

FAKE NEWS CLASSIFIER





FAKE NEWS CLASSIFIER

SUBMITTED BY

SAURABH UPADHYAY

## ACKNOWLEDGEMENT

First of all, I extol the Almighty for pouring his blessings on me and giving me potentiality and opportunity to carry the work to its end with a success.

“It’s impossible to prepare a project report without the help and fillip of some people and certainly this project report is of no exception.”

At the commencement of this project report I would like to evince my deepest sense of gratitude to Ms. Astha Mishra, my honored mentor. Without her guidance, insightful decision, valuable comments and correction it would not have possible to reach up to this mark.

I would like to draw my gratitude to Flip Robo and Data Trained for providing me a suitable environment and guidance to complete my work. Last but not least thanks to the brilliant authors from where I have got the idea to carry out the project.

References were taken from various articles from Medium, KDnuggets, Towards Data Science, Machine Learning Mastery, Analytics Vidya, American Statistical Association, Research Gate and documentations of Python and Sklearn.

Saurabh Upadhyay

CONTENTS

|  |  |
| --- | --- |
| ACKNOWLEDGEMENT | 03 |
| CHAPTER 1: INTRODUCTION | 07-08 |
| CHAPTER 2: ANALYTICAL PROBLEM FRAMING | 09-16 |
| CHAPTER 3: MODEL DEVELOPMENT | 17-25 |
| CHAPTER 4: CONCLUSION | 26 |

LIST OF FIGURES

|  |  |
| --- | --- |
| FIGURES | PAGE NO |
| Fig.1 THE DATASET | 10 |
| Fig.2 SHAPE & COLUMNS | 11 |
| Fig.3 DUPLICATES & NULL VALUES | 11 |
| Fig.4 HEATMAP OF NULL VALUES | 11 |
| Fig.5 DROPPING COLUMNS AND NULL VALUES | 12 |
| Fig.6 NEW COLUMNS FOR STRING LENGTH | 12 |
| Fig.7 LOSS & NEW SHAPE | 13 |
| Fig.8 FUNCTION FOR PROCESSING DATA | 13 |
| Fig.9 SAMPLE TESTING | 14 |
| Fig.10 TEST RESULTS | 14 |
| Fig 11: FUNCTION IMPLEMENTATION | 14 |
| Fig.12 FUNCTION RESULT | 15 |
| Fig.13 TF-IDF | 16 |
| Fig.14 X & Y PREPARATION | 17 |
| Fig.15 LOGISTIC REGRESSION | 19 |
| Fig.16 CROSS VALIDATION OF LOGISTIC REGRESSION | 19 |
| Fig.17 HYPERPARAMETER TUNING OF PASSIVE AGGRESSIVE CLASSIFIER | 19 |
| Fig.18: PASSIVE AGGRESSIVE CLASSIFIER | 20 |
| Fig.19 CROSS VALIDATION OF PASSIVE AGGRESSIVE CLASSIFIER | 20 |
| Fig.20: AUC CURVE OF PASSIVE AGGRESSIVE CLASSIFIER | 20 |

|  |  |
| --- | --- |
| Fig.21 HEATMAP OF CONFUSION MATRIX OF PASSIVE AGGRESSIVE CLASSIFIER | 21 |
| Fig.22 RESULTS | 22 |
| Fig.23 PIE PLOT OF LABEL | 22 |
| Fig 24. AUC\_ROC CURVES | 23 |
| Fig 25. HEATMAP OF CONFUSION MATRIXS | 24 |
| Fig 26. SAVING MODEL | 25 |

CHAPTER-1

INTRODUCTION

1.1 BUSINESS PROBLEM

In current world scenario it’s easy to find news but it’s becoming harder and harder to guarantee the authenticity of that piece of news. These kinds of news are swarming all over corrupting everything. 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.

1.2 BACKGROUND

The rate of production of fake news have increased exponentially. In the past news obtained from newspaper, radio or TV were considered as the best and authentic source of information about the real world and ongoing situations but now everything has changed. In the run of popularity and ill mind set the media houses and social media are spreading fake news. It’s becoming harder and harder to say whether a piece of news is real or fabricated. Its becoming a new weapon of destruction.

The effect of fake news can be seen everywhere. The fake news leads to communal disturbance, character assassination, mental trauma, sometimes it is used as a weapon to achieve some illicit plans etc. these are like wild fire which spread too quickly and difficult to control. Its really hard for a human being to differentiate between a fake news and an authentic news.

However, this news can be classified into fake or authentic news group with the help of python and algorithms of Machine Learning and Natural Language Processing.

1.3 MOTIVATION

The project was the first provided to me by FlipRobo as a part of the internship programme. The exposure to real world data and the opportunity to deploy my skillset in solving a real time problem has been the primary objective. However, the motivation for taking this project was that it is relatively a new field of research. Here we have many options but less concrete solutions. The main motivation was to classify the news in order to bring awareness and reduce unwanted chaos.

CHAPTER-2

ANALYTICAL PROBLEM FRAMING

2.1 ANALYTICAL MODELING OF PROBLEM

People can get infected with fake news very quickly with misleading words and images and post them without any fact-checking. The social media life has been used to distribute counterfeit data, which has a significant negative influence on individual consumers and on a wider community. The fake news problem is tackled using a machine learning algorithm

The dataset provided here has a shape of (20800, 6). Which means it has 20800 rows and 6 columns. Here the target or the dependent variable named “Label” have two distinct values 0 and 1. Where 0 represents the news which are not fake or authentic while 1 represents category of fake news. As the target column ‘Label’ is giving binary outputs and all the independent variables has text so it is clear that it is supervised machine learning problem where we can use, we can use the techniques of NLP and classification-based algorithms of Machine learning.

