International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Fake News Detection using NLP

Vattumilli Komal Venugopal, Prathyush Pulla, Khandapu Snehith

Vellore Institute of Technology,India

ABSTRACT:

Preading of Fake-News as if it is the original has become an evolving problem now. In an attempt to give a form of originality to the Fake-News, various social media are playing an active and considerable role. Later the same news becomes authentic when it finds its way through traditional news sources like Television and Radio. Social media platforms make the news spicy and exaggerate it by giving unwanted emphasis to it. They even go to the extent of adding linguistic tools in the process of magnification. As a step to curb the same novel Fake-News detectors like Text blob, Natural Language and SciPy Toolkits are developed. The rate of effectiveness of the mentioned software is estimated and assessed as 63.333%. This technique is used to detect the Fake-News. This paper throws limelight on technical analysis, technical linguistics works, and classifier performance, and the results of the same are also included. The paper comes to the conclusion based on how the current implemented system will prove to be a solution and effective influence mining system.

Keywords: Scipy, Natural Language, Text blob, Fake News

1.Introduction

Fake news can also alter the political landscape of a country. In the process of addressing this growing [problem researchers and experts have come out with a possible solution like Natural language Processing (NLP). The usage of words can be analyzed judiciously and manipulation can be restricted to promote operative accuracy. As a part of the process, the model check will be done by the professional journalist to encourage and enhance the accuracy of originality. Every coin has two sides when there is an advantage surely there will be a setback too. Two major setbacks of this model checks are Firstly detection is possible when the Fake-News is under-written. If the news is unrelated to the given headline and is biased it doesn't reach the next stage. This is called CLICKBAIT. Secondly, the excessive usage of inflammatory phrases and provocative words in the article by giving the least importance to the original content and focusing more on the evil effects of the news. In such cases, the argumentative language and content will be attacked by researchers with a tool called state-of-the-art or FAKE BOX in simple language. Three classes of attacks have been formulated in this procedure. FACT DISTORTION, SUBJECT-OBJECT EXCHANGE, and CAUSE CONFONDING. Based on these attacks FAKE BOX comes out with low accuracy.

The job of detectors is getting tough as they are facing strict challenges. So, religious news or political news can be rejected as a result. Open platforms like Twitter in the USA and TaoTio in China are abundantly and excessively used by the people. Segregating the false news from the original one is still manual work. To minimize the adverse effect of this problem, some form of fact-based knowledge must be implemented along with NLP-based models. A crowd-sourced knowledge graph solution is also adopted to know the reliability of the news.

2.Motivation

The main motto of our project 'FAKE-NEWS DETECTION USING NLP' is mainly to identify the articles in the news (or) social media and trace out if they are real or fake. The people are diverted in the wrong way and are made fools by the forwarded Fake-News in WhatsApp and other media apps. So, to stop sharing the Fake-News we need a precise model for the detection of the Fake-News. We have used different classifiers and different models for this prediction. The accuracy of detection in the current models is less. So, it motivated us to identify the better suitable model for the detection of real news. We have used the Count-Vectorizer to find the repetitive words and identify only the keywords that are repeated among different sentences. We used a data set containing the news and classified it as fake or real and trained and tested using different models and obtained better accuracy from the passive-aggressive classifier.

3.Algorithms

a. Multinomial Naïve Bayes

The Multinomial Naïve Bayes, which is also called the MultinomialNB tool is used for distributed data that is multinomial. It comes from one of the two classic naïve Bayes variants which are used in Text Classification.

Here the data is typically represented as word vector counts. The parameters are alpha float, Additive (Laplace/Lidstone) smoothing parameter (0 for no smoothing). fit_priorbool, the default value of it is true and it identifies whether to learn class probabilities or not.class_priorarra, the default value of it is None. The distribution is limited by the vectors $\theta_y = (\theta_{y1}, \ldots, \theta_{yn})$ for each of the class Y, where the value of N represents the number of features and θ_{yi} represents the probability P(xi | y) of feature I appearing in a sample belonging to class y.

