

MACHINE LEARNING LAB ASSIGNMENT

[DS7A-709]

WEEK-1

MASTER OF TECHNOLOGY

In

DATA SCIENCE

Submitted by

RAHUL SHARMA

Submitted To

DR. DINESH BHATI

(Lecturer)

SCHOOL OF DATA SCIENCE AND FORECASTING

(UNIVERSITY TEACHING DEPARTMENT)

DEVI AHILYA VISHWAVIDYALAYA

Indore (M.P.)

Milestone 1: Downloading NLP libraries (NLTK)

NLTK is a leading platform for building Python programs to work with human language data. It provides easy-to-use interfaces to over 50 corpora and lexical resources such as WordNet, along with a suite of text processing libraries for classification, tokenization, stemming, tagging, parsing, and semantic reasoning, wrappers for industrial-strength NLP libraries, and an active discussion forum.

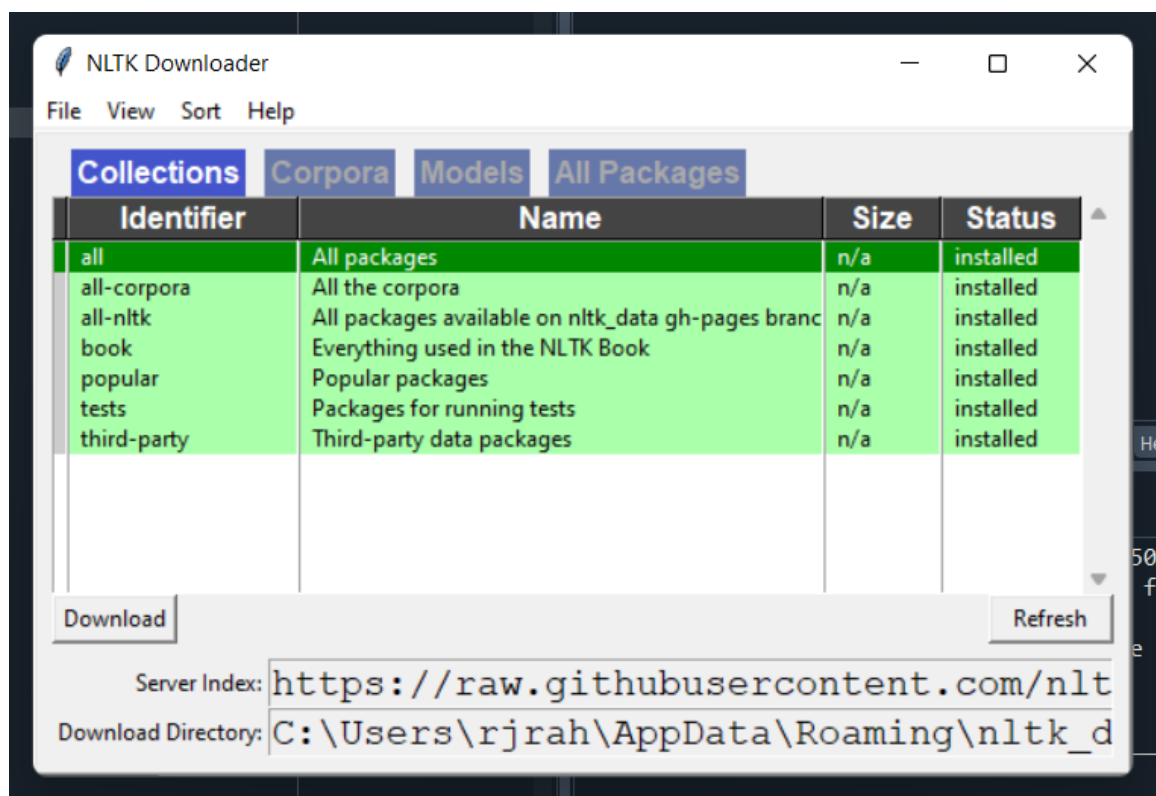
Interactive installer

Run the Python interpreter and type the commands:

```
import nltk
```

```
nltk.download()
```

A new window opened, showing the NLTK Downloader. Clicked on the File menu and select Change Download Directory where we want to download.



Installing the whole package. Once the package is installed.

Firstly, imported the corpus:

```
from nltk.corpus import names
```

```
print (names.words()[:10])
```

Output:

```
['Abagael', 'Abigail', 'Abbe', 'Abbey', 'Abbi', 'Abbie', 'Abby', 'Abigail', 'Abigail', 'Abigail']
```

Code#

```
print(len(names.words()))
```

Output:

```
7944
```

There are in total 7944 names

Stemming

Stemming is a process of reverting an inflected or derived word to its root form. Stemmers remove morphological affixes from words, leaving only the word stem.

PorterStemmer Interface

The Porter stemming algorithm is a process for removing the commoner morphological and inflexional endings from words in English.

Code#

```
from nltk.stem.porter import PorterStemmer
```

```
ps = PorterStemmer()
```

Test the stemmer on various pluralised words.

```
words=['machines', 'learning','stating', 'owned', 'agreed','reference']
```

```
for w in words:
```

```
    print(w, " : ", ps.stem(w))
```

Output:

```
machines : machin
```

```
learning : learn
```

```
stating : state
```

```
owned : own
```

```
agreed : agre
```

```
reference : refer
```

Noted that, stemming sometimes involves chopping off letters, if necessary, as we can see in machin.

Now import a lemmatization algorithm based on Wordnet corpus built-in, and initialize a lemmatizer.

WordNet Interface

WordNet is just another NLTK corpus reader, and can be imported like this:

Code#

```
from nltk.stem import WordNetLemmatizer  
wnl = WordNetLemmatizer()  
print("machines :", wnl.lemmatize("machines"))  
print("learning :", wnl.lemmatize("learning"))
```

Output:

```
machines : machine  
learning : learning
```

It turns out that this algorithm only lemmatizes on nouns by default.

Source Code##

Spyder (Python 3.9)

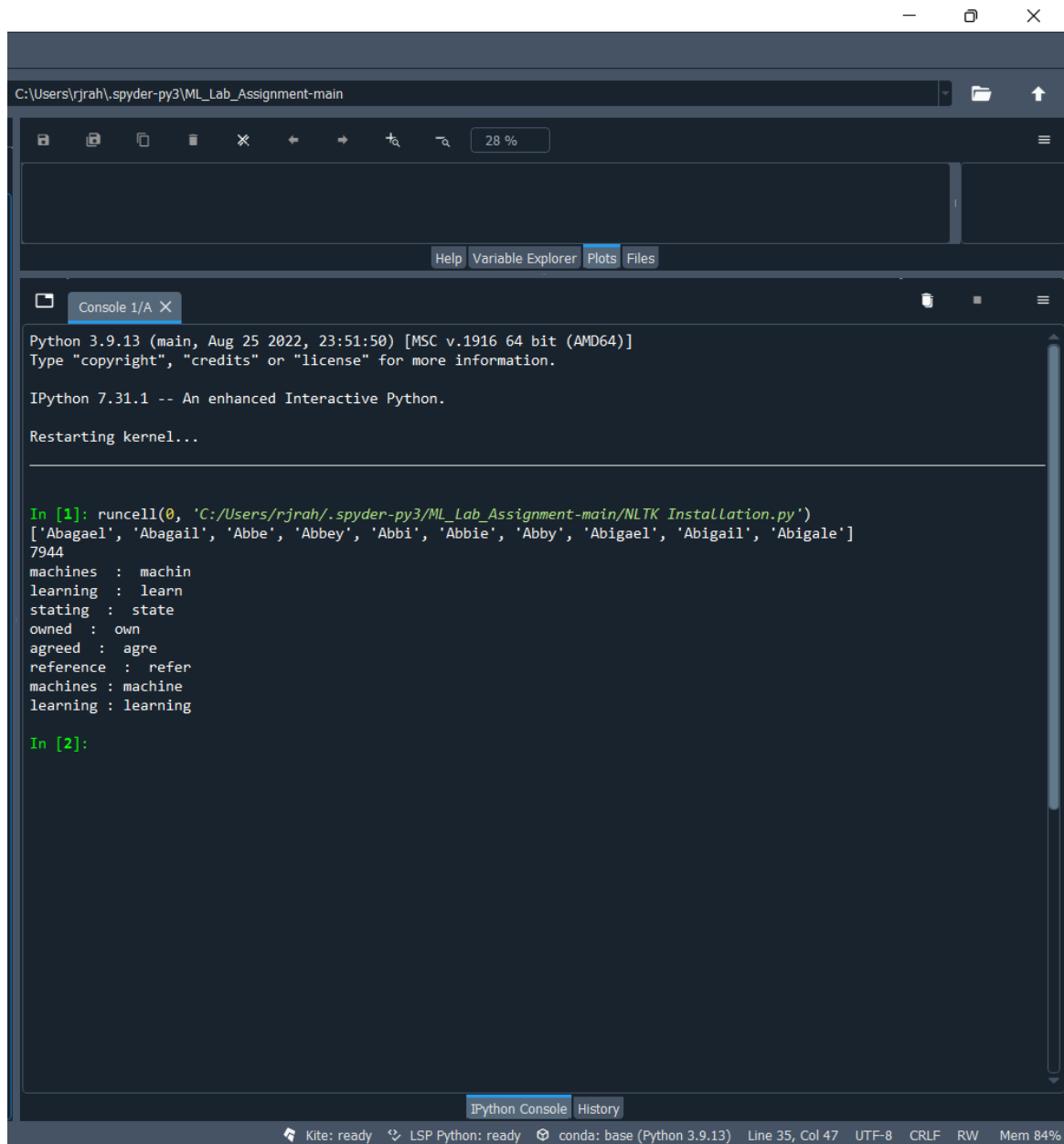
File Edit Search Source Run Debug Consoles Projects Tools View Help

C:\Users\rjrah\spyder-py3\ML_Lab_Assignment-main\NLTK Installation.py

NLTK Installation.py X Getting The 20 newspaper Dataset.py* X Clustering.py X Data Preprocessing.py X Topic Modelling.i

```
1  # -*- coding: utf-8 -*-
2  """
3  Created on Sun Sep  4 10:48:48 2022
4
5  @author: Rahul Sharma "M.Tech (Data Science)"
6  """
7
8  '''Downloading NLTK'''
9  # import nltk
10 # nltk.download()
11
12
13 '''importing names'''
14
15 from nltk.corpus import names
16 print(names.words()[:10])
17
18 '''length of words'''
19
20 print(len(names.words()))
21
22 '''    PorterStemmer    '''
23
24 from nltk.stem.porter import PorterStemmer
25 ps = PorterStemmer()
26 words=['machines', 'learning', 'stating', 'owned', 'agreed', 'reference']
27 for w in words:
28     print(w, " : ", ps.stem(w))
29
30
31 '''    WordNetLemmatizer    '''
32 from nltk.stem import WordNetLemmatizer
33 wnl = WordNetLemmatizer()
34 print("machines :", wnl.lemmatize("machines"))
35 print("Learning :", wnl.lemmatize("Learning"))
```

Output:



The screenshot displays the Spyder Python IDE interface. The top panel shows the file explorer with the current file path: `C:\Users\rjrah\.spyder-py3\ML_Lab_Assignment-main`. The middle panel contains tabs for `Help`, `Variable Explorer`, `Plots`, and `Files`. The bottom panel is the IPython console, titled `Console 1/A`. It shows the following output:

```
Python 3.9.13 (main, Aug 25 2022, 23:51:50) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 7.31.1 -- An enhanced Interactive Python.

Restarting kernel...

In [1]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/NLTK Installation.py')
['Abagael', 'Abagail', 'Abbe', 'Abbey', 'Abbi', 'Abbie', 'Abby', 'Abigael', 'Abigail', 'Abigale']
7944
machines : machin
learning : learn
stating  : state
owned   : own
agreed  : agre
reference : refer
machines : machine
learning : learning

In [2]:
```

The bottom status bar indicates the following information: `Kite: ready`, `LSP Python: ready`, `conda: base (Python 3.9.13)`, `Line 35, Col 47`, `UTF-8`, `CRLF`, `RW`, and `Mem 84%`.

