CPTS 315

COURSE PROJECT REPORT

FAKE NEWS Classification

Jiwoo Kim

# Introduction

Imagine this: You read some articles with tons of different news frame broadcast articles on the internet. You want to pick the best reliable news articles because you are city person. So, you click the link that says, 'breaking news.' We are facing tons of information of data in our daily life, but we need to pick the real news instead of the fake news. This is a fundamental problem that people did not have ability to consider which one is real or fake news because they did not have much information before reading the articles. It sounds like a mild annoyance that you would never want to read fake news article, right? The fake news is dangerous because it seems innocent or just an attempt at fun, a lot of it can be malicious and desperate. The reason we need to judge which news is more reliable because fake news is created to change people's beliefs, attitudes, or perceptions, so they will ultimately change their behavior. This is the reason why I get motivated about it.

With this context in mind, I set out to answer some questions regarding what we were doing to combat this problem with gaining a better understanding of how data mining could be a key tool in detect the fake news. The questions I started out with were

What libraries does python offer for data science?

Which classifier would be the most optimal classification of detecting the fake news?

What machine learning tools should I use to determine the accuracy of the fake news?

How to separate fake news and real news in my code?

I set about attempting to answer these questions through a series of trial and errors in coding and by Googling it. I faced challenge that which machine learning algorithm is the most suitable for my project. I discovered the capabilities of the sklearn python library and decided that I can use this to compare multiple classifiers on the dataset. Also, I found that the pandas library can help me to visualize data over the interface. I decided to try 4 different machine learning classifications to predict the accuracy of fake news, as well as authenticity of fake news from statement.

I chose to compare the four classifiers –

-Linear regression

-Decision tree classification

-Gradient boost classification

-Random forest classification

I then used the different vectorizers in sklearn and paired each of the above with TFIDF Vectorizer. The linear regression with the TFIDF vectorizer will be the best choice so I trained classifier to predict the label of any user input text. The details are elaborated in the sections that follow.

# Data Mining Task

Input Data

From the data.csv file, I measured range by using shape built in function and take 10 rows of each dataset. To do that, I used concat built in function to merge with tail of fake news csv file and true news csv file. After that I created new csv file to put 10 rows of each dataset known as "manual_testing_csv" file. Also, I used to drop built-in function to get the 10 rows of tail in each dataset and merge it together. I split the data into 90:10 to leave some test data for accuracy scoring and validation and take 25 percent data as a test set.

Task Details

First task was to build a working classifier with a high accuracy for the dataset. Then it evolved into a comparison study of the sklearn library's inbuilt classifiers and vectorizer combinations and building a simple application for real time classification.

The first step of the final project plan was reading through the articles about his particular problem and industry verified approaches to solving it. I used some knowledge from articles that what kind of state of the industry today, and what algorithms will be more efficient to classify texts.

Questions:

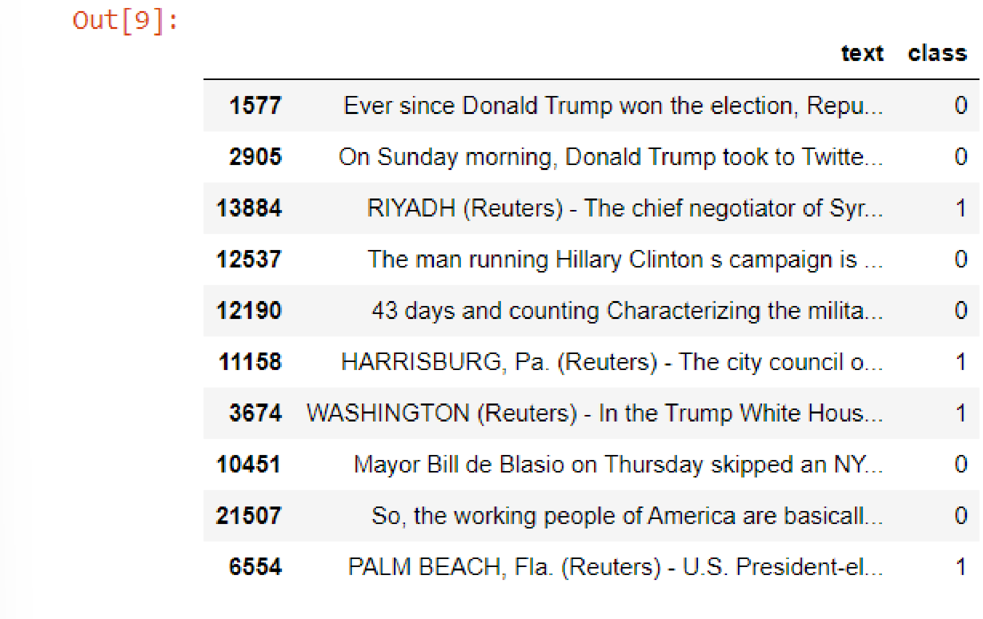
I kept asking myself that some of the broader questions are detailed in the introduction section and my proposal paper. Why was a particular classifier the best and is this the best for classify the fake news by statement? I am curious that how a particular classifier is being when it comes to detecting a text as fake? How can libraries optimize their training to be so quick? What data structures are included the classifiers in sklearn? These questions were what arose from trying to find out how to code and accomplish the tasks mentioned before. So that I could follow to improve my understanding of practical implementation of the algorithms.

