ROLL NO – 162010022

PRACTICAL – 6
CLUSTERING

PRE-REQUISITES:-

- PYTHON
- JUPYTER NOTEBOOK
- CONDA
- ANACONDA WITH INBUILT PACKAGES
  conda install -c anaconda
- TQDM : pip install tqdm
- NLTK : conda install -c anaconda nltk=3.2.2
- BOOKEH : conda install bokeh
- LDA : pip install lda
- PYLDAVIS : pip install pyldavis

BEGIN :-

1. **IMPORTS**

```
import pandas as pd
import numpy as np
from nltk.tokenize import word_tokenize,sent_tokenize
from nltk.corpus import stopwords
from string import punctuation
import re
from collections import Counter
```

2. **DEFINE tokenizer()**

```
def tokenizer(text):
    try:
        tokens_ = [word_tokenize(sent) for sent in sent_tokenize(text)]
        tokens = []
        for token_sen in tokens_:
            tokens += token_sen

        tokens = list(filter(lambda t: t.lower() not in stop,tokens))
        tokens = list(filter(lambda t: t not in punctuation,tokens))
        tokens = list(filter(lambda t: t not in [u"'s", u"n't", u"...", u"''", u'``',u'\u2014', u'\u2026',
                              u'\u2013'],tokens))
        filtered_tokens = []
        for token in tokens:
            if re.search("[a-zA-Z]",token):
                filtered_tokens.append(token)
        return filtered_tokens
    except Exception as e:
        print(e)
```

3. **DEFINE keywords()**

```
def keywords(source):
    tokens = dataset[dataset['source']==source]['tokens']
    alltokens = []
    for token in tokens:
        alltokens+=token
        count = Counter(alltokens)

return count.most_common(10)
```

4. stop = set(stopwords.words('english'))
5. dataset = pd.read_csv('rediff_realtime_news_201704_201706',delimiter = '\t',nrows=10000)
6. dataset.drop_duplicates(subset = ['summary'],inplace=True)
7. dataset = dataset[~dataset['summary'].isnull()]
8. dataset['length'] = dataset['summary'].map(len)
9. dataset = dataset[dataset['length'] > 140]
10. dataset.reset_index(inplace=True)
11. dataset.drop('index',inplace=True,axis=1)
12. dataset['tokens'] = dataset['summary'].map(tokenizer)
13. for summary,tokens in zip(dataset['summary'].head(5),dataset['tokens'].head(5)):

```
    print("Summary:",summary)
    print("Tokens:",tokens)
    print()
```

```
('Summary:', "NEW YORK (Reuters) - The Federal Reserve could begin shrinking its $4.5-trillion balance sheet as soo
n as this year, earlier than most economists expect, New York Fed President William Dudley said on Friday in the ce
ntral bank's most definitive comments on the question that looms over financial markets.")
('Tokens:', ['NEW', 'YORK', 'Reuters', 'Federal', 'Reserve', 'could', 'begin', 'shrinking', '4.5-trillion', 'balanc
e', 'sheet', 'soon', 'year', 'earlier', 'economists', 'expect', 'New', 'York', 'Fed', 'President', 'William', 'Dudl
ey', 'said', 'Friday', 'central', 'bank', 'definitive', 'comments', 'question', 'looms', 'financial', 'markets'])
()
('Summary:', 'A total 27,850 migrants and refugees landed in Europe in the first 89 days of this year, of whom 23,1
25 reached Italy, the UN migration agency International Organisation for Migration said on Friday.Although the over
all arrivals were a fraction of those in the same period of 2016 (165,697), 7,000 more people reached Italy by sea,
the IOM figures...')
('Tokens:', ['total', 'migrants', 'refugees', 'landed', 'Europe', 'first', 'days', 'year', 'reached', 'Italy', 'UN'
, 'migration', 'agency', 'International', 'Organisation', 'Migration', 'said', 'Friday.Although', 'overall', 'arriv
als', 'fraction', 'period', 'people', 'reached', 'Italy', 'sea', 'IOM', 'figures'])
()
('Summary:', 'The Central government on Friday urged the state utilities to hasten the process of completion of tra
nsmission projects in the pipeline in order to meet the power demand in the coming summer, an official statement sa
id.According to the Power Ministry, the all India peak demand during the upcoming summer is expected to be of the o
rder of 165...')
```

