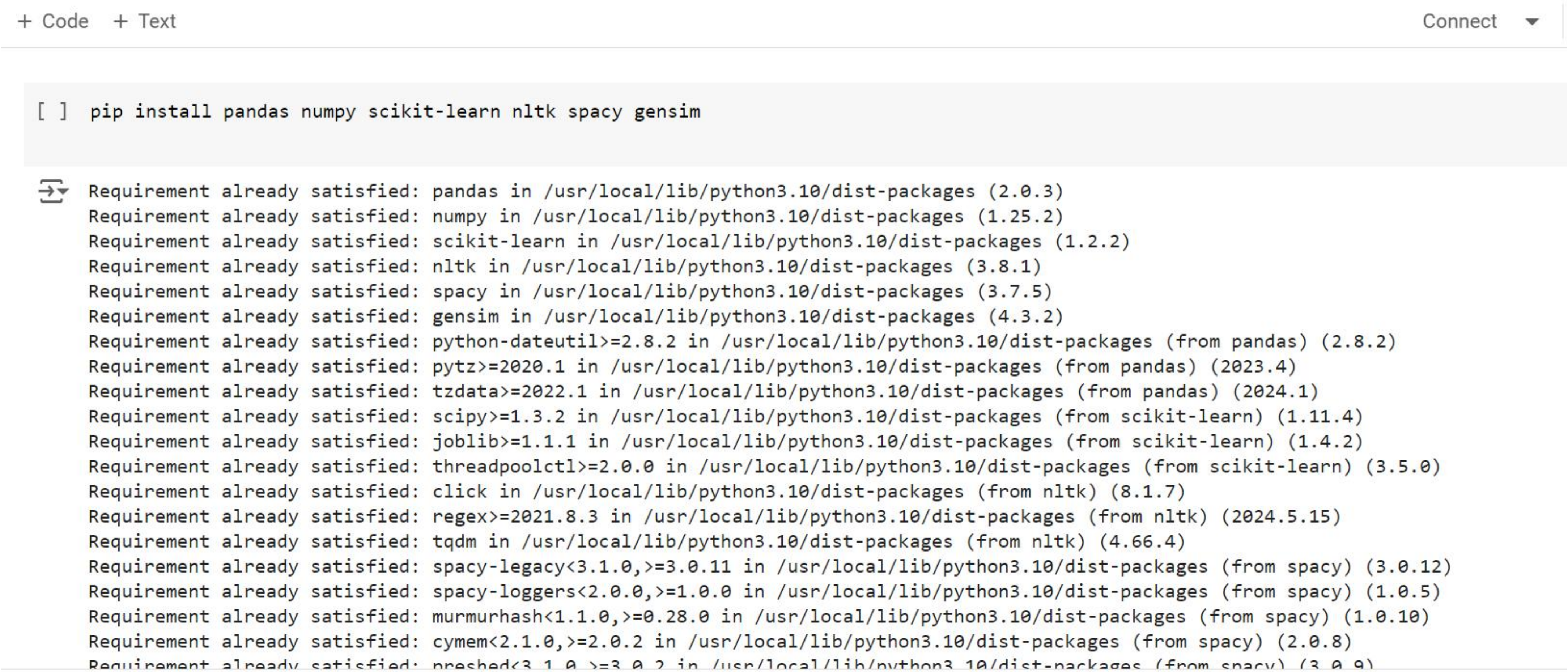
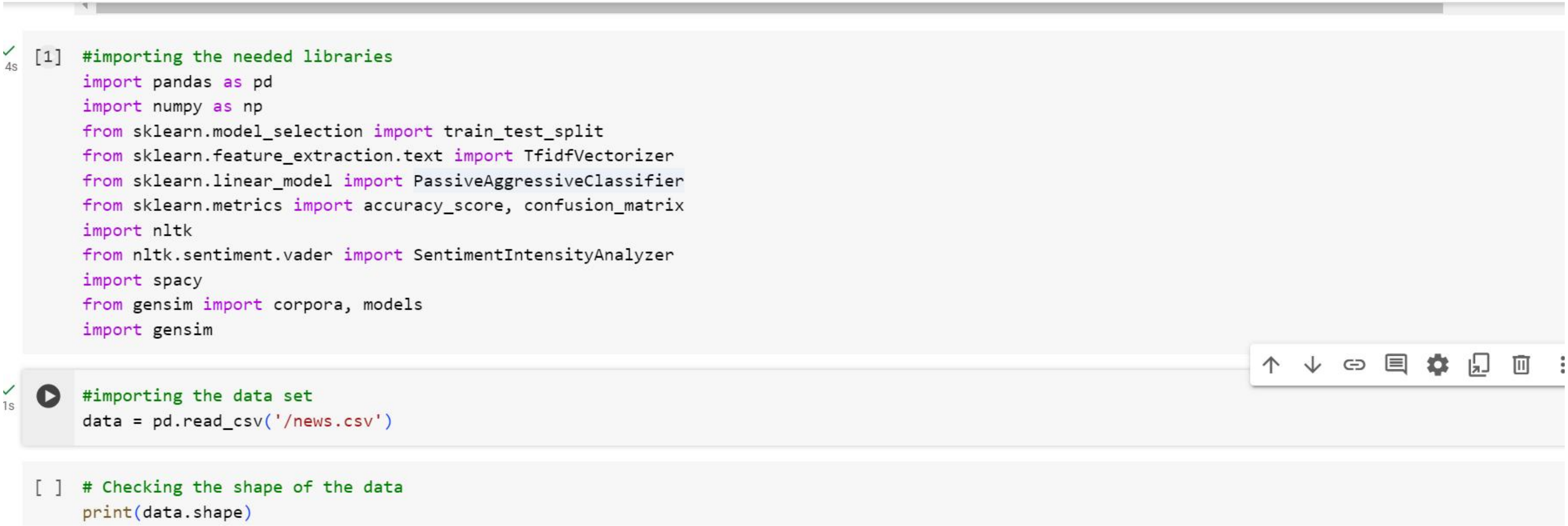


```
pip install pandas numpy scikit-learn nltk spacy gensim
```

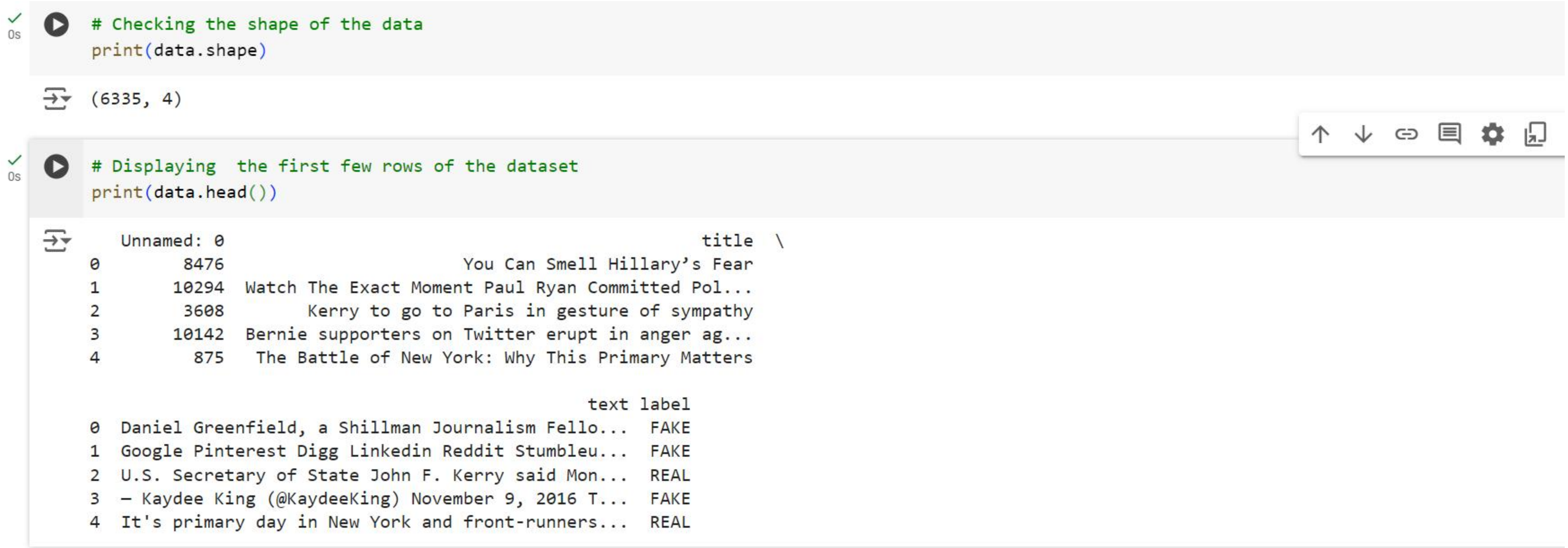
```
Requirement already satisfied: pandas in /usr/local/lib/python3.10/dist-packages (2.0.3)
Requirement already satisfied: numpy in /usr/local/lib/python3.10/dist-packages (1.25.2)
Requirement already satisfied: scikit-learn in /usr/local/lib/python3.10/dist-packages (1.2.2)
Requirement already satisfied: nltk in /usr/local/lib/python3.10/dist-packages (3.8.1)
Requirement already satisfied: spacy in /usr/local/lib/python3.10/dist-packages (3.7.5)
Requirement already satisfied: gensim in /usr/local/lib/python3.10/dist-packages (4.3.2)
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.10/dist-packages (from pandas) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.10/dist-packages (from pandas) (2023.4)
