

POS Tagging

Load the following libraries: pandas, numpy, matplotlib, nltk, import sent_tokenize and word_tokenize from nltk, import FreqDist from nltk.probability, import stopwords from nltk.corpus, import WordCloud from wordcloud, import Image from PIL

```
In [18]: !pip install wordcloud
```

```
Collecting wordcloud
  Obtaining dependency information for wordcloud from https://files.pythonhosted.org/packages/34/ac/72a4e42e76bf549df
d91791a6b10a9832f046c1d48b5e778be9ec012aa47/wordcloud-1.9.2-cp311-cp311-win_amd64.whl.metadata (https://files.pythonh
osted.org/packages/34/ac/72a4e42e76bf549dfd91791a6b10a9832f046c1d48b5e778be9ec012aa47/wordcloud-1.9.2-cp311-win
_amd64.whl.metadata)
    Using cached wordcloud-1.9.2-cp311-cp311-win_amd64.whl.metadata (3.4 kB)
Requirement already satisfied: numpy>=1.6.1 in c:\users\jerry\anaconda3\lib\site-packages (from wordcloud) (1.24.3)
Requirement already satisfied: pillow in c:\users\jerry\anaconda3\lib\site-packages (from wordcloud) (9.4.0)
Requirement already satisfied: matplotlib in c:\users\jerry\anaconda3\lib\site-packages (from wordcloud) (3.7.2)
Requirement already satisfied: contourpy>=1.0.1 in c:\users\jerry\anaconda3\lib\site-packages (from matplotlib->wordc
loud) (1.0.5)
Requirement already satisfied: cycler>=0.10 in c:\users\jerry\anaconda3\lib\site-packages (from matplotlib->wordclou
d) (0.11.0)
Requirement already satisfied: fonttools>=4.22.0 in c:\users\jerry\anaconda3\lib\site-packages (from matplotlib->word
cloud) (4.25.0)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\jerry\anaconda3\lib\site-packages (from matplotlib->word
cloud) (1.4.4)
Requirement already satisfied: packaging>=20.0 in c:\users\jerry\anaconda3\lib\site-packages (from matplotlib->wordcl
oud) (23.1)
Requirement already satisfied: pyparsing<3.1,>=2.3.1 in c:\users\jerry\anaconda3\lib\site-packages (from matplotlib->
wordcloud) (3.0.9)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\jerry\anaconda3\lib\site-packages (from matplotlib->w
ordcloud) (2.8.2)
Requirement already satisfied: six>=1.5 in c:\users\jerry\anaconda3\lib\site-packages (from python-dateutil>=2.7->mat
plotlib->wordcloud) (1.16.0)
Using cached wordcloud-1.9.2-cp311-cp311-win_amd64.whl (151 kB)
Installing collected packages: wordcloud
Successfully installed wordcloud-1.9.2
```

```
In [1]: import pandas as pd
import numpy as np

import nltk
from nltk import word_tokenize
from nltk import sent_tokenize

from nltk.probability import FreqDist
from nltk.corpus import stopwords

from wordcloud import WordCloud
from PIL import Image

import matplotlib.pyplot as plt
```

**Open the text file DuneMessiah.txt and save it in a variable called: text_file
open()**

```
In [2]: text_file = open("DuneMessiah.txt")
```

Read the data and save it in a variable called: text

```
In [3]: text = text_file.read()
```

Display the datatype of the variable "text"

```
In [4]: print (type(text))
```

```
<class 'str'>
```

Print the text :

```
In [5]: print(text)
```

them out in the fire-eddy pattern. Immediately, she got the Kwisatz Haderach of the Major Arcana and the card lay coupled with the Eight of Ships: the sibyl hoodwinked and betrayed. These were not cards of good omen: they spoke of concealed resources for her enemies.

She turned away from the cards, sat in agitation, wondering if Irulan might yet destroy them.

= = = = =

The Fremen see her as the Earth Figure, a demigoddess whose special charge is to protect the tribes through her powers of violence. She is Reverend Mother to their Reverend Mothers. To pilgrims who seek her out with demands that she restore virility or make the barren fruitful, she is a form of antimentat. She feeds on that proof that the "analytic" has limits. She represents ultimate tension. She is the virgin-harlot -- witty, vulgar, cruel, as destructive in her whims as a coriolis storm.