14. dataset = dataset[~dataset['source'].isnull()]

```
for source in set(dataset['source']):
    print("Source:",source)
    print("Top 10 keywords:",keywords(source))
    print('----')
```

```
('Source:', 'SME Times')
('Top 10 keywords:', [('Friday', 14), ('said', 9), ('Bank', 5), ('US', 5), ('Rs', 4), ('India', 4), ('trade', 4), ('
April', 4), ('Jio', 3), ('Global', 3)])
----
```

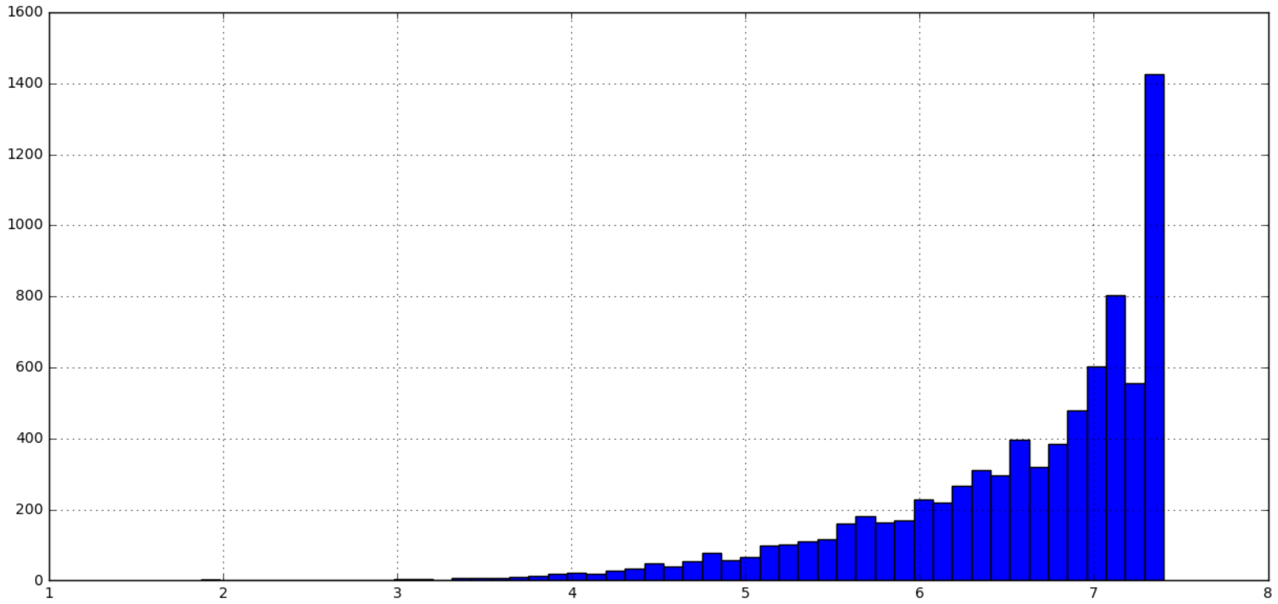
15. dataset.shape

```
dataset.shape

(6630, 8)
```

16. from sklearn.feature_extraction.text import TfidfVectorizer

```
vectorizer =
TfidfVectorizer(min_df=10,max_features=10000,tokenizer=tokenizer,ngram_range=(1,2))
vz= vectorizer.fit_transform(list(dataset['summary']))

tfidf = dict(zip(vectorizer.get_feature_names(),vectorizer.idf_))

tfidf = pd.DataFrame(columns=['tfidf']).from_dict(dict(tfidf),orient='index')

tfidf.columns = ['tfidf']

tfidf.tfidf.hist(bins=50,figsize=(15,7))

tfidf.sort_values(by=['tfidf'],ascending=True).head(30)

tfidf.sort_values(by=['tfidf'],ascending=False).head(30)
```

	tfidf
sovereignty	7.401616
said meanwhile	7.401616
drunken	7.401616
municipal elections	7.401616
four college	7.401616
abducted	7.401616
little bit	7.401616
aggressively	7.401616
posters	7.401616
state-run xinhua	7.401616
five-year	7.401616
degrees normal	7.401616
sought push	7.401616
superstars	7.401616
ramesh	7.401616
meat sellers	7.401616
commercial vehicles	7.401616
unacceptable	7.401616
recall	7.401616
government school	7.401616



