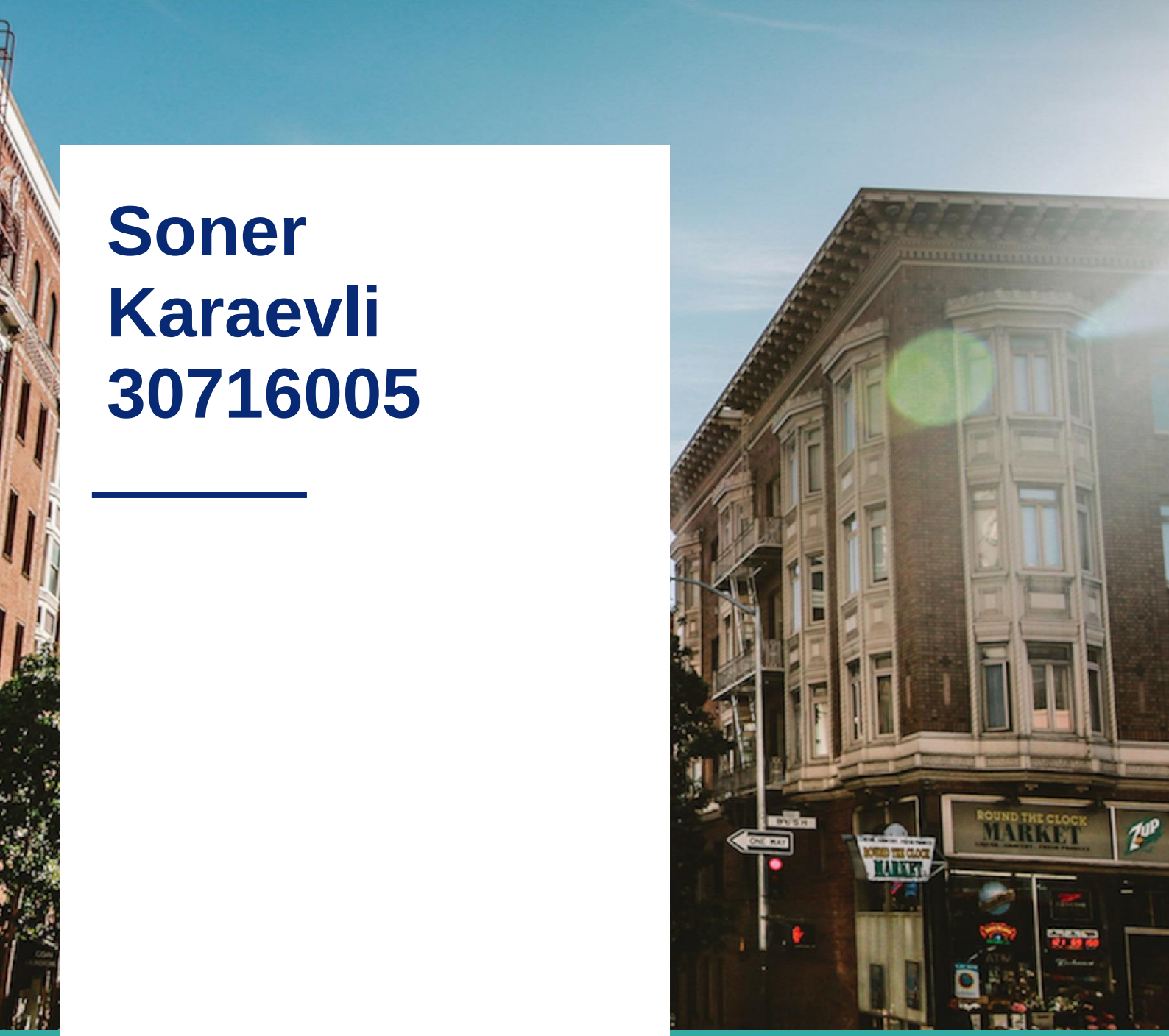
Soner Karaevli
30716005



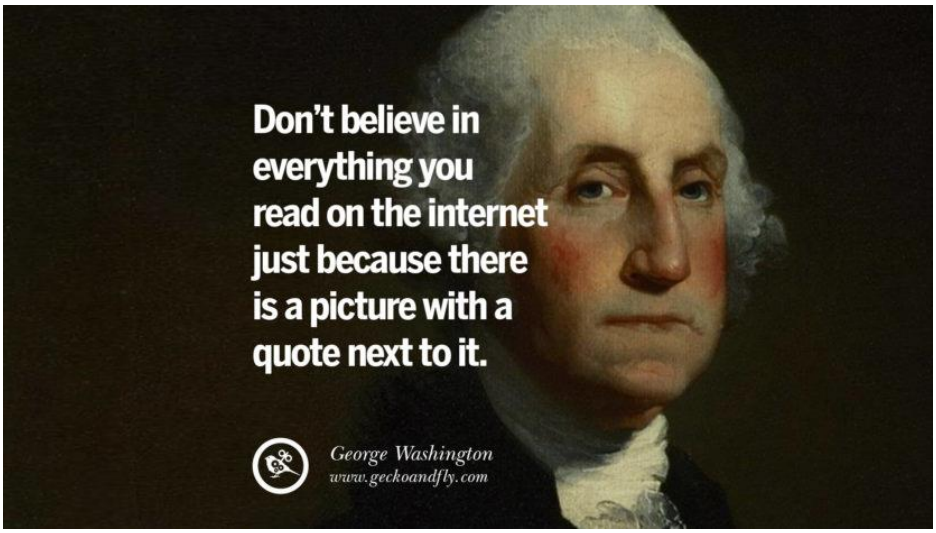
Fake News Detection

Data Mining Project

a) Introduction

About the project

Nowadays, social media is very significant for the communication. Since, 2000's the social media apps are developed and they access the billions of people.
In this world where information spreads so quickly through social media, fake news can spread very quickly. According to researchers, many social events can be created by using fake news today.

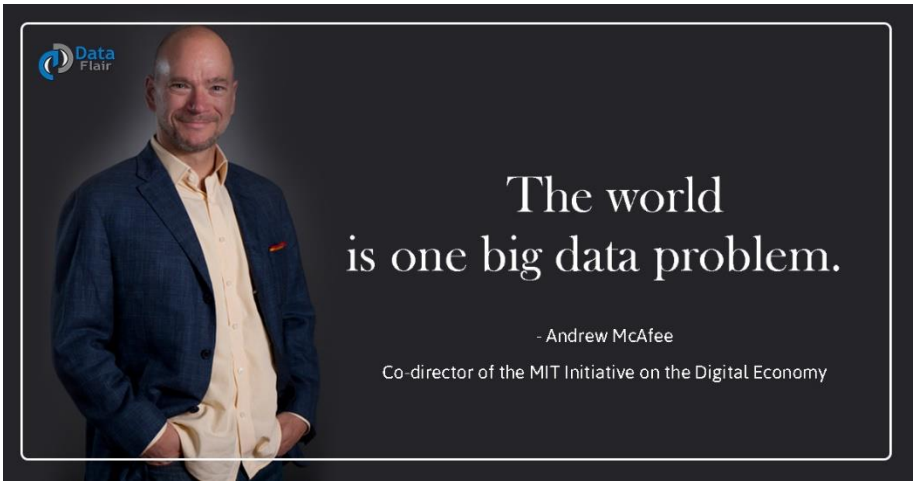


For example, the result of a presidential election can be changed with fake news. These consequences may not only be from political events. There can even be a genocide with fake news. Large masses can be incited to crime, and when you look at this picture, all kinds of bad events and crimes can be done with fake news.
Therefore, fake news has a big impact on society. Most social media apps have developed various models to prevent the spread of fake news. In this project, we will develop a model to detect fake news spread on social media.

"Alternative facts and fake news are just other names for propaganda"
— Johnny Corn

