Fake News Detection Project

Submitted by:

Jaideep Pitale

ACKNOWLEDGMENT

Following are the websites which we used during this project for reference and study purpose:

https://towardsdatascience.com/overfitting-vs-underfitting-a-complete-example-d05dd7e19765

https://www.analyticsvidhya.com/blog/2016/01/ridge-lasso-regression-python-complete-tutorial/

INTRODUCTION

- Business Problem Framing

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.

- 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.

- Review of Literature

Following are the websites which we used during this project for reference and study purpose:

https://towardsdatascience.com/overfitting-vs-underfitting-a-complete-example-d05dd7e19765

https://www.analyticsvidhya.com/blog/2016/01/ridge-lasso-regression-python-complete-tutorial/

- Motivation for the Problem Undertaken

  This project is done as part of internship program between Data Trained and Flip Robo Technologies.

  Our goal is to build a model to identify whether a news is fake or not.

Analytical Problem Framing

- Mathematical/ Analytical Modeling of the Problem

There are 6 columns in the dataset. The description of each of the column is given below:

“id”: Unique id of each news article

“headline”: It is the title of the news.

“news”: It contains the full text of the news article

“Unnamed:0”: It is a serial number

“written_by”: It represents the author of the news article

“label”: It tells whether the news is fake (1) or not fake (0).

Dataset has 25116 records in all.

- Data Sources and their formats

There are 6 columns in the dataset. The description of each of the column is given below:

“id”: Unique id of each news article

“headline”: It is the title of the news.

“news”: It contains the full text of the news article

“Unnamed:0”: It is a serial number

“written_by”: It represents the author of the news article

“label”: It tells whether the news is fake (1) or not fake (0).

Dataset has 25116 records in all.

- Data Preprocessing Done

  We have deleted ‘id’, ‘headline’, ’written by’ and ‘Unnamed:0’ columns from data set.

  News column had 39 ‘NULL’ records. We replaced these records with ‘IGNORE TEXT’ text.

  Following is the pre-processing done on ‘news’ column:

  1. Convert all messages to lower case
  2. Replace email addresses with 'email'
  3. Replace URLs with 'web address'
  4. Replace money symbols with 'money symbol' (£ can by typed with ALT key + 156)
  5. Replace 10-digit phone numbers (formats include paranthesis, spaces, no spaces, dashes) with 'phonenumber'
  6. Replace numbers with 'numbr'
  7. Convert text into vectors using TF-IDF

- Data Inputs- Logic- Output Relationships

  There are 6 columns in our dataset including the output column ‘label’. For our model building purpose, we are only using ‘news’ column as input record and ‘label’ as output column.

- State the set of assumptions (if any) related to the problem under consideration

  None.

- Hardware and Software Requirements and Tools Used

  We are using Jupyter Notebook for coding purpose along with Python 3.7.9 version for our model building process.

  We are using below libraries for our model building process:

  ```
  import numpy as np

  import pandas as pd

  import sklearn

  import seaborn as sns

  import matplotlib.pyplot as plt

  from nltk.stem import WordNetLemmatizer

  import nltk

  from nltk.corpus import  stopwords

  import string
  ```

```
from sklearn.model_selection import train_test_split

from sklearn.metrics import accuracy_score, classification_report,
confusion_matrix, f1_score, roc_curve ,roc_auc_score, auc

from sklearn.linear_model import LogisticRegression

from sklearn.model_selection import cross_val_score,GridSearchCV

from sklearn.naive_bayes import MultinomialNB

from sklearn.tree import DecisionTreeClassifier

from sklearn.neighbors import KNeighborsClassifier

from sklearn.ensemble import RandomForestClassifier,
AdaBoostClassifier, GradientBoostingClassifier

from sklearn.naive_bayes import GaussianNB

from sklearn.linear_model import LogisticRegression

from sklearn.svm import SVC

from sklearn.tree import DecisionTreeClassifier

from sklearn.feature_extraction.text import TfidfVectorizer

from sklearn.ensemble import AdaBoostClassifier,
GradientBoostingClassifier

import pickle
```

Model/s Development and Evaluation

- Identification of possible problem-solving approaches (methods)

  'news' column is the one on which we have focused more as based on content of this column output was to be decided.

There is lot of data pre-processing that is being done on this column. Refer section ‘Data Preprocessing Done‘ for all details.

Then NLTK tool is used to process the comments. Stop word is used to remove unnecessary and more frequently used words so as to reduce the population of words for better model building.

