media sources. This model, based on `roberta-base`, is specifically designed to identify fake news, making it well-suited for our task.

For the Spanish dataset, we selected `PlanTL-GOB-ES/roberta-base-bne`, which is based on the RoBERTa base model and fine-tuned with the largest Spanish corpus known to date made by the National Library of Spain. This model's fine-tuning on language-specific data aims to enhance its ability to detect misleading information in Spanish texts.

## Results

The experiments are divided into:

* Base
  + Using the default settings but with the new scheduler and early stopping techniques (all experiments do this).
* CTop
  + Custom with new classification layer.
  + CTop <augment>-<reduce> where *augment* is the times the dimension is multiplied by 2 and the opposite goes to the *reduce* by diving per 2 the dimension that number of times.
* Trans
  + Translated corpus experiments.
  + Trans AC: Augmented corpus by translating the other language (*en* to *es* and the other way around) into the experiment language.
* Final
  + Combining what we think could be the best: alt models plut CTop 1-2.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Exp. name | Corpus language | Mean F1\_macro | Mean F1 | Mean F1-neg | Mean ACC | Mean prec | Mean recall | Mean MCC |
| Base | en | 0.876 | 0.913 | 0.838 | 0.887 | 0.916 | 0.914 | 0.757 |
| es | 0.839 | 0.888 | 0.789 | 0.854 | 0.864 | 0.914 | 0.683 |
| CTop 2-2 | en | 0.889 | 0.926 | 0.852 | 0.901 | 0.912 | 0.940 | 0.780 |
| es | 0.842 | 0.889 | 0.794 | 0.856 | 0.870 | 0.909 | 0.686 |
| CTop 1-2 | en | 0.885 | 0.923 | 0.847 | 0.897 | 0.908 | 0.939 | 0.771 |
| es | 0.833 | 0.882 | 0.783 | 0.848 | 0.865 | 0.902 | 0.670 |
| Trans  AC | en | 0.858 | 0.900 | 0.816 | 0.870 | 0.897 | 0.902 | 0.717 |
| **es** | **0.855** | **0.901** | **0.809** | **0.870** | **0.883** | **0.920** | **0.712** |
| Alt models | en | 0.885 | 0.922 | 0.849 | 0.897 | 0.917 | 0.928 | 0.772 |
| es | 0.838 | 0.880 | 0.796 | 0.849 | 0.886 | 0.876 | 0.678 |
| Final | **en** | **0.890** | **0.926** | **0.853** | **0.902** | **0.912** | **0.941** | **0.781** |
| es | 0.835 | 0.884 | 0.786 | 0.850 | 0.867 | 0.903 | 0.673 |

# Task 2 - Detecting elements of the oppositional narratives

In the second task, we focused exclusively on experiments that yielded better results to ensure the highest possible accuracy and efficiency in detecting elements of oppositional narratives. By concentrating on the top-performing experiments, we leveraged the insights and optimizations already proven effective in the first task. This strategic decision allowed us to streamline our efforts and resources, reducing the time and computational power required to achieve high-quality results.

## Experiments

The experiments are divided into:

* Base
  + Using the default settings but with the new scheduler and early stopping techniques (all experiments do this).
* CTop (described in section above)
* Alt models (described in section above)
* Final
  + Also the combination of alt models and CTop 1-2

## Results

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Exp. name | Corpus language | Mean F1 | Mean prec | Mean recall |
| Base | en | 0.523 | 0.449 | 0.672 |
| es | 0.463 | 0.430 | 0.520 |
| CTop  1-2 | en | 0.420 | 0.347 | 0.562 |
| es | 0.449 | 0.384 | 0.576 |
| Alt models | **en** | **0.554** | **0.493** | **0.667** |
| es | 0.501 | 0.474 | 0.558 |
| Final | en | 0.427 | 0.363 | 0.556 |
| **es** | **0.515** | **0.463** | **0.623** |

# Conclusions

In our project we implemented several strategies, including data augmentation, custom classification layers, and specialized models, to enhance the performance of our classifiers. This experiments yielded varying results across different configurations and tasks, revealing valuable insights into what approaches were most effective.

The first task showed that the best-performing strategy involved the use of custom classification layers, specifically the CTop 2-2 configuration. This method significantly improved the model's ability to capture nuanced patterns within the text, leading to higher performance metrics. For English, this approach resulted in a Mean F1\_macro of 0.889 and a Mean MCC of 0.780, while for Spanish, it achieved a Mean F1\_macro of 0.842 and a Mean MCC of 0.686. These results indicate that expanding and then compressing the dimensionality of the text embeddings allowed the model to distill more relevant information, thus enhancing its classification capabilities.

Additionally, data augmentation through translation and backtranslation also contributed positively, particularly in the Spanish corpus. This technique introduced linguistic variability, which helped the model generalize better by exposing it to different phrasings and expressions. However, the results from this method, while beneficial, did not surpass those obtained from the custom classification layers.

On the other hand, some approaches did not perform as expected. The baseline models, even with the implementation of a learning rate scheduler and early stopping mechanisms, showed comparatively lower performance. This was particularly evident in the English corpus, where the Mean MCC was only 0.757. This suggests that while these techniques helped prevent overfitting and improved training efficiency, they were not sufficient on their own to achieve optimal performance.

In the second task we streamlined our efforts by concentrating on the top-performing experiments from the first task. This strategic focus allowed us to ensure the highest possible accuracy and efficiency. Among the various approaches, the use of alternative models, specifically tailored for detecting fake news, yielded the best results. For English, the alternative models achieved a Mean F1 of 0.554, while for Spanish, the Mean F1 was 0.501. These models, built on the RoBERTa architecture and fine-tuned on large corpora specifically curated for fake news detection, proved to be adept at identifying subtle indicators of misinformation, thus enhancing their performance in our tasks.

Nevertheless, some results were below our expectations. For instance, the CTop configuration, despite its success in the first task, did not perform as well in detecting elements of oppositional narratives. This outcome highlights the challenge of transferring successful strategies from one task to another, indicating that certain methods may be more effective for specific types of text classification problems.

Overall, we consider our work a success for several reasons. First, we demonstrated the effectiveness of advanced techniques such as custom classification layers and data augmentation in improving model performance. The significant improvements in metrics, particularly with the CTop 2-2 configuration, validate our approach and showcase the potential of these methods in text classification tasks.

Moreover, the specialized models tailored for fake news detection proved to be valuable assets, particularly in the context of detecting oppositional narratives. By leveraging these pre-existing models, we could achieve higher accuracy and efficiency, underscoring the importance of selecting the right tools for specific tasks.

While some results did not meet our expectations, these outcomes provided valuable lessons. They highlighted the need for continuous experimentation and adaptation, as not all techniques will perform equally well across different tasks. The insights gained from these experiments will guide future research and optimization efforts.