b) Literature

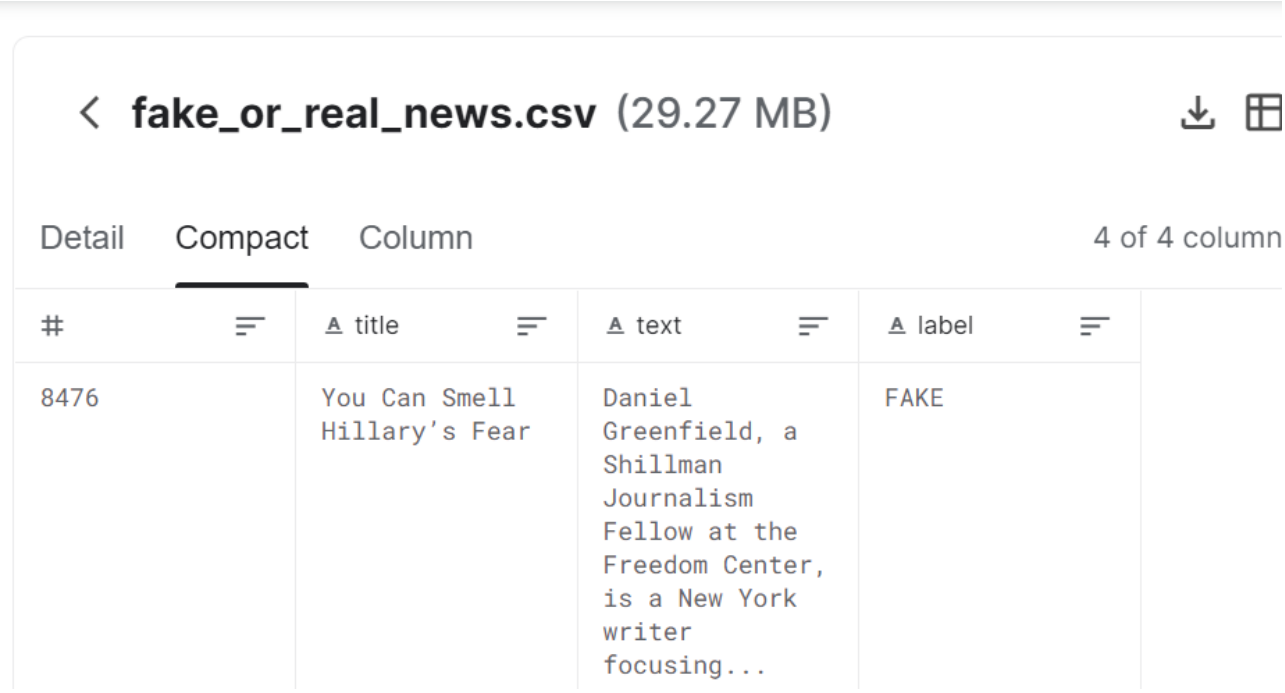
Fake news has been a problem for many years. In ancient times, various solutions and methodologies were developed to solve this problem. Today, it is believed that this issue will be solved with the developments in the "IT" sector.



Data Mining will be the solution source of this problem today. Many data scientists and software developers have developed various libraries and frameworks to solve this problem. They haven't been fully successful because Data Science needs much more improvement. Still, the results are satisfactory nowadays.

c) Structure of the solution proposed

The solution is actually to transfer the data you receive with data mining methods to a model and train it. In this project, I got the data set from kaggle. The data set consists of 4 columns. The first column consists of the ID numbers of the news. The second column consists of the headlines of the news. The third column consists of the content of the news. And the fourth column states that the news is true or false. The data set consists mostly of political news and has been collected from twitter and other websites.



- Python is very strong and useful programming language for the Data Science.

That's why, i used the Python. And first of all, we need the get data.
In my project, to getting and reading data, i used the pandas library.

```
# csv dosyalarını okumak için gereken kütüphane
import pandas as pd
```

- After that, we need to get data into two parts, it is test and train. So i used the "train_test_split" function in the sklearn library. Scikit-learn or Sklearn is a Python-based library for building machine learning models.

```
#data seti bölmek için kullanılan kütüphane
from sklearn.model_selection import train_test_split
```

- Than we create the pipeline. Pipeline is the tool or object that you can work on the data and run multiple text mining functions at the same time(it's mine comment).

```
#pipeline oluşturmak için kullanılan kütüphane
from sklearn.pipeline import Pipeline
```

- TfidfVectorizer is transforms text to feature vectors that can be used as input to estimator. For the stopword, we need to add this library.

```
#verileri serileştirmek için kullanılan kütüphaneyi ekliyorum
from sklearn.feature_extraction.text import TfidfVectorizer
```

- Added the naive bayes library. We will use this approach for the project.