Requirement already satisfied: tzdata>=2022.1 in /usr/local/lib/python3.10/dist-packages (from pandas) (2024.1)
Requirement already satisfied: scipy>=1.3.2 in /usr/local/lib/python3.10/dist-packages (from scikit-learn) (1.11.4)
Requirement already satisfied: joblib>=1.1.1 in /usr/local/lib/python3.10/dist-packages (from scikit-learn) (1.4.2)
Requirement already satisfied: threadpoolctl>=2.0.0 in /usr/local/lib/python3.10/dist-packages (from scikit-learn) (3.5.0)
Requirement already satisfied: click in /usr/local/lib/python3.10/dist-packages (from nltk) (8.1.7)
Requirement already satisfied: regex>=2021.8.3 in /usr/local/lib/python3.10/dist-packages (from nltk) (2024.5.15)
Requirement already satisfied: tqdm in /usr/local/lib/python3.10/dist-packages (from nltk) (4.66.4)
Requirement already satisfied: spacy-legacy<3.1.0,>=3.0.11 in /usr/local/lib/python3.10/dist-packages (from spacy) (3.0.12)
Requirement already satisfied: spacy-loggers<2.0.0,>=1.0.0 in /usr/local/lib/python3.10/dist-packages (from spacy) (1.0.5)
Requirement already satisfied: murmurhash<1.1.0,>=0.28.0 in /usr/local/lib/python3.10/dist-packages (from spacy) (1.0.10)
Requirement already satisfied: cymem<2.1.0,>=2.0.2 in /usr/local/lib/python3.10/dist-packages (from spacy) (2.0.8)
Requirement already satisfied: preshed<3.1.0,>=3.0.2 in /usr/local/lib/python3.10/dist-packages (from spacy) (3.0.9)
```



```
#importing the needed libraries
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.linear_model import PassiveAggressiveClassifier
from sklearn.metrics import accuracy_score, confusion_matrix
