

# Project proposal

## Authors

- Aitor González Marfil
- Carlos Domínguez Becerril

## Title

Fake news detection in combination with machine learning

## Tutor

Gorka Azkune Galparsoro

## Description

Fake news and rumors are rampant on social media, and believing in them can cause significant harm. In this project, we focus on detecting fake news about Covid-19 that are published on Twitter, Facebook, Instagram, PolitiFact, and so on.

The work is based on the following two papers:

- <https://arxiv.org/ftp/arxiv/papers/2011/2011.03327.pdf> (the main one, includes dataset)
- <https://arxiv.org/pdf/2101.03545.pdf>

## Related work

Different papers using the dataset are reviewed to compare the results and obtain ideas for the architecture developed. The list of papers can be found in the [papers with code](#) webpage.

## Dataset

**Dataset URL:** [https://github.com/diptamath/covid\\_fake\\_news/tree/main/data](https://github.com/diptamath/covid_fake_news/tree/main/data)

### Information about the dataset:

- **Splits:**

Split	Real	Fake	Total
Training	3360	3060	6420
Validation	1120	1020	2140
Test	1120	1020	2140
<b>Total</b>	5600	5100	10700

Table 3: distribution of data across classes and splits. Note that the data is class-wise balanced and the class-wise distribution is similar across splits.

- **Sources of the posts:** Twitter, Instagram, Facebook, Politifact, NewsChecker, Boomlive, and so on.
- **Dataset structure:** For each example, the dataset contains 3 different columns. The id of the text, the label (fake or real), and the text.

## Algorithm architecture

We are going to extract different features from the text and use them to decide whether a text is fake news or not.

The features that we are going to use:

- **Text analysis:** Use a transformer to analyze the text and obtain whether it is fake or not. We are going to fine-tune a transformer like BERT (to be decided). Note that this is going to be one more feature to be used.
- **Sentiment analysis:** Analyze the sentiment of the text using a pre-trained model.

Other features that can be used (to be analyzed during development):

- **Word/character count:** Count the number of words/characters. Differentiating between short texts (Twitter, Instagram...) and long texts (Politifact, NewsChecker, ...) might be important because usually social media posts are shorter than the news.
- **Formality:** Check whether the text is formal or not (using a pre-trained model). Social media posts tend to be more informal than news from a newspaper.
- **The number of sentences:** Check the number of sentences that the text has.
- **Sentences length:** Check the length of the sentences. Maybe short sentences are used to make the topic vaguer.
- **Punctuation:** number of spaces, paragraphs, hashtags, and so on
- **Sources:** Check whether the message sends you to another source using URLs.

**Note:** The number of features may vary during the development.

With the extracted features we are going to use both, classical classifiers like decision tree, support vector machine, and so on, and deep learning classifiers to decide whether a text is fake news or not.

## Evaluation

To evaluate the task, we are going to use the following metrics (the ones used in the papers):

- **Accuracy**
- **Precision**
- **Recall**
- **F1-Score**

## Expected results

We expect that using deep neural networks for extracting complex features (text analysis, sentiment analysis, and formality) and combining them with more simple features (word/character count, punctuation, sources) can help to improve the results.

Taking into account that we do not know if it is going to work as expected, we are going to analyze and compare the performance of several configurations using both classical machine learning and deep learning.