Here we will use NLP techniques like word tokenization, lemmatization and tfidf vectorizer then those processed data will be used to create best model using various classification based supervised machine learning algorithms like Logistic Regression, Passive Aggressive Classifier, Multinomial NB, Complement NB, Random Forest Classifier.

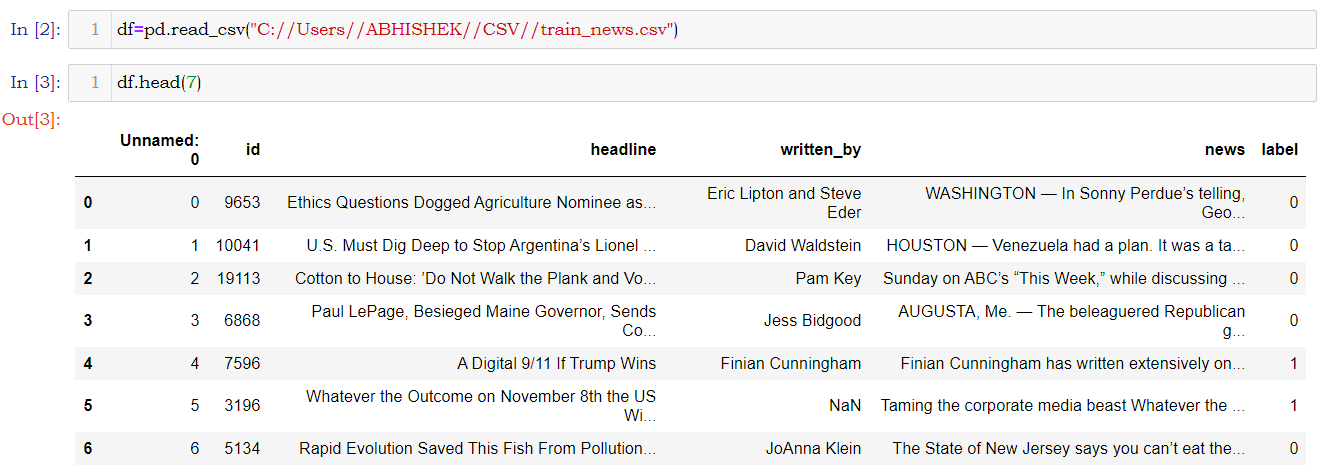
The passive Aggressive Classifier belongs to the family of online machine learning algorithms and it is very much helpful in processing large scale data. It remains passive for a correct classification and turns aggressive in case of a misclassification. Its aim is to make updates that corrects the loss causing very little change in the weight vector.

2.2 DATA SOURECE AND FORMATS

The data was provided by FlipRobo in CSV format. After loading the dataset into Jupyter Notebook using Pandas after loading with help of df.head() **[Fig. 1]** it can be seen that there are six columns named as “*Unnamed: 0, id, headline, written\_by, news, label*”. The metadata table is provided below [**Table 1**].

|  |  |
| --- | --- |
| **Unnamed: 0** | **SERIAL NUMBER/ INDEX NUMBER** |
| **Id** | **UNIQUE ID OF EACH NEWS ARTICLE** |
| **Headline** | **THE TITLE OF THE NEWS** |
| **Written\_By** | **THE AUTHOR OF THE NEWS ARTICLE** |
| **News** | FULL TEXT OF THE NEWS ARTICLE |
| **Label** | **IT TELLS WHETHER THE NEWS IS FAKE (1) OR NOT FAKE (0)** |