import nltk
from nltk.sentiment.vader import SentimentIntensityAnalyzer
import spacy
from gensim import corpora, models
import gensim
```

```
#importing the data set
data = pd.read_csv('/news.csv')
```

```
# Checking the shape of the data
print(data.shape)
```



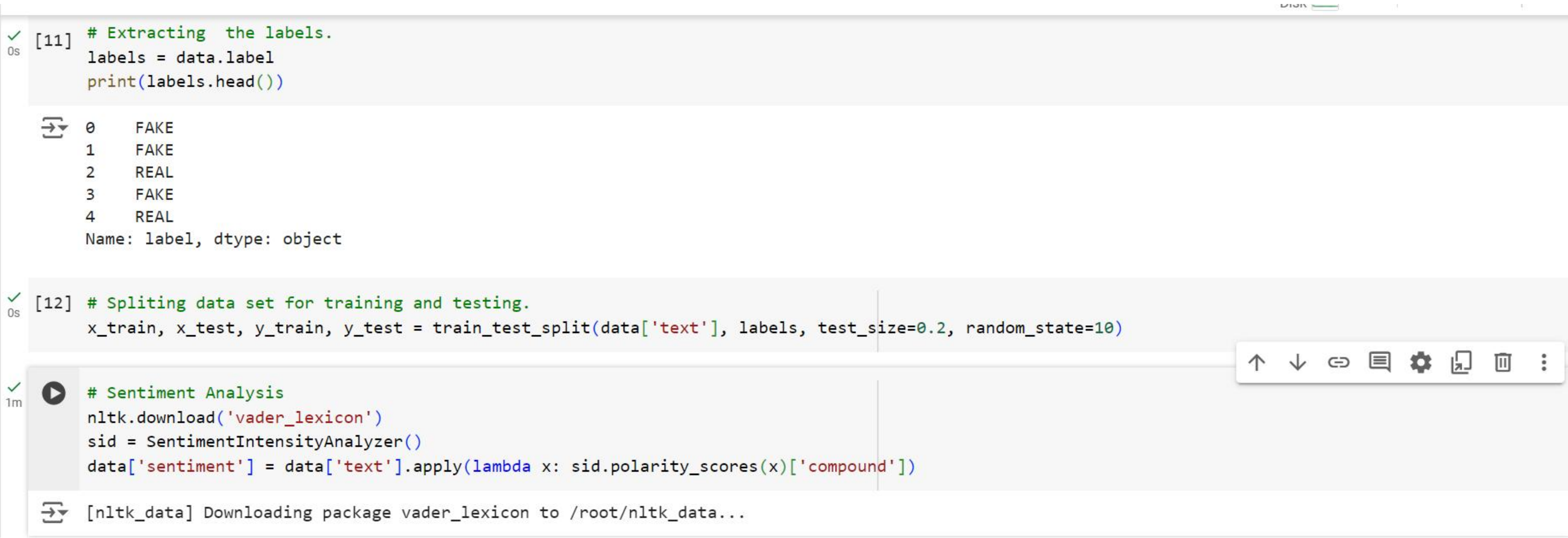
```
# Checking the shape of the data
print(data.shape)
```

```
(6335, 4)
```

```
# Displaying  the first few rows of the dataset
print(data.head())
```

```
   Unnamed: 0                                              title  \
0        8476                       You Can Smell Hillary’s Fear   
1       10294  Watch The Exact Moment Paul Ryan Committed Pol...   
2        3608        Kerry to go to Paris in gesture of sympathy   
3       10142  Bernie supporters on Twitter erupt in anger ag...   