17. from sklearn.decomposition import TruncatedSVD
svd = TruncatedSVD(n_components=6,random_state=0)
svd_tfidf = svd.fit_transform(vz)

svd_tfidf.shape

from sklearn.manifold import TSNE

tsne_model = TSNE(n_components=2,verbose=1,random_state=0)

tsne_tfidf = tsne_model.fit_transform(svd_tfidf)

tsne_tfidf.shape

```
[t-SNE] Computing pairwise distances...
[t-SNE] Computing 91 nearest neighbors...
[t-SNE] Computed conditional probabilities for sample 1000 / 6630
[t-SNE] Computed conditional probabilities for sample 2000 / 6630
[t-SNE] Computed conditional probabilities for sample 3000 / 6630
[t-SNE] Computed conditional probabilities for sample 4000 / 6630
[t-SNE] Computed conditional probabilities for sample 5000 / 6630
[t-SNE] Computed conditional probabilities for sample 6000 / 6630
[t-SNE] Computed conditional probabilities for sample 6630 / 6630
[t-SNE] Mean sigma: 0.000000
[t-SNE] Error after 100 iterations with early exaggeration: 1.563398
[t-SNE] Error after 375 iterations: 1.457381

(6630, 2)
```

18. import bokeh.plotting as bp
from bokeh.models import HoverTool, BoxSelectTool
from bokeh.plotting import figure, show, output_notebook

output_notebook()
plot_tfidf = bp.figure(plot_width=700, plot_height=600, title="tf-idf clustering of the news",
tools="pan,wheel_zoom,box_zoom,reset,hover,previewsave",
x_axis_type=None, y_axis_type=None, min_border=1)

tfidf_df = pd.DataFrame(tsne_tfidf, columns=['x', 'y'])
fidf_df['summary'] = dataset['summary']

BokehJS 0.12.15 successfully loaded.

19. print(tfidf_df)

	x	y	summary
0	5.409187	-7.574306	NEW YORK (Reuters) - The Federal Reserve could...
1	1.219611	-4.841058	A total 27,850 migrants and refugees landed in...
2	2.303575	2.896371	The Central government on Friday urged the sta...
3	6.925687	3.965021	Venezuela's powerful attorney general on Frida...
4	6.241896	5.305988	SUPREME COURT NOMINEE Two Democratic senators ...
5	-6.160031	-4.619103	Rana Daggubati, who started his acting career ...
6	6.756569	8.708511	The Cabinet Committee on Economic Affairs (CCE...
7	1.787366	2.954256	India's three armed services are short of over...
8	-0.247471	3.134241	Pakistan's electronic media watchdog today imp...
9	10.562601	-1.043369	Jharkhand Chief Minister Raghubar Das today we...
10	6.451645	8.886172	The Union Cabinet, chaired by Prime Minister N...
11	-1.393102	2.181424	The second 1,000 MW atomic power reactor at th...
12	6.718653	8.676555	The Union Cabinet on Friday approved the propo...
13	1.914529	-10.305219	Elon Musk's SpaceX on Thursday salvaged half o...
14	-8.311554	5.701379	WASHINGTON (Reuters) - U.S. President Donald T...
15	-0.713049	-7.073674	Kaushalya Devi, 38, is a resident of slums of ...
16	3.021193	-9.704532	Bharat Petroleum Corp (BPCL), Hindustan Petrol...
17	3.412300	-4.019393	CHICAGO (Reuters) - Researchers have begun the...

20. plot_tfidf.scatter(x='x', y='y', source=tfidf_df)
hover = plot_tfidf.select(dict(type=HoverTool))
hover.tooltips={"summary": "@summary"}
show(plot_tfidf)

21. import warnings

```
warnings.filterwarnings("ignore", category=DeprecationWarning)

from sklearn.cluster import MiniBatchKMeans

num_clusters = 30
kmeans_model = MiniBatchKMeans(n_clusters=num_clusters, init='k-means++', n_init=1,
                         init_size=1000, batch_size=1000, verbose=False, max_iter=1000)
kmeans = kmeans_model.fit(vz)
kmeans_clusters = kmeans.predict(vz)
kmeans_distances = kmeans.transform(vz)
```