-St. Alia of the Knife as taken from The Irulan Report Alia stood like a black-robed sentinel figure on the south platform of her temple, the Fane of the Oracle which Paul's Fremen cohorts had built for her against a wall of his stronghold.

She hated this part of her life, but knew no way to evade the temple without bringing down destruction upon them all. The pilgrims (damn them!) grew

print the length of the text

```
In [6]: print (len(text))
```

430811

Tokenize the text by sentences sent_tokenize()

```
In [7]: #Tokenize the text by sentences :  
sentences = sent_tokenize(text)
```

How many sentences are there?

```
In [8]: print (len(sentences))
```

8667

Print the sentences

```
In [9]: print(sentences)
```

["x0cConverted to "PDF" by\n" by ->MKM<\n>MKM<--\nDune Messiah\n\x0cFrank Herbert\nCopyright 1969\nExcerpts from the Death Cell Interview with Bronso of IX\nQ: What led you to take your particular approach to a history of Muad'dib?", 'A: Why should I answer your questions?', 'Q: Because I will preserve your words.', 'A: Ahhh!', 'The ultimate appeal to a historian!', 'Q: Will you cooperate then?', 'A: Why not?', "But you'll never understand what inspired my Analysis of\nHistory.", 'Never.', 'You Priests have too much at stake to .', '.', '.', 'Q: Try me.', 'A: Try you?', 'Well, Again .', '.', '.', 'why not?', 'I was caught by the shallowness of\nthe common view of this planet which arises from its popular name: Dune.', 'Not Arrakis, notice, but Dune.', 'History is obsessed by Dune as desert, as\nbirthplace of the Fremen.', "Such history concentrates on the customs which\nbegan to grow out of water scarcity and the fact that Fremen led semi-nomadic lives\nin stillsuits which recovered most of their body's moisture.", 'Q: Are these things not true, then?', 'A: They are surface truth.', 'As well ignore what lies beneath the surface as .', '.', '.', 'as try to understand my birthplanet, Ix, without exploring how we derived\nour name from the fact that we are the ninth planet of our sun.', 'No .', '.', '.', 'no.', 'It\nis not enough to see Dune as a place of savage storms.', 'It is not enough to\ntalk about the threat posed by the gigantic sandworms.', 'Q: But such things are crucial to the Arrakeen character!', 'A: Crucial?', 'Of course.', 'But they produce a one-vie w planet in the same way\nthat Dune is a one-crop planet because it is the sole and exclusive source of\nthe spice, melange.', 'Q: Yes.', 'Let us hear you expand on the sacred spice.', 'A: Sacred!', 'As with all things sacred, it gives with one hand and takes with\nthe other.', 'It extends life and allows the adept to foresee his future, but it ties\nhim to a cruel addiction and marks his eyes as yours are marked: total blue\nwithout any white.', 'Y our eyes, your organs of sight, become one thing\nwithout contrast, a single view.', 'Q: Such heresy brought you to this cell! I was brought to this cell by your Priests! As with all priests, you learned\nnearly to all

Tokenize the text with words. Assign them to a variable called: words

```
In [10]: words = word_tokenize(text)
```