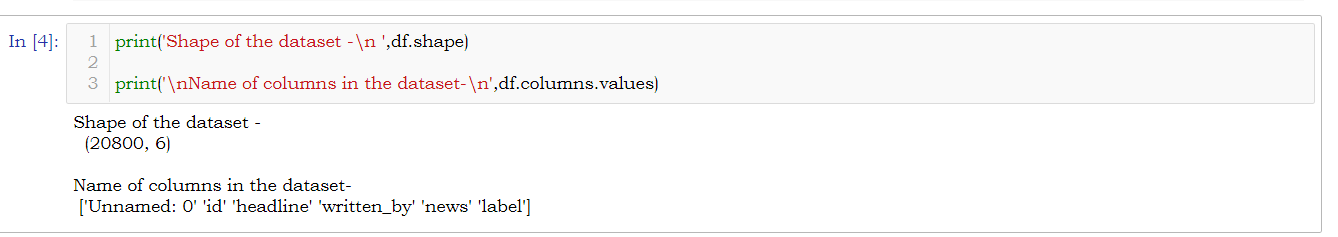
**(Table 1: METADATA)**



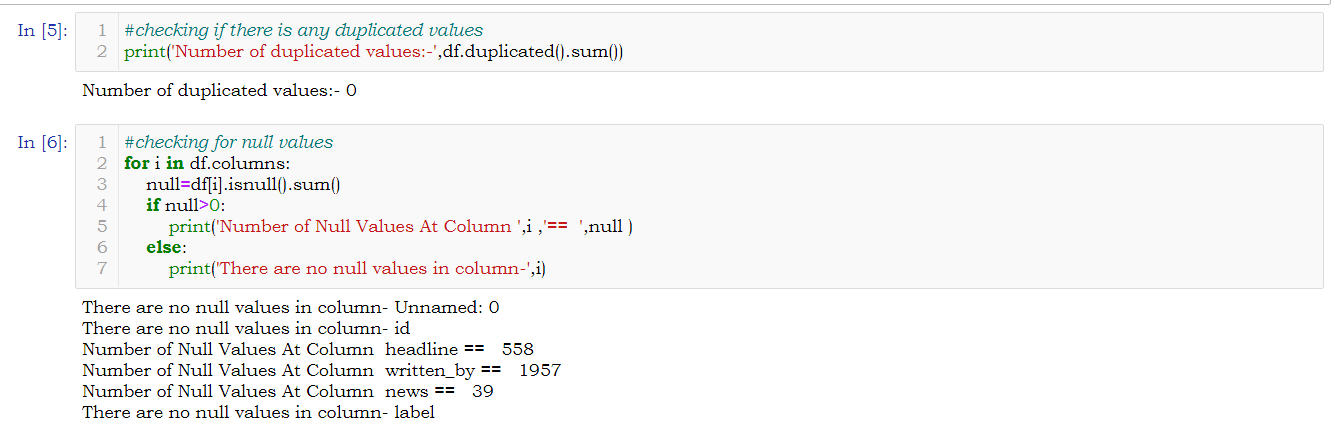
**(Fig 1. DATASET)**

As mentioned earlier the shape of the dataset is (20800, 6). The shape of the dataset in form of a tuple can be accessed using df.shape() [Fig 2]. The dataset has no duplicated values but have few null values. The number of duplicated values can be seen using df.duplicated().sum() and the null values can be seen using df.isnull().sum() [Fig 3].

The null values can also be visualized with help of seaborn and matplotlib library [Fig 4].



(Fig 2. SHAPE AND COLUMNS)



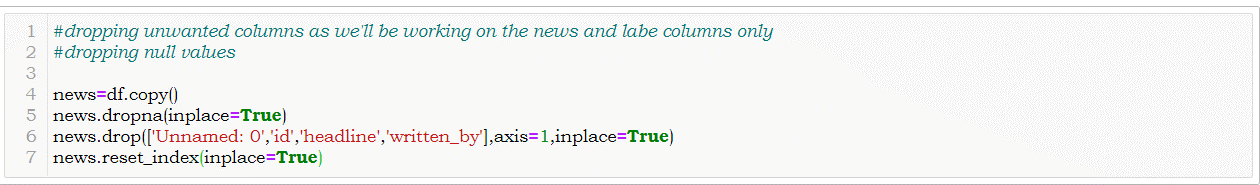
(Fig 3. DUPLICATES AND NULL VALUES)



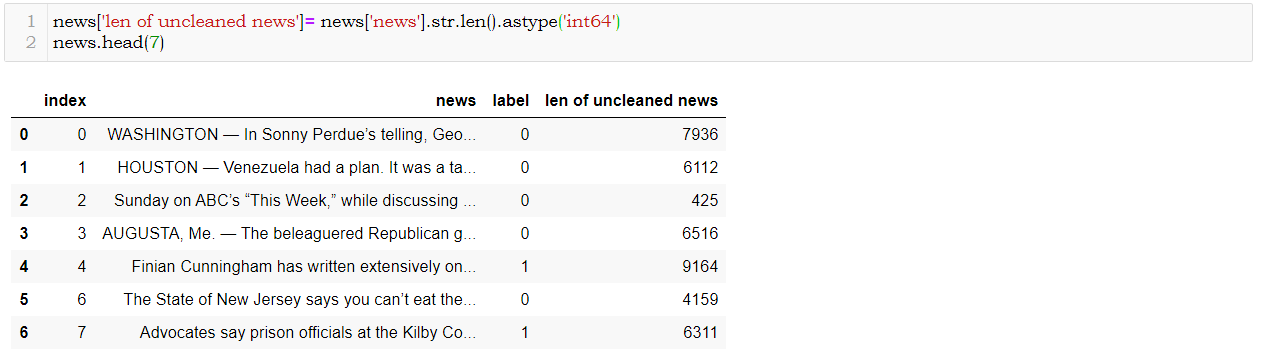
(Fig 4. HEATMAP OF NULL VALUES)

2.3 DATA PREPROCESSING

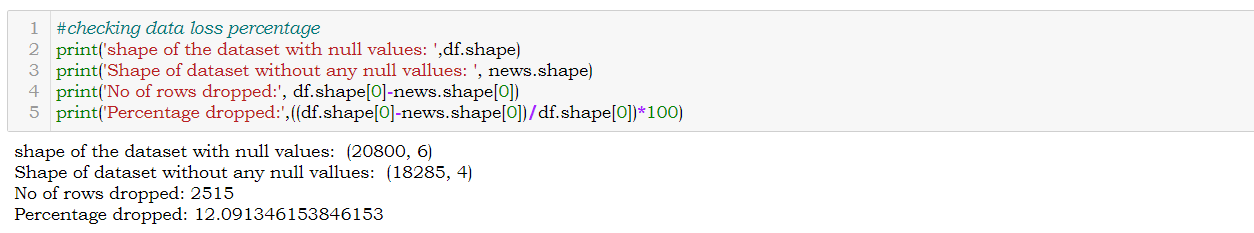
After the dataset is loaded and the shape, null values and duplicated values were checked then the data- set is further treated where the unwanted columns like “*Unnamed: 0, id, headline, written\_by”* removed as we will work on the columns “*news, label”.* So a copy of the dataset was made using df.copy() and these columns were dropped from the new dataset created using df.drop(). As we have found there are numbers of null values and null values are not good for modelling. So these values were dropped using df.dropna().After dropping the null values the index was reset . [Fig 5]. After removing these null values and unwanted columns a new column representing the string length of the ‘news’ column is added to the dataset [Fig 6]. It’ll help to know the earlier length of the strings in ‘news’ columns and new length of those strings after processing. A calculation showed that total of 2515 null values were dropped and the shape of the dataset after removing unwanted columns and null values became (18285,4) [Fig 7].