4         875   The Battle of New York: Why This Primary Matters   

                                                text label  
0  Daniel Greenfield, a Shillman Journalism Fello...  FAKE  
1  Google Pinterest Digg Linkedin Reddit Stumbleu...  FAKE  
2  U.S. Secretary of State John F. Kerry said Mon...  REAL  
3  — Kaydee King (@KaydeeKing) November 9, 2016 T...  FAKE  
4  It's primary day in New York and front-runners...  REAL  
```



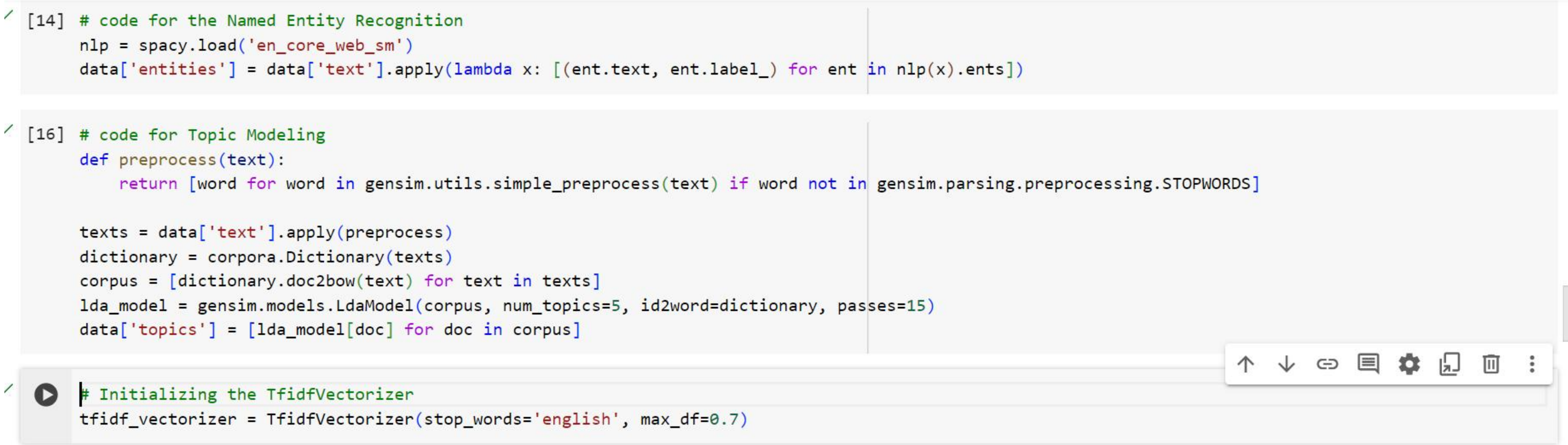
```
[11] # Extracting  the labels.
     labels = data.label
     print(labels.head())
```

```
0    FAKE
1    FAKE
2    REAL
3    FAKE
4    REAL
Name: label, dtype: object
```

```
[12] # Spliting data set for training and testing.
     x_train, x_test, y_train, y_test = train_test_split(data['text'], labels, test_size=0.2, random_state=10)
```

```
# Sentiment Analysis
nltk.download('vader_lexicon')
sid = SentimentIntensityAnalyzer()
data['sentiment'] = data['text'].apply(lambda x: sid.polarity_scores(x)['compound'])
```

```
[nltk_data] Downloading package vader_lexicon to /root/nltk_data...