How many words are there?

```
In [11]: print (len(words))
```

94575

Print the words

```
In [12]: print (words)
```

```
'a', 'historian', '!', 'Q', ':', 'Will', 'you', 'cooperate', 'then', '?', 'A', ':', 'Why', 'not', '?', 'But', 'yo  
u', "'ll", 'never', 'understand', 'what', 'inspired', 'my', 'Analysis', 'of', 'History', '.', 'Never', '.', 'Yo  
u', 'Priests', 'have', 'too', 'much', 'at', 'stake', 'to', '.', '.', 'Q', ':', 'Try', 'me', '.', 'A', ':',  
'Try', 'you', '?', 'Well', '.', 'Again', '.', '.', '.', 'why', 'not', '?', 'I', 'was', 'caught', 'by', 'the', 'sh  
allowness', 'of', 'the', 'common', 'view', 'of', 'this', 'planet', 'which', 'arises', 'from', 'its', 'popular',  
'name', ':', 'Dune', '.', 'Not', 'Arrakis', ',', 'notice', ',', 'but', 'Dune', '.', 'History', 'is', 'obsessed',  
'by', 'Dune', 'as', 'desert', ',', 'as', 'birthplace', 'of', 'the', 'Fremen', '.', 'Such', 'history', 'concentrat  
es', 'on', 'the', 'customs', 'which', 'grew', 'out', 'of', 'water', 'scarcity', 'and', 'the', 'fact', 'that', 'Fr  
emen', 'led', 'semi-nomadic', 'lives', 'in', 'stillsuits', 'which', 'recovered', 'most', 'of', 'their', 'body',  
"'s", 'moisture', '.', 'Q', ':', 'Are', 'these', 'things', 'not', 'true', '.', 'then', '?', 'A', ':', 'They', 'ar  
e', 'surface', 'truth', '.', 'As', 'well', 'ignore', 'what', 'lies', 'beneath', 'that', 'surface', 'as', '.',  
'.', '.', 'as', 'try', 'to', 'understand', 'my', 'birthplanet', ',', 'Ix', ',', 'without', 'exploring', 'how', 'w  
e', 'derived', 'our', 'name', 'from', 'the', 'fact', 'that', 'we', 'are', 'the', 'ninth', 'planet', 'of', 'our',  
'sun', '.', 'No', '.', '.', '.', 'no', '.', 'It', 'is', 'not', 'enough', 'to', 'see', 'Dune', 'as', 'a', 'place',  
'of', 'savage', 'storms', '.', 'It', 'is', 'not', 'enough', 'to', 'talk', 'about', 'the', 'threat', 'posed', 'b  
y', 'the', 'gigantic', 'sandworms', '.', 'Q', ':', 'But', 'such', 'things', 'are', 'crucial', 'to', 'the', 'Arrak  
een', 'character', '!', 'A', ':', 'Crucial', '?', 'Of', 'course', '.', 'But', 'they', 'produce', 'a', 'one-view',  
'planet', 'in', 'the', 'same', 'way', 'that', 'Dune', 'is', 'a', 'one-crop', 'planet', 'because', 'it', 'is', 'th  
e', 'sole', 'and', 'exclusive', 'source', 'of', 'the', 'spice', '.', 'melange', '.', 'Q', ':', 'Yes', '.', 'Let',  
'us' 'hear' 'you' 'expand' 'on' 'the' 'sacred' 'snice' ' ' 'A' ' ' 'Sacred' ' ' 'A' 'with' 'all'
```

using a variable "fdist" Find the frequency distribution
`FreqDist()`

```
In [13]: fdist = FreqDist(words)
```

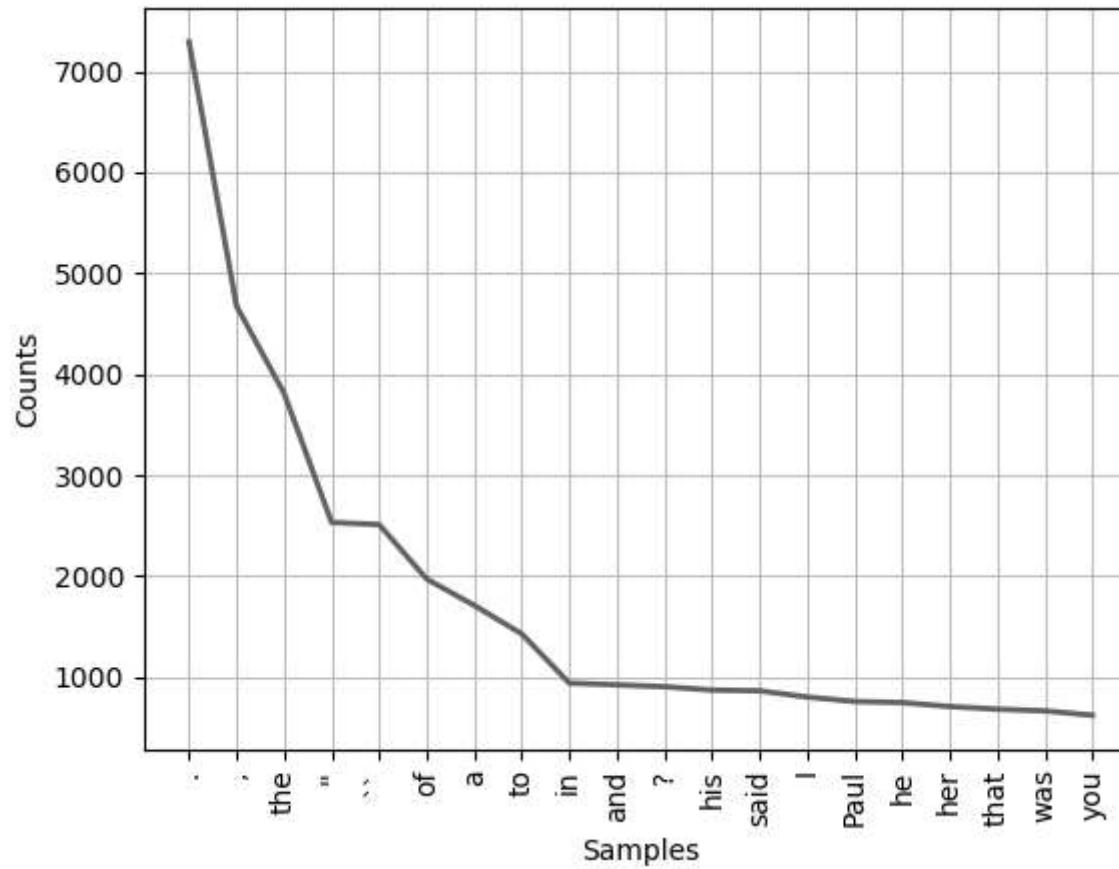
Print the 20 most common words

```
In [14]: fdist.most_common(20)
```

```
Out[14]: [('.', 7292),  
          (',', 4675),  
          ('the', 3811),  
          ('''', 2532),  
          ('``', 2511),  
          ('of', 1972),  
          ('a', 1711),  
          ('to', 1428),  
          ('in', 941),  
          ('and', 924),  
          ('?', 905),  
          ('his', 872),  
          ('said', 865),  
          ('I', 803),  
          ('Paul', 759),  
          ('he', 748),  
          ('her', 709),  
          ('that', 683),  
          ('was', 666),  
          ('you', 623)]
```

