Using Machine Learning for News Verification

Gerardo Ernesto Rolong Agudelo[1], Octavio José Salcedo Parra[1,2(✉)], and Javier Medina[3]

[1] Faculty of Engineering, Universidad Distrital "Francisco José de Caldas", Bogotá D.C., Colombia
grolong@correo.udistrital.edu.co, osalcedo@udistrital.edu.co
[2] Department of Systems and Industrial Engineering, Faculty of Engineering, Universidad Nacional de Colombia, Bogotá D.C., Colombia
ojsalcedop@unal.edu.co
[3] Faculty of Engineering, GEFEM Research Group, Universidad Distrital "Francisco José de Caldas", Bogotá D.C., Colombia
rmedina@udistrital.edu.co

Abstract. The news fakes are issued with the intention of misleading, manipulating personal decisions, discredit or exalt an institution, entity or person or obtain economic gains or political revenue. They are related to propaganda and post-truth. Fake news, by presenting falsehoods as if they were real, are considered a threat to the credibility of serious media and professional journalists. The dissemination of false news in order to influence the behavior of a community has antecedents since antiquity, but given that its scope is directly related to the means of reproduction of information specific to each historical stage, its area and speed of propagation was scarce in the historical stages prior to the appearance of the mass media.

Keywords: Fake news · Machine learning · NLTK · Sklearn

1 Introduction

It is not possible to make a formal definition of a false news [1] since false news has become a place in today's society, from apparently innocuous publications on social networks [2] to web pages completely dedicated to the production of false information, but made in such a way that they can masterfully imitate some of the most recognized newspapers and news channels.

2 Related Work

Promising results were obtained with the previous research taking into account the existing limitations in terms of the reduced volume of the available information. This is only one of the aspects that affect this complex problem since this is a classification task that seeks optimal balance between the accuracy of the obtained classifications and the

T. K. Dang et al. (Eds.): FDSE 2018, LNCS 11251, pp. 251–257, 2018.
https://doi.org/10.1007/978-3-030-03192-3_19

computational cost [3]. It can be approached in two different ways since the dataset provides two types of information: it provides visual readings from the patient's vital signs that show the states of the body's various systems. Since these signals are so complex, it is necessary to train the system to recognize the current situation of the patient based on those images. In [4] two previously trained convolutional neural networks (CNN) were used and the machine managed a success rate of 83.2% when trying to classify images in 10 special categories: "ceremony", "concert", "demonstration", "football", "picnic", "race cars", "reunion", "swimming", "tennis", "traffic".

The other type of information that the dataset provides is a logbook that contains the records from the performed procedures to stabilize the patient. In previous work, where it is sought to predict future results by studying only the present that changes, a statistical analysis has been proposed in order to deliver success or failure rates [5]. This is only being explored up to now.

On the other hand in [6] present a review of several existing methods for the detection of false news, on the one hand there are works focused on the processing of news content and its form, those based on knowledge use external sources to verify the information exposed in the news. Those based on style seek to find within the news signs of language that demonstrates subjectivity or disappointment.

The study done in [7] makes an analysis of how bots have been used to spread false news on social networks like twitter and facebook.

In [8] they present a study of Deep Learning using natural language processing for the detection of false news; thus, different models are presented, and an assessment is made of which may be the best option to obtain adequate results.

3 Methodology

In the study carried out natural language processing (PLN) is used as a Python computational tool; This programming language uses different libraries and platforms, among them its PANDAS natural language processing library (Python Data Analysis Library) which is an open source library with BSD license that provides data structures and data analysis tools. Additionally, NLTK was used, which is a set of libraries and programs oriented to natural language processing and Scikit-learn which is a specialized machine learning library for classification, regression and clustering. The three libraries mentioned above have been designed to operate in conjunction with the other Numpy and Scipy libraries which were also included in the program.

