FAKE NEWS PROJECT

Submitted by:

ANJANA.P

ACKNOWLEDGMENT

I would like to express my appreciation to team Fliprobo for giving such a data for analysis, with a full-length description of the project. My mentor Ms.Khushboo Garg has helped me in many stages of this project where I was stuck with problems. I use this opportunity to thank her for helping me at the right time without any delay.

Also, this project made me search for a lot of data's in several webpages and sites, that helped me to rectify my doubts and, I was able to study more about NLP

INTRODUCTION

Business Problem Framing

We are assigned with a new project about fake and real news detector.Fake news is a form of news consisting of deliberate disinformation or hoaxes spread via traditional news media or online social media.Our Project is that ,we need to build a system to identify fake and real news from various online sites and social medias. We were also given a set of dataset which contains some news.

The dataset has five columns named headlines, news,written by and also the label column.

It is a basic Natural Language Processing project based on sentiment analysis. Now we have to classify them as fake and real using machine learning and NLP techniques

Conceptual Background of the Domain Problem

The authenticity of Information has become a longstanding issue affecting businesses and society, both for printed and digital media. On social networks, the reach and effects of information spread occur at such a fast pace and so amplified that distorted, inaccurate, or false information acquires a tremendous potential to cause real-world impacts, within minutes, for millions of users. Recently, several public concerns about this problem and some approaches to mitigate the problem were expressed.

In this project, we are given a dataset in the fake-news_data.zip folder. The folder contains a CSV files train_news.csv and we have to use the train_news.csv data to build a model to predict whether a news is fake or not fake. We have to try out different models on the dataset, evaluate their performance, and finally report the best model got on the data and its performance

Review of Literature

First of all the data is saved in a csv file. Then its shape, datatypes, column value counts are all checked to get an outline of the data collected.Presence of Null values are also checked and should be cleared if anyData visualization is done for more clarification and understanding of the data.Data is cleaned by removing all punctuations and symbols.All the cases are converted to lower and stop words are removed.Word cloud is used to display the most frequent words appeared in the news.Train test split is done and various models are performed and the best model which gave maximum accuracy is selected as the final model and is saved as pickle file.

.

Motivation for the Problem Undertaken

The sheer scale of social media and online sites has led to myriad problems for the current generation. One of the major problems I have come across is the circulation of bogus news articles.And in todays world spurious news articles can cause panic and create chaos in our societies.The yello journals and the websites are aware that people would be blindly attracted towards catchy titles and images. So its our responsibility to put an end to this .Hope this analysis may help poor and innocent people not to get cheated in these fake journals.

Analytical Problem Framing

Mathematical/ Analytical Modeling of the Problem

In the describe function we have checked mean, std.deviation, minimum, maximum, 25 percentile, 50 percentile, 75 percentile of each attribute columns.

Mean is the average, median is the central value and mode is the frequency.
Percentile is the value below which the percentage of data falls.

We also use evaluation matrix like confusion matrix, accuracy score classification report, auc-roc. These can be expressed in mathematical formulas as:

Accuracy = TP+TN / TP+FN+FP+TN
Recall (True positive rate TPR) = TP / TP+FN
False negative rate (FNR) = FN / TP+FN
Precision =TP / TP+FP
F1 Score = precision – Recall / Precision +Recall
Specificity = TN / FP+TN

Data Sources and their formats

FlipRobo technologies have provided this dataset for detailed analysis and classification of fake and real news which are now a days very common in yellow journals as well as social medias. The data collected was in an excel sheet with a very detailed description of each columns with it. The data is converted into csv file and loaded in Jupyter notebook first. There are 6 columns and 20800 rows in this dataset. The data was in object datatypes. The output column name is ‘label’, which is filled with only 0’s and 1 value. 0’s represent the news is real and 1’s represent the news is fake.Thus, it is understood that Logistic regression and classification methods should be used for the model prediction.We have to test various models and select the more accurate one to predict whether the news is fake or not fake..

Data Preprocessing Done

First the important libraries for preprocessing and also csv file for analysis is imported. Shape and datatypes of columns are checked. There are 20800 rows and 6 columns in our dataset. They are object datatypes. Unnecessary columns like id and written_by are dropped as they provide no necessary information for our analysis. Null values are checked and should be

cleared if there is any. Extra columns for original length of news and legth after removal of stop words is added. News column data is converted to lower case and also all the punctuations and stop words are removed from it.Word cloud is used to display the most frequent words present both in fake news and real news.The data needs to be converted into a format that can be interpreted by a machine learning algorithm since these algorithms donot work well with textual data. Hence we need to convert it into a form that will enable the algorithm to discern patterns and meaningful insights from the data. Inorder to achieve this I used Tfidf vectorizer. Then maximum r_state is found and then various models like logistic regression, multinomial NB, Randomforest classifier, decision tree classifier, auc_roc score are found and the best model is saved as pickle file.