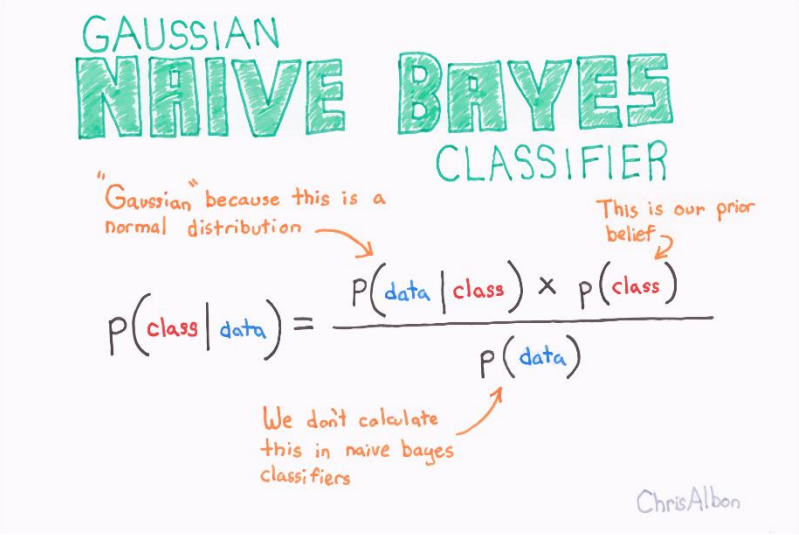
```
# Naive Bayes Kütüphanelerini ekliyorum
from sklearn.naive_bayes import MultinomialNB, GaussianNB, BernoulliNB
```

d) How to use software?

- The project was made using the python programming language and libraries. So project can work with a dataset and computer that can compile Python codes. These are the minimum requirements for the project to work.

e) RFC for the Framework Algorithm

- I applied classification to classify the data. So , new information can be categorized by checking at previous information. I used the Naive Bayes approach to classify.
- Bayes' Theorem is the conditional probability calculation formula introduced by Thomas Bayes in 1812.



- Naive-Bayes calculates probabilities for all inputs and classifies according to high probability. And than, it's categorizes according to previously taught data. It's a supervised learning.

Now, i am going to describe my project how to work step by step.

- First I read and get my dataset and assign the "text" and "label" columns in the data set to the variables.

```
# csv dosyalarını okumak için gereken kütüphane
import pandas as pd

# Dataseti içe aktarıyoruz, text ve label'ı  değişkene atıyorum.
news = pd.read_csv('data.csv')
X = news['text']
y = news['label']
```

- Than i divided into two parts "test" and "train" to train dataset.

```
#data seti bölmek için kullanılan kütüphane
from sklearn.model_selection import train_test_split
# Dataseti eğitmek için test ve train şeklinde bölüyorum
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.22)
```

- Than i create pipeline that name of "seri". I am get out the stopwords in my dataset and applying the Multinomial or Bernoulli Naive Bayes approach.

```
# TfidfVectorizer kütüphanesi data'yı seri hale getiriyor.
# Pipeline oluşturuyorum ve içinde gereksiz kelimeleri stopwords ile çıkarıyorum. Daha sonra Naive Bayes uyguluyorum.
seri = Pipeline([('tfidf', TfidfVectorizer(stop_words='english')),
                 #  ('bernoli', BernoulliNB()),
                    ('nbmodel1',  MultinomialNB()),
                  ])
```

- Than i train the machine and predict for test data.

```
# Makineyi Eğitiyorum
seri.fit(X_train, y_train)

# Test verileri için tahmin yapiyorum
tahmin = seri.predict(X_test)
```