22. for (i, summary) in enumerate(dataset.summary):

```
  if(i < 5):
    print("Cluster " + str(kmeans_clusters[i]) + ": " + summary +
       "(distance: " + str(kmeans_distances[i][kmeans_clusters[i]]) + ")")
    print('---------------------------------------------------------------')
```

```
Cluster 0: NEW YORK (Reuters) - The Federal Reserve could begin shrinking its $4.5-trillion balance sheet as soon as
this year, earlier than most economists expect, New York Fed President William Dudley said on Friday in the central
bank's most definitive comments on the question that looms over financial markets.(distance: 0.993136373439)
-----------------------------------------------------------------
Cluster 0: A total 27,850 migrants and refugees landed in Europe in the first 89 days of this year, of whom 23,125 r
eached Italy, the UN migration agency International Organisation for Migration said on Friday.Although the overall a
rrivals were a fraction of those in the same period of 2016 (165,697), 7,000 more people reached Italy by sea, the I
OM figures...(distance: 0.997421577173)
-----------------------------------------------------------------
Cluster 0: The Central government on Friday urged the state utilities to hasten the process of completion of transmi
ssion projects in the pipeline in order to meet the power demand in the coming summer, an official statement said.Ac
cording to the Power Ministry, the all India peak demand during the upcoming summer is expected to be of the order o
f 165...(distance: 0.992651098565)
-----------------------------------------------------------------
Cluster 0: Venezuela's powerful attorney general on Friday broke ranks with President Nicolas Maduro's government af
ter the judiciary annulled congress, a rare show of internal dissent as protests and international condemnation grew
.(distance: 0.997840926978)
-----------------------------------------------------------------
Cluster 0: SUPREME COURT NOMINEE Two Democratic senators voice opposition to Trump's Supreme Court nominee, Neil Gor
such, ahead of an expected contentious confirmation fight next week on the Senate floor.(distance: 0.995013090529)
-----------------------------------------------------------------
```

23. sorted_centroids = kmeans.cluster_centers_.argsort()[:, ::-1]

```
terms = vectorizer.get_feature_names()
for i in range(num_clusters):
  print("Cluster %d:" % i)
  aux = ''
  for j in sorted_centroids[i, :10]:
    aux += terms[j] + ' | '
  print(aux)
  print()
```

```
Cluster 0:
said | india | state | minister | today | police | first | government | new | saturday |
()
Cluster 1:
tunnel | modi | kashmir | prime minister | prime | minister | jammu | narendra modi | narendra | minister narendra
|
()
Cluster 2:
michael | george | tribute | loved | cover | always | called | touching | sigh | lights |
()
Cluster 3:
tiwari | party would | delhi bjp | tickets | manoj | apparently | sitting | bid | beat | give |
()
Cluster 4:
spell | plastics | good news | discovery | capable | novel | media report | report said | environment | identified
|
()
```

24. tsne_kmeans = tsne_model.fit_transform(kmeans_distances)

25. import numpy as np

colormap = np.array(["#6d8dca", "#69de53", "#723bca", "#c3e14c", "#c84dc9", "#68af4e", "#6e6cd5",
"#e3be38", "#4e2d7c", "#5fdfa8", "#d34690", "#3f6d31", "#d44427", "#7fcdd8", "#cb4053", "#5e9981",
"#803a62", "#9b9e39", "#c88cca", "#e1c37b", "#34223b", "#bdd8a3", "#6e3326", "#cfbdce", "#d07d3c",
"#52697d", "#7d6d33", "#d27c88", "#36422b", "#b68f79"])

plot_kmeans = bp.figure(plot_width=700, plot_height=600, title="KMeans clustering of the news",
tools="pan,wheel_zoom,box_zoom,reset,hover,previewsave",
x_axis_type=None, y_axis_type=None, min_border=1)

26. import numpy as np

colormap = np.array(["#6d8dca", "#69de53", "#723bca", "#c3e14c", "#c84dc9", "#68af4e", "#6e6cd5",
"#e3be38", "#4e2d7c", "#5fdfa8", "#d34690", "#3f6d31", "#d44427", "#7fcdd8", "#cb4053", "#5e9981",
"#803a62", "#9b9e39", "#c88cca", "#e1c37b", "#34223b", "#bdd8a3", "#6e3326", "#cfbdce", "#d07d3c",
"#52697d", "#7d6d33", "#d27c88", "#36422b", "#b68f79"])

plot_kmeans = bp.figure(plot_width=700, plot_height=600, title="KMeans clustering of the news",
tools="pan,wheel_zoom,box_zoom,reset,hover,previewsave",
x_axis_type=None, y_axis_type=None, min_border=1)

