# Approaches to the Profiling Fake News Spreaders on Twitter Task in English and Spanish

Jacobo López Fernández[1] and Juan Antonio López Ramírez[2]

Universitat Politècnica de València
jalofer1@posgrado.upv.es, jualora1@inf.upv.es

**Abstract.** This paper discusses the decisions made approaching PANs Profiling Fake News Spreaders on Twitter Task at CLEF 2020. We briefly describe how we combined author tweets to create samples that do or do not represent a Fake News Spreader. We decided to handle both languages proposed for this task: Spanish and English; and the methodologies that we suggested were Linear Support Vector Machines (SVMs) and Gradient Boosting, respectively. Other approaches such as Long Short-Term Memory (LSTM) were taken into account in the process of finding a model with the best accuracy results and these were also reported in this paper. We made use of the cross-validation scenario to obtain accuracy results due to the reduced amount of data. We have managed to achieve average accuracy scores of 0.735 for the Spanish language identification task and 0.685 for the English language identification task.

**Keywords:** author profiling, fake news, multilingual, social media, spreaders

## 1 Introduction

The trust in information read through social media has steadily increased in the last few years. However, allow their accounts to publish and propagate misinformation with severe consequences for our society. First of all, we should make it clear that there are different types of misinformation and disinformation, such as fake news, satire or rumours that go viral in online social networks [5]. In addition, psycho-linguistic information as emotion, sentiment or informal language should be previously analised. Exploiting information extracted from user profiles and user interactions, we should be able to classify them depending on the information obtained.

A great amount of fake news and rumors are propagated in online social networks with the aim, usually, to deceive users and formulate specific opinions [15]. Users play a critical role in the creation and propagation of fake news

online by consuming and sharing articles with inaccurate information either intentionally or unintentionally.

To prevent dissemination of misinformation and disinformation we may profile fake news spreaders automatically. Author profiling is based on the detection of certain characteristics in profiles making use of linguistic pattern recognition techniques.

This paper discusses the decisions made approaching PANs' Profiling Fake News Spreaders on Twitter Task at CLEF 2020 [11]. The task consists in, given a Twitter feed, determine whether its author is keen to be a fake news spreader. So, it focuses on identifying possible fake news spreaders on social media as a first step towards preventing fake news from being propagated among online users. The task has a multilingual perspective, so that includes tweets in English and Spanish. It is defined as a binary classification task.

## 2 Related Work

Fake news detection has attracted a lot of research attention in the last years. Guess et al. [7] made an approach to this field by doing research during the elections, in particular the 2016 US election procedure. In that paper, they propose a system that obtained features from polls published on Facebook. Popat et al. [9] suggested an end-to-end model to evaluate trust on random texts, without human supervision. Subsequently, they presented a biLSTM neural network model which aggregates signals from external evidence articles, the language of these articles and the trustworthiness of their sources.

Shu et al. [14] pointed that fake news spreaders cannot be profiled precisely based only on text content, but we should understand the correlation between user profiles on social media and fake news. They state that social engagements should be used as auxiliary information to improve fake news detection systems. In addition, Sliva et al. [13], distinguished that, approaching content from a data mining perspective, we could identify patterns that could mark a text as fake. These patterns, such as clearness which make the text more readable and could convince the receiver even when it is fake. Collecting this kind of information produces a huge, unestructured, incomplete and noisy data; difficult and expensive to manage. Giachanou et al. [6] proposed EmoCred that incorporates emotions that are expressed in the claims into an LSTM network to differentiate between fake and real claims.

## 3 Fake News Spreaders Detection Systems

First, we apply the same type of preprocessing for both, the English and Spanish tasks data, following the next steps:

- Load tweets from XML files.
- Concatenate tweets forming a chain for every author. All tweets in this chain are separated by a blank space. We apply this technique on the English dataset and the Spanish dataset.

With the concatenated data, we vectorized our samples. The vectorizers used to perform this task were *CountVectorizer* and *TfidfVectorizer* [12]. CountVectorizer creates valuable data from counting words in samples while TfidfVectorizer takes into consideration more common words in detriment of those which are less common.

Concurrently, the tokenizer selected was *casual_tokenize*, an implementation of TweetTokenizer from NLTK, due to its suitability to manage characters and expressions commonly used on the Twitter social network.

After all these transformations, we ended up getting a feature matrix for each of the languages proposed, English and Spanish.