Plot the graph for fdist

```
In [15]: #Plot the graph for fdist :  
import matplotlib.pyplot as plt  
%matplotlib inline  
fdist.plot(20)
```



```
Out[15]: <Axes: xlabel='Samples', ylabel='Counts'>
```

Create an empty list to store words and call it: words_no_punc

```
In [16]: words_no_punc = []
```

Using a for loop, remove punctuation marks and make them lowercase w.lower()

Hint: for w in words: if w.isalpha(): words_no_punc.append(w.lower())

```
In [17]: #Removing punctuation marks :  
for w in words:  
    if w.isalpha():  
        words_no_punc.append(w.lower())
```

Print the words without punctuation marks

```
In [18]: print (words_no_punc)
```

```
'o', 'discuss', 'this', 'as', 'a', 'phenomenon', 'to', 'expound', 'upon', 'their', 'mental', 'arts', 'and', 'thei  
r', 'genetic', 'program', 'you', 'do', 'want', 'attention', 'called', 'to', 'the', 'fact', 'that', 'was', 'the',  
'sisterhood', 'hoped', 'for', 'captive', 'messiah', 'that', 'he', 'was', 'their', 'kwisatz', 'haderach', 'befor  
e', 'he', 'was', 'your', 'prophet', 'q', 'if', 'i', 'had', 'any', 'doubts', 'about', 'your', 'death', 'sentence',  
'you', 'have', 'dispelled', 'them', 'a', 'i', 'can', 'only', 'die', 'once', 'q', 'there', 'are', 'deaths', 'and',  
'there', 'are', 'deaths', 'a', 'beware', 'lest', 'you', 'make', 'a', 'martyr', 'of', 'me', 'i', 'do', 'not', 'thi  
nk', 'tell', 'me', 'does', 'know', 'what', 'you', 'do', 'in', 'these', 'dungeons', 'q', 'we', 'do', 'not', 'trou  
ble', 'the', 'holy', 'family', 'with', 'trivia', 'a', 'laughter', 'and', 'for', 'this', 'paul', 'atreides', 'fough  
t', 'his', 'way', 'to', 'a', 'niche', 'among', 'the', 'fremen', 'for', 'this', 'he', 'learned', 'to', 'control',  
'and', 'ride', 'the', 'sandworm', 'it', 'was', 'a', 'mistake', 'to', 'answer', 'your', 'questions', 'q', 'but',  
'i', 'will', 'keep', 'my', 'promise', 'to', 'preserve', 'your', 'words', 'a', 'will', 'you', 'really', 'then', 'l  
isten', 'to', 'me', 'carefully', 'you', 'fremen', 'degenerate', 'you', 'priest', 'with', 'no', 'god', 'except',  
'yourself', 'you', 'have', 'much', 'to', 'answer', 'for', 'it', 'was', 'a', 'fremen', 'ritual', 'which', 'gave',  
'paul', 'his', 'first', 'massive', 'dose', 'of', 'melange', 'thereby', 'opening', 'him', 'to', 'visions', 'of',  
'his', 'futures', 'it', 'was', 'a', 'fremen', 'ritual', 'by', 'which', 'that', 'same', 'melange', 'awakened', 'th  
e', 'unborn', 'alia', 'in', 'the', 'lady', 'womb', 'have', 'you', 'considered', 'what', 'it', 'meant', 'for', 'al  
ia', 'to', 'be', 'born', 'into', 'this', 'universe', 'fully', 'cognitive', 'possessed', 'of', 'all', 'her', 'moth  
er', 'memories', 'and', 'knowledge', 'no', 'rape', 'could', 'be', 'more', 'terrifying', 'q', 'without', 'the', 's  
acred', 'melange', 'would', 'not', 'have', 'become', 'leader', 'of', 'all', 'fremen', 'without', 'her', 'holy',  
'experience', 'alia', 'would', 'not', 'be', 'alia', 'a', 'without', 'your', 'blind', 'fremen', 'cruelty', 'you',
```