Milestone 2: The 20 Newsgroups data set

The 20 newsgroups dataset comprises around 18000 newsgroups posts on 20 topics split into two subsets: one for training (or development) and the other one for testing (or for performance evaluation). The split between the train and test set is based on a message posted before and after a specific date.

comp.graphics comp.os.ms-windows.misc comp.sys.ibm.pc.hardware comp.sys.mac.hardware comp.windows.x	rec.autos rec.motorcycles rec.sport.baseball rec.sport.hockey	sci.crypt sci.electronics sci.med sci.space
misc.forsale	talk.politics.misc talk.politics.guns talk.politics.mideast	talk.religion.misc alt.atheism soc.religion.christian

The data available here are in .tar.gz bundles. You will need tar and gunzip to open them 20news-bydate.tar.gz - 20 Newsgroups sorted by date; duplicates and some headers removed (18846 documents)

Milestone 3: Getting the data

To load the data, we can import the loader function for the 20 newsgroups data as follows:

```
from sklearn.datasets import fetch_20newsgroups
```

```
groups = fetch_20newsgroups()
```

Milestone 4: Thinking about features

Code#

```
print(groups.keys())
```

```
print(groups['target_names'])
```

```
print(groups.target)
```

Output:

```
dict_keys(['data', 'filenames', 'target_names', 'target', 'DESCR'])
```

```
['alt.atheism', 'comp.graphics', 'comp.os.ms-windows.misc', 'comp.sys.ibm.pc.hardware',  
'comp.sys.mac.hardware', 'comp.windows.x', 'misc.forsale', 'rec.autos', 'rec.motorcycles',  
'rec.sport.baseball', 'rec.sport.hockey', 'sci.crypt', 'sci.electronics', 'sci.med', 'sci.space',
```

```
'soc.religion.christian', 'talk.politics.guns', 'talk.politics.mideast', 'talk.politics.misc',  
'talk.religion.misc']
```

```
[7 4 4 ... 3 1 8]
```

Code#

```
import numpy as np  
  
print(np.unique(groups.target))  
  
print(groups.data[0])
```

Output:

```
[ 0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19]
```

```
From: lerxst@wam.umd.edu (where's my thing)
```

```
Subject: WHAT car is this!?
```

```
Nntp-Posting-Host: rac3.wam.umd.edu
```

```
Organization: University of Maryland, College Park
```

```
Lines: 15
```

```
I was wondering if anyone out there could enlighten me on this car I saw  
the other day. It was a 2-door sports car, looked to be from the late 60s/  
early 70s. It was called a Bricklin. The doors were really small. In addition,  
the front bumper was separate from the rest of the body. This is  
all I know. If anyone can tellme a model name, engine specs, years  
of production, where this car is made, history, or whatever info you  
have on this funky looking car, please e-mail.
```

```
Thanks,
```

```
- IL
```

```
---- brought to you by your neighborhood Lerxst ----
```

Code#

```
print(groups.target[0])  
  
print(groups.target_names[groups.target[0]])  
  
print(len(groups.data[0]))  
  
print(len(groups.data[1]))
```

Output:

```
7
```

```
rec.autos
```

```
721
```

```
858
```


Source Code##

Spyder (Python 3.9)

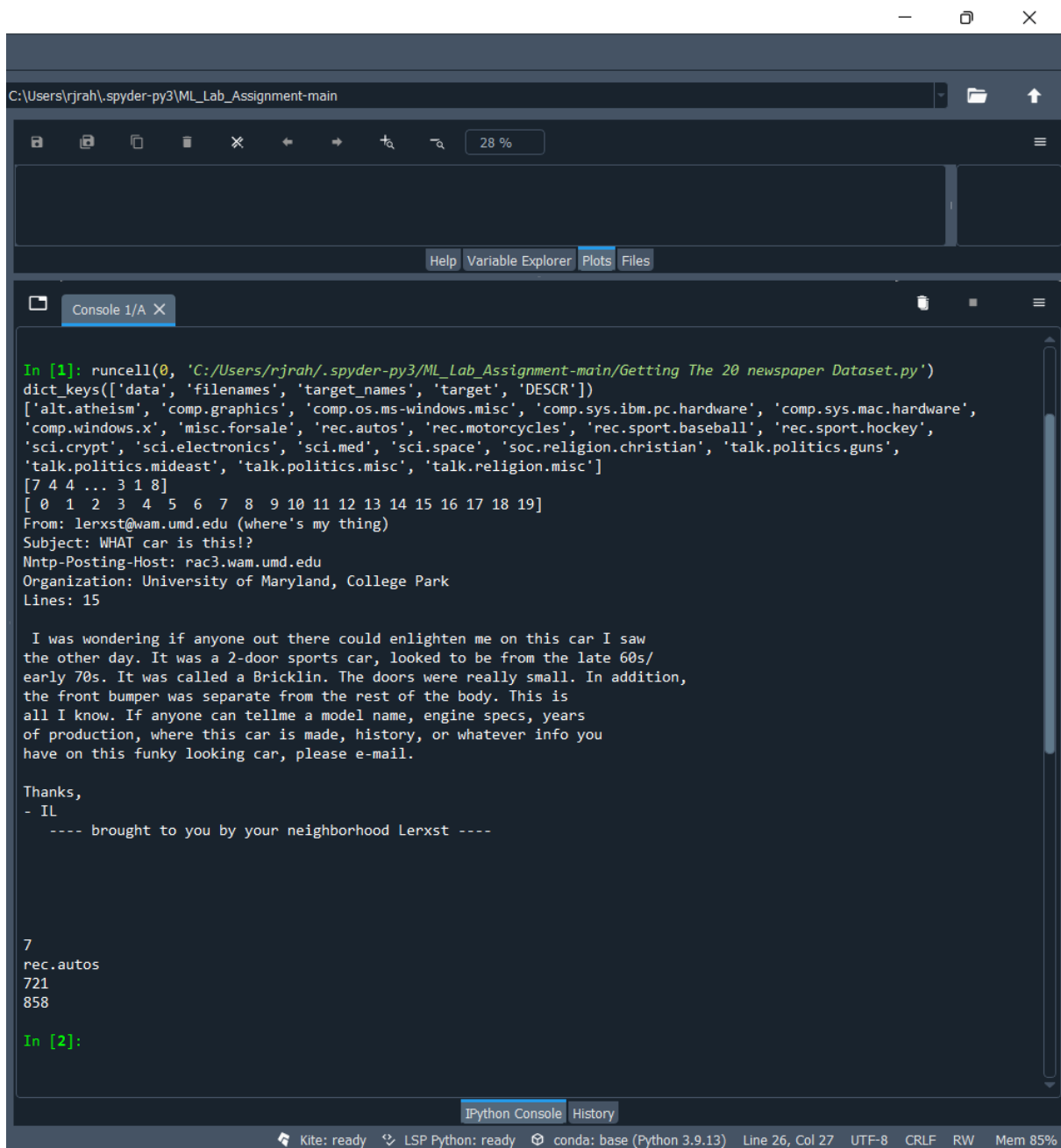
```
File Edit Search Source Run Debug Consoles Projects Tools View Help

C:\Users\rjrah\spyder-py3\ML_Lab_Assignment-main\Getting The 20 newspaper Dataset.py

NLTK Installation.py × Getting The 20 newspaper Dataset.py* × Clustering.py × Data Preprocessing.py × Topic Modelling.py ×

1  # -*- coding: utf-8 -*-
2  """
3  Created on Fri Sep  9 08:39:45 2022
4
5  @author: Rahul Sharma "M.Tech (Data Science)"
6  """
7
8  '''fetching 20 newsgroups dataset'''
9
10 from sklearn.datasets import fetch_20newsgroups
11 groups = fetch_20newsgroups()
12
13
14 '''thinking about feature'''
15
16 print(groups.keys())
17 print(groups['target_names'])
18 print(groups.target)
19
20 import numpy as np
21 print(np.unique(groups.target))
22 print(groups.data[0])
23 print(groups.target[0])
24 print(groups.target_names[groups.target[0]])
25 print(len(groups.data[0]))
26 print(len(groups.data[1]))
```

Output:



```
C:\Users\rjrah\spyder-py3\ML_Lab_Assignment-main

dict_keys(['data', 'filenames', 'target_names', 'target', 'DESCR'])
['alt.atheism', 'comp.graphics', 'comp.os.ms-windows.misc', 'comp.sys.ibm.pc.hardware', 'comp.sys.mac.hardware',
'comp.windows.x', 'misc.forsale', 'rec.autos', 'rec.motorcycles', 'rec.sport.baseball', 'rec.sport.hockey',
'sci.crypt', 'sci.electronics', 'sci.med', 'sci.space', 'soc.religion.christian', 'talk.politics.guns',
'talk.politics.mideast', 'talk.politics.misc', 'talk.religion.misc']
[7 4 4 ... 3 1 8]
[ 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19]
From: lerxst@wam.umd.edu (where's my thing)
Subject: WHAT car is this!?
Nntp-Posting-Host: rac3.wam.umd.edu
Organization: University of Maryland, College Park
Lines: 15

I was wondering if anyone out there could enlighten me on this car I saw
the other day. It was a 2-door sports car, looked to be from the late 60s/
early 70s. It was called a Bricklin. The doors were really small. In addition,
the front bumper was separate from the rest of the body. This is
all I know. If anyone can tellme a model name, engine specs, years
of production, where this car is made, history, or whatever info you
have on this funky looking car, please e-mail.

Thanks,
- IL
---- brought to you by your neighborhood Lerxst ----

7
rec.autos
721
858

In [2]:
```

Kite: ready LSP Python: ready conda: base (Python 3.9.13) Line 26, Col 27 UTF-8 CRLF RW Mem 85%

Milestone 5: Visualization

It's good to visualize to get a general idea of how the data is structured, what possible issues may arise, and if there are any irregularities that we have to take care of:

Source Code#

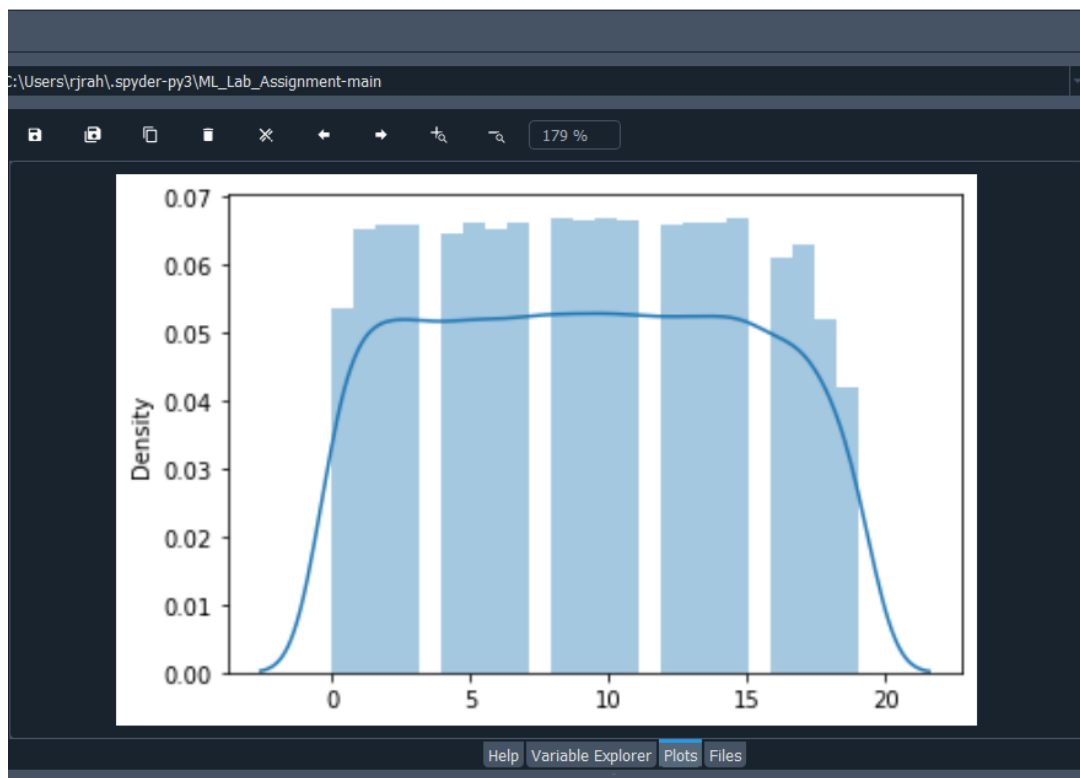
```
Spyder (Python 3.9)
File Edit Search Source Run Debug Consoles Projects Tools View Help

C:\Users\rjrah\spyder-py3\ML_Lab_Assignment-main\Visualization.py

NLTK Installation.py X Getting The 20 newspaper Dataset.py* X Visualization.py* X Clustering.py X Data Preprocessing.py X

1  # -*- coding: utf-8 -*-
2  """
3  Created on Fri Sep  9 14:42:59 2022
4
5  @author: Rahul Sharma "M.Tech (Data Science)"
6  """
7  from sklearn.datasets import fetch_20newsgroups
8  groups = fetch_20newsgroups()
9
10 """Visualization"""
11 import seaborn as sns
12 sns.distplot(groups.target)
13 import numpy as np
14 import matplotlib.pyplot as plt
15 plt.show()
16
```

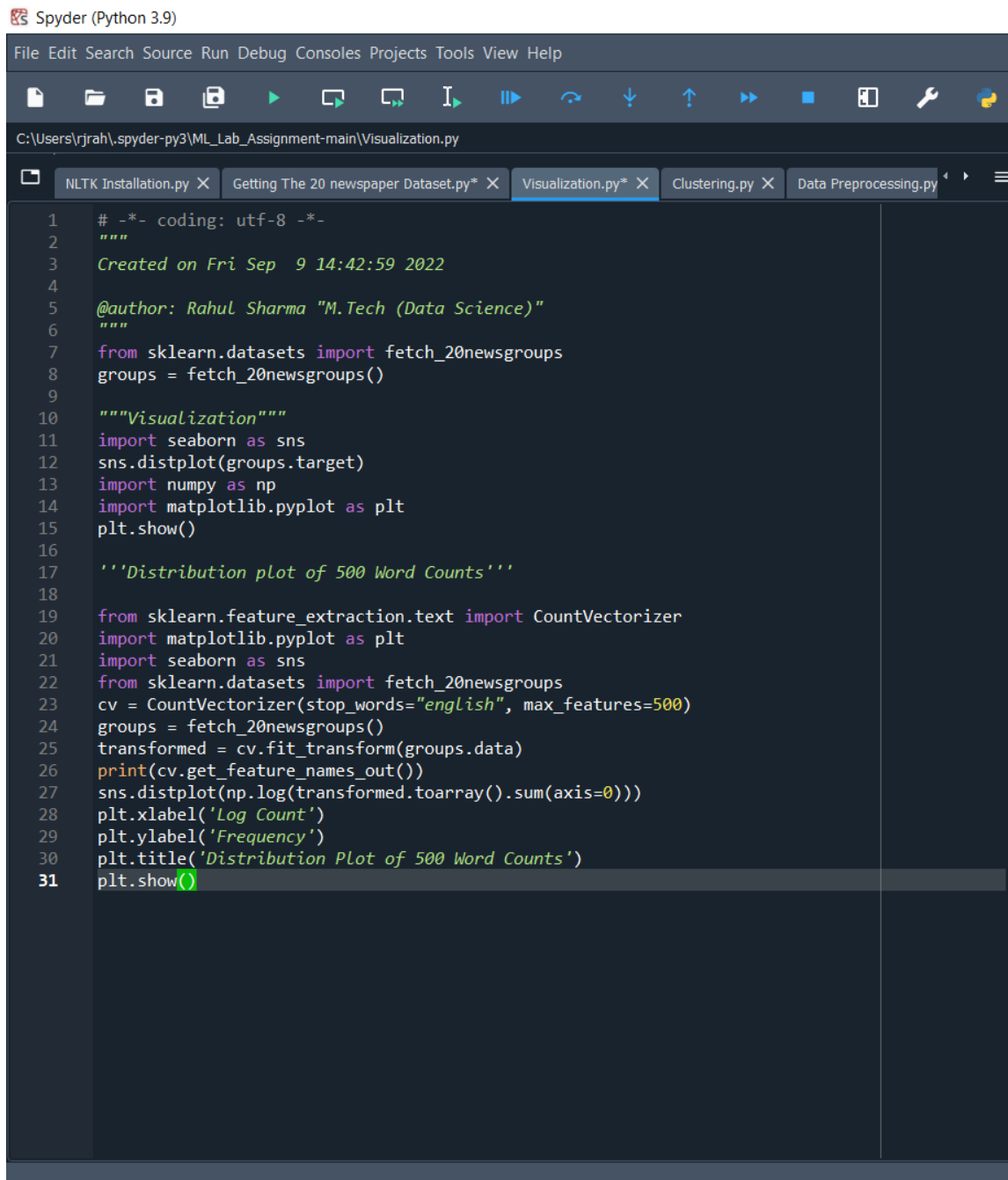
Output:



We can see, that the distribution is (approximately) uniform.

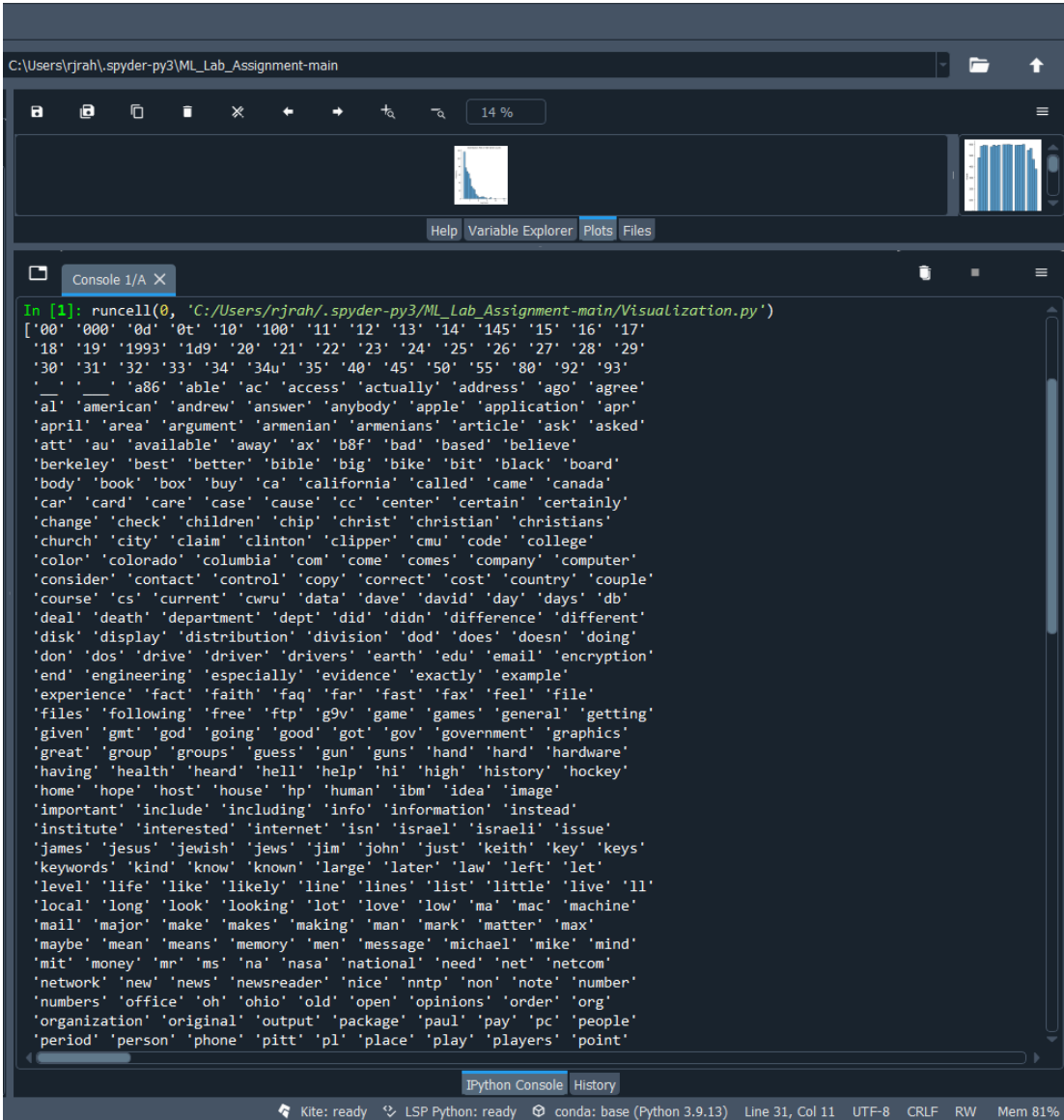
The following code displays a histogram of the 500 highest word counts:

Source code#



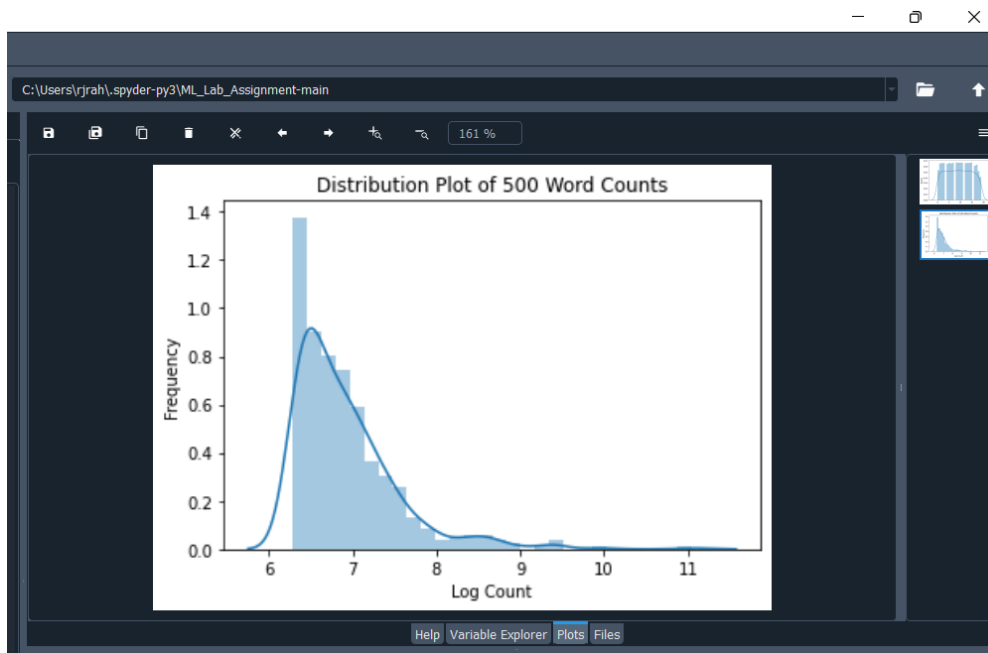
```
1  # -*- coding: utf-8 -*-
2  """
3  Created on Fri Sep  9 14:42:59 2022
4
5  @author: Rahul Sharma "M.Tech (Data Science)"
6  """
7  from sklearn.datasets import fetch_20newsgroups
8  groups = fetch_20newsgroups()
9
10  """Visualization"""
11  import seaborn as sns
12  sns.distplot(groups.target)
13  import numpy as np
14  import matplotlib.pyplot as plt
15  plt.show()
16
17  '''Distribution plot of 500 Word Counts'''
18
19  from sklearn.feature_extraction.text import CountVectorizer
20  import matplotlib.pyplot as plt
21  import seaborn as sns
22  from sklearn.datasets import fetch_20newsgroups
23  cv = CountVectorizer(stop_words="english", max_features=500)
24  groups = fetch_20newsgroups()
25  transformed = cv.fit_transform(groups.data)
26  print(cv.get_feature_names_out())
27  sns.distplot(np.log(transformed.toarray().sum(axis=0)))
28  plt.xlabel('Log Count')
29  plt.ylabel('Frequency')
30  plt.title('Distribution Plot of 500 Word Counts')
31  plt.show()
```

Output: list of 500 words



The screenshot shows a Jupyter Notebook window with a dark theme. The top bar displays the file path: `C:\Users\rjrah\.spyder-py3\ML_Lab_Assignment-main`. Below the toolbar, there is a small plot area showing a bar chart. The main console area displays the output of a Python script, which is a list of 500 words. The words are printed in a single line, separated by single quotes and commas. The words include: '000', '0d', '0t', '10', '100', '11', '12', '13', '14', '145', '15', '16', '17', '18', '19', '1993', '1d9', '20', '21', '22', '23', '24', '25', '26', '27', '28', '29', '30', '31', '32', '33', '34', '34u', '35', '40', '45', '50', '55', '80', '92', '93', 'a86', 'able', 'ac', 'access', 'actually', 'address', 'ago', 'agree', 'al', 'american', 'andrew', 'answer', 'anybody', 'apple', 'application', 'apn', 'april', 'area', 'argument', 'armenian', 'armenians', 'article', 'ask', 'asked', 'att', 'au', 'available', 'away', 'ax', 'b8f', 'bad', 'based', 'believe', 'berkeley', 'best', 'better', 'bible', 'big', 'bike', 'bit', 'black', 'board', 'body', 'book', 'box', 'buy', 'ca', 'california', 'called', 'came', 'canada', 'can', 'card', 'care', 'case', 'cause', 'cc', 'center', 'certain', 'certainly', 'change', 'check', 'children', 'chip', 'christ', 'christian', 'christians', 'church', 'city', 'claim', 'clinton', 'clipper', 'cmu', 'code', 'college', 'color', 'colorado', 'columbia', 'com', 'come', 'comes', 'company', 'computer', 'consider', 'contact', 'control', 'copy', 'correct', 'cost', 'country', 'couple', 'course', 'cs', 'current', 'cwru', 'data', 'dave', 'david', 'day', 'days', 'db', 'deal', 'death', 'department', 'dept', 'did', 'didn', 'difference', 'different', 'disk', 'display', 'distribution', 'division', 'dod', 'does', 'doesn', 'doing', 'don', 'dos', 'drive', 'driven', 'drivers', 'earth', 'edu', 'email', 'encryption', 'end', 'engineering', 'especially', 'evidence', 'exactly', 'example', 'experience', 'fact', 'faith', 'faq', 'far', 'fast', 'fax', 'feel', 'file', 'files', 'following', 'free', 'ftp', 'g9v', 'game', 'games', 'general', 'getting', 'given', 'gmt', 'god', 'going', 'good', 'got', 'gov', 'government', 'graphics', 'great', 'group', 'groups', 'guess', 'gun', 'guns', 'hand', 'hard', 'hardware', 'having', 'health', 'heard', 'hell', 'help', 'hi', 'high', 'history', 'hockey', 'home', 'hope', 'host', 'house', 'hp', 'human', 'ibm', 'idea', 'image', 'important', 'include', 'including', 'info', 'information', 'instead', 'institute', 'interested', 'internet', 'isn', 'israel', 'israeli', 'issue', 'james', 'jesus', 'jewish', 'jews', 'jim', 'john', 'just', 'keith', 'key', 'keys', 'keywords', 'kind', 'know', 'known', 'large', 'later', 'law', 'left', 'let', 'level', 'life', 'like', 'likely', 'line', 'lines', 'list', 'little', 'live', 'll', 'local', 'long', 'look', 'looking', 'lot', 'love', 'low', 'ma', 'mac', 'machine', 'mail', 'major', 'make', 'makes', 'making', 'man', 'mark', 'matter', 'max', 'maybe', 'mean', 'means', 'memory', 'men', 'message', 'michael', 'mike', 'mind', 'mit', 'money', 'mr', 'ms', 'na', 'nasa', 'national', 'need', 'net', 'netcom', 'network', 'new', 'news', 'newsreader', 'nice', 'nntp', 'non', 'note', 'number', 'numbers', 'office', 'oh', 'ohio', 'old', 'open', 'opinions', 'order', 'org', 'organization', 'original', 'output', 'package', 'paul', 'pay', 'pc', 'people', 'period', 'person', 'phone', 'pitt', 'pl', 'place', 'play', 'players', 'point'.

Distribution Plot of 500 Word Counts



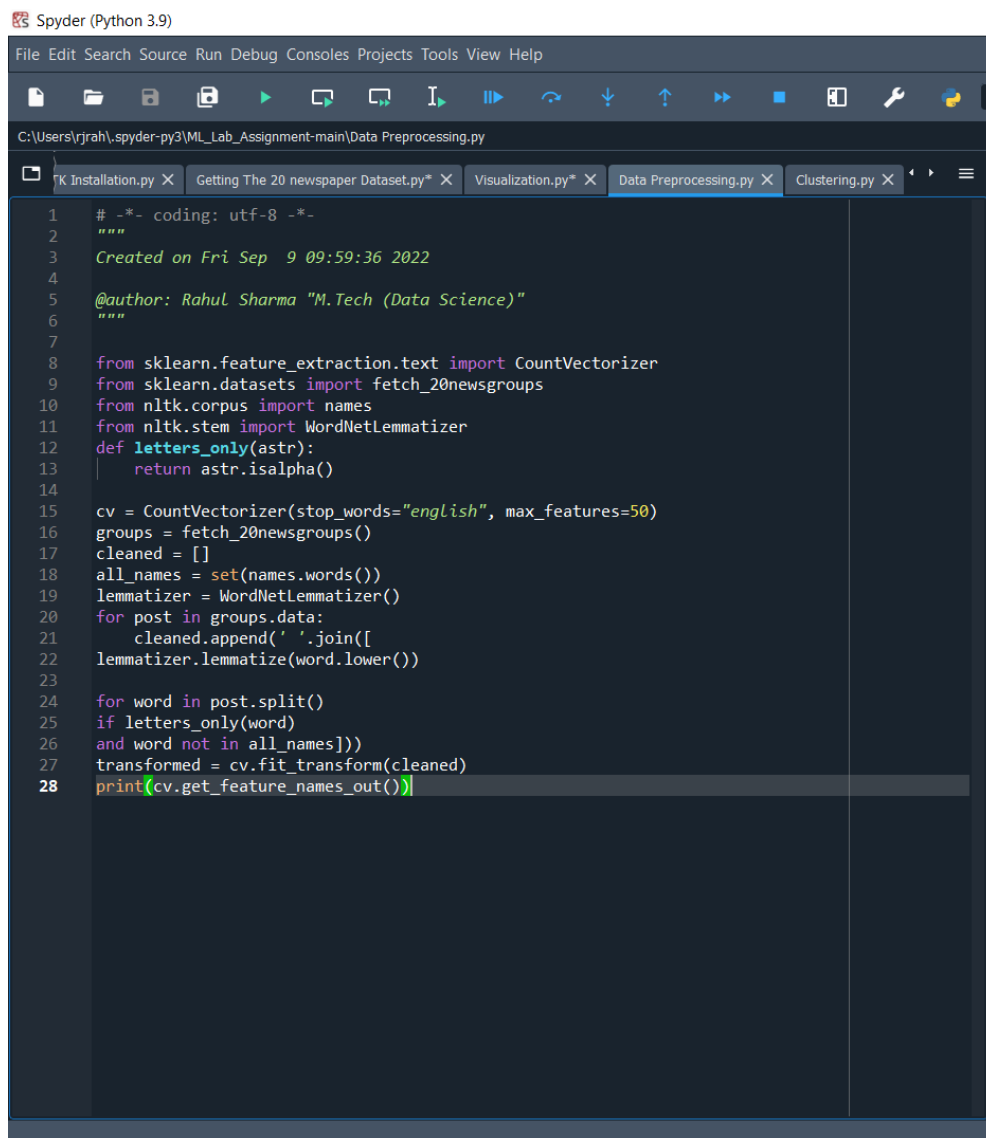
Milestone 6: Data Pre-processing

We see items, which are not words, such as 00 and 000. Maybe we should ignore items that contain only digits. However, 0d and 0t are also not words. We also see items as ___, so maybe we should only allow items that consist only of letters.