The relative frequency counting is obtained by the equation:

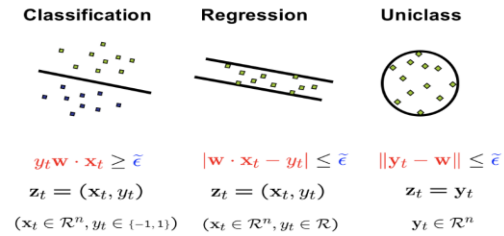
$$\hat{\theta}_{yi} = \frac{N_{yi} + \alpha}{N_y + \alpha n}$$

where Nyi represents the number of time features and i is the variable that appears in a sample of class y in the training set.

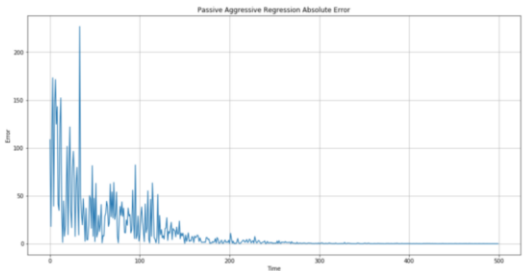
The smoothing prevents zero probabilities in further computations. Setting alpha=1 is called Laplace smoothing, while alpha<1 is called Lidstone smoothing.

b. Passive Aggressive Classifier

These are a family of algorithms that are used for large-scale learning. They work almost the same as one's Perceptron and they don't require parameters of learning rate. But it includes a regularization parameter C. Passive-aggressive Classifier can be implemented with loss='hinge' (or) loss = 'square_hinge' for primary classification. Passive-aggressive Regressor can be implemented with loss='epsilon_insensitive' or loss = 'squared_epsilon_insensitive' used for primary regression.



These algorithms which are discussed above are working based on the Hinge Loss function.



The quality of the regression particularly the length of the transient period when the error is at peaks can be minimized or controlled by picking better C and epsilon values.

c. Hashing Vectorizer

The Hashing Vectorizer collects text documents and converts them into a matrix of occurrences of tokens. At last, it forms a collection of all these documents in the format of text into a ScipPy. The matrix which has all the token occurrence counts is called a Sparse matrix. The Vectorizer is mainly based upon this as it has several advantages It will do the process by taking less memory and we don't need to store a local dictionary while implementing this. It is very fast and can be used in streaming as there is no chance of state computed during the process of fitting.

There are also some disadvantages. The inverse transform cannot be computed while using this method. There is a possibility of collisions when different tokens are mapped to the same index, few parameters are

input string {'filename', 'file', 'content'}, default='content'

encoding string, default='utf-8'

decode_error{'strict', 'ignore', 'replace'}, default='strict'

strip_accents{'ascii', 'unicode'}, default=None

lowercasebool, default=True

preprocessorcallable, default=None

tokenizercallable, default=None

token_patternstring

a=min_n

b=max_n

ngram_range : tuple (a,b), default=(1, 1)

analyser : string, {'word', 'char', 'char_wb'} or callable, default='word'

n_features : int, default=(2 ** 20)

binary : bool, default=False.

Norm : {'l1', 'l2'}, default='l2'

alternate_sign : bool, default=True

dtype : type, default=np.float64

4.Literature survey

[11] The spread of fake medical websites and ways to combat the same has been discussed and highlighted in our present work of ours. The techniques followed can be listed as follows: RTL – Recursive Trust Labelling consists of three vital stages.

i)Linguistic features identification

ii) graph-based classifier and finally

iii) recursive labeling mechanism.

These three elements have been used. The growth of the recursive marking component takes place based on the training data set, which is done by choosing extra test samples for every case. Based onthe fundamental classifier's predictions, the class labels and occasions the most grounded prediction scores have been chosen. Content-based methods: Through this method of computerized content generation, unreliable and unrealistic fake content can be traced by NLP techniques. This method also aids us in many other ways. Firstly, the selection or picking up extra test situations in the middle of every cycle is done and ultimately results in the growth of the recursive naming component. Keeping in view the substance and chart classifier is chosen and is included with the class names depending on the basic classifier's predictions. As a result, we find that Algorithms are not highly subjected to bigger training datasets. But are remarkable and outstandingly impressive and attractive. Features used in this procedure are tags related to HTML, text of URL, image hashes, and link information. After availing of all the above-mentioned features, the accuracy is 68%, which is considered reliable.