print the Length of words_no_punc

```
In [19]: print (len(words_no_punc))
```

73303

find the Frequency distribution of words_no_punc. Use the variabel fdist

```
In [20]: fdist = FreqDist(words_no_punc)
```

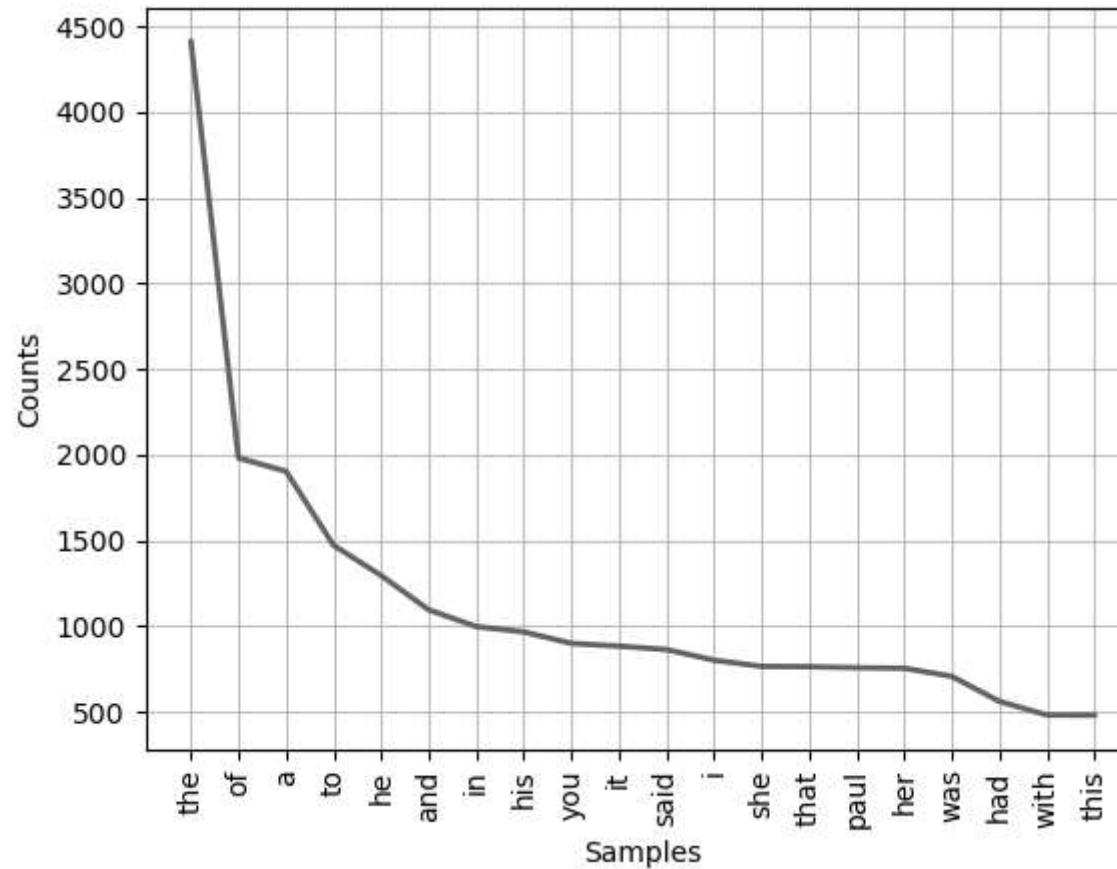
Display the 20 most common words

```
In [21]: fdist.most_common(20)
```

```
Out[21]: [('the', 4411),  
          ('of', 1983),  
          ('a', 1905),  
          ('to', 1473),  
          ('he', 1297),  
          ('and', 1097),  
          ('in', 999),  
          ('his', 968),  
          ('you', 901),  
          ('it', 885),  
          ('said', 865),  
          ('i', 803),  
          ('she', 767),  
          ('that', 765),  
          ('paul', 759),  
          ('her', 756),  
          ('was', 709),  
          ('had', 563),  
          ('with', 483),  
          ('this', 482)]
```

