

Fake News Detection on Twitter using NLP with Python

Theodoros Alexandros Chandrinou

Theodoros Zampatis

Eleni Koutrouli

Maria Anastasia Katikaridi

AphroditeTsalgaidou

Department of Informatics and Telecommunications
National and Kapodistrian University of Athens, Greece
 {tchandrin, tzampatis}@cosmote.gr, {ekou, mkatikaridi, atsalga}@di.uoa.gr

Abstract

The advancement of social networks has facilitated the sharing and spread of news among people all over the world. With the growth of these networks and of the volume of news shared daily, the phenomena of fake news have become stronger and widely spread. Over the past few years, big social networks like Twitter admit that fake and duplicate accounts, fake news and fake likes exists in their networks. This stems from the fact that the social network account owners have the ability to distribute false information, to support or attack an idea or a product, to promote or demote an election candidate, as well as to influence real network users in their decision making. Therefore misinformation detection in enhancing public trust and society stability becomes of critical importance. Along these lines, detection of misinformation is still a challenging problem for the Natural Language Processing community. In our work, we have utilized natural language processing and supervised machine learning in order to detect fake tweets using Python. In this paper, we will introduce our work and describe its current status.

CCS Concepts

• Machine Learning • Programming in Python • Natural Language Processing

Keywords

Natural Language Processing, Fake news, Twitter

1. Introduction

In our work, we explore the detection of fake tweets in Twitter using Natural Language Processing (NLP) with Python. We have studied a variety of approaches on the subject, e.g. [1]. This inspired us to combine these approaches with the goal to find out which combinations work better. Therefore, we have developed a software tool, which checks the success ratio of four different systems for fake news detection using four different datasets, resulting in a total of sixteen) ratios, one for each combination.

2. Experiments and Results

For the purpose outlined above, we utilized the [PHEME](#) Dataset [2], which includes thousands of real pre-labeled tweets exported

via [TweeterAPI](#) [3]. We created a python program, which parses the aforementioned dataset and saves all the tweets from it into .tsv files. We have four (4) different datasets differentiated by the following: • Should we accept the existence of duplicate tweets: Some of the same tweets may have been shared by various users/profiles. • Should we accept a third label for the truthfulness of the tweets besides “true” or “fake”, which is the “undefined” label. After choosing a .tsv file, sentiment analysis is performed in every tweet using **Sentiment Intensity Analyzer** algorithm [4]. Next, we process the results of this analysis and decide whether a tweet should be labeled as positive or negative.

Then, we use a pipeline in which the following steps are handled:

- Tokenization and Lemmatization into a bag of words using **NLTK**
- Vectorization using **Count Vectorizer** or **TF-IDF Vectorizer** using **Scikit-Learn**
- Classification using **Support Vector Machine** or **Multinomial Naive Bayes** using **Scikit-Learn**.

The combination of the linguistic and sentimental processing outputs different results based on the selection of the file, the vectorizer and the classifier. The success ratio of correctly labeled test data is ranged between 54,6% and 99,8%.

Our goal is to develop an effective software tool with a well-structured code, which will assist in highlighting the right combinations of well-known and effective algorithms. Results until now show that datasets with duplicates produce a great training set, therefore the success ratio is higher than 97%. After all, this is a more realistic scenario since a single tweet could be posted by many users.

REFERENCES

- [1] . Qi Su, Mingyu Wan, Xiaoqian Liu, Chu-Ren Huang, Motivations, Methods and Metrics of Misinformation Detection: An NLP Perspective, in Natural Language Processing Research, 2020
- [2] Kochkina, Elena; Liakata, Maria; Zubiaga, Arkaitz (2018): PHEME dataset for Rumour Detection and Veracity Classification. figshare. Dataset. <https://doi.org/10.6084/m9.figshare.6392078.v1> .
- [3] [Twitter API](https://developer.twitter.com/en/docs/twitter-api), <https://developer.twitter.com/en/docs/twitter-api>, accessed on 3/6/2021 .
- [4] Aditya Beri, Sentiment Analysis Using Vader: Interpretation and classification of emotions, <https://www.coursehero.com/file/91544252/doc8docx/>, accessed on 3/6/2021