

# FAKE NEWS PREDICTION USING TF-IDF VECTORIZER

<sup>1</sup>Raj.Shah, <sup>2</sup>Sanket.Magodia, <sup>3</sup>Yashpal.Zala, <sup>4</sup>Kanchan Dabre

<sup>1</sup>Bachelor of Engineering, <sup>2</sup>Bachelor of Engineering, <sup>3</sup>Bachelor of Engineering, <sup>4</sup>Professor  
Dept. of Computer Engineering  
Universal College of Engineering Mumbai, India, .

**Abstract:** The technology has improved a lot in the field of Mass media, the market trends and advancement in techniques are growing rapidly nowadays. But this information is not always accurate enough, since a more number of searching steps are required to obtain the REAL NEWS results from the Internet sources. This system “NewMors” overcomes this problem and provides an application that detects the hashtags from different Social Media Platforms and using Geotags and Algorithms of NLP, NewMors detects if the source of information is accurate enough to believe if so, the information is sent to NEWS section or else it’s Kept in RUMORS section. The focus is on developing an algorithm in a more intellectual way, that it can even recognize not so well grammatically defined sentences, misspelled words, incomplete phrases, etc. The responses are generated using classification algorithms and produce non textual responses so that it can be easily perceived by the users. This system also uses prediction algorithms to predict the future data like who will be next president of the USA from data obtained from different platforms, who will win the cricket match. Also using data our platform will provide some of the analysis and sentiment Reports.

**Index Terms - Natural Language Processing, Machine Learning, Knowledge Base, Prediction Algorithms.**

## I.INTRODUCTION

The main objective is to detect the fake news, which is a classic text classification problem with a straightforward proposition. It is needed to build a model that can differentiate between “Real” news and “Fake” news. These days’ fake news is creating different issues from sarcastic articles to fabricated news and plan government propaganda in some outlets. Fake news and lack of trust in the media are growing problems with huge ramifications in our society. Obviously, a purposely misleading story is “fake news” but lately blathering social media’s discourse is changing its definition. Some of them now use the term to dismiss the facts counter to their preferred viewpoints.

## II.SURVEY STUDY

The author of [1] in this paper has reviewed some papers on fake news detection they are as follows:-

The authors of [2] in their paper proposed the discovery of fraud using data labeled 'LIAR' with explicit and improved performance in finding false posts / news. The authors object to the use of corpus in state planning, opinion mining, rumor detection, and NLP political research.

The authors of [3] Introducing the Need to Find Deception. They used the ML method by combining news content and social media methods. The authors claim that performance is relative to what is described in the textbooks. Authors Use it via Facebook messenger Chabot. Three different data from the Italian news of Facebook were used. Both content-based methods with social and content features use the Boolean crowd to get expertise where it is used. The following methods when used are: 1. Supported content 2. Regular reversal of public signals. 3. Harmonic Boolean label crowdsourcing on social signals.

The authors at [4] have seen nearly 14 million messages re-typed nearly 400 billion times on Twitter during and after the U.S. Presidential campaign. And a selection of bots in 2016. Methods of classifying posts distributed by bots were defined.

The authors in [5] described Tabloidization in the form of Clicking. They described Click to spread the word as a way to quickly spread rumors and false information online. The authors discussed possible ways to automatically get clickbait as a deceptive method. Content indicators include lexical and semantic level analysis where used by authors.

Authors of [6] have a probabilistic, a binary classifier logistic regression. Before presenting the ROC curve (Receiver Operating Characteristic curve), the concept of confusion matrix must be understood. When we make a binary prediction, there can be 4 types of outcomes:

- We predict FAKE while we should have the class is actually FAKE: this is called a True Negative.
- We predict FAKE while we should have the class is actually TRUE this is called a False Negative.
- We predict TRUE while we should have the class is actually FAKE this is called a False Positive.
- We predict TRUE while we should have the class is actually TRUE: this is called a True Positive.

