AssignmentTopicModeling

June 15, 2021

[3]:

```
!ls
```

```
01-LDAdefault.py                 '05-perplexity .py'
02-TopicModelingUCIDataset.py     AssignmentTopicModeling.ipynb
03-LDAwithHyperParameters.py      Datasets
04-OnlineLDA.py                   Topic-Modeling.ipynb
```

[31]:

```
import pandas as pd
import numpy as np
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.decomposition import LatentDirichletAllocation






df = pd.read_csv("Datasets/dataset2.csv")
```

[32]:

```
df.head()
```

[32]:

```
                                              Review  \
0                          Everything from the weather
1  The hotel it is fantastic built by the sea, li…
2  One dream! Cozy  and comfortable Hotel!  The b…
3  Hotel concept is hard to grasp. They communica…
4                            This is a wonderful hotel

                                          Unnamed: 1  \
0                                              staff
1                                                NaN
2   since reception to the end of the stay! We we…
3                                                NaN
4      for a romantic escape. Every room has a theme

                   Unnamed: 2           Unnamed: 3  \
0                        food             property
1                         NaN                  NaN
2   as I have gluten aversion                  NaN
```

```
3                         NaN                   NaN
4           and is incredible   overlooking the sea

                                          Unnamed: 4  \
0                                          fire pits
1                                                NaN
2   all the employees already knew and were waiti…
3                                                NaN
4   the sustainable concept of the hotel is excel…

                           Unnamed: 5  \
0                               d cor
1                                 NaN
2   we were received in the fire pits
3                                 NaN
4                       modern design

                                          Unnamed: 6  \
0                                                spa
1                                                NaN
2   with some wine and all the guests were invite…
3                                                NaN
4   the staff and owners will make your stay memo…

                        Unnamed: 7 Unnamed: 8 Unnamed: 9 Unnamed: 10  \
0   rooms and beach were top notch        NaN        NaN         NaN
1                              NaN        NaN        NaN         NaN
2                              NaN        NaN        NaN         NaN
3                              NaN        NaN        NaN         NaN
4                              NaN        NaN        NaN         NaN

  Unnamed: 11 Unnamed: 12 Unnamed: 13 Unnamed: 14 Unnamed: 15 Unnamed: 16
0         NaN         NaN         NaN         NaN         NaN         NaN
1         NaN         NaN         NaN         NaN         NaN         NaN
2         NaN         NaN         NaN         NaN         NaN         NaN
3         NaN         NaN         NaN         NaN         NaN         NaN
4         NaN         NaN         NaN         NaN         NaN         NaN
```

[33]:

```
df.columns
```

[33]:

```
Index(['Review', 'Unnamed: 1', 'Unnamed: 2', 'Unnamed: 3', 'Unnamed: 4',
       'Unnamed: 5', 'Unnamed: 6', 'Unnamed: 7', 'Unnamed: 8', 'Unnamed: 9',
       'Unnamed: 10', 'Unnamed: 11', 'Unnamed: 12', 'Unnamed: 13',
       'Unnamed: 14', 'Unnamed: 15', 'Unnamed: 16'],
      dtype='object')
```

[34]:

```
data = df.drop(columns=['Unnamed: 1', 'Unnamed: 2', 'Unnamed: 3', 'Unnamed: 4',
       'Unnamed: 5', 'Unnamed: 6', 'Unnamed: 7', 'Unnamed: 8', 'Unnamed: 9',
       'Unnamed: 10', 'Unnamed: 11', 'Unnamed: 12', 'Unnamed: 13',
       'Unnamed: 14', 'Unnamed: 15', 'Unnamed: 16'])
```

[42]:

```
data.Review[0]
```

[42]:

```
' Everything from the weather'
```

[43]:

```
XDocs = data.Review
```

[45]:

```
## Embedding the vector
Tfidf = TfidfVectorizer()

MatrixX = Tfidf.fit_transform(XDocs)
```

[46]:

```
MatrixX
```

[46]:

```
<401x2176 sparse matrix of type '<class 'numpy.float64'>'
        with 10195 stored elements in Compressed Sparse Row format>
```

[47]:

```
MatrixX.toarray
```

[47]:

```
<bound method _cs_matrix.toarray of <401x2176 sparse matrix of type '<class
'numpy.float64'>'
        with 10195 stored elements in Compressed Sparse Row format>>
```

[50]:

```
features = Tfidf.get_feature_names()
```

[52]:

```
# features
```

[58]:

```
Alt =␣
 ↪LatentDirichletAllocation(n_components=20,random_state=0,max_iter=200,learning_method="onli
```

[58]:

```
LatentDirichletAllocation(learning_method='online', max_iter=200,
                          n_components=20, random_state=0)
```

[59]:

```
Alt.fit(MatrixX)
```

[59]:

```
LatentDirichletAllocation(learning_method='online', max_iter=200,
                          n_components=20, random_state=0)
```

[60]:

```
features = Tfidf.get_feature_names()
for tids, topic in enumerate(Alt.components_):
    print('topic ID: ', tids)
    print([features[i] for i in topic.argsort()[:-6:-1]])
```

```
topic ID:  0
['20', 'stars', 'architecture', 'to', 'congratulations']
topic ID:  1
['considering', 'entire', 'surely', 'terms', 'won']
topic ID:  2
['exceeded', 'expectations', 'rivals', 'world', 'yet']
topic ID:  3
['the', 'single', 'built', 'that', 'know']
topic ID:  4
['come', 'back', 'will', 'we', 'lot']
topic ID:  5
['the', 'we', 'and', 'to', 'for']
topic ID:  6
['location', 'great', 'we', 'the', 'weather']
topic ID:  7
['dear', 'all', 'excellent', 'restaurant', 'holdings']
topic ID:  8
['extraordinary', 'dear', 'especially', 'maria', 'enjoy']
topic ID:  9
['oxal', 'fabulous', 'atlantic', 'complements', 'home']
topic ID:  10
['what', 'heavenly', 'place', 'wonderful', 'fabulous']
topic ID:  11
['beautiful', 'thank', 'wonderful', 'place', 'you']
topic ID:  12
['dears', 'delicacy', 'pure', 'my', 'fantastic']
topic ID:  13
['focusing', 'room', 'the', 'creative', 'hospitable']
topic ID:  14
['dreamful', 'fruit', 'selection', 'fresh', 'place']
topic ID:  15
['nice', 'dream', 'the', 'all', 'thank']
topic ID:  16
['daniela', 'peaceful', 'general', 'in', 'to']
topic ID:  17
['love', 'with', 'we', 'place', 'of']
topic ID:  18
['pleasant', 'very', 'beautiful', 'magical', 'warm']
topic ID:  19
['very', 'special', 'anniversary', 'of', 'one']
```

[ ]:

```
```