Neural Network, SVM, Naïve Bayes: It consists of features like n-grams, and anchor text. In this case, the precision is highly progressive as well as recommendable as it is 96%. Every research and theory is formulated based on certain assumptions. Similarly, here we assume that the good News link to Good News, and the bad News link to other bad one Techniques that are based on Graph- are designed to stand with link forms. Rumor Guage is pragmatic in the prediction of the Veracity and reliability of rumors on Twitter. The model applied is Hidden MaekovModel (HMMMs). Data-sets extraction is Twitter's historical API. The main idea behind this research is that the veracity of rumors is predictive. However, weak they are and after trusted verification too.

[12] The paper identifies the reliability and aptness of an article in news and investigates the possible sub-tasks to combat news with supporting proof. When an article is given the main aim is to decide the appropriate body and case of it to make it more authentic. The original thought should get polluted and an efficient answer for this issue is given by adopting neural, factual, and outer highlight methods. Neural inserting is done from the repetitive model, factual highlights obtain from the weighted n – gram pack of words display. Highlight designing will assist in deploying the handmade outer highlights. Finally, it is determined and found that the features are combined and thus resulting in the classification of the headline-body pair as 3 categories like dis-agree, agree, or unrelated. With this model, they conclude that this model outperforms different fake news which is to be tested.

Authors and Year (Reference)	**Title (Study)**	**Concept / Theoretical model/ Framework**	**Methodology used/ Implementation**	**Data-set details/ Analysis**	**Relevant Finding**	**Limitations/ Future Research/ Gaps identified**
Conroy, N. J., Rubin, V. L., and Chen, Y. (2015).	"Automatic deception detection: Methods for finding Fake-News".[3]	It contains the proceedings of the Association for Information Science and Technology. The accuracy and reliability of the news is considered to get preferred and ranked.	It is implemented using the linguistic cue approach and network analysis approach.	Structured data set such as text.	Two approaches have similarities like adopting various techniques of machine learning for the classifier's training that is best suited.	It is the combination of linguistic cue and machine learning based on network based behavioral data.
Wu, Liang, & Huan Liu. (2018)	"Tracing Fake-News Footprints: Characterizing Social Media Messages by How They Propagate. "[13]	Limelight on messages in the social network and showcase the same.	TRACEMINER is introduced and vividly used by social network structure users.	Structural data set such as text.	Adoption of Tracemineer expands the accuracy rate and deciphers the real world data-set more accurate than the traditional approaches.	To make the accuracy more reliable and promote the correctness the optimization methods are avail resulting in the performance of the physical world social information technology.
Shu, Kai, et al. (2017)	"Fake news detection on social media: A data mining perspective." [14]	Survey of counterfeit news distinction using web-based networking media.	Advantages of making use of the social media for news information and quality in comparison with the traditional way is made clear.	-	Fake news and their features, characteristics are made crystal clear.	

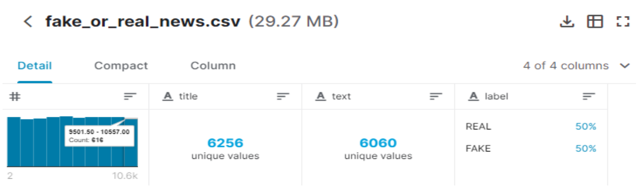
Bhatt, Gaurav, et al. (2018)	"Combining Neural, Statistical & External Features for Fake News Stance Identification." [12]	To enhance and promote accuracy fusion of original thought with factual, neural & outer highlights evolves as an answer to this issue.	Adopting methods like weighted n - gram pack of -words display.	Structural data set such as the text.	Differentiating the first-hand information as agree, disagree, discusser unreal.	The given model excels in tracing the Fake-News.
Abbasi, Ahmed, Fatemeh Zahedi, & Siddharth Kaza. (2012)	"Looping Trust Tag" [11]	Restriction of the spread of fake medical websites and curbing their sources too.	Hidden Markov Models	Twitter historical API.	Assessing the aptness of rumors on Twitter.	Before getting finalized the check is done on the grounds of reliability and veracity of the rumors.
Buntain, Cody, and Jennifer Golbeck(2017)	"Recognition of spam news in the viral Twitter thread." [9]	Spam new data is identified on Twitter data by using various modes of tests.	Machine learning, Deep learning models are standing as the solution to the current problem.	Twitter data-sets – CREDBANK and PHEME	-	Fake news challenges are successfully proposed by the recommended model.