### III. PROBLEM STATEMENT AND OBJECTIVE

Users finding news on Social media are not sure enough to believe on what they are reading as some people writes something using some popular hashtags like #news, #politics etc. to defame someone or spread some rumor to increase one’s sales of a product. Predict if the so called “news” on social media is actually a news or Rumor

The importance of disinformation within American political discourse was the subject of weighty attention, particularly following the American presidential election. The term ‘fake news’ became common parlance for the issue, particularly to describe factually incorrect and misleading articles published mostly for the purpose of making money through page views. In this paper, it is seeked to produce a model that can accurately predict the likelihood that a given article is fake news.

Facebook has been at the epicenter of much critique following media attention. They have already implemented a feature to flag fake news on the site when a user sees it; they have also said publicly they are working on distinguishing these articles in an automated way. Certainly, it is not an easy task. A given algorithm must be politically unbiased – since fake news exists on both ends of the spectrum and also give equal balance to legitimate news sources on either end of the spectrum. In addition, the question of legitimacy is a difficult one. However, in order to solve this problem, it is necessary to have an understanding of what Fake News is. Later, it is needed to look into how the techniques in the fields of machine learning, natural language processing help us to detect fake news.

### IV. PROPOSED SYSTEM

#### 4.1 Analysis/Framework/ Algorithm

In this system the model is built based on the count vectorizer or a tf idf matrix (i.e. word tallies relative to how often they are used in other articles in your dataset) can help. Since this problem is a kind of text classification, implementing a Naive Bayes classifier will be best as this is standard for text-based processing.

#### 4.2 Details of Hardware & Software

- Mobile phones (Android/iOS)
- Desktop

#### 4.3 Design details

The actual goal is in developing a model which is the text transformation (count vectorizer vs tfidf vectorizer) and choosing which type of text to use (headlines vs full text). Now the next step is to extract the most optimal features for count vectorizer or tf idf-vectorizer, this is done by using a n- number of the most used words, and/or phrases, lower casing or not, mainly removing the stop words which are common words such as “the”, “when”, and “there” and only using those words that appear at least a given number of times in a given text dataset.

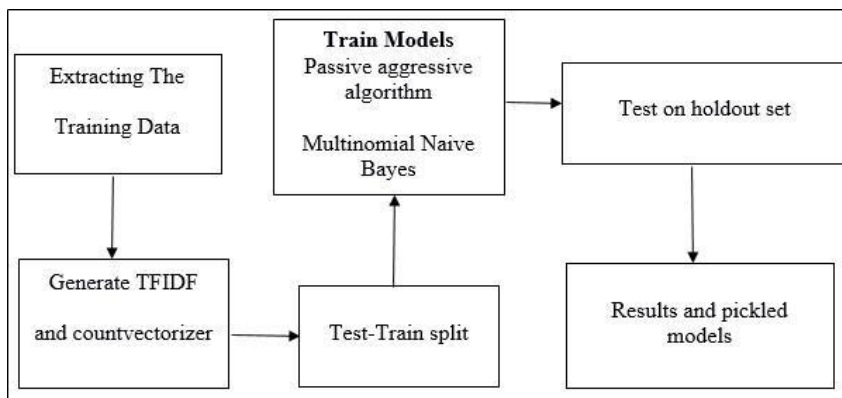


Fig. 1. Flow-Chart

### V. UML DIAGRAM

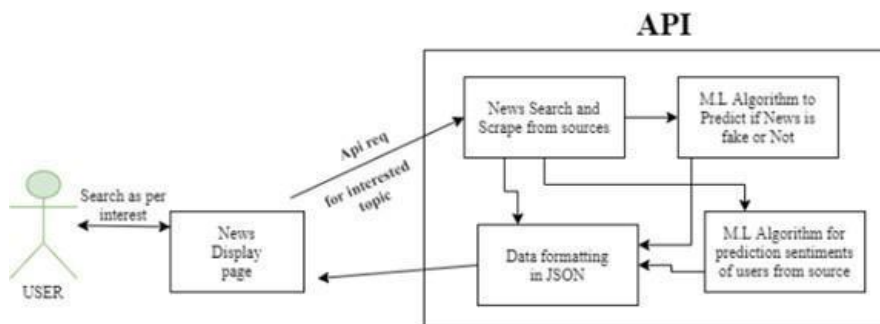


Fig. 2. The above diagram shows us the use case of the project, as in how each system would coordinate with one another during final deployment.