### 3.1 First approaches

The first method applied to classify our samples employed Recurrent Neural Networks (RNN), making use of pretrained *word embeddings* in order to help representing words as real-valued vectors and lead to better performance of our neural network system.

For the Spanish language task, the *embeddings* loaded by our system were the *Spanish Billion Words Corpus and Embeddings* [1] which had been trained using *word2vec* and consists of near 1 million words where each of them is represented as a vector with a size of 300.

For the English language task, the *embeddings* loaded by our system had been trained using *GloVe* from Stanford [8] and collected by Laurence Moroney, composed of near 6000 billion words where each of them is represented as a vector with a size of 100.

Once the *embeddings* were loaded, we trained our RNN model with LSTM and the following topology:

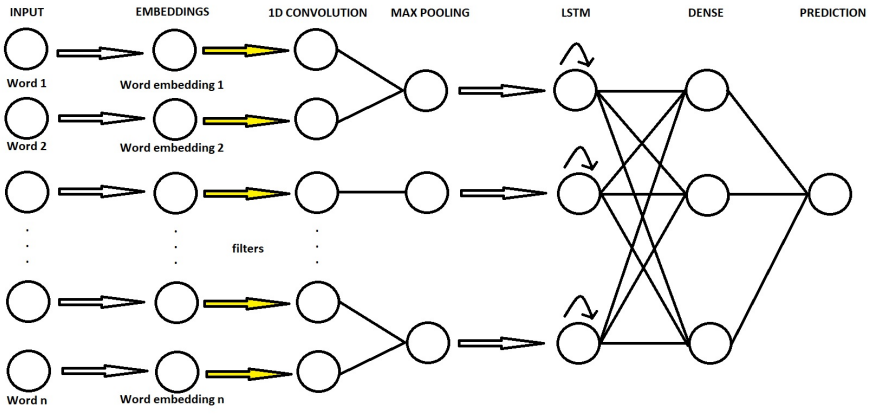


**Fig. 1.** Topology of convolutional RNN and LSTM.

### 3.2 Final systems

Despite the fact that the results obtained making use of RNN were close to those reported in the state of the art for this kind of tasks, we did not reach promising results as we will explain later in this paper. At that point, we made use of classifiers provided by the framework scikit-learn and chose the Gradient Boosting algorithm and the linear SVM algorithm for the English and Spanish tasks, respectively.

The main core of Gradient Boosting [4] consists of a predictive model based on decision trees, built step-by-step allowing the optimization of a differentiable loss function. For this function we made use of linear regression or 'sigmoid', called 'deviance' in the scikit-learn framework, whose mathematical expression is represented as the following:

$$\sigma(z) = \frac{1}{1 + e^{-z}} \tag{1}$$

We also experimented using the AdaBoost [3] algorithm along with the loss function, called 'exponential' in the scikit-learn framework. This algorithm focuses on classification problems and aims to convert a set of weak classifiers into a strong one. The final equation for classification can be represented as:

$$F(x) = sign(\sum_{m=1}^{M} \theta_m f_m(x)) \tag{2}$$

where $f_m$ stands for the $m_{th}$ weak classifier and $\theta_m$ is the corresponding weight. It is exactly the weighted combination of M weak classifiers. The function which gives the weight for the $m_{th}$ weak classifier is the following:

$$\theta_m = \frac{1}{2} ln(\frac{1 - \epsilon_m}{\epsilon_m}) \tag{3}$$

where $\epsilon_m$ is the lowest weighted classification error.

From another point of view, the main core of SVM [2] is based on the concept of separating a group of points (samples) into two different categories. As a consequence, our model had to be able to classify the sample correctly into its category. SVM looks for a hyperplane which optimally separates the points belonging the two classes. Subsequently, we look for the hyperplane with the longest distance (margin) to the closest points to it.

The equation of the hyperplane in the 'M' dimension can be given as:

$$y = b + \sum_{i=1}^{M} w_i x_i \tag{4}$$

where $w_i$ are vectors, b is biased term and $x_i$ are input variables.