Challenges:

Honestly, this project that relies heavily on a framework that I don't quite understand, I found myself struggling a lot in trying to understand and read through the documentations of the libraires, as well as articles. I found that many of python's packages are open source and have very little examples to base my implementation process. However, I found that Sklearn is well documented but the way of using some built-in functions were complicated at first. Another challenge was trying to do more complex things with the dataset when the result accuracy was already quite high. Thus, any improvements can only be meaningful when evaluated accuracy against a larger test set.

Technical Approach

First, I extract random text of articles from test.csv and fake.csv files by using pandas library. After that I drop 10 rows articles in each file, which contains title, text, subject, date, and class. And then I merge it over the new csv file called "manual_testing.csv." This file contains fake news articles for 10 items and true news articles for 10 items. After that, I drop "title", "subject", and "date" columns then shuffle it.



Out[9]:

| | text | class |
|---|---|---|
| **1577** | Ever since Donald Trump won the election, Repu... | 0 |
| **2905** | On Sunday morning, Donald Trump took to Twitte... | 0 |
| **13884** | RIYADH (Reuters) - The chief negotiator of Syr... | 1 |
| **12537** | The man running Hillary Clinton s campaign is ... | 0 |
| **12190** | 43 days and counting Characterizing the milita... | 0 |
| **11158** | HARRISBURG, Pa. (Reuters) - The city council o... | 1 |
| **3674** | WASHINGTON (Reuters) - In the Trump White Hous... | 1 |
| **10451** | Mayor Bill de Blasio on Thursday skipped an NY... | 0 |
| **21507** | So, the working people of America are basicall... | 0 |
| **6554** | PALM BEACH, Fla. (Reuters) - U.S. President-el... | 1 |

Now, I separate x = text section and y = class section then split each csv files according on that.

After that I used TfidVectorizer function because it helps us in dealing with most frequent words. This counts by a measure of how often they appear in our files. I used transform function to transform a given text into a vector based on the frequency of each word that occurs in the entire text.