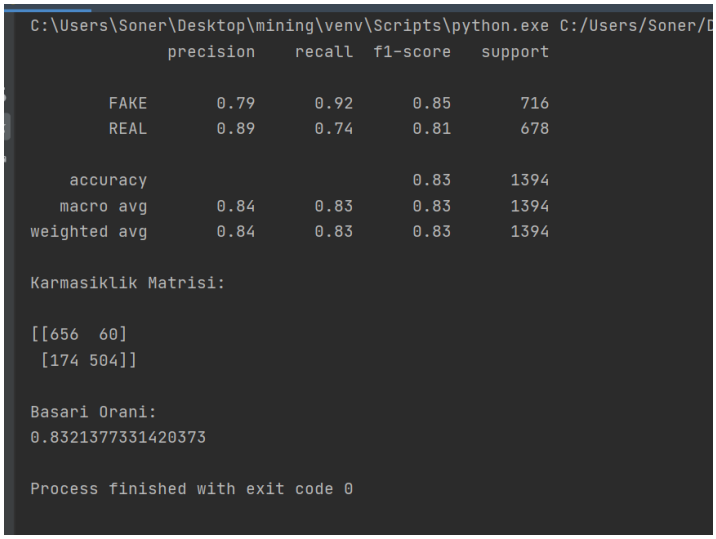
- Finally, I added the library "classification_report" to check the performance of my machine.

```
#performansı kontrol etmek için kullanılan kütüphaneyi ekliyorum.
from sklearn.metrics import classification_report, confusion_matrix
# Modelimizin performansını kontrol ediyorum.
print(classification_report(y_test, tahmin))
```

f) Runtime Examples and Results

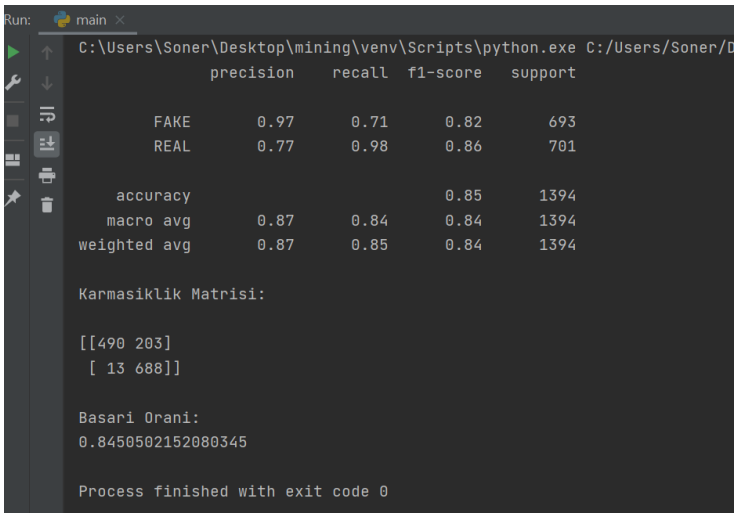
So as i told you, i used the Naive Bayes approach to classify. And for the classify, i used two Naive Bayes function for my machine.

For the Beurnolli Approach:



```
C:\Users\Soner\Desktop\mining\venv\Scripts\python.exe C:/Users/Soner/D
              precision    recall  f1-score   support

        FAKE       0.79      0.92      0.85       716
        REAL       0.89      0.74      0.81       678

    accuracy                           0.83      1394
   macro avg       0.84      0.83      0.83      1394
weighted avg       0.84      0.83      0.83      1394

Karmasiklik Matrisi:

[[656  60]
 [174 504]]

Basari Orani:
0.8321377331420373

Process finished with exit code 0
```

For the Multinominal Approach:



```
Run:  main
C:\Users\Soner\Desktop\mining\venv\Scripts\python.exe C:/Users/Soner/D
              precision    recall  f1-score   support

        FAKE       0.97      0.71      0.82       693
        REAL       0.77      0.98      0.86       701

    accuracy                           0.85      1394
   macro avg       0.87      0.84      0.84      1394
weighted avg       0.87      0.85      0.84      1394

Karmasiklik Matrisi:

[[490 203]
 [ 13 688]]

Basari Orani:
0.8450502152080345

Process finished with exit code 0
```

At the results, as you seen, the accuracy scores are so similar for thats functions. But for the other datasets, the results can be change.

g) Future Work

Consequently data science is still in its infancy. In this project, we developed a machine that detects fake news using data science, but the limits of what can be done are entirely up to our imagination.
New frameworks, libraries to be written and new open source projects to be made will help the development of data science over time.
Especially the developing "IT" sector and the algorithms to be created with machine learning models will appear not only with social media but in many areas of our lives.

h) References

- For the naive bayes theorem: https://tr.wikipedia.org/wiki/Naive_Bayes_s%C4%B1n%C4%B1fland%C4%B1r%C4%B1c%C4%B1s%C4%B1

- For the dataset: https://www.kaggle.com/hassanamin/textdb3

- For the scikit-learn libraries: https://scikit-learn.org/stable/modules/naive_bayes.html