(Fig 5. DROPPING COLUMNS AND NULL VALUES)

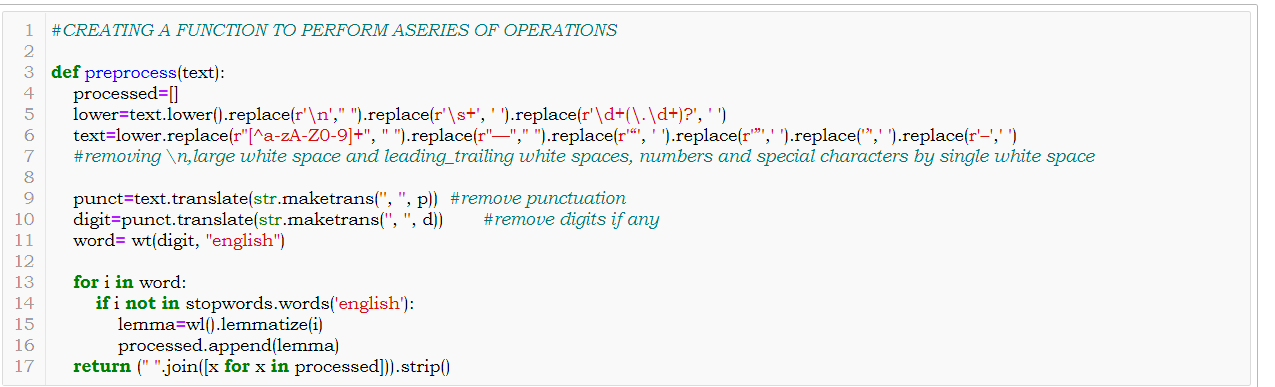


(Fig 6. NEW COLUMNS FOR STRING LENGTH)



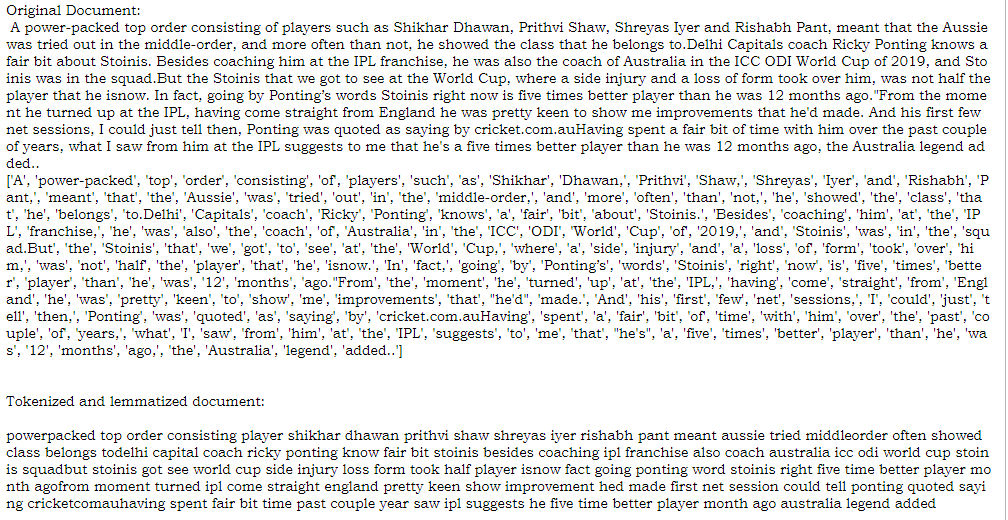
(Fig 7. LOSS & NEW SHAPE)

After the dataset is treated and freed from unwanted columns and null values the NLP techniques were implemented for processing the texts in the ‘news’ columns. In the preprocessing the string converted to lower case as it is easier to understand for the machine then from the string stopwords, special characters, digits were dropped using proper format. After those unnecessary characters were removed the string is tokenized using word\_tokenization() of NLTK library those tokenized words were lemmatized using wordnetLemmatizer() which brings back all words to their root form with proper meaning. Then again, those tokenized words were joined to form a string. All these operations were compiled inside a function [Fig 8]. After the function was created a test run was done on a sample text to check the effectiveness of the function so created [Fig 9-10]. After successful testing the entire ‘news’ column was processed using the function created to get a clear and pure form of data for further operations [Fig 11].

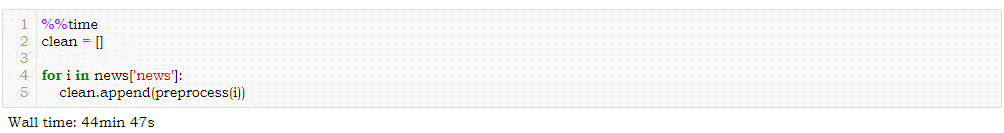
(Fig 8. FUNCTION FOR PROCESSING DATA)



(Fig 9. SAMPLE TESTING)



(Fig 10. TEST RESULTS)



(Fig 11. FUNCTION IMPLEMENTATION)

After the procedure was completed a list of cleaned data were obtained which was added to the dataset by column name ‘clean news’ and another column name ‘len of clean news’ was added showing the length of words in the ‘len of clean news’ column. Further calculation revelled

that there were total of 87771325 words were present in the news column after processing it became 58426659 [**Fig 12**].



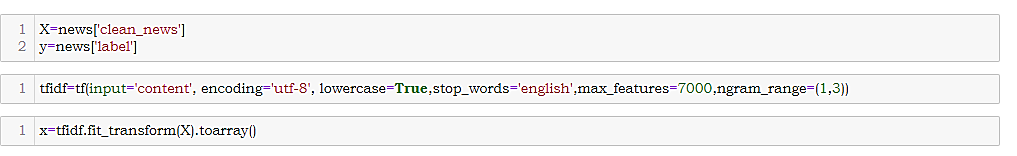
(Fig 12. FUNCTION RESULT)

After getting a cleaned data TF-IDF vectorizer will be used. It’ll help to transform the text data to feature vector which can be used as input in our modelling. The TFIDF stands for Term Frequency Inverse Document Frequency. It is a common algorithm to transform text into vectors or numbers. It measures the originality of a word by comparing the frequency of appearance of a word in a document with the number of documents the words appear in.