As for the classifiers, I chose

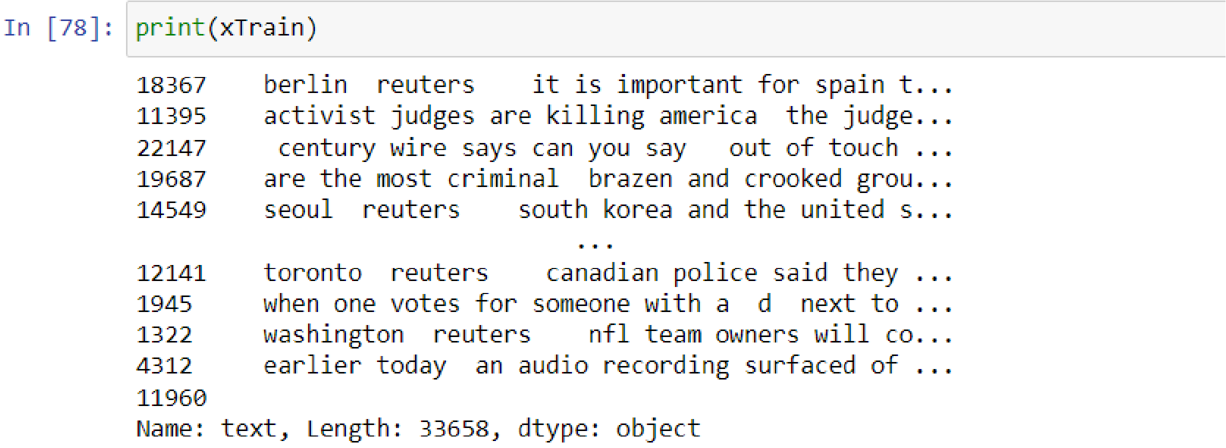
-Logistic Regression

-Decision Tree

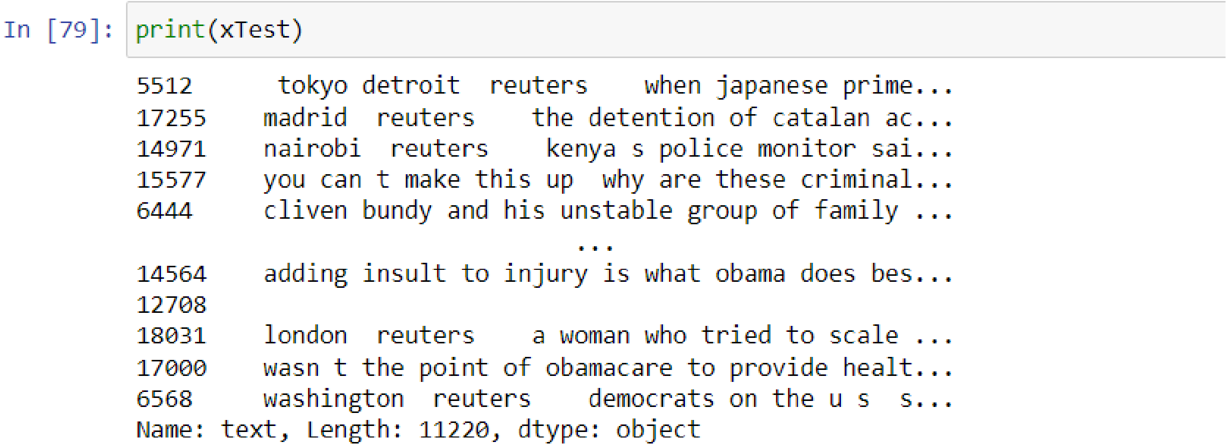
-Gradient Boosting

-Random Forest

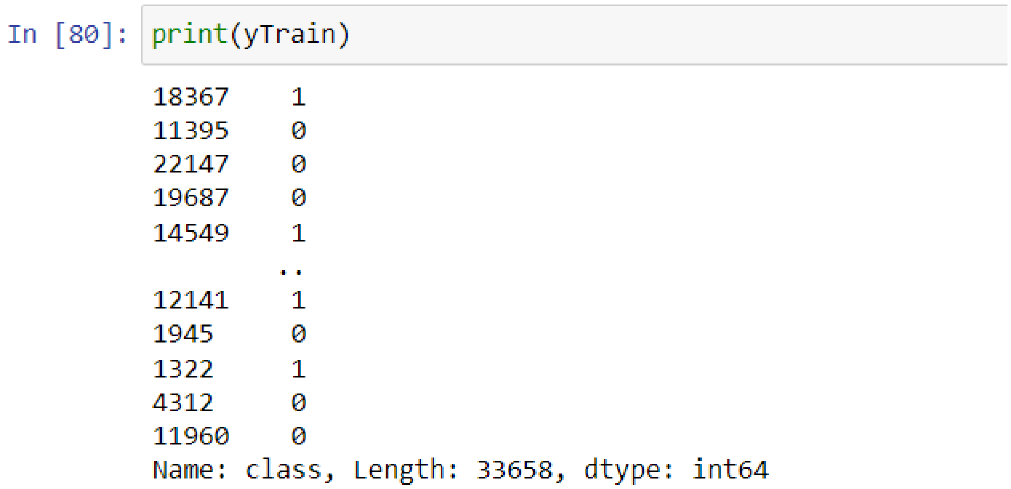
I found that xTrain and xTest values are represented text and yTrain and yTest are represented class shown as below.