```

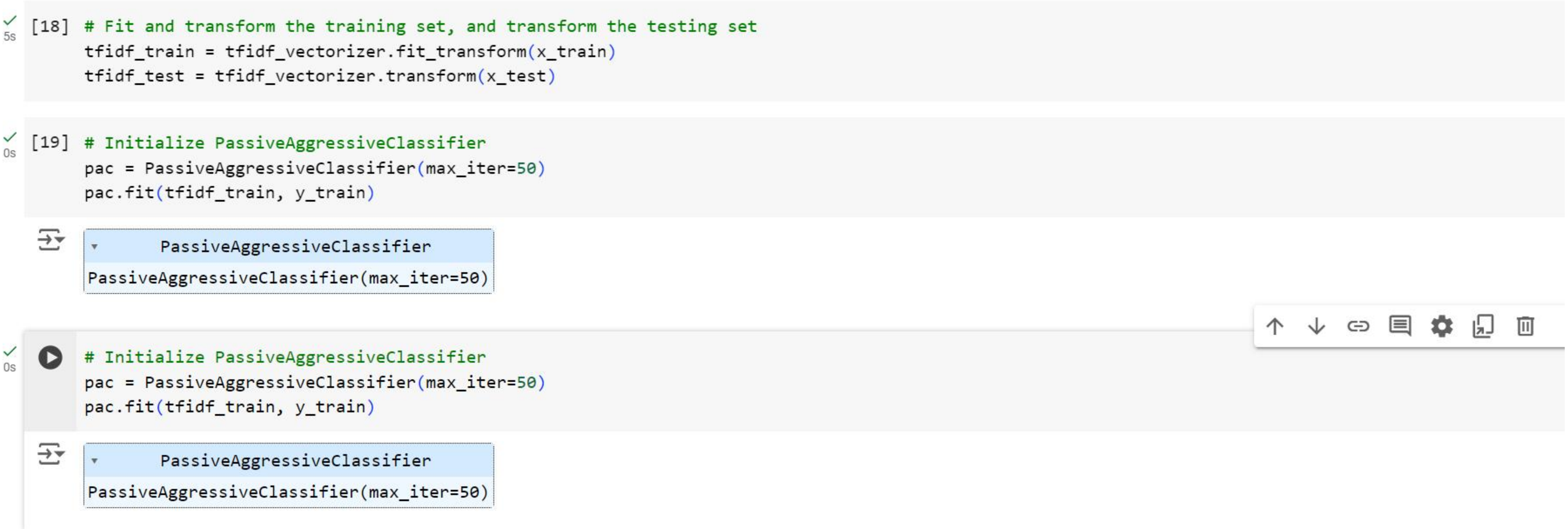


```
[14] # code for the Named Entity Recognition
     nlp = spacy.load('en_core_web_sm')
     data['entities'] = data['text'].apply(lambda x: [(ent.text, ent.label_) for ent in nlp(x).ents])
```

```
[16] # code for Topic Modeling
     def preprocess(text):
         return [word for word in gensim.utils.simple_preprocess(text) if word not in gensim.parsing.preprocessing.STOPWORDS]

     texts = data['text'].apply(preprocess)
     dictionary = corpora.Dictionary(texts)
     corpus = [dictionary.doc2bow(text) for text in texts]
     lda_model = gensim.models.LdaModel(corpus, num_topics=5, id2word=dictionary, passes=15)
     data['topics'] = [lda_model[doc] for doc in corpus]
```

```
# Initializing the TfidfVectorizer
tfidf_vectorizer = TfidfVectorizer(stop_words='english', max_df=0.7)
```



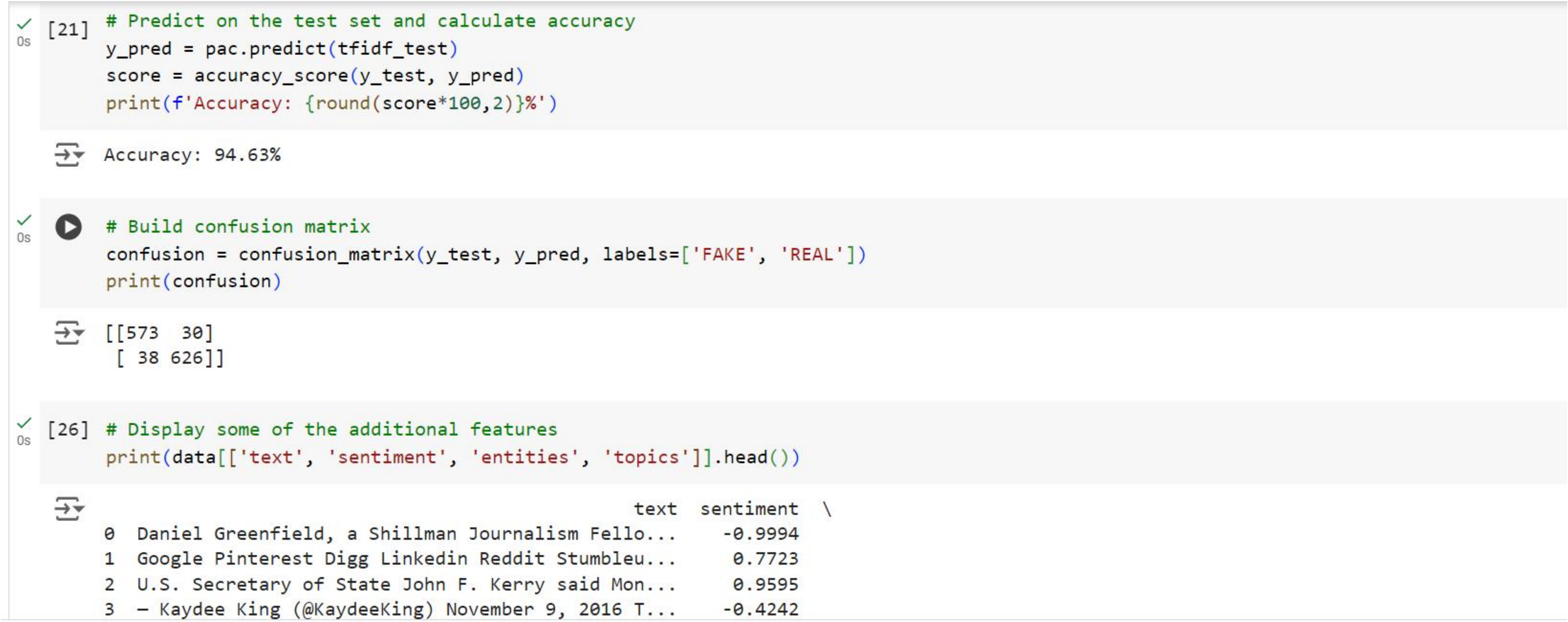
```
✓ [18] # Fit and transform the training set, and transform the testing set
5s     tfidf_train = tfidf_vectorizer.fit_transform(x_train)
       tfidf_test = tfidf_vectorizer.transform(x_test)
```

```
✓ [19] # Initialize PassiveAggressiveClassifier
0s     pac = PassiveAggressiveClassifier(max_iter=50)