Data Inputs- Logic- Output Relationships

There are many null values in our dataset.There are 1957 null values in written_by column and 558 in headline column and 39 in news column.written_by column is dropped as it contain no importand data for our analysis.The null values in headline columns are filled with the corresponding news in news column and null values in news column is treated with the corresponding headline in headline column.All the data in news column are converted to lowercases and also punctuations and stop words are removed from them as part of data cleaning.

1) NLTK library-stopwords are downloaded from nltk library.Stopwords are English words which doesnot add much meaning to a sentence. stopwords are also removed from our dataset .List of stopwords can be found in corpus module.To remove the stopwords from a data first divide the text into words and then remove the word if it exists in the list of stopwords provided by NLTK.

Porter stemmer- stemmers remove morphological affixes from words leaving only the word stem.It extracts the base of a modified word.so the efficiency of any content based spam filter can be significantly improved.
Corpus- it is a language resource consisting of large and structured set of texts.

Hardware and Software Requirements and Tools Used

I used intel core i3 processor, 4GB RAM and 64 bit operating system as hardware and windows 10, MS excel, MS word and python 3 Jupyter notebook as software for the completion of this project. In jupyter notebook various libraries are also used. They include pandas, numpy, matplotlib , seaborn , imblearn ,wordcloud and sklearn.

Model/s Development and Evaluation

Identification of possible problem-solving approaches (methods)

The major problems we dealt with this dataset are

Presence of null values-There were only 20800 rows in the dataset and so we cannot delete the null valued rows.Only thing we can do is to fill the null valued columns without affecting the accuracy of the data.So I filled null values of headlines with data in corresponding news column and viceversa.

Testing of Identified Approaches (Algorithms)

After removing punctuations and stop words Tf-idf vectoriser is used to convert the text into machine learning algorithm format.Tf-Idf is also known as Term Frequency –Inverse document frequency. It gives us a way to associate each word in a document with a number that represents how relevant each word is in that document. With Tf-Idf, instead of representing a term in a document by its raw frequency (number of occurrences) or its relative frequency (term count divided by document length), each term is weighted by dividing the term frequency by the number of documents in the corpus containing the word. The overall effect of this weighting scheme is to avoid a common problem when conducting text analysis: the most frequently used words in a document are often the most frequently used words in all of the documents. In contrast, terms with the highest Tf-Idf scores are the terms in a document that are distinctively frequent in a document, when that document is compared to other documents. I used TfidfVectorizer from the sklearn library to convert the text into a sparse matrix. This matrix represents the Tf-Idf values for all the words present in my data. The training and test data are now represented by the variables x and y.Since I now have the data ready for implementing the machine learning algorithm, I move to the next step which includes fitting my machine learning algorithm on the training data.

Run and Evaluate selected models

1)Maximum accuracy score corresponding to r_state is found using for loop.

```
from sklearn.metrics import accuracy_score
from sklearn.tree import DecisionTreeClassifier
dtc=  DecisionTreeClassifier()

max_score=0
for r_state in range(40,100):
    x_train,x_test,y_train,y_test=train_test_split(x,y,random_state=r_state,test_size=0.22)

    dtc.fit(x_train,y_train)
    y_pred=dtc.predict(x_test)
    acc_scr=accuracy_score(y_test,y_pred)
    if acc_scr>max_score:
        max_score=acc_scr
        final_r_state=r_state
print("max accuracy score corresponding to",final_r_state,"is",max_score)
```

```
max accuracy score corresponding to 70 is 0.8975087412587412
```

I got maximum accuracy score as 0.89 and the corresponding r_state is 70.

