Yash_Big_Data_Docker Copy-Copy1

May 20, 2023

[1]:
```
import nltk
nltk.download('stopwords')
```

```
[nltk_data] Downloading package stopwords to /home/jovyan/nltk_data…
[nltk_data]   Package stopwords is already up-to-date!
```

[1]:
```
True
```

[2]:
```
from nltk.corpus import stopwords
stop_words = stopwords.words('english')
```

[3]:
```
from pyspark import SparkConf
configuration = SparkConf().setAppName('RomeoAndJulietCounter')\
.setMaster('local[*]')
```

[4]:
```
from pyspark import SparkContext
sc = SparkContext(conf=configuration)
```

[5]:
```
from textblob.utils import strip_punc
tokenized = sc.textFile('RomeoAndJuliet.txt')\
.map(lambda line: strip_punc(line, all=True).lower())\
.flatMap(lambda line: line.split())
```

[6]:
```
filtered = tokenized.filter(lambda word: word not in stop_words)
```

[7]:
```
from operator import add
word_counts = filtered.map(lambda word: (word, 1)).reduceByKey(add)
```

[8]:
```
filtered_counts = word_counts.filter(lambda item: item[1] >= 60)
```

[9]:
```
from operator import itemgetter
sorted_items = sorted(filtered_counts.collect(),
key=itemgetter(1), reverse=True)
```

[10]:
```
max_len = max([len(word) for word, count in sorted_items])
for word, count in sorted_items:
    print(f'{word:>{max_len}}: {count}')
```

```
   romeo: 298
    thou: 277
  juliet: 178
     thy: 170
   nurse: 146
 capulet: 141
    love: 136
    thee: 135
   shall: 110
    lady: 109
   friar: 104
    come: 94
mercutio: 83
    good: 80
benvolio: 79
   enter: 75
      go: 75
    i’ll: 71
  tybalt: 69
   death: 69
   night: 68
lawrence: 67
     man: 65
    hath: 64
     one: 60
```

[11]:
```
import pandas as pd
```

[12]:
```
data_frame = pd.DataFrame(sorted_items, columns=['word','count'])
```

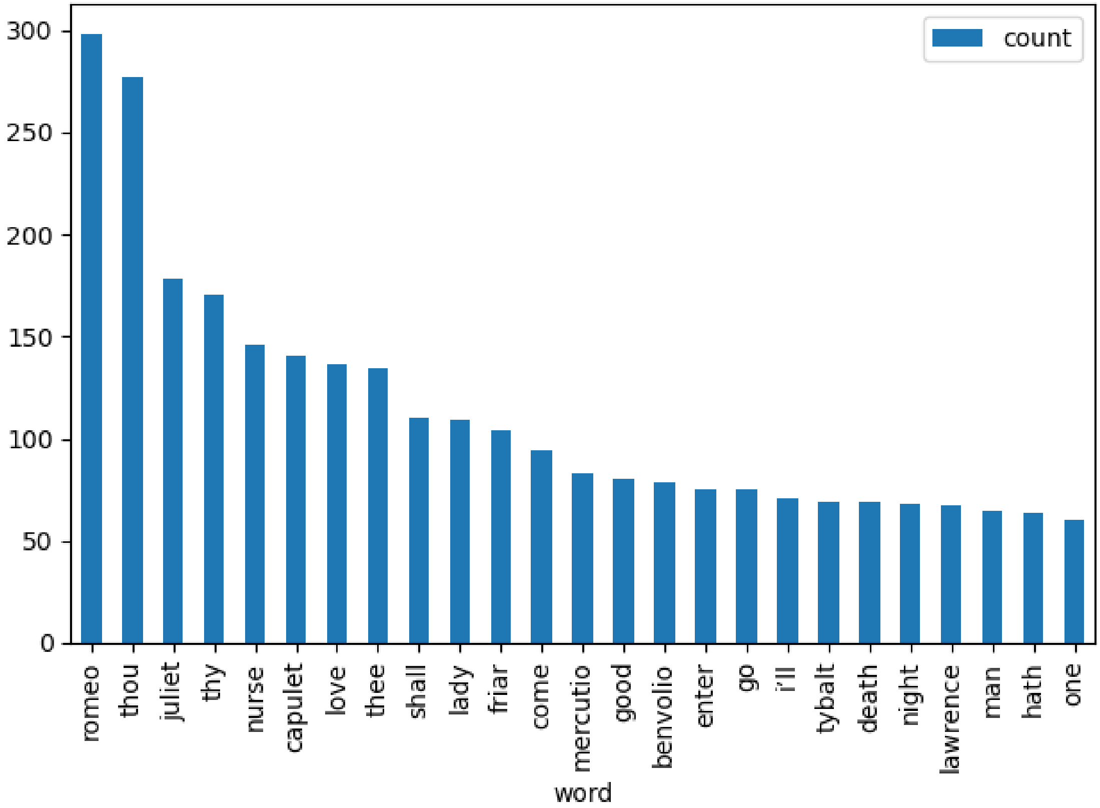
[13]:
```
data_frame
```

[13]:
```
        word  count
0      romeo    298
1       thou    277
2     juliet    178
3        thy    170
4      nurse    146
5    capulet    141
6       love    136
7       thee    135
8      shall    110
9       lady    109
10     friar    104
11      come     94
12  mercutio     83
13      good     80
```

```
14   benvolio     79
15      enter     75
16         go     75
17       i’ll     71
18     tybalt     69
19      death     69
20      night     68
21   lawrence     67
22        man     65
23       hath     64
24        one     60
```