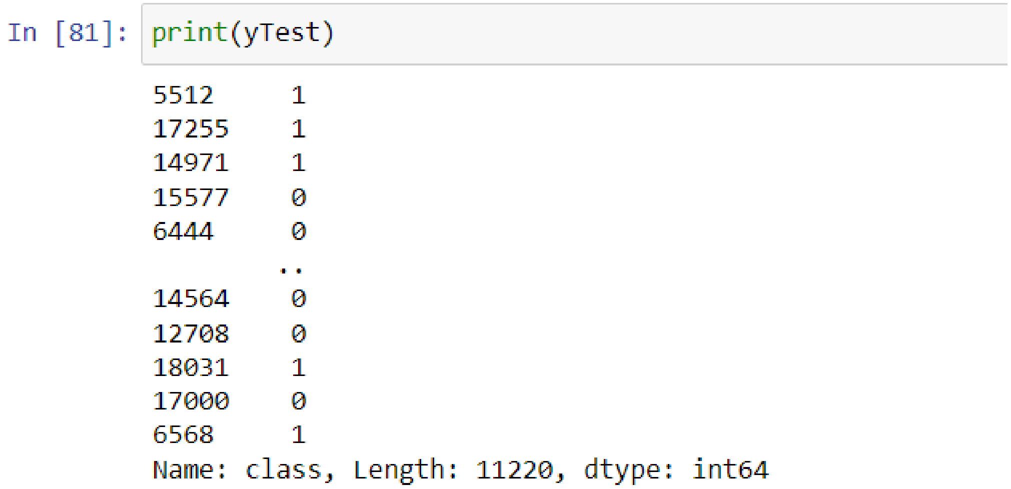
```
In [78]: print(xTrain)

18367    berlin  reuters    it is important for spain t...
11395    activist judges are killing america  the judge...
22147     century wire says can you say   out of touch ...
19687    are the most criminal  brazen and crooked grou...
14549    seoul  reuters    south korea and the united s...
                               ...                        
12141    toronto  reuters    canadian police said they ...
1945     when one votes for someone with a  d  next to ...
1322     washington  reuters    nfl team owners will co...
4312     earlier today  an audio recording surfaced of ...
11960                                                     
Name: text, Length: 33658, dtype: object
```



```
In [79]: print(xTest)

5512      tokyo detroit  reuters    when japanese prime...
17255    madrid  reuters    the detention of catalan ac...
14971    nairobi  reuters    kenya s police monitor sai...
15577    you can t make this up  why are these criminal...
6444     cliven bundy and his unstable group of family ...
                               ...                        
14564    adding insult to injury is what obama does bes...
12708                                                     
18031    london  reuters    a woman who tried to scale ...
17000    wasn t the point of obamacare to provide healt...
6568     washington  reuters    democrats on the u s  s...
Name: text, Length: 11220, dtype: object
```



```
In [80]: print(yTrain)

18367    1
11395    0
22147    0
19687    0
14549    1
        ..
12141    1
1945     0
1322     1
4312     0
11960    0
Name: class, Length: 33658, dtype: int64
```



```
In [81]: print(yTest)

5512     1
17255    1
14971    1
15577    0
6444     0
        ..
14564    0
12708    0
18031    1
17000    0
6568     1
Name: class, Length: 11220, dtype: int64
```

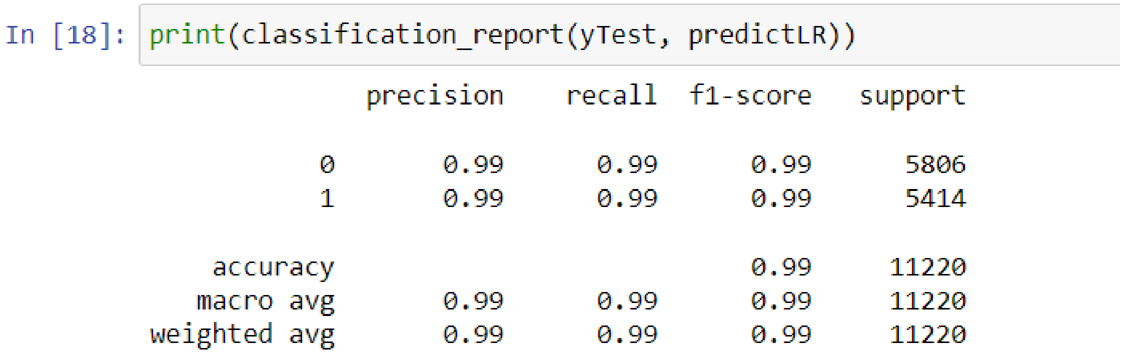
As images above, Now I can use text classification methods to get the accuracy between prediction and target values by vectorizing it.

