```
from bs4 import BeautifulSoup
from nltk.tokenize import PunktSentenceTokenizer
from nltk.corpus import stopwords

import nltk
nltk.download('punkt')
nltk.download('stopwords')

from nltk.corpus import webtext
nltk.download('webtext')
```

```
    [nltk_data] Downloading package punkt to /root/nltk_data...
    [nltk_data]   Unzipping tokenizers/punkt.zip.
    [nltk_data] Downloading package stopwords to /root/nltk_data...
    [nltk_data]   Unzipping corpora/stopwords.zip.
    [nltk_data] Downloading package webtext to /root/nltk_data...
    [nltk_data]   Unzipping corpora/webtext.zip.
    True
```

```
text = webtext.raw('/content/sample_data/test.txt')
sents = PunktSentenceTokenizer(text)
```

```
text
```

```
    'Open Editpad Wordpad and start creating the notes online. You can also copy-past
    e to edit text and save it for later use.\r\n\r\nBesides this, our online notepad
    allows users to upload files from a computer, Google Drive, and One Drive. (under
    development)\r\n\r\nYou can add, delete, and copy-paste text on the online text p
```

```
print(sents)
```

```
    <nltk.tokenize.punkt.PunktSentenceTokenizer object at 0x7fa8e69f41d0>
```

```
eng_stops = set(stopwords.words('english'))
word_c = word_tokenize(text)
```

```
    ---------------------------------------------------------------------------
    NameError                                 Traceback (most recent call last)
    <ipython-input-6-03867b31f028> in <module>()
          1 eng_stops = set(stopwords.words('english'))
    ----> 2 word_c = word_tokenize(text)

    NameError: name 'word_tokenize' is not defined
```

SEARCH STACK OVERFLOW

```
text = webtext.raw('/content/sample_data/test.txt')
sents = PunktSentenceTokenizer(text)
```

```
word_c
```

```
[word for word in word_c if word not in eng_stops]
```

```
len(word)
```

```
## Wordnet
```

```
from nltk.corpus import wordnet
nltk.download('wordnet')
nltk.download('omw-1.4')
```

```
[nltk_data] Downloading package wordnet to /root/nltk_data...
[nltk_data] Downloading package omw-1.4 to /root/nltk_data...
True
```

```
syn = wordnet.synsets('cookbook') [0]
syn.name()
```

```
'cookbook.n.01'
```

```
syn.definition()
```

```
'a book of recipes and cooking directions'
```

```
syn1 = wordnet.synsets('book')[1]
```

```
syn1.name()
```

```
'book.n.02'
```

```
syn1.lemmas()
```

```
[Lemma('book.n.02.book'), Lemma('book.n.02.volume')]
```

```
# Reference to try https://www.nltk.org/howto/wordnet.html
```

```
#Working with hypernyms
```

```
syn.hypernyms()
```

```
[Synset('reference_book.n.01')]
```

```
syn.hypernyms()[0].hyponyms()
```

```
[Synset('annual.n.02'),
 Synset('atlas.n.02'),
 Synset('cookbook.n.01'),
 Synset('directory.n.01'),
 Synset('encyclopedia.n.01'),
 Synset('handbook.n.01'),
 Synset('instruction_book.n.01'),
 Synset('source_book.n.01'),
 Synset('wordbook.n.01')]
```

```
syn.root_hypernyms()
```

```
[Synset('entity.n.01')]
```

```
syn.hypernym_paths()
```

```
[[Synset('entity.n.01'),
  Synset('physical_entity.n.01'),
  Synset('object.n.01'),
  Synset('whole.n.02'),
  Synset('artifact.n.01'),
  Synset('creation.n.02'),
  Synset('product.n.02'),
  Synset('work.n.02'),