### VI. TF-IDF VECTORIZER

TF (Term Frequency): The number of times a word appears in a document is its Term Frequency. A higher value means a term appears more often than others, and so, the document is a good match when the term is part of the search terms. IDF (Inverse Document Frequency): Words that occur many times in a document, but also occur many times in many others, may be irrelevant. IDF is a measure of how significant a term is in the entire corpus. The TfidfVectorizer converts a collection of raw documents into a matrix of TF-IDF features.

## VII. COLLECTING DATA

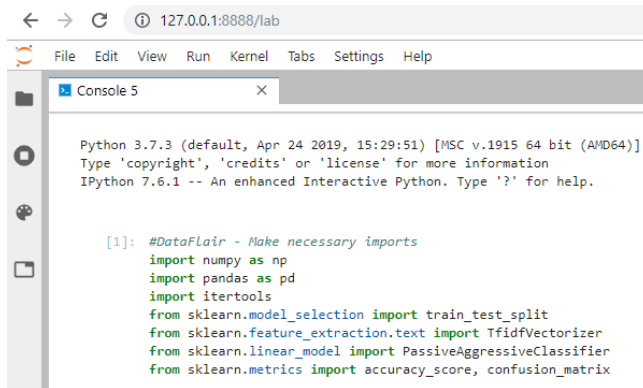
So, there must be two parts to the data-acquisition process, “fake news” and “real news”. Collecting the fake news was easy as Kaggle released a fake news dataset consisting of 13,000 articles published during the 2016 election cycle. Now the later part is very difficult. That is to get the real news for the fake news dataset. It requires huge work around many Sites because it was the only way to do web scraping thousands of articles from numerous websites. With the help of web scraping a total of 5279 articles, a real news dataset was generated, mostly from media organizations (New York Times, WSJ, Bloomberg, NPR, and the Guardian) which were published around 2015 – 2016.

Datasets The lack of manually labeled fake news datasets is certainly a bottleneck for advancing computationally intensive, text-based models that cover a wide array of topics. The dataset for the fake news challenge does not suit our purpose due to the fact that it contains the ground truth regarding the relationships between texts but not whether or not those texts are actually true or false statements. For our purpose, we need a set of news articles that is directly classified into categories of news types (i.e. real vs. fake or real vs. parody vs. clickbait vs. propaganda). For more simple and common NLP classification tasks, such as sentiment analysis, there is an abundance of labeled data from a variety of sources including Twitter, Amazon Reviews, and IMDb Reviews. Unfortunately, the same is not true for finding labeled articles of fake and real news. This presents a challenge to researchers and data scientists who want to explore the topic by implementing supervised machine learning techniques. I have researched the available datasets for sentence-level classification and ways to combine datasets to create full sets with positive and negative examples for document-level classification. Sentence Level produced a new benchmark dataset for fake news detection that includes 12,800 manually labeled short statements on a variety of topics. These statements come from politifact.com, which provides heavy analysis of and links to the source 14 documents for each of the statements. The labels for this data are not true and false but rather reflect the “sliding scale” of false news and have 6 intervals of labels. These labels, in order of ascending truthfulness, include ‘pants-fire’, ‘false’, ‘barely true’, ‘half-true’, ‘mostly-true’, and true. The creators of this database ran baselines such as Logistic Regression, Support Vector Machines, LSTM, CNN and an augmented CNN that used metadata. They reached 27% accuracy on this multiclass classification task with CNN that involved metadata such as speaker and party related to the text. Document Level There exists no dataset of similar quality to the Liar Dataset for document level classification of fake news.