# Evaluation Methodology

The input data for this project came from the FAKE NEWS Detection Dataset in Kaggle. The entire dataset contains a total 30000 articles extracted from various research sources on the internet. The accuracy of my application was quantitatively evaluated by the test data set's accuracy scores. I evaluated my learning goals based on what I can find answers from question that I asked. I enjoyed the study how words can classify fake statement or true statement by using machine learning classifications. The real-world perspective is that this is important to classify between fake and real because I found that they did not provide reliable resources as well as publication dates.

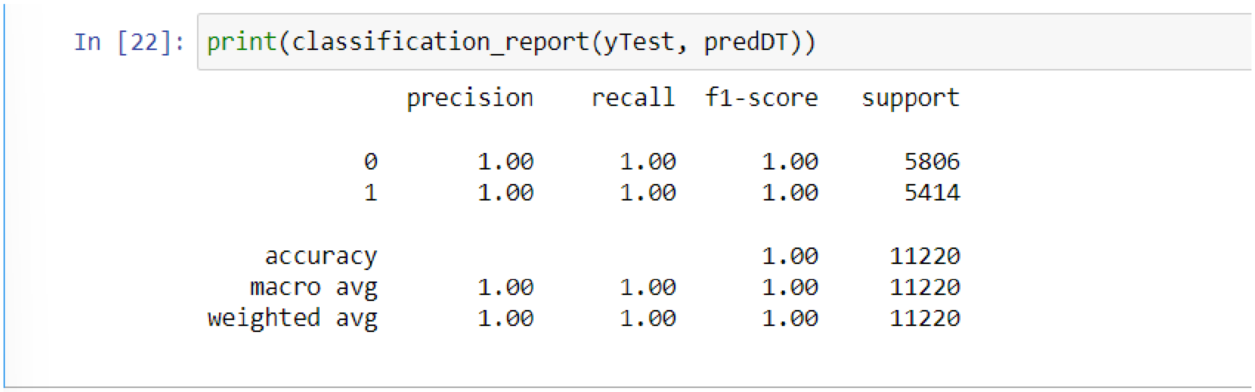
# Results and Discussion

I attained the values by using classification report built-in function which compare between target value and prediction value. Table are shown as below.

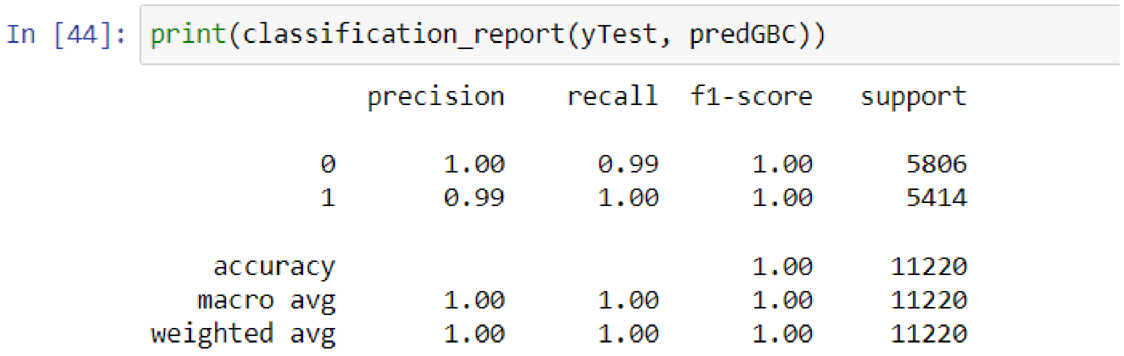


```
In [18]: print(classification_report(yTest, predictLR))

              precision    recall  f1-score   support

           0       0.99      0.99      0.99      5806
           1       0.99      0.99      0.99      5414

    accuracy                           0.99     11220
   macro avg       0.99      0.99      0.99     11220
weighted avg       0.99      0.99      0.99     11220
```