  Synset('publication.n.01'),
  Synset('book.n.01'),
  Synset('reference_book.n.01'),
  Synset('cookbook.n.01')]]
```

```
## Lemmas and S
```

```
lemmas = syn.lemmas()
lemmas
```

```
[Lemma('cookbook.n.01.cookbook'), Lemma('cookbook.n.01.cookery_book')]
```

```
len(lemmas)
```

```
2
```

```
lemmas[0].name()

     'cookbook'
```

```
lemmas[1].name()

     'cookery_book'
```

```
#Antonyms
```

```
gn2 = wordnet.synset('good.n.02')
```

```
gn2.definition()

     'moral excellence or admirableness'
```

```
evil = gn2.lemmas()[0].antonyms()[0]
```

```
evil

     Lemma('evil.n.03.evil')
```

```
evil.name

     <bound method Lemma.name of Lemma('evil.n.03.evil')>
```

```
evil.synset

     <bound method Lemma.synset of Lemma('evil.n.03.evil')>
```

```
import nltk
from nltk.corpus import wordnet
nltk.download("wordnet")
nltk.download('omw-1.4')
print(wordnet.synsets("computer"))
print(wordnet.synset("computer.n.01").definition())

     [Synset('computer.n.01'), Synset('calculator.n.01')]
     a machine for performing calculations automatically
     [nltk_data] Downloading package wordnet to /root/nltk_data...
     [nltk_data]   Package wordnet is already up-to-date!
     [nltk_data] Downloading package omw-1.4 to /root/nltk_data...
     [nltk_data]   Package omw-1.4 is already up-to-date!
```

```
print("Examples:", wordnet.synset("computer.n.01").examples())
```

```
Examples: []
```

```
print(wordnet.lemma('buy.v.01.buy').antonyms())
```

```
[Lemma('sell.v.01.sell')]
```

```
wordnet.synset("computer.n.01").lemma_names()
```

```
['computer',
 'computing_machine',
 'computing_device',
 'data_processor',
 'electronic_computer',
 'information_processing_system']
```

```
syn = wordnet.synset("computer.n.01")
print(syn.hyponyms)
print([lemma.name() for synset in syn.hyponyms() for lemma in synset.lemmas()])
```

```
<bound method _WordNetObject.hyponyms of Synset('computer.n.01')>
['analog_computer', 'analogue_computer', 'digital_computer', 'home_computer', 'node', 'c
```

```
vehicle = wordnet.synset('vehicle.n.01')
car = wordnet.synset('car.n.01')
print(car.lowest_common_hypernyms(vehicle))
```

```
[Synset('vehicle.n.01')]
```

```
syn1 = wordnet.synsets("football")
syn2 = wordnet.synsets("soccer")

for s1 in syn1:
  for s2 in syn2:
    print("Path similarity of: ")
    print(s1, '(', s1.pos(), ')', '[', s1.definition(), ']')
    print(s2, '(', s1.pos(), ')', '[', s2.definition(), ']')
    print("is ", s1.path_similarity(s2))
    print()
```

```
Path similarity of:
Synset('football.n.01') ( n ) [ any of various games played with a ball (round or oval)
Synset('soccer.n.01') ( n ) [ a football game in which two teams of 11 players try to ki
is  0.5
```

```
Path similarity of:
Synset('football.n.02') ( n ) [ the inflated oblong ball used in playing American footba
Synset('soccer.n.01') ( n ) [ a football game in which two teams of 11 players try to ki
is  0.05
```

```
!pip install inltk
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/publ
Collecting inltk
  Downloading inltk-0.9-py3-none-any.whl (13 kB)
Requirement already satisfied: scipy in /usr/local/lib/python3.7/dist-packages (from inl
Collecting async-timeout>=3.0.1