1) Logistic regression-It is a classification algorithm used to predict the probability of categorical dependent variable

```
: lr=LogisticRegression()
lr.fit(x_train,Y_train)
lr.score(x_train,Y_train)
predlr=lr.predict(x_test)
print("Accuracy score :",accuracy_score(Y_test,predlr))
print(confusion_matrix(Y_test,predlr))
print(classification_report(Y_test,predlr))
```

```
Accuracy score : 0.7676743897906083
[[32136  8224]
 [10527 29823]]
              precision    recall  f1-score   support

           0       0.75      0.80      0.77     40360
           1       0.78      0.74      0.76     40350

    accuracy                           0.77     80710
   macro avg       0.77      0.77      0.77     80710
weighted avg       0.77      0.77      0.77     80710
```

Logistic regression is giving an accuracy score -0.76

3)Random Forest Classifier –Random forest is a meta estimator that fits a number of decision tree classifiers on various sub samples of the dataset and uses averaging to improve the predictive accuracy and control overfitting.

```
#Random forest classifier

from sklearn.ensemble import RandomForestClassifier
rf=RandomForestClassifier(n_estimators=20,random_state=68)
rf.fit(x_train,y_train)
print('score:',rf.score(x_train,y_train))
predrf=rf.predict(x_test)
print('\n')
print("Accuracy score :",accuracy_score(y_test,predrf))
print(confusion_matrix(y_test,predrf))
print(classification_report(y_test,predrf))
```

```
score: 0.9999383629191322


Accuracy score : 0.8607954545454546
[[2109  222]
 [ 415 1830]]
              precision    recall  f1-score   support

           0       0.84      0.90      0.87      2331
           1       0.89      0.82      0.85      2245

    accuracy                           0.86      4576
   macro avg       0.86      0.86      0.86      4576
weighted avg       0.86      0.86      0.86      4576
```

Random Forest Classifier gives accuracy -0.86

4))Multinomial Naïve Bayes-It is a variant Naïve Bayes that follows Multinomial normal distribution and supports continuous data.

```
mnb=MultinomialNB()
mnb.fit(x_train,y_train)
predmnb=mnb.predict(x_test)
print("Accuracy score :",accuracy_score(y_test,predmnb))
print(confusion_matrix(y_test,predmnb))
print(classification_report(y_test,predmnb))
```

```
Accuracy score : 0.8446241258741258
[[2317   14]
 [ 697 1548]]
              precision    recall  f1-score   support

           0       0.77      0.99      0.87      2331
           1       0.99      0.69      0.81      2245

    accuracy                           0.84      4576
   macro avg       0.88      0.84      0.84      4576
weighted avg       0.88      0.84      0.84      4576
```

Multinomial NB is giving an accuracy score of 0.84

5)Decision tree classifier – A tree structure is constructed that breaks the dataset into smaller subsets eventually resulting in prediction.The root node partitions the data based on most influential feature partitioning.There are two measures for this. They are gini impurity and Entropy

```
dtc=DecisionTreeClassifier()
dtc.fit(x_train,y_train)
dtc.score(x_train,y_train)
preddtc=dtc.predict(x_test)
print("Accuracy score :",accuracy_score(y_test,preddtc))
print(confusion_matrix(y_test,preddtc))
print(classification_report(y_test,preddtc))
```

```
Accuracy score : 0.8946678321678322
[[2074  257]
 [ 225 2020]]
              precision    recall  f1-score   support

           0       0.90      0.89      0.90      2331
           1       0.89      0.90      0.89      2245

    accuracy                           0.89      4576
   macro avg       0.89      0.89      0.89      4576
weighted avg       0.89      0.89      0.89      4576
```

Decision Tree Classifier gives accuracy score 0.89

6)AUC_ROC –It is an important evaluation metrics to check any models performance.auc_roc graph is drawn with different threshold valued outputs based on true

positive rate and false positive rate. The more the area is under the curve, it shows that the model performs well.
We get an auc_score of 0.94
7)Cross Validation score-Logistic Regression is giving maximum accuracy score.So its cross validation score is found out and its mean score is 0.94 and std.deviation is 0.002

Key Metrics for success in solving problem under consideration

Using logistic regression,MultinomialNB, Decision Tree classifier,Random Forest classifier etc, Logistic regression is giving maximum accuracy score of 94%.So its cross validation score is checked.

```
from sklearn.model_selection import cross_val_score

lr=LogisticRegression()
score=cross_val_score(lr,x,y,cv=5)
print("score:",score)
print("Mean score:",score.mean())
print("Standard deviation:",score.std())
```

```
score: [0.94711538 0.94350962 0.94879808 0.94447115 0.94423077]
Mean score: 0.9456249999999999
Standard deviation: 0.0020019868090054745
```

Cross_val_score of Logistic Regression also gives a score of 94%. That means our data is not overfit.We can also see that std.deviation is very much low.
After checking cross_val_score The model is saved as pickle file as it is the best model.