Mathematically,

TF-IDF =TF(t\*d) \* IDF (t,d)

So here the dataset is divided into two parts X and Y. X represents the column ‘clean\_news’ and Y represents the column ‘label’. After the splitting the tfidf vectorizer was initialized and X is fitted into it and converted into array form. [Fig 13].



(Fig 13. TF-IDF)

2.4 HARDWARE & TOOL USED

In this project the below mentioned Hardware, IDE, Language, Packages were used.

|  |  |
| --- | --- |
| HARDWARE | LAPTOP: ASUS TUF A17  OS: WIN 10 HOME BASIC  PROCESSOR: AMD RYZEN 7 4800H  RAM: 16GB  VRAM: 6GB NVIDIA GTX 1660Ti |
| LANGUAGE | Python 3.8 |
| IDE | JUPYTER NOTEBOOK 6.0.3 |
| PACKAGES | PANDAS, NLTK, SKLEARN, MATPLOTLIB, SEABORN |

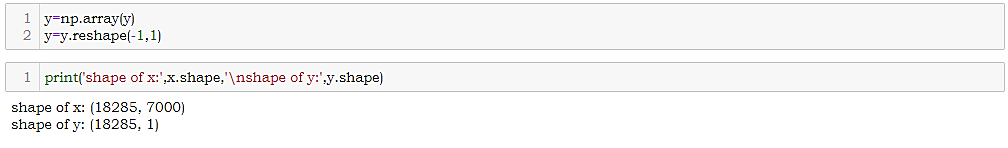
**(Table 2: HARDWARE & TOOLS)**

CHAPTER-3

DEVELOPMENT AND EVALUTION

3.1 IDENTIFACTION OF POSSIBLE PROBLEM-SOLVING APPROACHES

After TF-IDF implementation array conversion we have x and y for modelling. The shape of x found to be (18285,7000) and y was converted into an array and reshaped and the shape found to be (18285,) [Fig 14].



(Fig 14. X & Y PREPARATION)

3.2 TESTING OF IDENTIFIED ALGORITHMS

Here the output column 'label' is generating binary output so it’s a classification-based problem and we can use the following algorithms for modelling and the highest performing algorithms to get our final model;

* Logistic Regression()
* Passive Aggressive Classifier()
* Multinomial NB()
* Complement NB()
* Random Forest Classifier()

With the help of GridSearchCV hyper parameter tuning will be done and the best parameters for each model will be found. During modelling various metrices like f1 score, confusion matrix, accuracy score, classification report, roc curve, roc auc score, mean squared error, precision score, recall score, log loss will be used to determine the performance of the model.

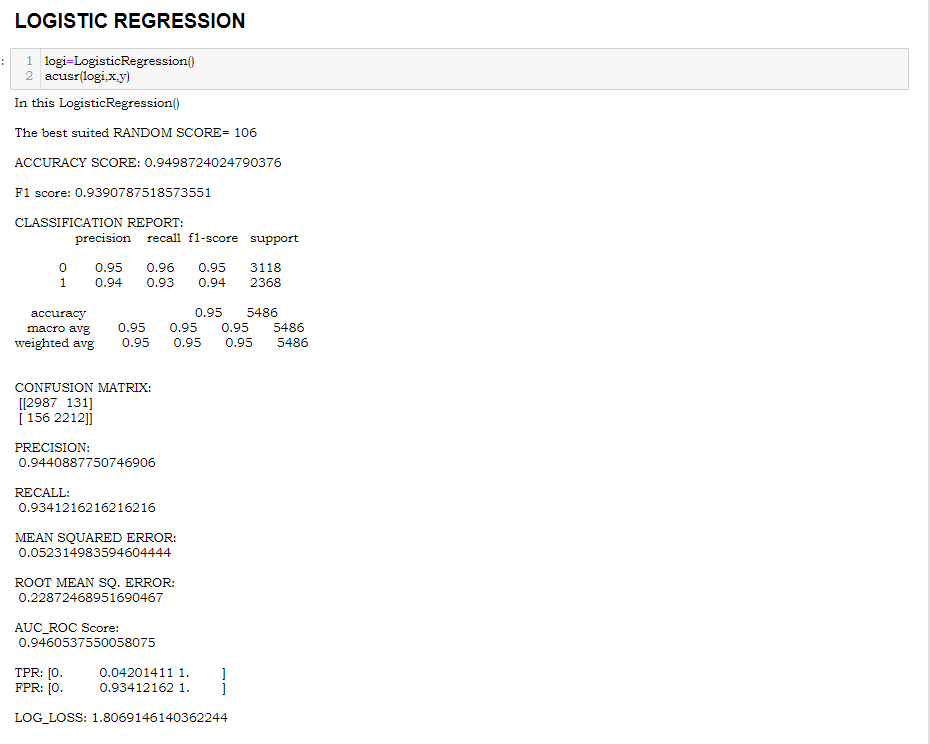
To check whether the model suffering from over fitting or underfitting cross val score will be used where ‘f1’ will be used as scorer. To view the best performing model AUC Curve will be used and heatmap will be used to visualize the confusion matrix. All the result will save to a DataFrame and at the end the best model will be saved using Joblib library.