[14]:

```
import matplotlib.pyplot as plt
axes = data_frame.plot.bar(x='word', y='count')
plt.gcf().tight_layout()
```



[15]:

```
from textblob import TextBlob
```

[16]:

```
from pathlib import Path
```

[17]:

```
blob = TextBlob(Path('RomeoAndJuliet.txt').read_text())
```

```
[18]: text = Path('RomeoAndJuliet.txt').read_text()
```

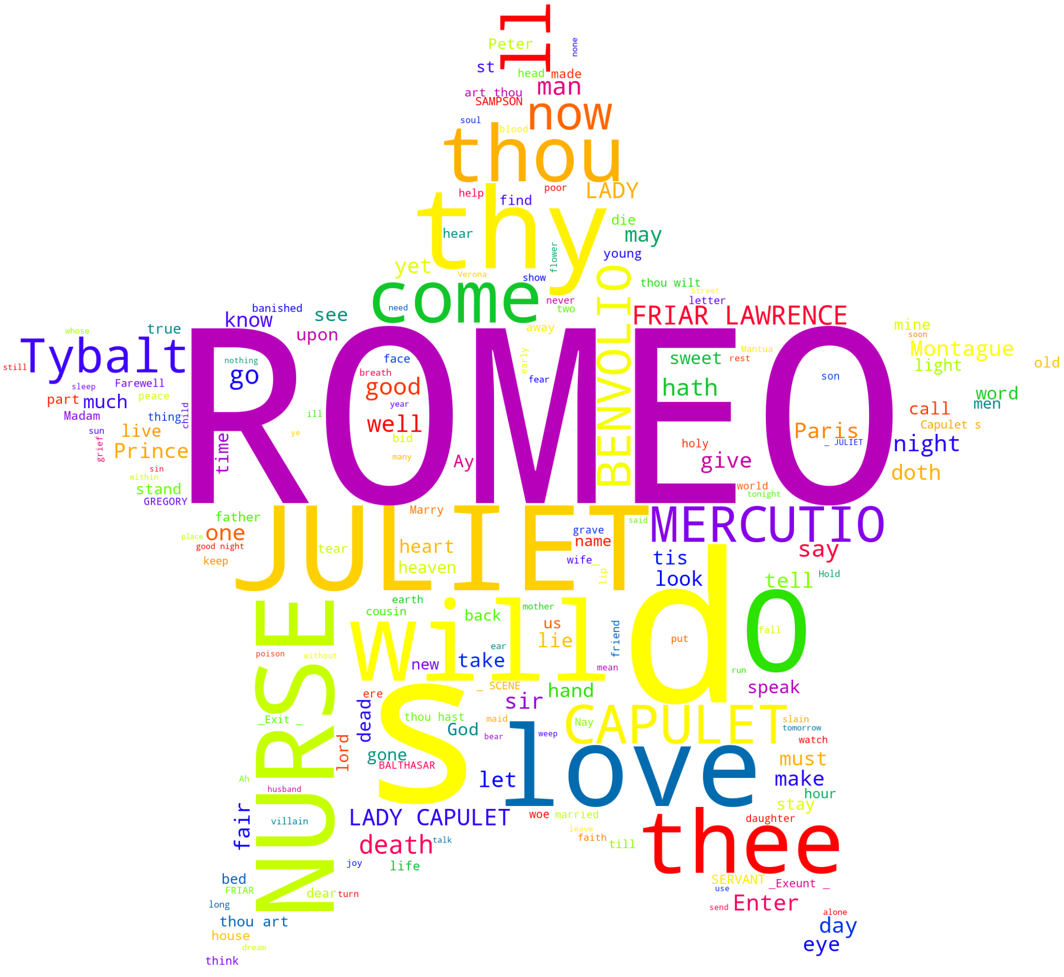
```
[19]: import imageio
      masked_shape = imageio.v3.imread('mask_star.png')
```

```
[20]: from wordcloud import WordCloud
```

```
[21]: wordcloud = WordCloud(width=1000, height=1000,
          colormap='prism', mask=masked_shape, max_words=27).generate(text)
```

```
[22]: from IPython.display import Image
      Image(filename='RomeoAndJulietStar.png', width=400)
```

[22]:



```
[ ]:
```