Genes trains a model on a specific subset of both the Kaggle dataset and the data from NYT and the Guardian. In his experiment, the topics involved in training and testing are restricted to Sports, Politics, Business and World news. We have collected data in a manner similar to that of Genes, but more cautious in that we control for more bias in the sources and topics. Because the goal of our project was to find patterns in the language that are indicative of real or fake news, having source bias would be detrimental to our purpose. Including any source bias in our dataset, i.e. patterns that are specific to NYT, The Guardian, or any of the fake news websites, would allow the model to learn to associate sources with real/fake news labels. Learning to classify sources as fake or real news is an easy problem, but learning to classify specific types of language and language patterns as fake or real news is not. The dataset also contained a decent amount of repetitive data and incomplete data, we removed any non-unique samples and also samples that appeared incomplete (i.e. lacked a source). This left us with approximately 12,000 samples of fake news. Since the Kaggle dataset does not contain positive examples, i.e. examples of real news, it is necessary to augment the dataset with such in order to either compare or perform supervised learning. Real news samples as suggested by, an acceptable approach would be to use the APIs from reliable sources like Twitter, including both text and images that are found in the document. We found that extra effort was required to ensure that we removed any source-specific patterns so that the model would not simply learn to identify how an article from Twitter is written or how an article from The other media platform is written. Instead, we wanted our model to learn more meaningful language patterns that are similar to real news reporting, regardless of the source.

## VIII. IMPLEMENTATION METHODOLOGY AND RESULTS

1. Make necessary imports as seen in Fig 3:



```

Python 3.7.3 (default, Apr 24 2019, 15:29:51) [MSC v.1915 64 bit (AMD64)]
Type 'copyright', 'credits' or 'license' for more information
IPython 7.6.1 -- An enhanced Interactive Python. Type '?' for help.

[1]: #DataFlair - Make necessary imports
import numpy as np
import pandas as pd
import itertools
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.linear_model import PassiveAggressiveClassifier
from sklearn.metrics import accuracy_score, confusion_matrix

```

Fig. 3.

2. Now, let's read the data into a DataFrame, and get the shape of the data and the first 5 records .

```

[2]: #Read the data
df=pd.read_csv('D:\DataFlair\news.csv')

#Get shape and head
df.shape
df.head()

```

```

[2]:
  Unnamed: 0  title  text  label
0      8476  You Can Smell Hillary's Fear  Daniel Greenfield, a Shillman Journalism Fello...  FAKE
1     10294  Watch The Exact Moment Paul Ryan Committed Pol...  Google Pinterest Digg LinkedIn Reddit Stumbleu...  FAKE
2      3608  Kerry to go to Paris in gesture of sympathy  U.S. Secretary of State John F. Kerry said Mon...  REAL
3     10142  Bernie supporters on Twitter erupt in anger ag...  — Kaydee King (@KaydeeKing) November 9, 2016 T...  FAKE
4        875  The Battle of New York: Why This Primary Matters  It's primary day in New York and front-runners...  REAL

```

Fig. 4.

3. And get the labels from the DataFrame.

```

[3]: #DataFlair - Get the Labels
labels=df.label
labels.head()

```

```

[3]: 0    FAKE
     1    FAKE
     2    REAL
     3    FAKE
     4    REAL
     Name: label, dtype: object

```

Fig. 5.

- Split the dataset into training and testing sets.

```
[4]: #DataFlair - Split the dataset
      x_train,x_test,y_train,y_test=train_test_split(df['text'], labels, test_size=0.2, random_state=7)
```

Fig. 6

- Let's initialize a TfidfVectorizer with stop words from the English language and a maximum document frequency of 0.7 (terms with a higher document frequency will be discarded). Stop words are the most common words in a language that are to be filtered out before processing the natural language data. And a TfidfVectorizer turns a collection of raw documents into a matrix of TF-IDF features. Now, fit and transform the vectorizer on the train set, and transform the vectorizer on the test set.

```
[5]: #DataFlair - Initialize a TfidfVectorizer
      tfidf_vectorizer=TfidfVectorizer(stop_words='english', max_df=0.7)

      #DataFlair - Fit and transform train set, transform test set
      tfidf_train=tfidf_vectorizer.fit_transform(x_train)
      tfidf_test=tfidf_vectorizer.transform(x_test)
```

Fig. 7.