5.Methodology

Our implementation requires python3 using Jupyter notebook and PHP 7 with a windows operating system having a wamp server installed. There are no Hardware requirements.

Software Requirements: Wamp Server, Windows OS, PHP, Python3, and Jupyter Notebook

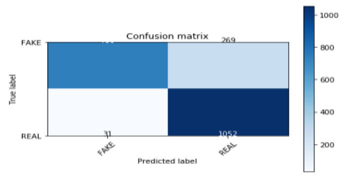
Dataset: We used the data-set 'real_or_fake.csv which contains the different news and is classified as fake or real in different columns. The data-set is obtained from the Kaggle website. Data-set we took contains 6265 unique values among which 50% are real and the other 50% are classified as fake. [11]



Dataset Structure (Kaggle)

The Naive Bayes multinomial classifier is ideal for discrete feature classification. It classifies the data based on the word counts from the data. The Multinomial distribution needs an integer number of features. Fractional counts like Term frequency identifiers like TF-IDF is applicable in practice. Pandas, sklearn, and numpy are the modules which are used in our project. The data-set is split into 0.33 test size using test_train_split. We have trained the model using the Naïve Bayes and the Passive Aggressive Classifier. The Hashing Vectorizer is done to take unique data for training. So that the repetitive data is not trained and it will save the time and improve the accuracy of the model.

a) Training with the Count-Vectorizer using the Naïve Bayes classifier is done by fitting data and transforming. We have found the accuracy of 0.893 and the confusion matrix without normalization is obtained as [[865 143] [80 1003]].



Confusion matrix for Multi naïve Bayes (Count Vectorizer)

ls = 0

for i=0 and i<1 jump=0.1

nbc = MultinomialNB(alpha=alpha)

fit(tfidf_train, y_train)

p = peredict_tfidf_test

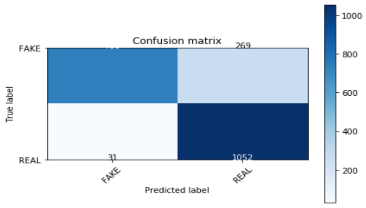
s = accuracy_score(y_test, pred)

if s >ls thenclassifier = nbc

else print("Alpha : {:.2f} Score: {:.5f}".format(alpha, score))

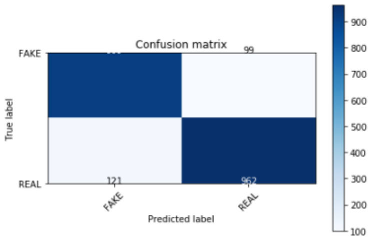
b) The above algorithm is the implementation of Naïve Bayes using the Tfidf-Vectorizer that gave an accuracy of 0.857.

The confusion matrix obtained is [[739 269] [31 1052]]



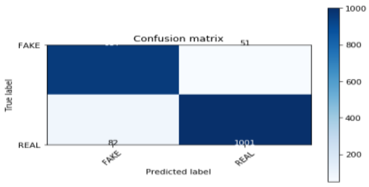
Confusion matrix for Multi Naïve Bayes Classifier (Tfidf- Vectorizer)

c) When the Passive aggressive classifier is implemented using the Count-Vectorizer that gave and accuracy of 0.895.
The confusion matrix obtained is [[909 99] [121 962]]



Confusion matrix for Passive Aggressive Classifier (Count Vectorizer)

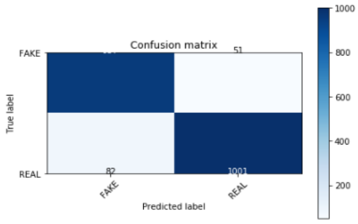
d) The accuracy of 0.936 is obtained using the passive-aggressive classifier and with Tfidf-Vectorizer which is considered as the best among all the models that we have implemented. So, we used this model to test the new data that is given in the portal that we have developed. The confusion matrix is [[957 51] [82 1001]]



Confusion matrix for Passive Aggressive Classifier (tfidf Vectorizer)

6.Result and Discussion

We have implemented the Multi Naïve Bayes and Passive Aggressive Classifiers and in each of the classifiers, we have implemented using the Count Vectorizer and the term Frequency-Vectorizer (Tfidf-Vectorizer). We got the accuracy of 0.857, 0.893 for tfidf and count vectorizers while implementing using the Multi Naïve Bayes. Also, while implementing using the Passive Aggressive classifier we got the accuracies of 0.936 and 0.895 for the Tfidf Vectorizer and Count Vectorizers. On comparing the accuracies of both the classifiers it is found that the Passive Aggressive Classifier using Tfidf Vectorizer gives better accuracy.