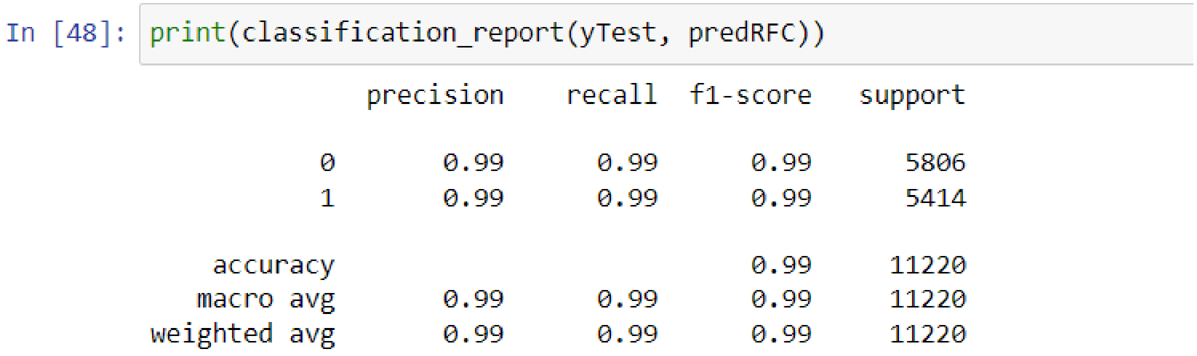
Same processes are going with rest of the classifiers as shown above.



```
In [22]: print(classification_report(yTest, predDT))

              precision    recall  f1-score   support

           0       1.00      1.00      1.00      5806
           1       1.00      1.00      1.00      5414

    accuracy                           1.00     11220
   macro avg       1.00      1.00      1.00     11220
weighted avg       1.00      1.00      1.00     11220
```

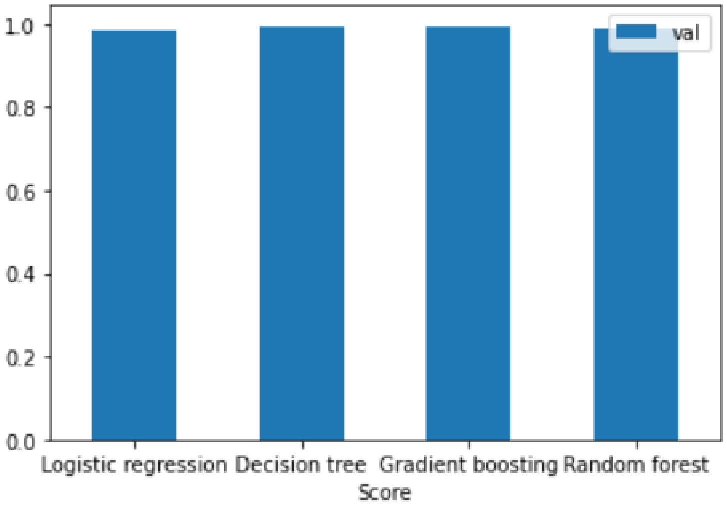


```
In [44]: print(classification_report(yTest, predGBC))

              precision    recall  f1-score   support

           0       1.00      0.99      1.00      5806
           1       0.99      1.00      1.00      5414

    accuracy                           1.00     11220
   macro avg       1.00      1.00      1.00     11220
weighted avg       1.00      1.00      1.00     11220
```

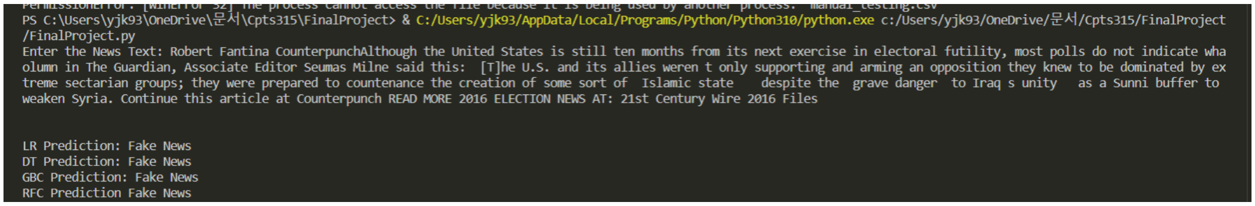


```
In [48]: print(classification_report(yTest, predRFC))

              precision    recall  f1-score   support

           0       0.99      0.99      0.99      5806
           1       0.99      0.99      0.99      5414

    accuracy                           0.99     11220
   macro avg       0.99      0.99      0.99     11220
weighted avg       0.99      0.99      0.99     11220
```

I found that most of the classifications had a solid accuracy. Now I can evaluate the accuracy by using machine learning classifications like I mentioned above. I used 4 different machine learning classifications to improve the accuracy of the score. I created bar to more visualize which machine learning classification will fit the best among the many data mining classifications.