The posts contain names such as andrew as well. We can filter names with the Names corpus from NLTK.

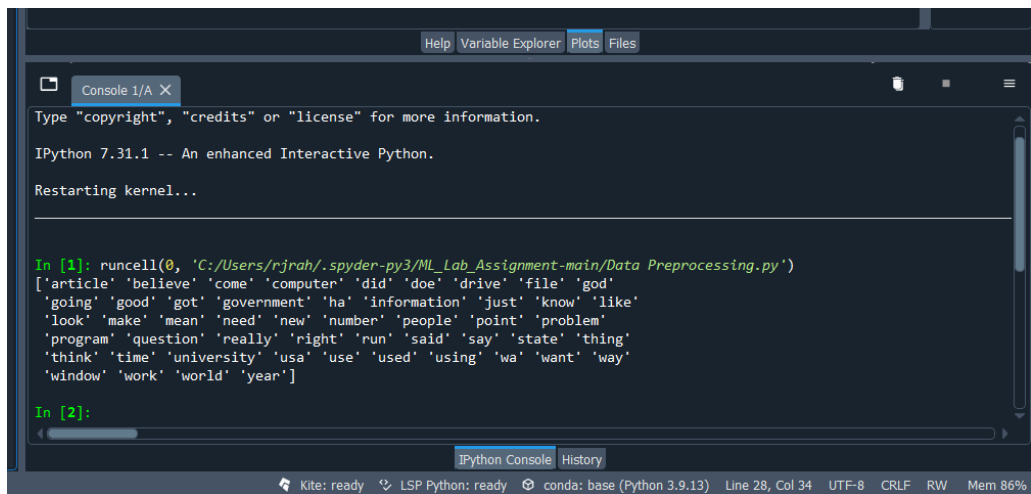
With every filtering we apply, we have to make sure that we don't lose information. Finally, we see words that are very similar, such as include and including, and make and makes.

Source Code#

A screenshot of the Spyder Python IDE interface. The title bar reads 'Spyder (Python 3.9)'. The menu bar includes 'File', 'Edit', 'Search', 'Source', 'Run', 'Debug', 'Consoles', 'Projects', 'Tools', 'View', and 'Help'. The toolbar contains icons for file operations (new, open, save, close), running (run, debug, interrupt), and other utilities. The file explorer on the left shows the project path: 'C:\Users\vrjah\spyder-py3\ML_Lab_Assignment-main\Data Preprocessing.py'. The editor window has several tabs: 'K Installation.py', 'Getting The 20 newspaper Dataset.py', 'Visualization.py', 'Data Preprocessing.py' (active), and 'Clustering.py'. The active tab displays a Python script for data preprocessing. The script includes a docstring with metadata, imports for CountVectorizer, fetch_20newsgroups, names, WordNetLemmatizer, and a custom letters_only function. It then fetches the dataset, cleans it by removing non-letter characters and words not in the NLTK names corpus, and finally prints the feature names after vectorization.

```
1  # -*- coding: utf-8 -*-
2  """
3  Created on Fri Sep  9 09:59:36 2022
4
5  @author: Rahul Sharma "M.Tech (Data Science)"
6  """
7
8  from sklearn.feature_extraction.text import CountVectorizer
9  from sklearn.datasets import fetch_20newsgroups
10 from nltk.corpus import names
11 from nltk.stem import WordNetLemmatizer
12 def letters_only(astr):
13     return astr.isalpha()
14
15 cv = CountVectorizer(stop_words="english", max_features=50)
16 groups = fetch_20newsgroups()
17 cleaned = []
18 all_names = set(names.words())
19 lemmatizer = WordNetLemmatizer()
20 for post in groups.data:
21     cleaned.append(' '.join([
22         lemmatizer.lemmatize(word.lower())
23     ]))
24 for word in post.split():
25     if letters_only(word)
26     and word not in all_names]))
27 transformed = cv.fit_transform(cleaned)
28 print(cv.get_feature_names_out())
```

Output:



```
Help Variable Explorer Plots Files

Console 1/A X
Type "copyright", "credits" or "license" for more information.

IPython 7.31.1 -- An enhanced Interactive Python.

Restarting kernel...

In [1]: runcell(0, 'C:/Users/rjrahl/.spyder-py3/ML_Lab_Assignment-main/Data_Preprocessing.py')
['article' 'believe' 'come' 'computer' 'did' 'doe' 'drive' 'file' 'god'
'going' 'good' 'got' 'government' 'ha' 'information' 'just' 'know' 'like'
'look' 'make' 'mean' 'need' 'new' 'number' 'people' 'point' 'problem'
'program' 'question' 'really' 'right' 'run' 'said' 'say' 'state' 'thing'
'think' 'time' 'university' 'usa' 'use' 'used' 'using' 'wa' 'want' 'way'
>window' 'work' 'world' 'year']

In [2]:
```

Kite: ready LSP Python: ready conda: base (Python 3.9.13) Line 28, Col 34 UTF-8 CRLF RW Mem 86%

Milestone 7: Clustering

Clustering is the task of dividing the population or data points into several groups such that data points in the same groups are more similar to other data points in the same group and dissimilar to the data points in other groups. It is a collection of objects based on similarity and dissimilarity between them.

Source Code##

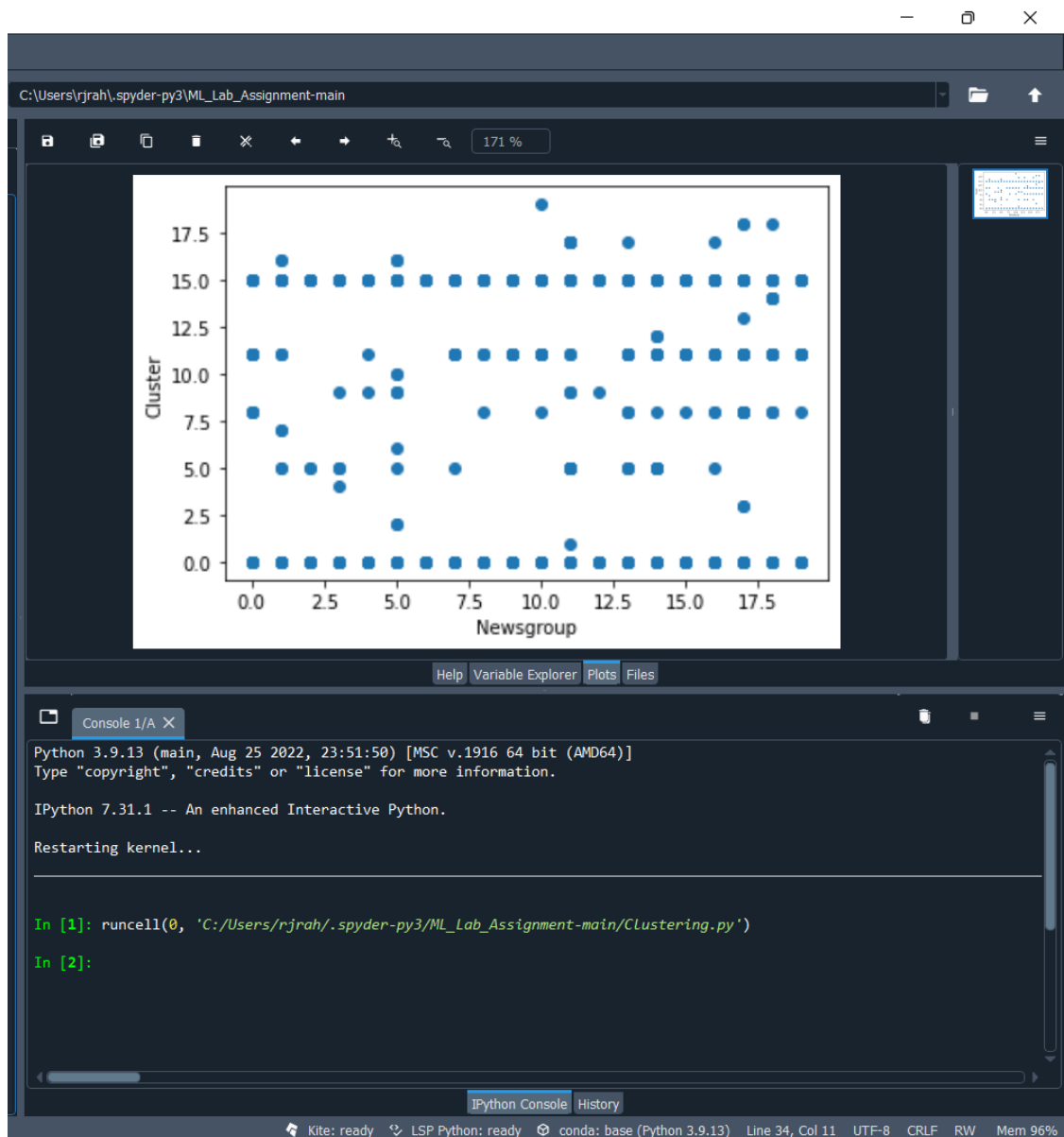
```
Spyder (Python 3.9)
File Edit Search Source Run Debug Consoles Projects Tools View Help

C:\Users\rjrah\spyder-py3\ML_Lab_Assignment-main\Clustering.py

K Installation.py X Getting The 20 newspaper Dataset.py* X Visualization.py* X Data Preprocessing.py X Clustering.py X

1  #-*- coding: utf-8 -*-
2  """
3  Created on Fri Sep  9 14:35:24 2022
4
5  @author: Rahul Sharma "M.Tech (Data Science)"
6  """
7  from sklearn.feature_extraction.text import CountVectorizer
8  from sklearn.datasets import fetch_20newsgroups
9  from nltk.corpus import names
10 from nltk.stem import WordNetLemmatizer
11 from sklearn.cluster import KMeans
12 import matplotlib.pyplot as plt
13 def letters_only(astr):
14     return astr.isalpha()
15
16 cv = CountVectorizer(stop_words="english", max_features=500)
17 groups = fetch_20newsgroups()
18 cleaned = []
19 all_names = set(names.words())
20 lemmatizer = WordNetLemmatizer()
21 for post in groups.data:
22     cleaned.append(' '.join([
23         lemmatizer.lemmatize(word.lower())
24         for word in post.split()
25         if letters_only(word)
26         and word not in all_names]))
27 transformed = cv.fit_transform(cleaned)
28 km = KMeans(n_clusters=20)
29 km.fit(transformed)
30 labels = groups.target
31 plt.scatter(labels, km.labels_)
32 plt.xlabel('Newsgroup')
33 plt.ylabel('Cluster')
34 plt.show()
```