Hence, we have used this model for testing the new news whether it is real or fake.

7.Conclusion

We have successfully created a portal for the detection of Fake-News identification the model works with an accuracy of 0.936 with a training data set, and it is observed that the accuracy keeps on increasing with the new data it is trained whenever runs. There is a scope for development of the project by implementing real-world use cases such as detecting fake forward messages in social media and stopping them automatically even if it is not reported by users.

APPENDIX

Code used for testing the input in the website

```
import sys
import pandas as pd
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.linear_model import PassiveAggressiveClassifier
from sklearn.model_selection import train_test_split
df = pd.read_csv ("C:/Users/komal/Desktop/Fake-News-Detection-master/fake_or_real_news.csv")
df = df.set_index("Unnamed: 0")
y = df.label
df.drop("label", axis=1)
X_train, X_test, y_train, y_test = train_test_split(df['text'], y, test_size=0.33, random_state=53)
tfidf_vectorizer = TfidfVectorizer(stop_words='english', max_df=0.7)
tfidf_train = tfidf_vectorizer.fit_transform(X_train)
tfidf_test = tfidf_vectorizer.transform(X_test)
t=[sys.argv[1]]
test = tfidf_vectorizer.transform(t)
linear_clf = PassiveAggressiveClassifier(50)
linear_clf.fit(tfidf_train, y_train)
pred = linear_clf.predict(test)
if(pred[0]=='FAKE'):
print(0)
else:
print(1)
```

REFERENCES

1. R. K. Kaliyar, "Fake News Detection Using A Deep Neural Network," 2018 4th International Conference on Computing Communication and Automation (ICCCA), Greater Noida, India, 2018, pp. 1-7, DOI: 10.1109/CCAA.2018.8777343.
2. Ruchansky, N., Seo, S., and Liu, Y. (2017). CSI: A hybrid deep model for fake news detection. In Proceedings of the 2017 ACM Conference on Information and Knowledge Management.
3. Conroy, N. J., Rubin, V. L., & Chen, Y. (2015). Automatic deception detection: Methods for finding fake news. Proceedings of the Association for Information Science and Technology, 52(1), 1-4.
4. Ibrishimova, Marina & Li, K.F.. (2020). A Machine Learning Approach to Fake News Detection Using Knowledge Verification and Natural Language Processing. 10.1007/978-3-030-29035-1_22.
5. Zhou, Zhixuan, et al. “Fake News Detection via NLP Is Vulnerable to Adversarial Attacks.” Proceedings of the 11th International Conference on Agents and Artificial Intelligence (2019): n. pag. Crossref. Web.
6. Abbasi, Ahmed, Fatemeh Zahedi, and Siddharth Kaza. "Detecting fake medical websites using recursive trust labeling." ACM Transactions on Information Systems (TOIS)30.4 (2012): 22
7. Liu, D., Wang, Y., Jia, Y., Li, J., and Yu, Z. (2014). Lsdh: a hashing approach for large-scale link prediction in microblogs. In Twenty-Eighth AAAI Conference on Artificial Intelligence.
8. McIntire, G. (2018). Fake real news dataset. In GeorgeMcIntire’sGithub.
9. Buntain, Cody, and Jennifer Golbeck. "Automatically Identifying Fake News in Popular Twitter Threads." Smart Cloud (Smart Cloud), 2017 IEEE International Conference on. IEEE, 2017.
10. Conroy, N. J., Rubin, V. L., and Chen, Y. (2015). Automatic deception detection: methods for finding fake news. In Proceedings of the 78th ASIS&T Annual Meeting.
11. Abbasi, Ahmed, Fatemeh Zahedi, and Siddharth Kaza. "Detecting fake medical websites using recursive trust labeling." ACM Transactions on Information Systems (TOIS)30.4 (2012)
12. Bhatt, Gaurav, et al. "Combining Neural, Statistical and External Features for Fake News Stance Identification." Companion of the Web Conference 2018 on the Web Conference 2018. International World Wide