- Next, we'll initialize a PassiveAggressiveClassifier. This is. We'll fit this on tfidf\_train and y\_train. Then, we'll predict on the test set from the TfidfVectorizer and calculate the accuracy with accuracy\_score() from sklearn.metrics.

```
[6]: #DataFlair - Initialize a PassiveAggressiveClassifier
      pac=PassiveAggressiveClassifier(max_iter=50)
      pac.fit(tfidf_train,y_train)

      #DataFlair - Predict on the test set and calculate accuracy
      y_pred=pac.predict(tfidf_test)
      score=accuracy_score(y_test,y_pred)
      print(f'Accuracy: {round(score*100,2)}%')

      Accuracy: 92.82%
```

Fig. 8

## IX. RESULTS

We got an accuracy of 92.82% with this model. Finally, let's print out a confusion matrix to gain insight into the number of false and true negatives and positives.

```
[6]: #DataFlair - Initialize a PassiveAggressiveClassifier
      pac=PassiveAggressiveClassifier(max_iter=50)
      pac.fit(tfidf_train,y_train)

      #DataFlair - Predict on the test set and calculate accuracy
      y_pred=pac.predict(tfidf_test)
      score=accuracy_score(y_test,y_pred)
      print(f'Accuracy: {round(score*100,2)}%')

      Accuracy: 92.82%
```

```
[7]: #DataFlair - Build confusion matrix
      confusion_matrix(y_test,y_pred, labels=['FAKE','REAL'])

[7]: array([[589,  49],
           [ 42, 587]], dtype=int64)
```

---

```
[ ]:
```

## X. CONCLUSION

**Contributions** The main contribution of this project is support for the idea that machine learning could be useful in a novel way for the task of classifying fake news. Our findings show that after much pre-processing of a relatively small dataset, a simple CNN is able to pick up on a diverse set of potentially subtle language patterns that a human may (or may not) be able to detect. Many of these language patterns are intuitively useful in a human manner of classifying fake news. Some such intuitive patterns that our model has found to indicate fake news include generalizations, colloquialisms and exaggerations. Likewise, our model looks for indefinite or inconclusive words, referential words, and evidence words as patterns that characterize real news. Even if a human could detect these patterns, they are not able to store as much information as a CNN model, and therefore, may not understand the complex relationships between the detection of these patterns and the decision for classification. As such, this seems to be a really good start on a tool that would be useful to augment human's ability to detect Fake News. Other contributions of this project include the creation of a dataset for the task and the creation of an application that aids in the visualization and understanding of the neural nets classification of a given body text. This application could be a tool for humans trying to classify fake news, to get indications of which words might cue them into the correct classification.

## XI. ACKNOWLEDGEMENT

We take this opportunity to express our deep sense of gratitude to our project guide and project coordinator, Mrs. Kanchan Dabre, for her continuous guidance and encouragement throughout the duration of our project work. It is because of her experience and wonderful knowledge; we can fulfill the requirement of completing the project within the stipulated time. We would also like to thank Dr. Jitendra Saturwar, head of the computer engineering department for his encouragement, whole-hearted cooperation and support. We would also like to thank our Principal Dr. J. B. Patil and the management of Universal College of Engineering, Vasai, Mumbai for providing us all the facilities and the work friendly environment. We acknowledge with thanks, the assistance provided by departmental staff, library and lab attendants

## REFERENCES

- [1] Manzoor, S. I., Singla, J., & Nikita. (2019). Fake News Detection Using Machine Learning approaches: A systematic Review. 2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI). doi:10.1109/icoei.2019.8862770
- [2] Wang, W. Y. (2017). " liar, liar pants on fire": A new benchmark dataset for fake news detection. arXiv preprint arXiv:1705.00648.
- [3] Della Vedova, M. L., Tacchini, E., Moret, S., Ballarin, G., DiPierro, M., & de Alfaro, L. (2018, May). Automatic Online Fake News Detection Combining Content and Social Signals. In 2018 22nd Conference of Open Innovations Association (FRUCT) (pp. 272-279). IEEE.
- [4] Shao, C., Ciampaglia, G. L., Varol, O., Flammini, A., & Menczer, F. (2017). The spread of fake news by social bots. arXiv preprint arXiv:1707.07592, 96-104.
- [5] Chen, Y., Conroy, N. J., & Rubin, V. L. (2015, November). Misleading online content: Recognizing clickbait as false news. In Proceedings of the 2015 ACM on Workshop on Multimodal Deception Detection (pp. 15-19). ACM.
- [6] Jain, A., & Kasbe, A. (2018). Fake News Detection. 2018 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS). doi:10.1109/sceecs.2018.8546944.