27. print(kmeans_df)

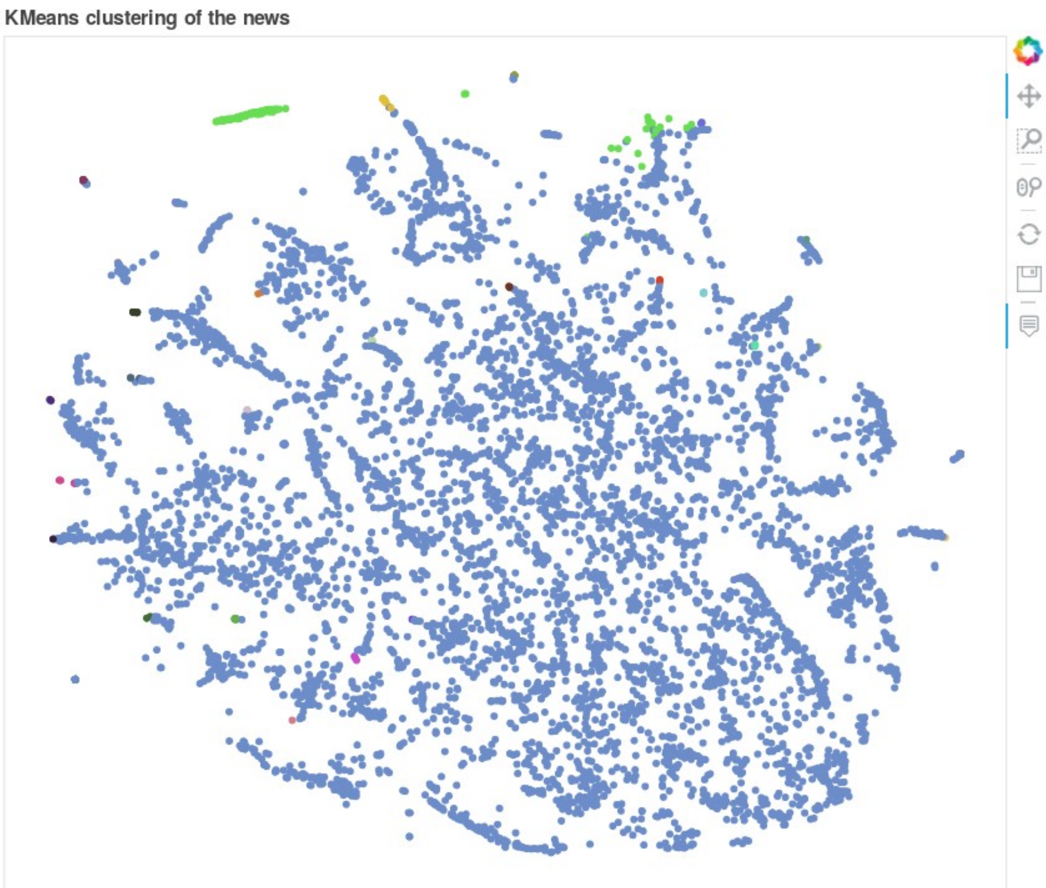
```
            x          y  cluster  \
0    2.862722   2.754873        0
1   10.104860  -3.467533        0
2   -5.375391  -7.152985        0
3    4.462104  -3.533770        0
4    6.123255  -8.566491        0
5    7.986605  -5.610218        0
6    5.321031   7.434730        0
7    3.862046  -4.898439        0
8    0.652972  -1.363594        0
9    6.945661  -1.727704        0
10   5.372495   7.566505        0
11  -2.505390   2.555026        0
12   5.371672   7.453019        0
13  10.918354  -0.578794        0
14  -3.858720   5.815642        0
15   9.503067 -10.693848        0
16  -5.182462  -7.499461        0
17  -0.512179   0.436635        0
```

```
17    CHICAGO (Reuters) - Researchers have begun the...   #6d8dca
18    New Delhi:The service charge exemption on rail...   #6d8dca
19    Thiruvananthapuram:Seventy two years aftercomi...   #6d8dca
20    Thiruvananthapuram:Public transport servicesin...   #6d8dca
21    New Delhi: The price of petrol is cut by Rs3.7...   #6d8dca
22    YANGON (Reuters) - The leader of a Rohingya Mu...   #6d8dca
23    BERLIN (Reuters) - A transition period offered...   #6d8dca
24    BEIRUT (Reuters) - Prime Minister Saad al-Hari...   #6d8dca
25    Rio Olympics silver-medallist PV Sindhu beat S...   #6d8dca
26    CAPE CANAVERAL, Fla. (Reuters) - Elon Musk’s S...   #6d8dca
27    WASHINGTON (Reuters) - President Donald Trump ...   #6d8dca
28    STOCKHOLM (Reuters) - Swedish prosecutors inve...   #6d8dca
29    WASHINGTON (Reuters) - Comcast Corp , Verizon ...   #6d8dca
...                                                 ...       ...
6600  [USA], April 3 (ANI):Pakistan's newly-appointe...   #6d8dca
6601  Bilateral talks in areas such as education, tr...   #6d8dca
6602  The Varanasi mayor has issued orders making it...   #6d8dca
6603  The Border Security Force (BSF) on Monday seiz...   #6d8dca
6604  Senior Congress leader Kamal Nath today dubbed...   #6d8dca
```