3.3 TESTING OF IDENTIFIED ALGORITHMS

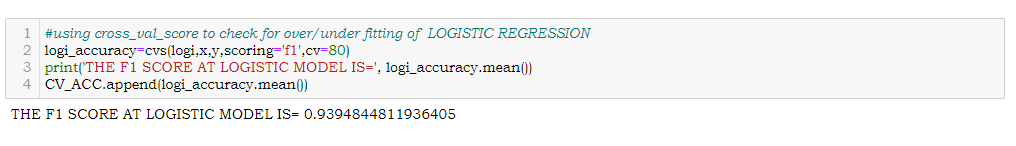
Here I have created a number of list named as F1, ACCURACY, PRECISION, RECALL, RMSE, MSE, AUC, TPR, FPR, CV\_ACC, LOG\_LOSS to hold the values of matrices like f1 scores, accuracy scores, precision values, recall values, root mean squared error values, mePage | 18an squared error values, auc scores, tpr values, fpr values, cross validation with f1 values, log loss values respectively.

Here I have generated a function which will find the best random score for the model in a range of 25 to 180. It’ll also show the confusion matrix, accuracy score, classification report, roc curve, auc, roc auc score, mean squared error, precision score, recall score, tpr, fpr, f1 score, log loss value for an algorithm at the best random score. The values obtained will be added to their respective lists.

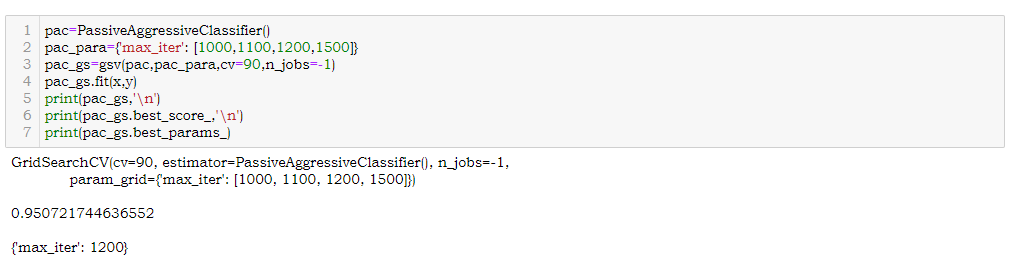
Below are few images of function performed on algorithms. Showing the metrics, heatmap of confusion matrix, AUC ROC curve, Cross validated values.



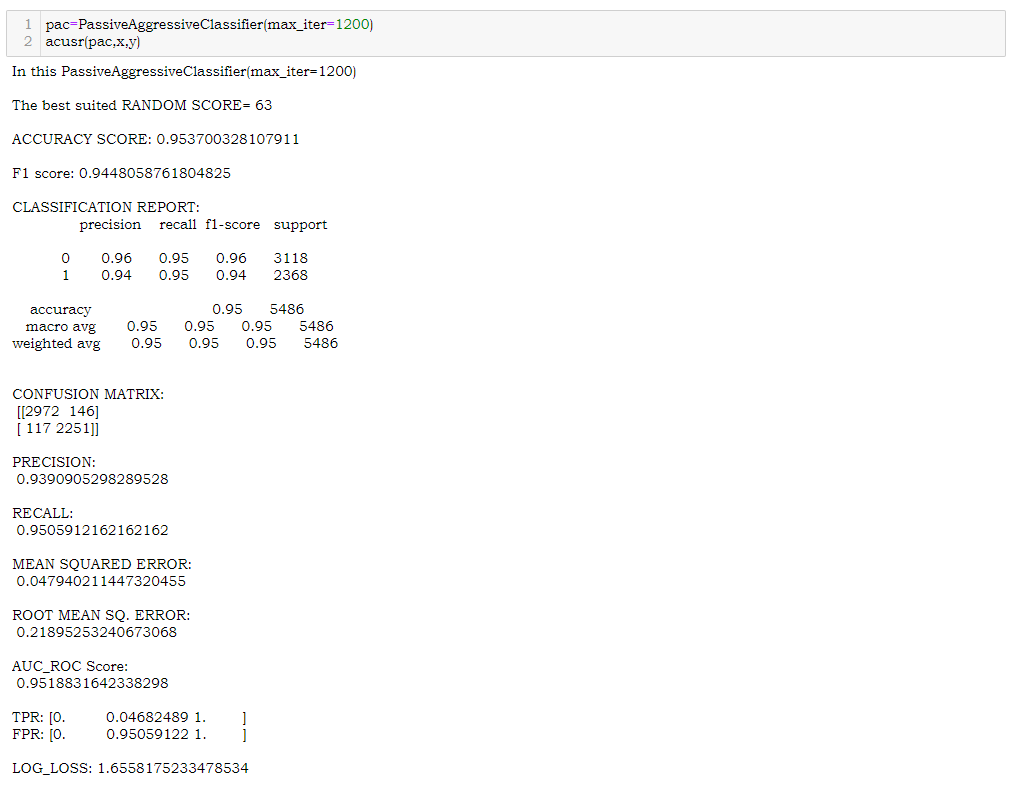
(Fig 15. LOGISTIC REGRESSION)



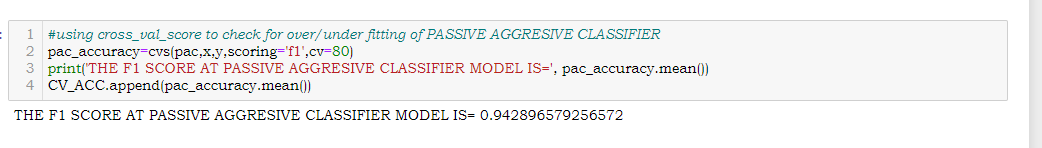
(Fig 16. CROSS VALIDATION OF LOGISTIC REGRESSION)



