```
from bs4 import BeautifulSoup as bs
import requests
import pandas as pd
import matplotlib.pyplot as plt

urls=["https://indianexpress.com/article/cities/chandigarh/four-killed-two-separate-accidents-8459637/",
      "https://indianexpress.com/article/world/16-killed-66-injured-multi-vehicle-collision-china-report-8425715/",
      "https://www.ndtv.com/india-news/bus-conductor-falls-dies-after-driver-applies-sudden-brake-3280562",
      "https://www.thehindu.com/news/national/karnataka/youth-dies-in-road-accident/article66639396.ece"]
headers={'user-agent':'https://blog.feedspot.com/auto_accident_blogs/'}

for url in urls:
  page=requests.get(url, headers=headers)

  soup=bs(page.content,'html.parser')

  Contents=soup.findAll("p")

Scrapped_data=[]
for content in Contents:
  Text=content.get_text()
  Scrapped_data.append([Text])

df=pd.DataFrame(Scrapped_data,columns=["Text"])

display(df)

df.to_json("Accident.json")

df["Text"][14]
```

	Text
0	To enjoy additional benefits
1	CONNECT WITH US
2	March 19, 2023 09:54 pm | Updated 09:54 pm IS...
3	\nCOMMents\n
4	SHARE\n
5	\nREAD LATER\n
6	A 23-year-old youth died in a road accident in...
7	Prateek Hongal was riding his bike on his way ...
8	His brother, Praveen Hongal, has blamed the ne...
9	A case has been registered.
10	\nCOMMents\n
11	SHARE\n
12	BACK TO TOP
13	Comments have to be in English, and in full se...
14	We have migrated to a new commenting platform....

```
'We have migrated to a new commenting platform. If you are already a registered u
ser of The Hindu and logged in, you may continue to engage with our articles. If
you do not have an account please register and login to post comments. Users can
```

```
!pip install pyLDAvis
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Collecting pyLDAvis
  Downloading pyLDAvis-3.4.0-py3-none-any.whl (2.6 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 2.6/2.6 MB 22.7 MB/s eta 0:00:00
Requirement already satisfied: scipy in /usr/local/lib/python3.9/dist-packages (from pyLDAvis) (1.10.1)
Requirement already satisfied: gensim in /usr/local/lib/python3.9/dist-packages (from pyLDAvis) (3.6.0)
Requirement already satisfied: scikit-learn>=1.0.0 in /usr/local/lib/python3.9/dist-packages (from pyLDAvis) (1.2.2)
Requirement already satisfied: setuptools in /usr/local/lib/python3.9/dist-packages (from pyLDAvis) (63.4.3)
Requirement already satisfied: jinja2 in /usr/local/lib/python3.9/dist-packages (from pyLDAvis) (3.1.2)
Requirement already satisfied: numpy>=1.22.0 in /usr/local/lib/python3.9/dist-packages (from pyLDAvis) (1.22.4)
Collecting joblib>=1.2.0
  Downloading joblib-1.2.0-py3-none-any.whl (297 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 298.0/298.0 KB 25.0 MB/s eta 0:00:00
Requirement already satisfied: numexpr in /usr/local/lib/python3.9/dist-packages (from pyLDAvis) (2.8.4)
Requirement already satisfied: pandas>=1.3.4 in /usr/local/lib/python3.9/dist-packages (from pyLDAvis) (1.4.4)
Collecting funcy
  Downloading funcy-1.18-py2.py3-none-any.whl (33 kB)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.9/dist-packages (from pandas>=1.3.4->pyLDAvis) (2022.7.1)
Requirement already satisfied: python-dateutil>=2.8.1 in /usr/local/lib/python3.9/dist-packages (from pandas>=1.3.4->pyLDAvis) (2.8
```

```
Requirement already satisfied: threadpoolctl>=2.0.0 in /usr/local/lib/python3.9/dist-packages (from scikit-learn>=1.0.0->pyLDAvis)
Requirement already satisfied: six>=1.5.0 in /usr/local/lib/python3.9/dist-packages (from gensim->pyLDAvis) (1.15.0)
Requirement already satisfied: smart-open>=1.2.1 in /usr/local/lib/python3.9/dist-packages (from gensim->pyLDAvis) (6.3.0)
Requirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python3.9/dist-packages (from jinja2->pyLDAvis) (2.1.2)
Installing collected packages: funcy, joblib, pyLDAvis
  Attempting uninstall: joblib
    Found existing installation: joblib 1.1.1
    Uninstalling joblib-1.1.1:
      Successfully uninstalled joblib-1.1.1
ERROR: pip's dependency resolver does not currently take into account all the packages that are installed. This behaviour is the so
pandas-profiling 3.2.0 requires joblib~=1.1.0, but you have joblib 1.2.0 which is incompatible.
Successfully installed funcy-1.18 joblib-1.2.0 pyLDAvis-3.4.0
```

```
import numpy as np
import json
import glob

#Gensim
import gensim
import gensim.corpora as corpora
from gensim.utils import simple_preprocess
from gensim.models import CoherenceModel

#spacy
import spacy
import nltk
from nltk.corpus import stopwords

#Vis
import pyLDAvis
import pyLDAvis.gensim

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