To obtain news for the study, a public data set located in a github repository was used https://github.com/GeorgeMcIntire/fake_real_news_dataset compiled in equal parts for ten thousand five hundred and fifty-eight (10558) news items collected in total between the years 2015 and 2017 written in English with their title, full text and false or true label which were taken from different media, making scrapping processes in news web portals for half of real news and news from a published dataset in Kaggle conformed only by false news.

So once having the dataset, the methodology consisted of three fundamental stages; the pre-processing that involved transforming the dataset from a .csv file to a Python object belonging to Pandas; a data frame to be able to deal with it efficiently. Subsequently, for processing, the data was changed so that the first half of the data with false label and the second half with a true label were not simply what would cause impartiality when applying the machine learning methods. Once this is done, groups of data are taken to make training and test sets with which tokenisation algorithms are executed so that the result is processed by the Multinomial Naive Bayes algorithm of the Scikit-Learn package and finally an array was made in analysis. of confusion to make analysis of the results obtained.

4 Design

To begin with the processing of the data, it was necessary to use the read_csv() function of the Pandas library, passing the path of the file in which the.csv file is located, which converts to the Data Frame format. For the creation of the test and training sets, the train_test_split() function of the sklearn library was used, which takes as parameters the column with which the learning will be done, the type of classification that must be determined, the size with which will be the test set and a random to scramble the data.

Subsequently, sets "bags" of features are gathered, which are words or subsets of words with which you can extract the frequencies that have the word within the paragraphs belonging to the news texts with two different functions CountVectorizer(), but first it is necessary to do a new cleaning, since at the time of applying machine learning one looks for to see a relation between the veracity of the news and the words that more frequently appear in this one; as is logical there will be many occurrences of "stop words" is words like "that, in, on" these words that serve as connectors and to give structure to the sentences but semantically does not have a great meaning, so it becomes necessary to get rid of those "stop words" and then if you can proceed to build structures made up of all the words that are part of the news.

Having the sets of words conformed, the NaiveBayes() function is sent as arguments so that it makes the process of determining from the word bag and the training sets if the news should be classified as false or true and subsequently an open source function is used to graph a confusion matrix in which the main diagonal shows the quantity of correctly classified news and the ones that are not seen outside the diagonal.

5 Implementation

The first thing that was done so that the program functions correctly is to import the necessary libraries so that all the functions used are recognized by the interpreter.

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
from sklearn.metrics import confusion_matrix
from sklearn.feature_extraction.text import CountVector-
izer
from sklearn.feature_extraction.text import TfidfVector-
izer
from sklearn.naive_bayes import MultinomialNB
import pickle
import nltk
import numpy as np
import matplotlib.pyplot as plt
import itertools
```

The file is then imported into a DataFrame from the Pandas library and formatted to make it easy to manipulate the data using the following commands.

```
features =
pd.read_csv("fake_or_real_news.csv",usecols=['text', 'la-
bel'])
```

Thus, the parts that we will be using for this study of the news, the text and its actual classification have been stored in features; First, mixing them is done to avoid that the classification is affected by the order of the news. These two columns are separated and used to create the training and test sets.

```
features.sample(frac=1)

trainig_set = features.text[:1900]
label_train = features.label[:1900]
test_set = features.text[1900:]
label_test = features.label[1900:]
```

The following commands make the word arrays to be generated do not contain stop words.

```
count_vectorizer = CountVectorizer(stop_words='english')
tfidf_vectorizer = TfidfVectorizer(stop_words='english',
max_df=0.7)
```

Then other counVectorizer functions are used to do a tokenization and frequency count of the tokens and the result is put into matrices made up of the tokens of the test set and the evaluation set.

```
count_train = count_vectorizer.fit_transform(trainig_set)
count_test = count_vectorizer.transform(test_set)
```

To contrast, another way of counting the frequency of the tokens is used and again applied to the test and training sets.