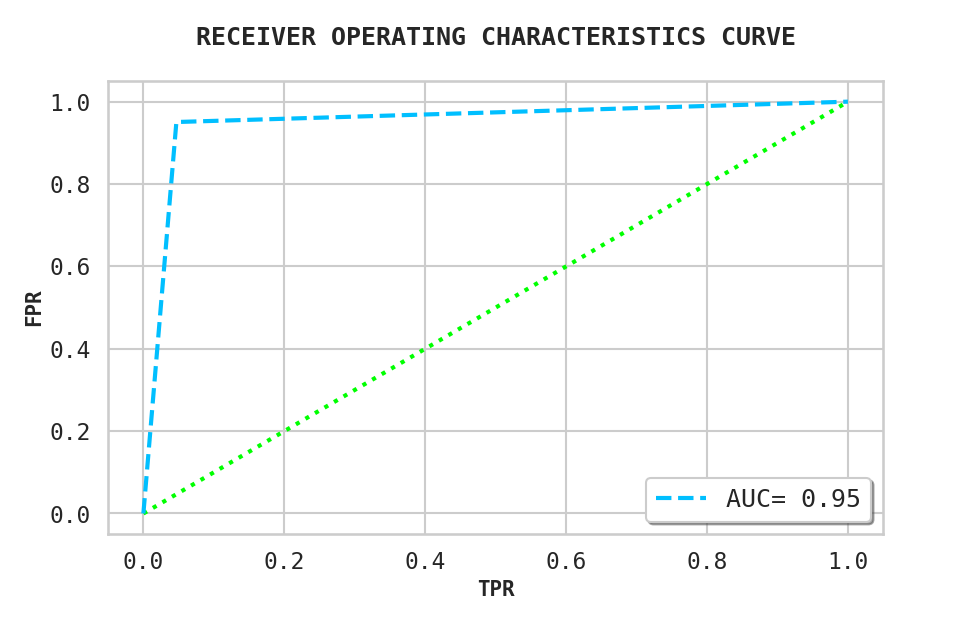
(Fig 17. HYPERPARAMETER TUNING OF PASSIVE AGGRESSIVE CLASSIFIER)



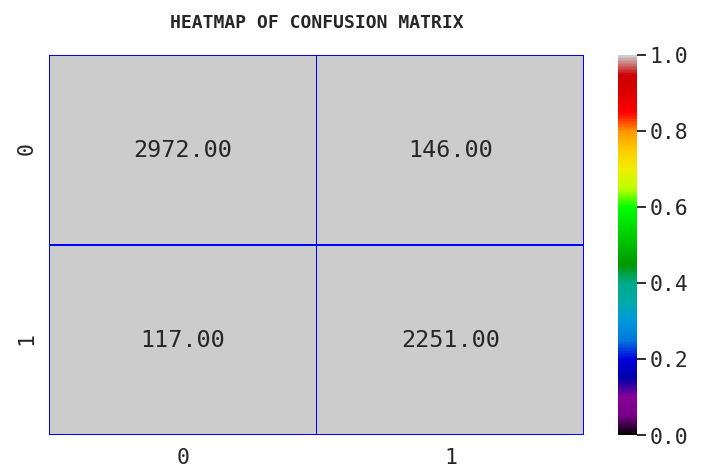
(Fig 18. PASSIVE AGGRESSIVE CLASSIFIER)



(Fig 19. CROSS VALIDATION OF PASSIVE AGGRESSIVE CLASSIFIER)



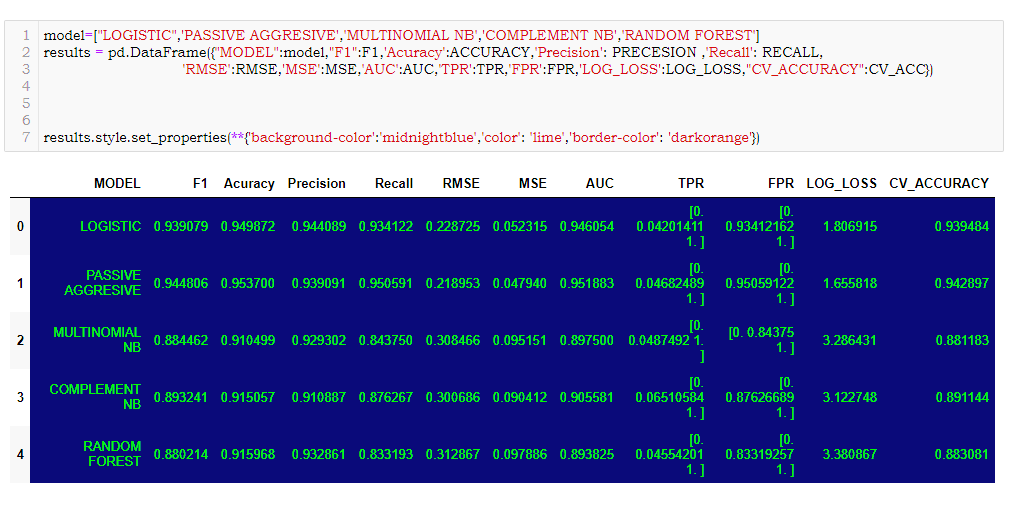
(Fig 20. AUC CURVE OF PASSIVE AGGRESSIVE CLASSIFIER)



(Fig 21. HEATMAP OF CONFUSION MATRIX OF PASSIVE AGGRESSIVE CLASSIFIER)