  Downloading async_timeout-4.0.2-py3-none-any.whl (5.8 kB)
Requirement already satisfied: pyyaml in /usr/local/lib/python3.7/dist-packages (from in
Collecting typing
  Downloading typing-3.7.4.3.tar.gz (78 kB)
     |████████████████████████████████| 78 kB 3.8 MB/s
Requirement already satisfied: spacy>=2.0.18 in /usr/local/lib/python3.7/dist-packages (
Requirement already satisfied: pandas in /usr/local/lib/python3.7/dist-packages (from in
Requirement already satisfied: fastprogress>=0.1.19 in /usr/local/lib/python3.7/dist-pac
Collecting fastai==1.0.57
  Downloading fastai-1.0.57-py3-none-any.whl (233 kB)
     |████████████████████████████████| 233 kB 13.1 MB/s
Collecting nvidia-ml-py3
  Downloading nvidia-ml-py3-7.352.0.tar.gz (19 kB)
Requirement already satisfied: requests in /usr/local/lib/python3.7/dist-packages (from
Requirement already satisfied: Pillow in /usr/local/lib/python3.7/dist-packages (from in
Requirement already satisfied: numpy>=1.15 in /usr/local/lib/python3.7/dist-packages (fr
Collecting sentencepiece
  Downloading sentencepiece-0.1.96-cp37-cp37m-manylinux_2_17_x86_64.manylinux2014_x86_64
     |████████████████████████████████| 1.2 MB 29.3 MB/s
Requirement already satisfied: packaging in /usr/local/lib/python3.7/dist-packages (from
Requirement already satisfied: matplotlib in /usr/local/lib/python3.7/dist-packages (fro
Requirement already satisfied: beautifulsoup4 in /usr/local/lib/python3.7/dist-packages
Collecting aiohttp>=3.5.4
  Downloading aiohttp-3.8.1-cp37-cp37m-manylinux_2_5_x86_64.manylinux1_x86_64.manylinux_
     |████████████████████████████████| 1.1 MB 36.7 MB/s
Requirement already satisfied: numexpr in /usr/local/lib/python3.7/dist-packages (from i
Collecting bottleneck
  Downloading Bottleneck-1.3.5-cp37-cp37m-manylinux_2_5_x86_64.manylinux1_x86_64.manylin
     |████████████████████████████████| 355 kB 52.3 MB/s
Requirement already satisfied: torch>=1.0.0 in /usr/local/lib/python3.7/dist-packages (f
Requirement already satisfied: torchvision in /usr/local/lib/python3.7/dist-packages (fr
Requirement already satisfied: typing-extensions>=3.7.4 in /usr/local/lib/python3.7/dist
Collecting multidict<7.0,>=4.5
  Downloading multidict-6.0.2-cp37-cp37m-manylinux_2_17_x86_64.manylinux2014_x86_64.whl
     |████████████████████████████████| 94 kB 3.6 MB/s
Collecting yarl<2.0,>=1.0
  Downloading yarl-1.7.2-cp37-cp37m-manylinux_2_5_x86_64.manylinux1_x86_64.manylinux_2_1
     |████████████████████████████████| 271 kB 56.0 MB/s
Collecting asynctest==0.13.0
  Downloading asynctest-0.13.0-py3-none-any.whl (26 kB)
Collecting aiosignal>=1.1.2
  Downloading aiosignal-1.2.0-py3-none-any.whl (8.2 kB)
Requirement already satisfied: attrs>=17.3.0 in /usr/local/lib/python3.7/dist-packages (
Collecting frozenlist>=1.1.1
  Downloading frozenlist-1.3.0-cp37-cp37m-manylinux_2_5_x86_64.manylinux1_x86_64.manylin
     |████████████████████████████████| 144 kB 54.4 MB/s
Requirement already satisfied: charset-normalizer<3.0,>=2.0 in /usr/local/lib/python3.7/
Requirement already satisfied: blis<0.8.0,>=0.4.0 in /usr/local/lib/python3.7/dist-packa
Requirement already satisfied: langcodes<4.0.0,>=3.2.0 in /usr/local/lib/python3.7/dist-
```

```
import nest_asyncio
```