Furthermore, given a group of points $S = (x_1, c_1), ..., (x_N, c_N)$ and a constant C>0, we should obtain weights $\theta \in \Re^d$, $\theta_0 \in \Re$ and the tolerance parameter $\varsigma \in \Re^N$ to minimize the following expression:

$$\frac{1}{2}\theta^t\theta + C\sum_{n=1}^{N}\varsigma_n \tag{5}$$

Dependant of the following two expressions:

$$c_n(\theta^t x_n + \theta_0) \geq 1 - \varsigma_n, 1 \leq n \leq N \tag{6}$$

$$\varsigma_n \geq 0, 1 \leq n \leq N \tag{7}$$

At last, the loss function employed was the hinge function, represented as:

$$L(y) = max(0, 1 - t \cdot y) \tag{8}$$

where y is the prediction and t is the intended output.

## 4 Experimental Setup

In this section we discuss about the dataset provided and the experimental adjustments we made.

### 4.1 Dataset

The given dataset is composed of two folders, a folder for the Spanish language and a folder for the English language, which contain:

- A XML file by author or Twitter user profile. There are 100 tweets in each XML file.
- A text file with the list of authors and the ground truth.

### 4.2 Experimental Settings

The submission of our system was made from the TIRA platform [10]. As the participants were only provided with the training data, we applied cross-validation into 10 folds in order to test our system. Different classifiers were tested such as Support Vector Machine, Gaussian Naive-Bayes, Gradient Boosting, Stochastic Gradient Descent, K-nearest Neigbours and two Neural Network approaches (Multilayer Perceptron and Convolutional Recurrent Neural Networks with LSTM). Regarding the English task we obtained the best results with a Gradient Boosting model and regarding the Spanish task we obtained the best results with a linear SVM model. Concerning SVM, we set the penalty parameter 'C' to 100 and the tolerance parameter to 0.01, with the maximum number of stages set to 100. In our SVM implementation we operated with the hinge loss function whereas in our Gradient Boosting implementation we operated with the deviance function. Learning rate was set to 0.01 for Gradient

Boosting and the number of boosting stages to perform was 250. To implement our system we made use of the scikit-learn framework [1].

For the Spanish language task we used a language model based on bigrams and trigrams where punctuation marks are processed and a vocabulary is built considering the top 1000 features, ordered by frequency from the whole corpus.

## 5 Results

Table 1 shows the results achieved by our system in the Spanish and English tasks in terms of precision over the training data. As we mentioned earlier in this article, we are using a 10 fold cross-validation where the average precision of the 10 folds gives the final precision result for each classifier. The best results in terms of precision were those given by the linear SVM model for the Spanish language task and the Gradient Boosting model for the English language task. We should mention that the results shown in this table were obtained without modifying any hyperparameter of the two classifiers.

| | Accuracy-English | Accuracy-Spanish |
|---|---|---|
| SVM | 0.63 | 0.74 |
| linear SVM | 0.64 | 0.83 |
| Naive Bayes | 0.64 | 0.70 |
| Gradient Boosting | 0.71 | 0.76 |
| SGD | 0.67 | 0.78 |
| Nearest Neighbors | 0.59 | 0.74 |
| MLP | 0.65 | 0.82 |
| RNN with LSTM | 0.60 | 0.66 |

**Table 1.** Accuracy scores obtained from Cross-Validation on the training set.

Table 2 shows the results obtained by modifying parameters with the Gradient Boosting algorithm for the English language. The results display the positive influence that reducing the learning rate to 0.01 and increasing the number of stages to 250 has on the system. As far as we are concerned, alterating the loss function did not report any change in the system results and was left as its default value: logistic regression. We should indicate that hyperparameters were as well adjusted in the linear SVM model for the Spanish language, but the disparity with the original precision results was considered trivial, achieving results close to 0.84.

Table 3 shows the results achieved by our Fake News Spreaders detection system in English and Spanish in terms of accuracy on the training set in TIRA. We observe that our system performs better for the Spanish tweets compared to English.

[1] `https://scikit-learn.org`

| Loss | Learning rate | Stages | Accuracy |
|---|---|---|---|
| deviance | 0.1 | 100 | 0.71 |
| deviance | 0.1 | 250 | 0.71 |
| deviance | 0.01 | 100 | 0.70 |
| deviance | 0.01 | 250 | 0.73 |
| deviance | 0.001 | 100 | 0.65 |
| deviance | 0.001 | 250 | 0.68 |
| exponential | 0.1 | 100 | 0.72 |
| exponential | 0.1 | 250 | 0.71 |
| exponential | 0.01 | 100 | 0.70 |
| exponential | 0.01 | 250 | 0.73 |
| exponential | 0.001 | 100 | 0.65 |
| exponential | 0.001 | 250 | 0.68 |