3.4 METRICE OF EVALUATION

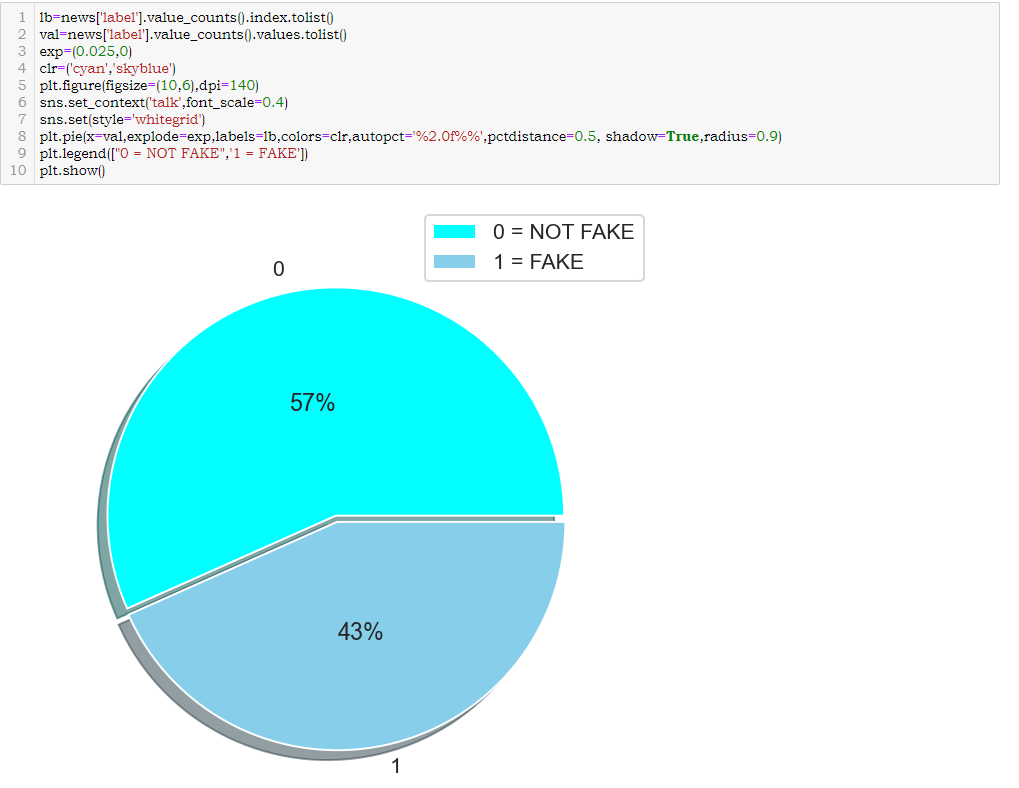
In the modeling I have chosen metrices like F1 score, Accuracy score, Precision, Recall, Mean Squared Error, Root Mean Square Error, Classification Report, Confusion Matrix, AUC, tpr, fpr and Cross val Score with f1 as scorer as my evaluation criteria. All the values were stored in a list and later they were saved in form of a DataFrame for proper evaluation and visualization of the values. Basing on the values the best model has been selected [Fig 21].



(Fig 22. RESULTS)

3.5 VISUALIZATION

Visualization plays a crucial role in EDA as well as during modelling. It gives a better idea about the things going on beautifully. Below are the few visualizations used during this project to understand the dataset and performance of the algorithms.



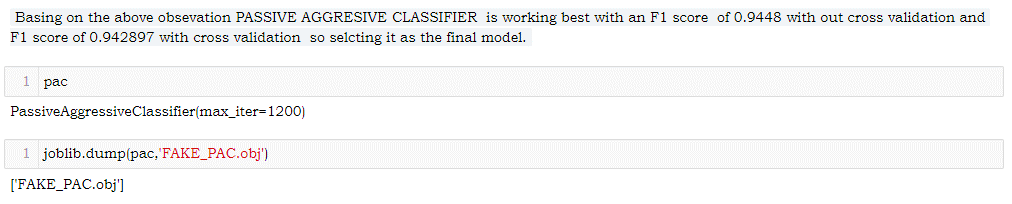
(Fig 23. PIE PLOT OF LABEL)

|  |  |
| --- | --- |
| AUC ROC CURVE – LOGISTIC REGRESSION | AUC ROC CURVE – MULTINOMIAL NB |
|  |  |
| AUC ROC CURVE – PAC | AUC ROC CURVE – COMPLEMENT NB |
|  |  |
| AUC ROC CURVE – RFC | |
| (Fig 24. AUC\_ROC CURVES) | |

|  |  |
| --- | --- |
| CONFUSION MATRIX – LOGISTIC REG | CONFUSION MATRIX – MULTINOMIAL NB |
|  |  |
| CONFUSION MATRIX – PAC | CONFUSION MATRIX – COMPLEMENT NB |
|  |  |
| CONFUSION MATRIX – RFC | |
| (Fig 25. HEATMAP OF CONFUSION MATRIXS) | |

3.6 INTERPRETATION

Basing on the result obtained ‘Passive Aggressive Classifier’ have performed well and has given better result as compared to other models. Here the Accuracy score is 95.37, Precision is 93.09, F1 score is 94.48, AUC value is 95.18, and after cross validation the F1 score becomes 94.48. As this model have all the desired characters and performance so Passive Aggressive Classifier has been selected as final model and it will be saved using joblib library.



(Fig 26. SAVING MODEL)

CHAPTER-4

CONCLUSION

4.1 KEY FINDINGS

From the above analysis the below mentioned results were achieved which depicts the

chances and conditions of a news being a fake;

* With the increasing popularity of social media, more and more people consume news from social media instead of traditional news media. However, social media has also been used to spread fake news, which has strong negative impacts on individual users and broader society.

4.2 LEARNING OUTCOMES

* It is possible to classify news content into the required categories of authentic and fake news however there will be always a bias to this kind of classification which depends on the behavioural pattern of the listener. However, using this kind of project an awareness can be created to know what is fake and authentic.

4.3 LIMITATION AND SCOPE OF WORK

Nothing is perfect and this project is of no exception. There are certain areas which can be enhanced.Fake news detection is an emerging research area with few public datasets. So a lot of works need to be done on this field.