This graph will show that decision tree classification and Gradient boosting classification are the best algorithm to improve accuracy and efficiency.



```
PermissionError: [WinError 32] The process cannot access the file because it is being used by another process: manual_testing.csv
PS C:\Users\yjk93\OneDrive\문서\Cpts315\FinalProject> & C:/Users/yjk93/AppData/Local/Programs/Python/Python310/python.exe c:/Users/yjk93/OneDrive/문서/Cpts315/FinalProject
/FinalProject.py
Enter the News Text: Robert Fantina CounterpunchAlthough the United States is still ten months from its next exercise in electoral futility, most polls do not indicate wha
olumn in The Guardian, Associate Editor Seumas Milne said this:  [T]he U.S. and its allies weren t only supporting and arming an opposition they knew to be dominated by ex
treme sectarian groups; they were prepared to countenance the creation of some sort of  Islamic state   despite the  grave danger  to Iraq s unity   as a Sunni buffer to
weaken Syria. Continue this article at Counterpunch READ MORE 2016 ELECTION NEWS AT: 21st Century Wire 2016 Files


LR Prediction: Fake News
DT Prediction: Fake News
GBC Prediction: Fake News
RFC Prediction Fake News
```

This is my compiler program result looks like. I simply user to type the text of the news articles then there are four different machine learning classifications will tell this is fake news or not. Since, I enter the fake news text article, it prints out this is a fake news.

Some of the questions I researched were:

1. Why is a decision tree classification most efficient for text classification?

   It is because the cost of using the tree is logarithmic in the number of data points that can train the tree. Also, it can perform well even if its assumptions are somewhat violated by the true model from which the data were generated.

2. What are the disadvantages of decision trees including?

   Decision tree learners can create over-complex trees that do not generalize the data well which is overfitting. So that setting the minimum number of samples required at a leaf node or setting the maximum depth of the tree are necessary to avoid this problem. It can also generate complex if you have small variations. Also, there are concepts that are hard to learn because decision tree does not express easily, such as XOR, or multiplexer problems.

3. What is the state of the art in text classification for fake news and what would be the better performance among those classifications?
   For the state of the art in text classification would be Long short-term memory, Attention Mechanism, IndRNN, Attention-Based Bidirection LSTM, Hierarchical Attention Networks for Text Classification, Adversarial Training Methods for Supervised Text Classification, Convolutional Neural Networks for Sentence Classification and RMDL. All these models have comparable performances.

# Lesson Learned

While I was doing this project myself, I found that data mining is one of the most important concepts in my industry, as well as our world. The reason behind is that our world is highly dependent on data which needs to be polish it well by IT engineers. I learned that if we can program how to learn something to computer, we can get better outcomes as well as performance in smart ways. When I pick the different types of machine learning classifications, I did not know which one is the most efficient classifiers until I programed.

It would have been great to try out the classifier on a larger data set because if you more data then it is more likely to get accurate data at the end. I challenge that how to deduct outcomes with your own guidelines and measures of success for a project and be able to explain what I have done in my project by words. I assume that it can encourage to move forward a lot quickly and be able to communicate with coworkers about what I am capable of and what kind of logic needs to keep in mind to resolve this challenge in the future.

Overall, I think the project gave me an experience I can draw on at my future career and job. It was a good knowledge to know how to approach the challenge from zero and be able to make it outcomes by training and testing it. Now that understanding what data is telling us is the most state-of-the-art technology you can have.

# Bibliography

Simon, L. (2018). Fake News Detection Using Machine Learning.

Retrieved from https://matheo.uliege.be/bitstream/2268.2/8416/1/s134450_fake_news_detection_using_machine_learning.pdf