WordNetLemmatizer is used to break entire comment into separate meaningful words.

TfidfVectorizer is used to Convert text into vectors using TF-IDF.

- Testing of Identified Approaches (Algorithms)

  Following is the list of algorithms on which this program is tested:

  a. LogisticRegression
  b. DecisionTreeClassifier
  c. KNeighborsClassifier
  d. RandomForestClassifier
  e. GradientBoostingClassifier

.

- Run and Evaluate selected models

```
models = [LogisticRegression(),DecisionTreeClassifier(),KNeighborsClassifier(),RandomForestClassifier()]

for i in models:
    print(i)
    #i.fit(x_train, y_train)
    i.fit(x_train, y_train)
    y_pred_test = i.predict(x_test)
    print('Test accuracy of:',i,'is {}'.format(accuracy_score(y_test,y_pred_test)))
```

```
LogisticRegression()
Test accuracy of: LogisticRegression() is 0.9479166666666666
DecisionTreeClassifier()
Test accuracy of: DecisionTreeClassifier() is 0.8878205128205128
KNeighborsClassifier()
Test accuracy of: KNeighborsClassifier() is 0.6065705128205128
RandomForestClassifier()
Test accuracy of: RandomForestClassifier() is 0.9439102564102564
```

- Key Metrics for success in solving problem under consideration

  Accuracy score and AUC_ROC curve is used measure the performance of the model.

  Cross validation method is also used to check the performance of model which splits the data into training and test mode into 5 parts

  AdaBoostClassifier and GradientBoostingClassifier is also used to enhance the performance of the model.

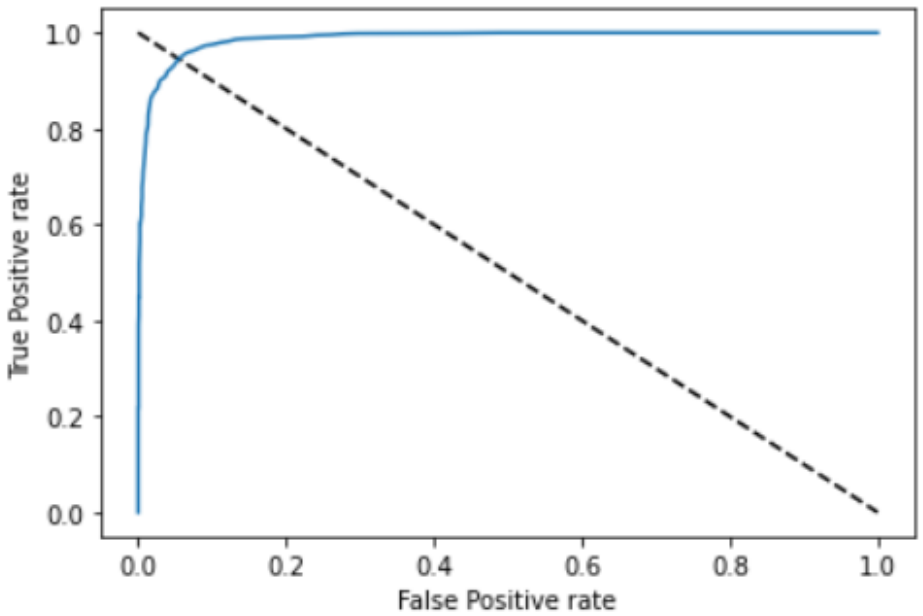
- Visualizations

  Heatmap is used to check for any null values in dataset.

```
sns.heatmap(raw_train_data.isnull()) #Heatmap also ir
```

```
<AxesSubplot:>
```

AUC_ROC curve used to check the performance of the model.



- Interpretation of the Results

Heatmap is used to check for any null values in dataset and also to check the correlation between columns in dataset.

AUC_ROC curve used to check the performance of the model

CONCLUSION

- Key Findings and Conclusions of the Study

From given test dataset, 94.39 % of news are not fake news.

- Learning Outcomes of the Study in respect of Data Science

1. Heatmaps are very useful for data visualization as it depicts the data in a very simple and user-friendly manner.
2. RandomForestClassifier algorithm gives better performance than DecisionTree algorithm.

3. RandomForestClassifier model gives the best performance of model at 94.39%.
4. AUC ROC curve is very useful for checking the performance of model.

- ## Limitations of this work and Scope for Future Work

  In future scope we can work on proper cleaning of data. We can optimize the solution/ performance by making sure that data with outliers and mostly positive data is provided to algorithms. More data visualization methods can be used to describe and explain the data