       pac.fit(tfidf_train, y_train)
```

```
▾ PassiveAggressiveClassifier
PassiveAggressiveClassifier(max_iter=50)
```

```
✓      # Initialize PassiveAggressiveClassifier
0s     pac = PassiveAggressiveClassifier(max_iter=50)
       pac.fit(tfidf_train, y_train)
```

```
▾ PassiveAggressiveClassifier
PassiveAggressiveClassifier(max_iter=50)
```



```
# Predict on the test set and calculate accuracy
y_pred = pac.predict(tfidf_test)
score = accuracy_score(y_test, y_pred)
print(f'Accuracy: {round(score*100,2)}%')
```

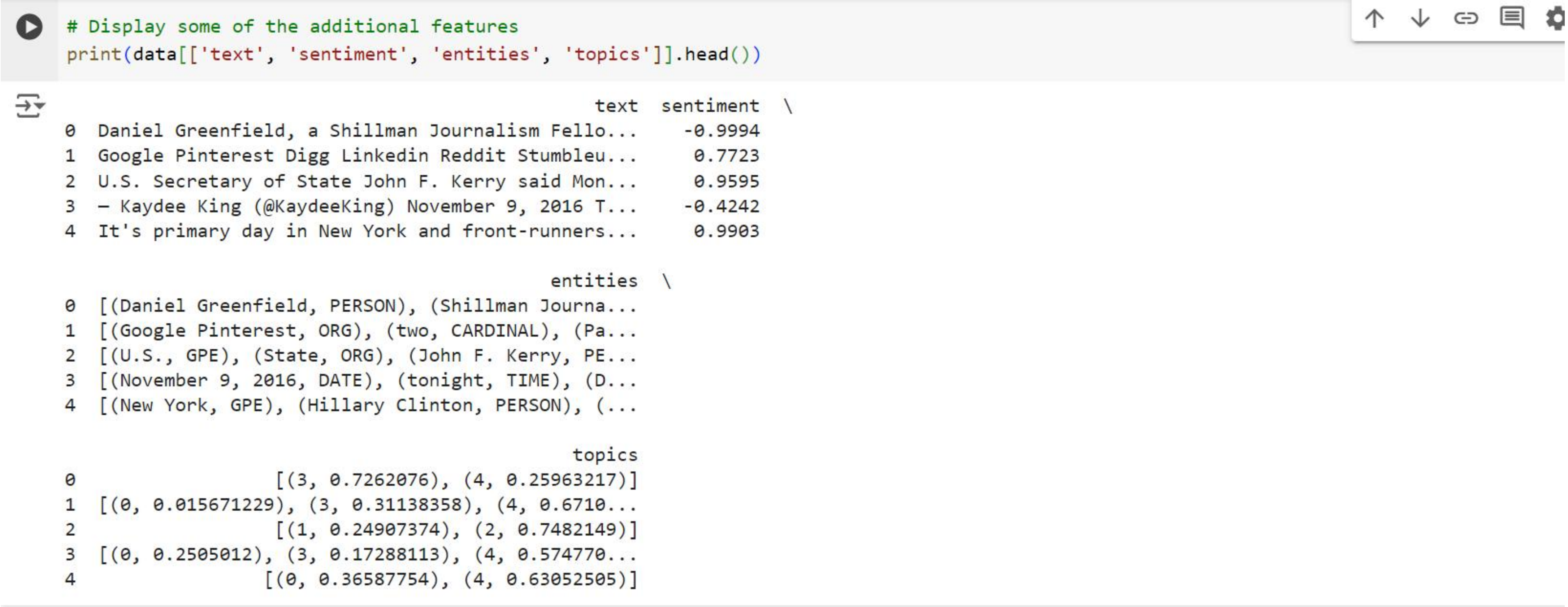
```
Accuracy: 94.63%
```

```
# Build confusion matrix
confusion = confusion_matrix(y_test, y_pred, labels=['FAKE', 'REAL'])
print(confusion)
```

```
[[573  30]
 [ 38 626]]
```

```
# Display some of the additional features
print(data[['text', 'sentiment', 'entities', 'topics']].head())
```

```
                                                text  sentiment  \
0  Daniel Greenfield, a Shillman Journalism Fello...    -0.9994
1  Google Pinterest Digg Linkedin Reddit Stumbleu...     0.7723
2  U.S. Secretary of State John F. Kerry said Mon...     0.9595