```
/usr/local/lib/python3.9/dist-packages/torch/cuda/__init__.py:497: UserWarning: Can't initialize NVML
  warnings.warn("Can't initialize NVML")
```

```
def load_data(file):
  with open(file,"r",encoding="utf-8") as f:
    data=json.load(f)
  return(data)

def write_data(file,data):
  with open(file,"w",encoding="utf-8") as f:
    json.dump(data,f,indent=4)
```

```
nltk.download("stopwords")
print(stopwords)
```

```
<WordListCorpusReader in '.../corpora/stopwords' (not loaded yet)>
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Unzipping corpora/stopwords.zip.
```

```
data=load_data("/content/Accident.json")["Text"]
splited_data=[]
for i in data:
  data_sp=data[i].split()
  splited_data.append(data_sp)
```

```
data=splited_data[0]+splited_data[1]+splited_data[2]+splited_data[3]+splited_data[4]+splited_data[5]+splited_data[6]+splited_data[7]+spli
```

```
['To',
 'enjoy',
 'additional',
 'benefits',
 'CONNECT',
 'WITH',
 'US',
 'March',
 '19,',
 '2023',
 '09:54',
 'pm',
 '|',
 'Updated',
 '09:54',
```

```
        'pm',
        'IST',
        '-',
        'Belagavi',
        'COMMents',
        'SHARE',
        'READ',
        'LATER',
        'A',
        '23-year-old',
        'youth',
        'died',
        'in',
        'a',
        'road',
        'accident',
        'in',
        'Belagavi',
        'on',
        'Saturday',
        'night',
        'and',
        'his',
        'family',
        'members',
        'are',
        'saying',
        'that',
        'an',
        'unscientifically',
        'built',
        'road',
        'hump',
        'claimed',
        'his',
        'life.',
        'Prateek',
        'Hongal',
        'was',
        'riding',
        'his',
        'bike',
        '  '
```

```
def lemmatization(texts,allowed_postages=["NOUN","ADJ","VERB","ADV"]):
  nlp=spacy.load("en_core_web_sm",disable=["senter","ner","parser"])
  texts_out=[]
  for text in texts:
    doc=nlp(text)
    new_text=[]
    for token in doc:
      if token.pos_ in allowed_postages:
        new_text.append(token.lemma_)
    final=" ".join(new_text)
    texts_out.append(final)
  return (texts_out)

lemmatized_texts=lemmatization(data)
lemmatized_texts
```

```
     ['',
      'enjoy',
      'additional',
      'benefit',
      'connect',
      '',
      '',
      '',
      '',
      '',
      '',
      'pm',
      '|',
      'update',
      '',
      'pm',
      '',
      '',
      'belagavi',
      'comment',
      'share',
      'read',
      'later',
      '',
      'year old',
      'youth',
      'die',
      '',
      '',
```

```
        'road',
        'accident',
        '',
        'belagavi',
        '',
        '',
        'night',
        '',
        '',
        'family',
        'member',
        '',
        'say',
        '',
        '',
        'unscientifically',
        'build',
        'road',
        'hump',
        'claim',
        '',
        'life',
        '',
        '',
        '',
        'ride',
        '',
        'bike',
        '',
```

```
def gen_words(texts):
  final=[]
  for text in texts:
    new=gensim.utils.simple_preprocess(text,deacc=True)
    final.append(new)
  return(final)

data_words=gen_words(lemmatized_texts)
print(data_words)
```

```
[[], ['enjoy'], ['additional'], ['benefit'], ['connect'], [], [], [], [], [], [], ['pm'], [], ['update'], [], ['pm'], [], [], ['bel
```

```
id2word=corpora.Dictionary(data_words)
corpus=[]
for text in data_words:
  new = id2word.doc2bow(text)
  corpus.append(new)

print(corpus)

word = id2word
print(word)
```

```
[[], [(0, 1)], [(1, 1)], [(2, 1)], [(3, 1)], [], [], [], [], [], [], [(4, 1)], [], [(5, 1)], [], [(4, 1)], [], [], [(6, 1)], [(7, 1
Dictionary(79 unique tokens: ['enjoy', 'additional', 'benefit', 'connect', 'pm']...)
```

```
lda_model= gensim.models.ldamodel.LdaModel(corpus=corpus,id2word=id2word,num_topics=30,random_state=100,update_every=1,
                                           chunksize=100,passes=10,alpha="auto")



pyLDAvis.enable_notebook()
vis=pyLDAvis.gensim.prepare(lda_model,corpus,id2word,mds="mmds",R=30)
vis
```

```
/usr/local/lib/python3.9/dist-packages/pyLDAvis/_prepare.py:243: FutureWarning: In
  default_term_info = default_term_info.sort_values(
/usr/local/lib/python3.9/dist-packages/sklearn/manifold/_mds.py:299: FutureWarning
  warnings.warn(
```

Selected Topic: 0 Previous Topic Next Topic Clear Topic