28. print (colormap[kmeans_clusters])

```
['#6d8dca' '#6d8dca' '#6d8dca' ..., '#6d8dca' '#6d8dca' '#6d8dca']
```

29. plot_kmeans.scatter(x = 'x', y = 'y',source = kmeans_df,color='color')
hover = plot_kmeans.select(dict(type=HoverTool))
hover.tooltips={"summary": "@summary", "cluster":"@cluster"}
show(plot_kmeans)



30. import lda
from sklearn.feature_extraction.text import CountVectorizer

31. import logging
logging.getLogger("lda").setLevel(logging.WARNING)

32. cvectorizer = CountVectorizer(min_df=4, max_features=10000, tokenizer=tokenizer, ngram_range=(1,2))

33. cvz = cvectorizer.fit_transform(dataset['summary'])
n_topics = 20
n_iter = 2000
lda_model = lda.LDA(n_topics=n_topics, n_iter=n_iter)
X_topics = lda_model.fit_transform(cvz)

34. n_top_words = 8
topic_summaries = []

topic_word = lda_model.topic_word_ # get the topic words
vocab = cvectorizer.get_feature_names()
for i, topic_dist in enumerate(topic_word):
    topic_words = np.array(vocab)[np.argsort(topic_dist)][:-(n_top_words+1):-1]
    topic_summaries.append(' '.join(topic_words))
    print('Topic {}: {}'.format(i, ' '.join(topic_words)))

```
Topic 0: election party commission evms election commission said pradesh assembly
Topic 1: said one like would time years also people
Topic 2: trump president said china donald us donald trump u.s.
Topic 3: india said indian april assam air airport dalai
Topic 4: party bjp congress delhi minister leader said people
Topic 5: new india first company services said business jio
Topic 6: police post said appeared appeared first first times state times
Topic 7: minister prime modi prime minister kashmir tunnel jammu narendra
Topic 8: india open first final ipl indian sindhu cricket
Topic 9: said police people two city fire three road
Topic 10: bank state pradesh uttar pradesh uttar state bank said sbi
Topic 11: court supreme supreme court liquor said state high order
Topic 12: said university education new may students infosys board
Topic 13: per rs cent per cent march percent said year
Topic 14: april rsquo v league saturday rdquo ldquo goa
Topic 15: said pakistan country security government rights would state
Topic 16: india said minister indian countries finance new development
```

35. tsne_lda = tsne_model.fit_transform(X_topics)

36. doc_topic = lda_model.doc_topic_
lda_keys = []
for i, tweet in enumerate(dataset['summary']):
    lda_keys += [doc_topic[i].argmax()]

37. plot_lda = bp.figure(plot_width=700, plot_height=600, title="LDA topic visualization",
    tools="pan,wheel_zoom,box_zoom,reset,hover,previewsave",
    x_axis_type=None, y_axis_type=None, min_border=1)

38. lda_df = pd.DataFrame(tsne_lda, columns=['x','y'])
lda_df['summary'] = dataset['summary']

39. lda_df['topic'] = lda_keys
lda_df['topic'] = lda_df['topic'].map(int)
lda_df['color'] = colormap[lda_keys]

40. plot_lda.scatter(source=lda_df, x='x', y='y', color='color')
hover = plot_lda.select(dict(type=HoverTool))
hover.tooltips={"summary":"@summary", "topic":"@topic"}
show(plot_lda)