Output/Plot:



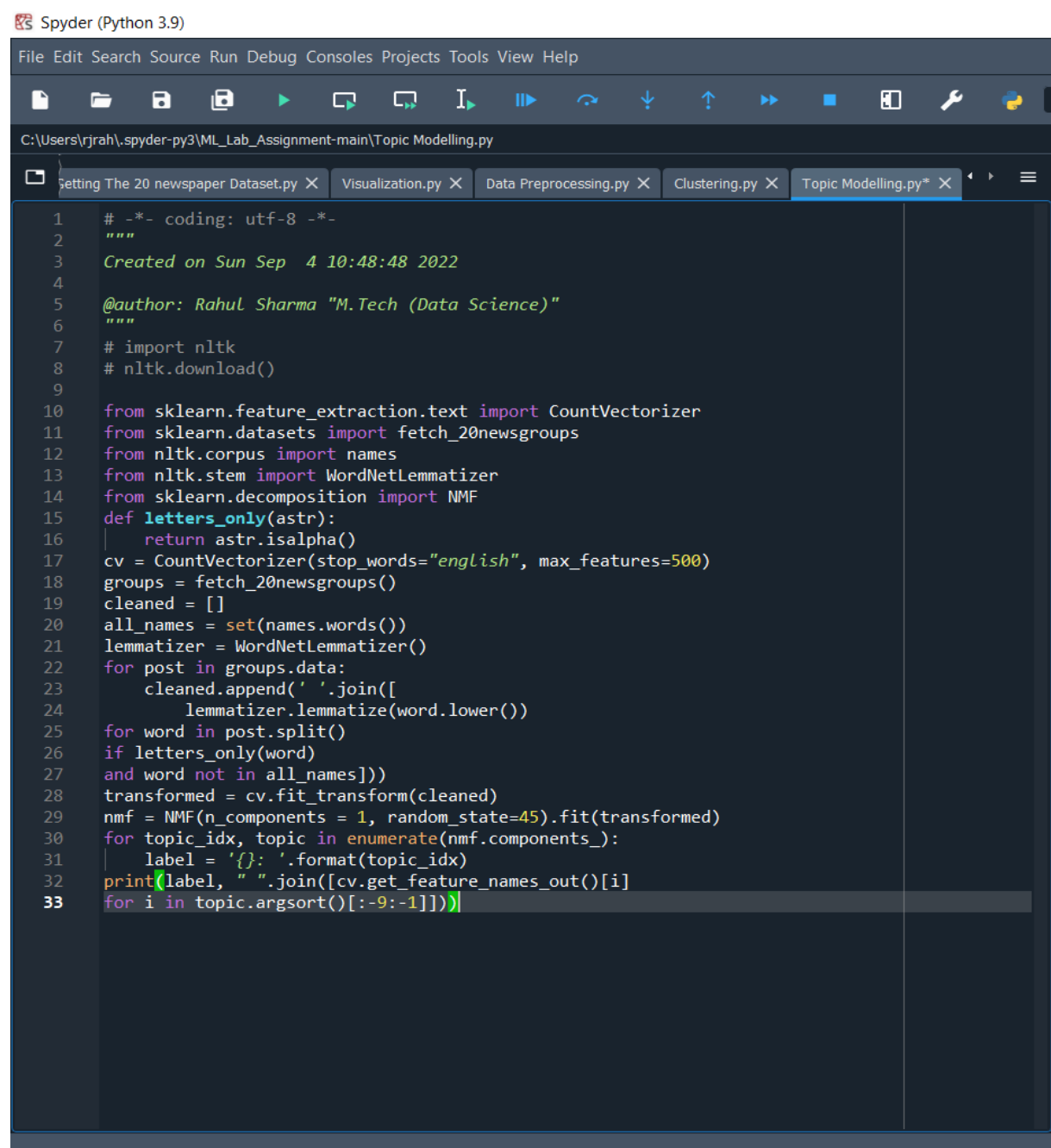
Milestone 8: Topic Modeling

Topic Modeling falls under unsupervised machine learning where the documents are processed to obtain the relative topics. It is a very important concept of the traditional Natural Processing Approach because of its potential to obtain a semantic relationship between words in the document clusters. In addition, it has numerous other applications in NLP.

Now let us have a look at the Non-Negative Matrix Factorization. NMF is a statistical method to reduce the dimension of the input corpora. It uses the factor analysis method to provide comparatively less weightage to the words with less coherence.

Source Code#

Note: “By changing the value n_component with a positive integer”



The screenshot shows the Spyder Python IDE interface. The top menu bar includes File, Edit, Search, Source, Run, Debug, Consoles, Projects, Tools, View, and Help. Below the menu is a toolbar with various icons for file operations and execution. The main window displays a Python script titled 'Topic Modelling.py' located at 'C:\Users\rjrah\spyder-py3\ML_Lab_Assignment-main\Topic Modelling.py'. The script is a Jupyter-style notebook with a dark background and syntax highlighting. It includes a header with a comment about the file's creation date and author, followed by imports for sklearn, nltk, and CountVectorizer. The main logic is in the 'letters_only' function, which processes the 20 Newsgroups dataset, filters out non-alphabetic characters, and applies NMF with 1 component. The script ends with a loop to print the top 9 words for each topic.

```
1  # -*- coding: utf-8 -*-
2  """
3  Created on Sun Sep  4 10:48:48 2022
4
5  @author: Rahul Sharma "M.Tech (Data Science)"
6  """
7  # import nltk
8  # nltk.download()
9
10 from sklearn.feature_extraction.text import CountVectorizer
11 from sklearn.datasets import fetch_20newsgroups
12 from nltk.corpus import names
13 from nltk.stem import WordNetLemmatizer
14 from sklearn.decomposition import NMF
15 def letters_only(astr):
16     return astr.isalpha()
17 cv = CountVectorizer(stop_words="english", max_features=500)
18 groups = fetch_20newsgroups()
19 cleaned = []
20 all_names = set(names.words())
21 lemmatizer = WordNetLemmatizer()
22 for post in groups.data:
23     cleaned.append(' '.join([
24         lemmatizer.lemmatize(word.lower())
25     for word in post.split()
26     if letters_only(word)
27     and word not in all_names]))
28 transformed = cv.fit_transform(cleaned)
29 nmf = NMF(n_components = 1, random_state=45).fit(transformed)
30 for topic_idx, topic in enumerate(nmf.components_):
31     label = '{}: '.format(topic_idx)
32     print(label, " ".join([cv.get_feature_names_out()[i]
33     for i in topic.argsort()[:-9:-1]]))
```

Output:

- 0: wa ha people know just like use think
- 1: db bit data stuff place add time line
- 2: file use ha program image available version information
- 3: file image program available use entry version window
- 4: drive disk hard controller support card scsi head
- 5: ha think people new make president doe just
- 6: new anonymous ha use information computer hockey service
- 7: image file available version data color program software
- 8: window widget use available application version set server
- 9: anonymous use information computer internet user service posting
- 10: people god ha doe just article like make
- 11: gun state use law new united control government
- 12: gun state use law new control united national
- 13: space nasa data ha center national research technology
- 14: anonymous posting information internet user computer email service
- 15: people like just think want good make know
- 16: people like just think know want thing right
- 17: widget application value data set type use return
- 18: health medical center number university child research use
- 19: ha article new like time year doe wa
- 20: key encryption government technology law use security chip
- 21: president think going ha wa know said russian
- 22: use need ground power box usually used doe
- 23: russian government new american program state way support
- 24: ibm scsi memory color use card hardware program
- 25: anonymous posting internet user information email computer service
- 26: la win san year game period power shot
- 27: new military war state time united world attack
- 28: encryption technology government law new use access data

29: drive scsi tape single using driver mode data

30: god atheist believe doe belief religion religious say

31: god atheist believe doe belief religion religious say

32: god atheist believe doe belief religion religious say

33: university center research new medical science article institute

34: university center research new medical institute study science

35: water city new car division san list high

36: good like think just make time really better

37: article university know post read doe usa science

38: good car like better make time year just

39: drive scsi single head mode set using model

40: israeli arab question jew human true attack state

41: like just dod make time look thing really

42: know just like going went come woman came

43: graphic send mail message server support computer package

44: graphic send mail message server support computer package

45: article university post read doe usa discussion world

46: right israeli state human government arab second law

47: power period second game shot got play goal

48: dod time question number computer member list bike

49: article university post read usa world discussion opinion

50: doe say argument make question word point mean

51: just time going maybe make come start got

52: information san said police group league political including

53: israeli arab jew question human attack policy true

54: make thing little le difference better way sure

55: university computer science general department engineering thanks state

56: article university post read world usa discussion opinion

57: university science computer state general department thanks engineering

58: gun control crime rate weapon house american united

59: people say said come tell government live woman
60: graphic send mail message server support package line
61: just start maybe going want really thought woman
62: information san said police group league political including
63: want thing need work help really going school
64: san said information police group league political including
65: game team got win play shot year run
66: article university post read usa world discussion opinion
67: new york year question sale change service old
68: article usa read world post opinion discussion sure
69: san information police said group league political government
70: say word believe mean thing did christian point
71: point way human case child different line better
72: car driver speed change better number high point
73: want need help let really thing better point
74: chip unit clipper serial number disk algorithm bit
75: article post usa world read opinion discussion sure
76: university science general thanks department computer engineering texas
77: say word believe mean christian did act point
78: believe belief christian truth evidence church claim different
79: need help phone kind able needed thanks bike
80: israeli arab human attack policy true state fact
81: university science general department thanks engineering computer texas
82: say word believe christian mean did act day
83: wa did thought used later order seen early
84: want need help let really life better reason
85: university science general thanks department engineering computer texas
86: power period second san special le play shot
87: time long lot better having able order actually
88: card driver video support mouse mode board bus

89: did death let fact money body place man

90: woman men world muslim life religion man great

91: believe belief christian truth evidence claim different church

92: like look really sound better going long little

93: gun rate crime city death study control difference

94: really better lot player probably little best big

95: going come way mean sure working look kind

96: point way different line better algorithm issue view

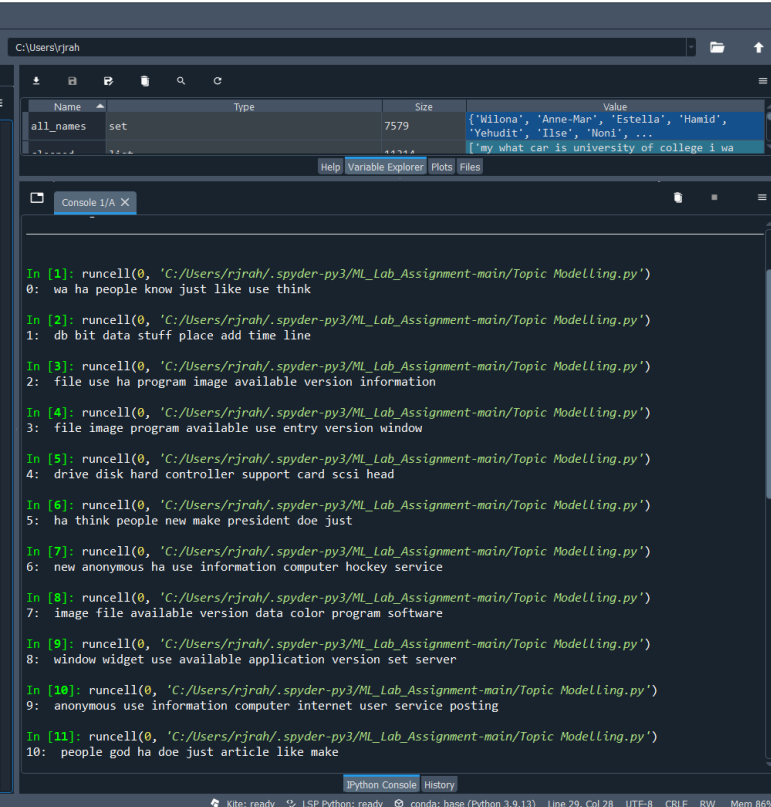
97: just maybe start woman big thought probably getting