3  — Kaydee King (@KaydeeKing) November 9, 2016 T...    -0.4242
```



```
# Display some of the additional features
print(data[['text', 'sentiment', 'entities', 'topics']].head())
```

```
                                             text  sentiment  \
0  Daniel Greenfield, a Shillman Journalism Fello...    -0.9994
1  Google Pinterest Digg Linkedin Reddit Stumbleu...     0.7723
2  U.S. Secretary of State John F. Kerry said Mon...     0.9595
3  — Kaydee King (@KaydeeKing) November 9, 2016 T...    -0.4242
4  It's primary day in New York and front-runners...     0.9903

                                         entities  \
0  [(Daniel Greenfield, PERSON), (Shillman Journa...
1  [(Google Pinterest, ORG), (two, CARDINAL), (Pa...
2  [(U.S., GPE), (State, ORG), (John F. Kerry, PE...
3  [(November 9, 2016, DATE), (tonight, TIME), (D...
4  [(New York, GPE), (Hillary Clinton, PERSON), (...

                                           topics
0                [(3, 0.7262076), (4, 0.25963217)]
1  [(0, 0.015671229), (3, 0.31138358), (4, 0.6710...
2                [(1, 0.24907374), (2, 0.7482149)]
3  [(0, 0.2505012), (3, 0.17288113), (4, 0.574770...
4               [(0, 0.36587754), (4, 0.63052505)]
```