Visualizations

```
sns.countplot(d1["label"])
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x21f6edcfe80>
```

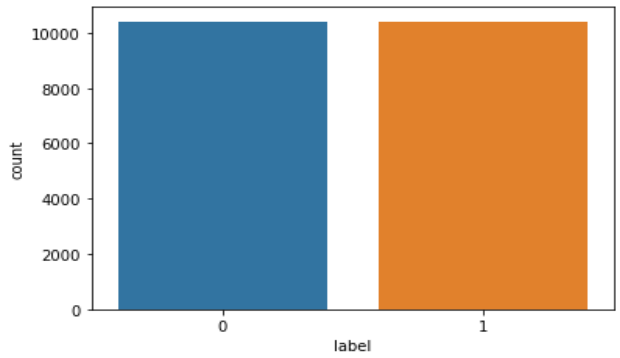

Fig:1 shows the countplot of label.The counts of fake news and real news can be inferred from this figure.There are 10413 fake news and 10387 real news in our dataset.

```
#Number of characters present in each headline
d1['headline'].str.len().hist(by=d1['label'])
```

```
array([<matplotlib.axes._subplots.AxesSubplot object at 0x0000021F6F576D30>,
       <matplotlib.axes._subplots.AxesSubplot object at 0x0000021F6F5AA220>],
      dtype=object)
```

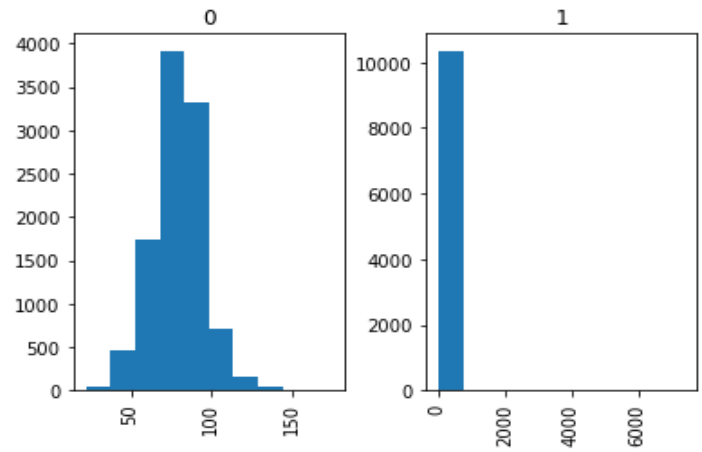

Fig:2 shows the number of characters present in each of the headline columns.We can see characters map for both fake and real news separate.

```
#Number of characters present in each news
d1['news'].str.len().hist(by=d1['label'])
```

```
array([<matplotlib.axes._subplots.AxesSubplot object at 0x0000021F6F664E50>,
       <matplotlib.axes._subplots.AxesSubplot object at 0x0000021F6F6F3D90>],
      dtype=object)
```

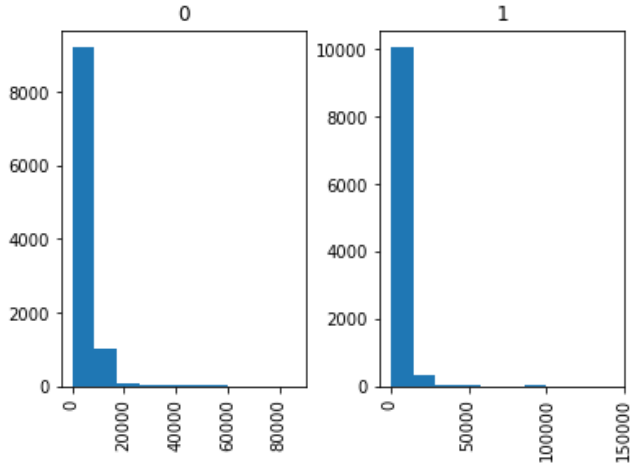

Fig:3 shows the characters present in the news column for both fake and real news.