98: government political federal free sure private local country

99: article read world usa opinion post discussion sure

100: state united public national political federal member local

Screenshots of the above given Output



The screenshot displays the Spyder Python IDE interface. At the top, the file explorer shows the path 'C:/Users/rjrah'. Below it, the variable explorer shows a variable named 'all_names' of type 'set' with a size of 7579. The value of 'all_names' is a list of names: 'Wilona', 'Anne-Mar', 'Estella', 'Hamid', 'Yehudit', 'Ilse', 'Moni', and 'my what can is university of college i wa'. Below the variable explorer, the console window shows the output of 11 runcell commands. Each command is followed by a line of text output. The status bar at the bottom indicates 'Kite: ready', 'LSP Python: ready', 'conda: base (Python 3.9.13)', 'Line 29, Col 28', 'UTF-8', 'CRLF', 'RW', and 'Mem 86%'.

```

In [1]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
0: wa ha people know just like use think

In [2]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
1: db bit data stuff place add time line

In [3]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
2: file use ha program image available version information

In [4]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
3: file image program available use entry version window

In [5]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
4: drive disk hard controller support card scsi head

In [6]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
5: ha think people new make president doe just

In [7]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
6: new anonymous ha use information computer hockey service

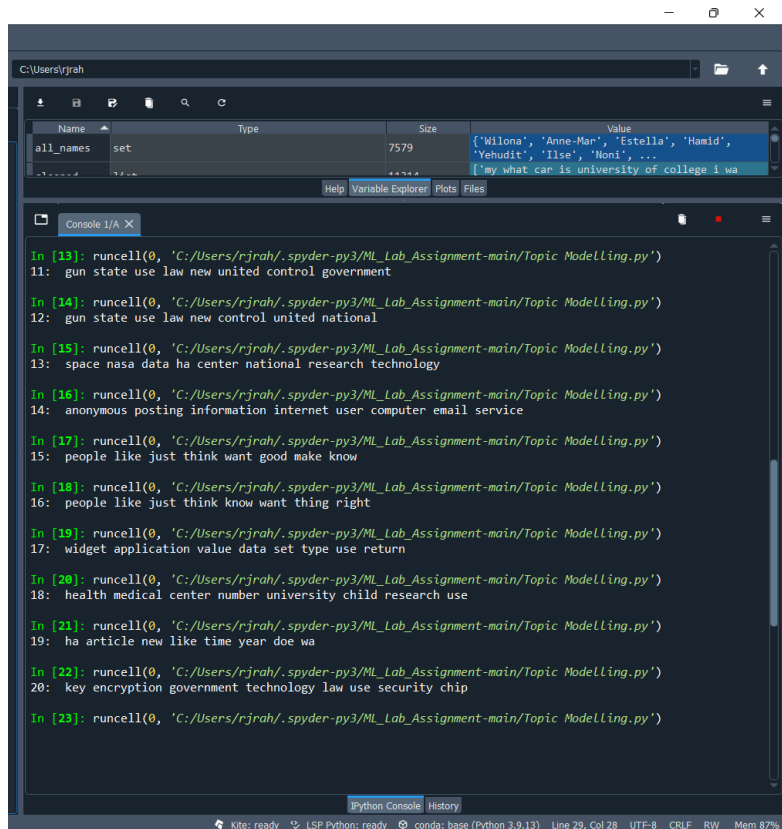
In [8]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
7: image file available version data color program software

In [9]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
8: window widget use available application version set server

In [10]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
9: anonymous use information computer internet user service posting

In [11]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
10: people god ha doe just article like make

```



C:\Users\rjrah

Name	Type	Size	Value
all_names	set	7579	{'Wilona', 'Anne-Mar', 'Estella', 'Hamid', 'Yehudit', 'Ilse', 'Noni', ...}

Help Variable Explorer Plots Files

Console 1/A X

```
In [13]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
11: gun state use law new united control government

In [14]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
12: gun state use law new control united national

In [15]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
13: space nasa data ha center national research technology

In [16]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
14: anonymous posting information internet user computer email service

In [17]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
15: people like just think want good make know

In [18]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
16: people like just think know want thing right

In [19]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
17: widget application value data set type use return

In [20]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
18: health medical center number university child research use

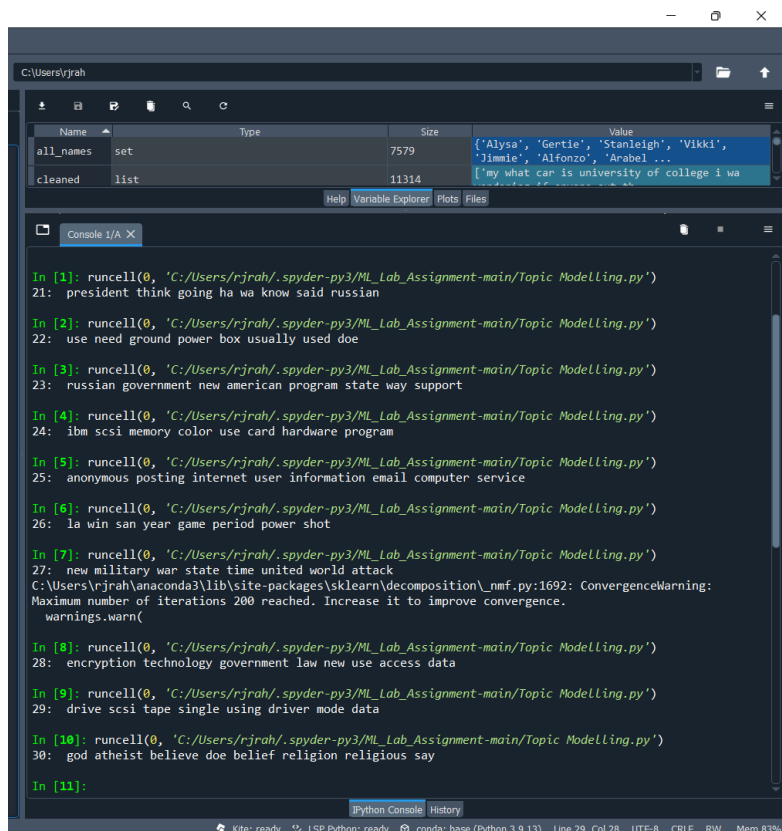
In [21]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
19: ha article new like time year doe wa

In [22]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
20: key encryption government technology law use security chip

In [23]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
```

Python Console History

Kite: ready LSP Python: ready conda: base (Python 3.9.13) Line 29, Col 28 UTF-8 CRLF RW Mem 87%



C:\Users\rjrah

Name	Type	Size	Value
all_names	set	7579	{'Alysa', 'Gertie', 'Stanleigh', 'Vikki', 'Jimmie', 'Alfonzo', 'Arabel ...}
cleaned	list	11314	['my what car is university of college i wa ...']

Help Variable Explorer Plots Files

Console 1/A X

```
In [1]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
21: president think going ha wa know said russian

In [2]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
22: use need ground power box usually used doe

In [3]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
23: russian government new american program state way support

In [4]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
24: ibm scsi memory color use card hardware program

In [5]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
25: anonymous posting internet user information email computer service

In [6]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
26: la win san year game period power shot

In [7]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
27: new military war state time united world attack
C:\Users\rjrah\anaconda3\lib\site-packages\sklearn\decomposition\_nmf.py:1692: ConvergenceWarning:
Maximum number of iterations 200 reached. Increase it to improve convergence.
  warnings.warn(

In [8]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
28: encryption technology government law new use access data

In [9]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
29: drive scsi tape single using driver mode data

In [10]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
30: god atheist believe doe belief religion religious say

In [11]:
```

Python Console History

Kite: ready LSP Python: ready conda: base (Python 3.9.13) Line 29, Col 28 UTF-8 CRLF RW Mem 83%

The screenshot shows the Spyder IDE interface. The top panel is the Variable Explorer, displaying two variables: 'all_names' (a set with 7579 elements) and 'cleaned' (a list with 11314 elements). The bottom panel is the IPython console, showing a series of 11 interactive commands and their outputs. The console output includes the Python version (3.9.13), the IPython version (7.31.1), and the results of 11 'runcell' commands, each followed by a line of text.

Name	Type	Size	Value
all_names	set	7579	{'Erhard', 'Xeven', 'Aamin', 'Garrott', 'Terry', 'Lorianna', 'Arvie', ...}
cleaned	list	11314	['my what car is university of college i wa ...']

```
Python 3.9.13 (main, Aug 25 2022, 23:51:50) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license()" for more information.

IPython 7.31.1 -- An enhanced Interactive Python.

In [1]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
31: god atheist believe doe belief religion religious say

In [2]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
32: god atheist believe doe belief religion religious say

In [3]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
33: university center research new medical science article institute

In [4]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
34: university center research new medical institute study science

In [5]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
35: water city new car division san list high

In [6]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
36: good like think just make time really better

In [7]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
37: article university know post read doe usa science

In [8]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
38: good car like better make time year just

In [9]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
39: drive scsi single head mode set using model

In [10]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
40: israeli arab question jew human true attack state

In [11]:
```

The screenshot shows the Spyder IDE interface. The top panel is the Variable Explorer, displaying two variables: 'all_names' (a set with 7579 elements) and 'cleaned' (a list with 11314 elements). The bottom panel is the IPython console, showing a series of 11 interactive commands and their outputs. The console output includes the Python version (3.9.13), the IPython version (7.31.1), and the results of 11 'runcell' commands, each followed by a line of text.