```
tfidf_train = tfidf_vectorizer.fit_transform(trainig_set)
tfidf_test = tfidf_vectorizer.transform(test_set).
```

6 Discussion and Results Analysis

After running the Naive Bayes algorithm with the two forms of tokenization Count-Vectorizer and TfidfVectorizer the following percentages of certainty were obtained:

– Count Vectorizer: certainty: 0.881
– TfidfVectorizer: certainty 0.848

Likewise, the confusion matrices were plotted with the results of both classifications, producing the following results:

TfidfVectorizer (Fig. 1):

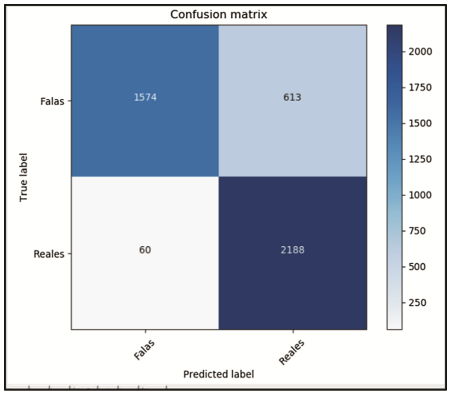

Fig. 1. Confusion matrix: TfidfVectorizer. Source: Authors

Count Vectorizer (Fig. 2):

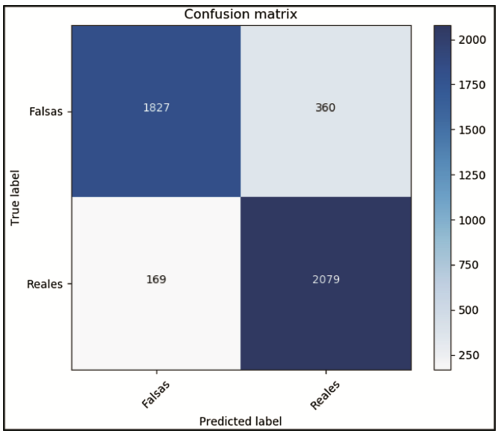

Fig. 2. Confusion matrix: Count Vectorizer. Source: Authors

This shows that it was more effective to use CountVectorizer as a classification method, since it successfully classified 89.3% of the news correctly classified as false 1827 and as true 2079.

The model used, however, proves not to have the same effectiveness as others, since in (Chiu, Gokcen, Wang, & Yan, nd) for example they use a model based on Support Vector Machines and achieve an average success rate of 95% and in (Chaudhry, Baker, & Thun-Hohenstein, nd) make an approximation using deep neural networks and achieve a certainty of up to 97.3% in the classification process.

7 Conclusions

The news has a large number of characteristics that can be evaluated and to reach a certainty greater than 95% is necessary to consider them to address an objective such as news classification is a complex task even using a standard procedure of text classification.

References

1. Mauri, M., Jonathan, G., Tommaso, V., Michele, M.: A field guide to fake news (2017)
2. Mele, N., Lazer, D., Baum, M., Grinberg, N., Friedland, L., Joseph, K., Hobbs, W., Mattsson, C.: Combating fake news: an agenda for research and action, May 2017
3. Gerazov, B., Conceicao, R.C.: Deep learning for tumour classification in homogeneous breast tissue in medical microwave imaging. In: IEEE EUROCON (2017)
4. Affonso, C., Rossi, A.L.D., Vieira, F.H.A., de Carvalho, A.C.P.D.L.F.: Deep learning for biological image classification. Expert Syst. Appl. **85**, 114122 (2017). https://doi.org/10.1016/j.eswa.2017.05.039
5. Yudin, D., Zeno, B.: Event recognition on images by fine-tuning of deep neural networks (2018). https://doi.org/10.1007/978-3319-68321-8_49

6. Shu, K., Wang, S., Sliva, A., Tang, J., Liu, H.: Fake news detection on social media: a data mining perspective. ACM SIGKDD Explor. Newslett. **19** (2017)
7. Shao, C., Ciampaglia, G.L., Varol, O., Flammini, A., Menczer, F.: The spread of fake news by social bots (2017). arXiv:1707.07592
8. Bajaj, S.: The Pope Has a New Baby! Fake News Detection Using Deep Learning (n.d.). https://web.stanford.edu/class/cs224n/reports/2710385.pdf