**Table 2.** Accuracy scores obtained by modifying hyperparameters of Gradient Boosting Classifier.

| | Accuracy |
|---|---|
| English | 0.98 |
| Spanish | 0.9967 |

**Table 3.** Accuracy scores of our system on the official training set using cross-validation.

Table 4 shows the results achieved by our Fake News Spreaders detection system in English and Spanish in terms of accuracy on the official test set. As we observed with the training data, our system performs better for the Spanish tweets compared to English.

| | Accuracy |
|---|---|
| English | 0.685 |
| Spanish | 0.735 |

**Table 4.** Accuracy scores of our system on the official test set.

## 6 Conclusions

In this paper we described our system for PANs Profiling Fake News Spreaders on Twitter Task at CLEF 2020. Regarding the English language task we proposed a system trained with Gradient Boosting algorithm while for the Spanish language task we proposed a system based on linear SVM. The state of the art tells us that Neural Networks are currently the best solution for this kind of classification tasks, but the results that we achieved do not match with this statement. We can consider that some tasks are not suitable to be addressed

with this kind of systems so far, as we saw with our implementation of Convolutional Recurrent Neural Network with LSTM. Eventually, a traditional machine learning algorithm performed better.

Our results showed that the input data processing considerably conditions performance in our system. In addition, from our results we can observe that our Spanish language solution performs better compared to our English language solution, as we managed to achieve 0.735 and 0.685 accuracy respectively.

## References


1. Cristian Cardellino. Spanish billion words corpus and embeddings.
2. Theodoros Evgeniou and Massimiliano Pontil. Support vector machines: Theory and applications. volume 2049, pages 249–257, 01 2001.
3. Yoav Freund and Robert E. Schapire. A short introduction to boosting. *Journal of Japanese Society for Artificial Intelligence*, 14, 1999.
4. Jerome H. Friedman. Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, 29:1189–1232, 2000.
5. Bilal Ghanem, Paolo Rosso, and Francisco Rangel. An Emotional Analysis of False Information in Social Media and News Articles. *ACM Transactions on Internet Technology (TOIT)*, 20(2):1–18, 2020.
6. Anastasia Giachanou, Paolo Rosso, and Fabio Crestani. Leveraging emotional signals for credibility detection. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR'19, page 877–880, New York, NY, USA, 2019. Association for Computing Machinery.
7. Andrew Guess, Jonathan Nagler, and Joshua Tucker. Less than you think: Prevalence and predictors of fake news dissemination on facebook. *Science Advances*, 5:eaau4586, 01 2019.
8. Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar, October 2014. Association for Computational Linguistics.
9. Kashyap Popat, Subhabrata Mukherjee, Andrew Yates, and Gerhard Weikum. Declare: Debunking fake news and false claims using evidence-aware deep learning. *CoRR*, abs/1809.06416, 2018.
10. Martin Potthast, Tim Gollub, Matti Wiegmann, and Benno Stein. TIRA Integrated Research Architecture. In Nicola Ferro and Carol Peters, editors, *Information Retrieval Evaluation in a Changing World*. Springer, September 2019.
11. Francisco Rangel, Anastasia Giachanou, Bilal Ghanem, and Paolo Rosso. Overview of the 8th Author Profiling Task at PAN 2020: Profiling Fake News Spreaders on Twitter. In Linda Cappellato, Carsten Eickhoff, Nicola Ferro, and Aurélie Névéol, editors, *CLEF 2020 Labs and Workshops, Notebook Papers*. CEUR-WS.org, September 2020.
12. Omid Shahmirzadi, Adam Lugowski, and Kenneth Younge. Text similarity in vector space models: A comparative study. *CoRR*, abs/1810.00664, 2018.
13. Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. Fake news detection on social media: A data mining perspective. *ACM SIGKDD Explorations Newsletter*, 19, 08 2017.

14. Kai Shu, Suhang Wang, and Huan Liu. Understanding user profiles on social media for fake news detection. In *Proceedings - IEEE 1st Conference on Multimedia Information Processing and Retrieval, MIPR 2018*, pages 430–435. Institute of Electrical and Electronics Engineers Inc., June 2018. 1st IEEE Conference on Multimedia Information Processing and Retrieval, MIPR 2018 ; Conference date: 10-04-2018 Through 12-04-2018.
15. Xinyi Zhou and Reza Zafarani. A survey of fake news: Fundamental theories, detection methods, and opportunities. 12 2018.