Name	Type	Size	Value
all_names	set	7579	{'Brody', 'Teddi', 'Elli', 'Hans-Peter', 'Myrtice', 'Binnie', 'Emalynn' ...}
cleaned	list	11314	['my what car is university of college i wa ...']

```
Python 3.9.13 (main, Aug 25 2022, 23:51:50) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license()" for more information.

IPython 7.31.1 -- An enhanced Interactive Python.

In [1]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
41: like just dod make time look thing really

In [2]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
42: know just like going went come woman came

In [3]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
43: graphic send mail message server support computer package

In [4]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
44: graphic send mail message server support computer package

In [5]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
45: article university post read doe usa discussion world

In [6]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
46: right israeli state human government arab second law

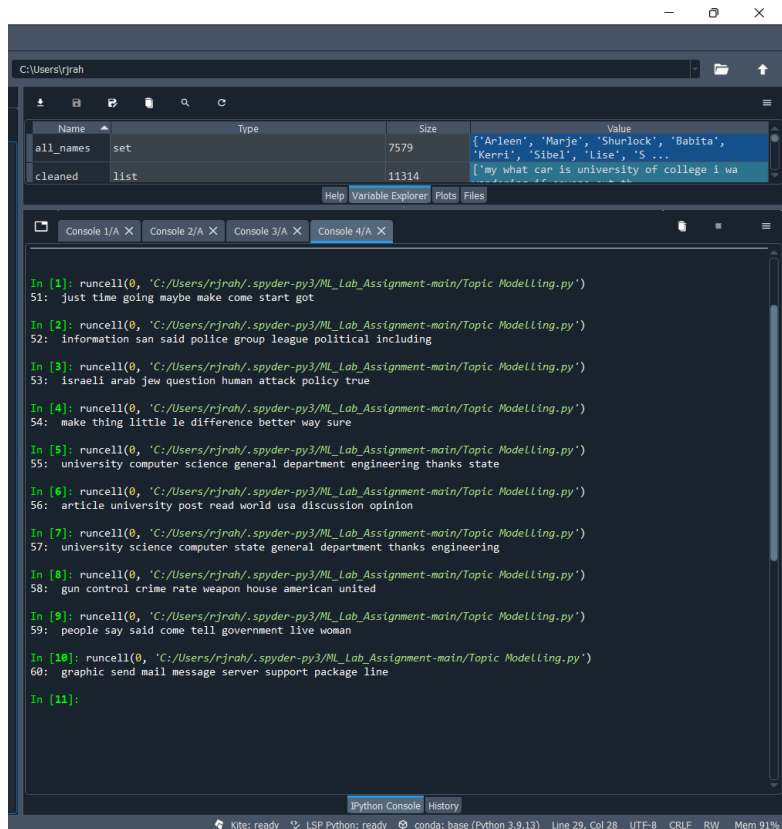
In [7]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
47: power period second game shot got play goal

In [8]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
48: dod time question number computer member list bike

In [9]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
49: article university post read usa world discussion opinion

In [10]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
50: doe say argument make question word point mean

In [11]: |
```



The screenshot shows a Jupyter Notebook interface with a dark theme. At the top, the file explorer shows the path 'C:\Users\rjrah'. Below it, the variable explorer displays two variables: 'all_names' (a set with 7579 elements) and 'cleaned' (a list with 11314 elements). The console shows the execution of 11 cells of code, each starting with 'In [X]:'. The code cells contain various text-based operations and data manipulations. The status bar at the bottom indicates 'Kite: ready', 'LSP Python: ready', 'conda: base (Python 3.9.13)', 'Line 29, Col 28', 'UTF-8', 'CRLF', 'RW', and 'Mem 91%'.

Name	Type	Size	Value
all_names	set	7579	{'Arleen', 'Marje', 'Shurlock', 'Babita', 'Kerri', 'Sibel', 'Lise', 'S ...
cleaned	list	11314	['my what car is university of college i wa ...

```
In [1]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
51: just time going maybe make come start got

In [2]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
52: information san said police group league political including

In [3]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
53: israeli arab jew question human attack policy true

In [4]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
54: make thing little le difference better way sure

In [5]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
55: university computer science general department engineering thanks state

In [6]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
56: article university post read world usa discussion opinion

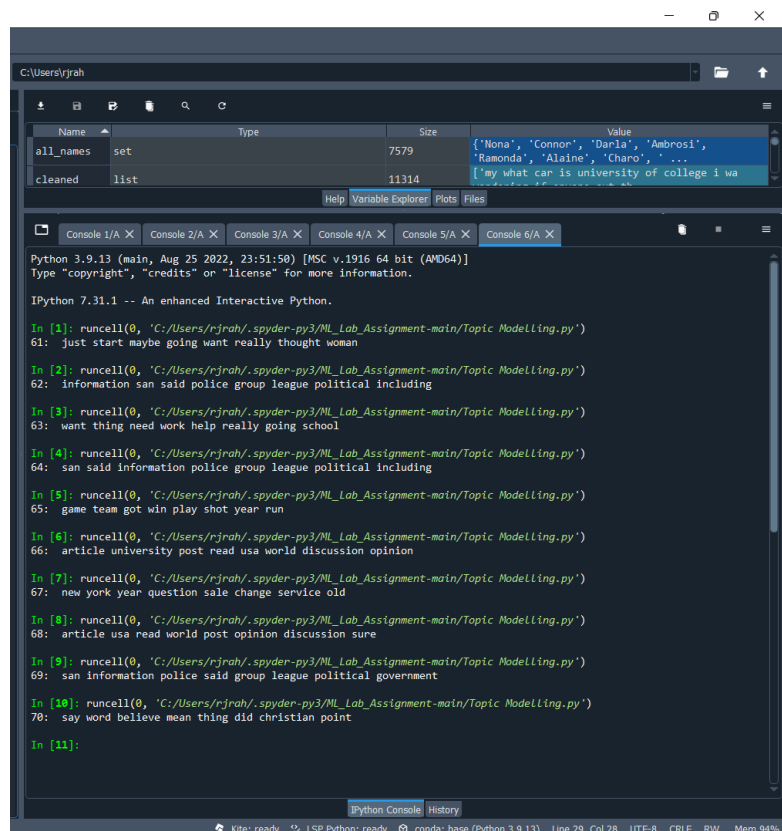
In [7]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
57: university science computer state general department thanks engineering

In [8]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
58: gun control crime rate weapon house american united

In [9]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
59: people say said come tell government live woman

In [10]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
60: graphic send mail message server support package line

In [11]:
```



The screenshot shows a Jupyter Notebook interface with a dark theme. At the top, the file explorer shows the path 'C:\Users\rjrah'. Below it, the variable explorer displays two variables: 'all_names' (a set with 7579 elements) and 'cleaned' (a list with 11314 elements). The console shows the execution of 11 cells of code, each starting with 'In [X]:'. The code cells contain various text-based operations and data manipulations. The status bar at the bottom indicates 'Kite: ready', 'LSP Python: ready', 'conda: base (Python 3.9.13)', 'Line 29, Col 28', 'UTF-8', 'CRLF', 'RW', and 'Mem 94%'.

Name	Type	Size	Value
all_names	set	7579	{'Nona', 'Connon', 'Darla', 'Ambrosi', 'Ramonda', 'Aline', 'Charo', ' ...
cleaned	list	11314	['my what car is university of college i wa ...

```
Python 3.9.13 (main, Aug 25 2022, 23:51:50) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license()" for more information.

IPython 7.31.1 -- An enhanced Interactive Python.

In [1]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
61: just start maybe going want really thought woman

In [2]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
62: information san said police group league political including

In [3]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
63: want thing need work help really going school

In [4]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
64: san said information police group league political including

In [5]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
65: game team got win play shot year run

In [6]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
66: article university post read usa world discussion opinion

In [7]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
67: new york year question sale change service old

In [8]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
68: article usa read world post opinion discussion sure

In [9]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
69: san information police said group league political government

In [10]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
70: say word believe mean thing did christian point

In [11]:
```

C:\Users\rjrah

Name	Type	Size	Value
all_names	set	7579	{'Jared', 'Halie', 'Dugan', 'Emilee', 'Roxy', 'Lana', 'Jacinthe', 'Kri ...
cleaned	list	11314	['my what car is university of college i wa

Help Variable Explorer Plots Files

Console 1/A X Console 2/A X Console 3/A X Console 4/A X Console 5/A X Console 6/A X Console 7/A X

```
Python 3.9.13 (main, Aug 25 2022, 23:51:50) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 7.31.1 -- An enhanced Interactive Python.

In [1]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
71: point way human case child different line better

In [2]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
72: car driver speed change better number high point

In [3]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
73: want need help let really thing better point

In [4]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
74: chip unit clipper serial number disk algorithm bit

In [5]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
75: article post usa world read opinion discussion sure

In [6]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
76: university science general thanks department computer engineering texas

In [7]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
77: say word believe mean christian did act point

In [8]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
78: believe belief christian truth evidence church claim different

In [9]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
80: israeli arab human attack policy true state fact

In [10]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
79: need help phone kind able needed thanks bike

In [11]:
```

Python Console History

Kite: ready LSP Python: ready conda: base (Python 3.9.13) Line 29, Col 28 UTF-8 CRLF RW Mem 87%

C:\Users\rjrah

Name	Type	Size	Value
all_names	set	7579	{'Gustav', 'Hank', 'Colene', 'Kristy', 'Yehudit', 'Erin', 'Vittoria', ...
cleaned	list	11314	['my what car is university of college i wa

Help Variable Explorer Plots Files

Console 2/A X Console 3/A X Console 4/A X Console 5/A X Console 6/A X Console 7/A X Console 8/A X

```
Python 3.9.13 (main, Aug 25 2022, 23:51:50) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 7.31.1 -- An enhanced Interactive Python.

In [1]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
81: university science general department thanks engineering computer texas

In [2]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
82: say word believe christian mean did act day

In [3]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
83: wa did thought used later order seen early

In [4]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
84: want need help let really life better reason

In [5]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
85: university science general thanks department engineering computer texas

In [6]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
86: power period second san special le play shot

In [7]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
87: time long lot better having able order actually

In [8]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
88: card driver video support mouse mode board bus

In [9]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
89: did death let fact money body place man

In [10]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
90: woman men world muslim life religion man great

In [11]:
```

Python Console History

Kite: ready LSP Python: ready conda: base (Python 3.9.13) Line 29, Col 28 UTF-8 CRLF RW Mem 85%

```
In [1]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
91: believe belief christian truth evidence claim different church

In [2]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
92: like look really sound better going long little

In [3]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
93: gun rate crime city death study control difference

In [4]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
94: really better lot player probably little best big

In [5]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
95: going come way mean sure working look kind
C:\Users\rjrah\anaconda3\lib\site-packages\sklearn\decomposition\_nmf.py:1692: ConvergenceWarning:
Maximum number of iterations 200 reached. Increase it to improve convergence.
warnings.warn(

In [6]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
96: point way different line better algorithm issue view

In [7]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
97: just maybe start woman big thought probably getting

In [8]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
98: government political federal free sure private local country

In [9]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
99: article read world usa opinion post discussion sure

In [10]: runcell(0, 'C:/Users/rjrah/.spyder-py3/ML_Lab_Assignment-main/Topic Modelling.py')
100: state united public national political federal member local

In [11]:
```

GitHub-Repository:

This link redirect to my all-ML Lab Assignment's source code repository on GitHub

https://github.com/cse-rahulsharma/ML_Lab_Assignment.git