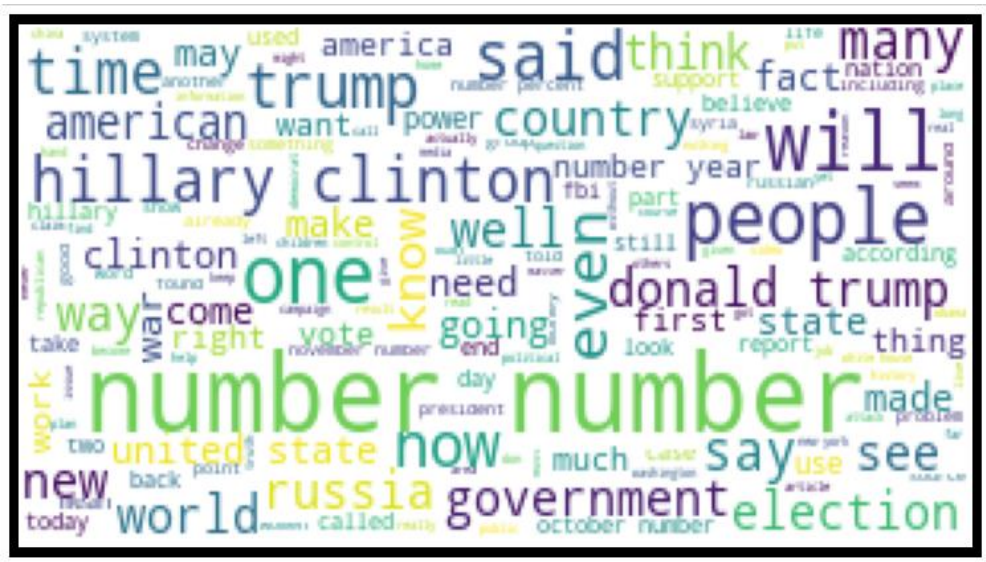

Fig:4 shows the most frequent words appeared in fake news

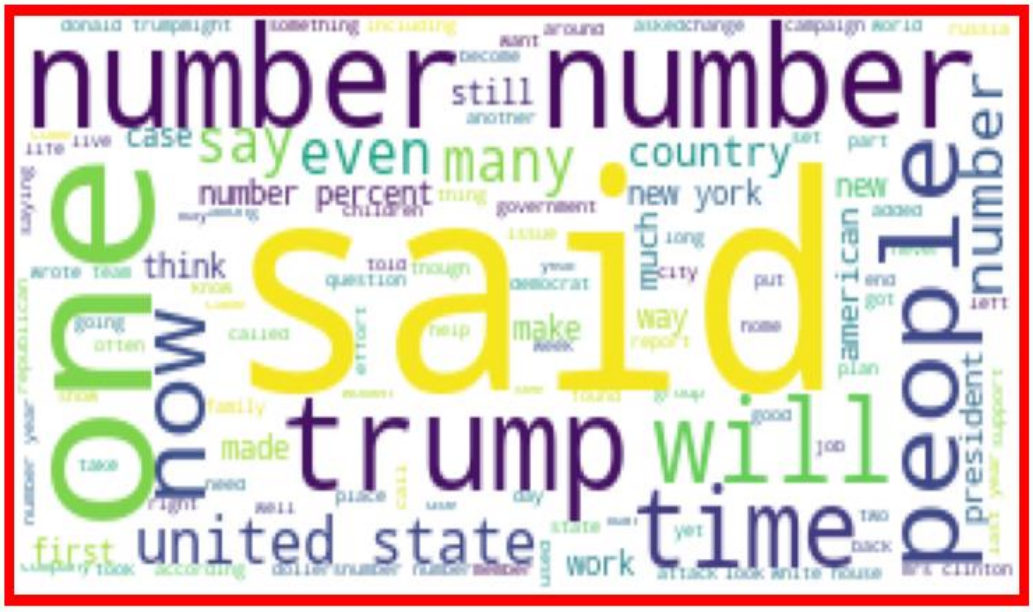

Fig:5 shows most frequent words appeared in real news

Interpretation of the Results

The news after removing punctuations and stopwords are converted to vector form to make it machine readable and then they are splitted to train and test and various models are performed to predict whether the news is fake or not. We tested logistic regression model, decision tree classifier, random forest classifier, auc_roc etc. We get maximum accuracy for logistic model and we used it as our selected model.

CONCLUSION

We used Natural Language Process method to solve this problem. Wide spreading of fake news is a common problem occurring nowadays in every platforms online and offline. This fast spreading of fake news like forest fire can cause chaos. It can even cause many harms to individuals. So it is very important nowadays to identify these yellow journals and block them. This is project is a small approach to identify them .We collected the news samples from online sites and first of all the data is cleaned and preprocessed. Then various models are performed and found that Logistic regression model gives the best accuracy score.

Limitations of this work and Scope for Future Work

Fake news is not something new however. As technology evolves and advances over time ,the detection of fake news also becomes more challenging as social media continues to dominate our every day lives and hence accelerating the speed of which fake news travels. Other than the sheer speed of how fast fake news travels, it is also more challenging to detect it simply because of how attractive most fake news articles are titled. Fake news

detection is still a challenge even to deep learning methods such as CNN and RNN because the content of fake news is planned in a way it resembles the truth so as to deceive readers.

With the help of artificial intelligence, we can control and limit the spread of such misinformation more quickly and efficiently as compared to manual efforts.