Plot the 20 most common words on a grpah:

```
In [22]: fdist.plot(20)
```



```
Out[22]: <Axes: xlabel='Samples', ylabel='Counts'>
```

**Print the List of stopwords from NLTK in English
assign them to a variable called: stopwords**

hint:

stopwords = stopwords.words("english") then print them.

```
In [23]: #List of stopwords
```

```
stopwords = stopwords.words("english")
print(stopwords)
```

```
['i', 'me', 'my', 'myself', 'we', 'our', 'ours', 'ourselves', 'you', "you're", "you've", "you'll", "you'd", 'your', 'yours', 'yourself', 'yourselves', 'he', 'him', 'his', 'himself', 'she', "she's", 'her', 'hers', 'herself', 'it', 'it's', 'its', 'itself', 'they', 'them', 'their', 'theirs', 'themselves', 'what', 'which', 'who', 'whom', 'this', 'that', "that'll", 'these', 'those', 'am', 'is', 'are', 'was', 'were', 'be', 'been', 'being', 'have', 'has', 'had', 'having', 'do', 'does', 'did', 'doing', 'a', 'an', 'the', 'and', 'but', 'if', 'or', 'because', 'as', 'until', 'while', 'of', 'at', 'by', 'for', 'with', 'about', 'against', 'between', 'into', 'through', 'during', 'before', 'after', 'above', 'below', 'to', 'from', 'up', 'down', 'in', 'out', 'on', 'off', 'over', 'under', 'again', 'further', 'then', 'once', 'here', 'there', 'when', 'where', 'why', 'how', 'all', 'any', 'both', 'each', 'few', 'more', 'most', 'other', 'some', 'such', 'no', 'nor', 'not', 'only', 'own', 'same', 'so', 'than', 'too', 'very', 's', 't', 'can', 'will', 'just', 'don', "don't", 'should', "should've", 'now', 'd', 'll', 'm', 'o', 're', 've', 'y', 'ain', 'aren', "aren't", 'could', 'n', "couldn't", 'didn', "didn't", 'doesn', "doesn't", 'hadn', "hadn't", 'hasn', "hasn't", 'haven', "haven't", 'isn', "isn't", 'ma', 'mightn', "mightn't", 'mustn', "mustn't", 'needn', "needn't", 'shan', "shan't", 'shouldn', "shouldn't", 'wasn', "wasn't", 'weren', "weren't", 'won', "won't", 'wouldn', "wouldn't"]
```

Create an empty list to store clean words and call it: clean_words

```
In [24]: clean_words = []
```

Using a for loop, remove stopwords

hint: for w in words_no_punc: if w not in stopwords: clean_words.append(w)

```
In [25]: for w in words_no_punc:
    if w not in stopwords:
        clean_words.append(w)
```

print the clean_words

```
In [26]: print(clean_words)
```

display the lenght of the clean_words

```
In [27]: print(len(clean_words))
```

38427

find the Frequency distribution and call it: fdist

```
In [28]: #Frequency distribution :  
fdist = FreqDist(clean_words)
```

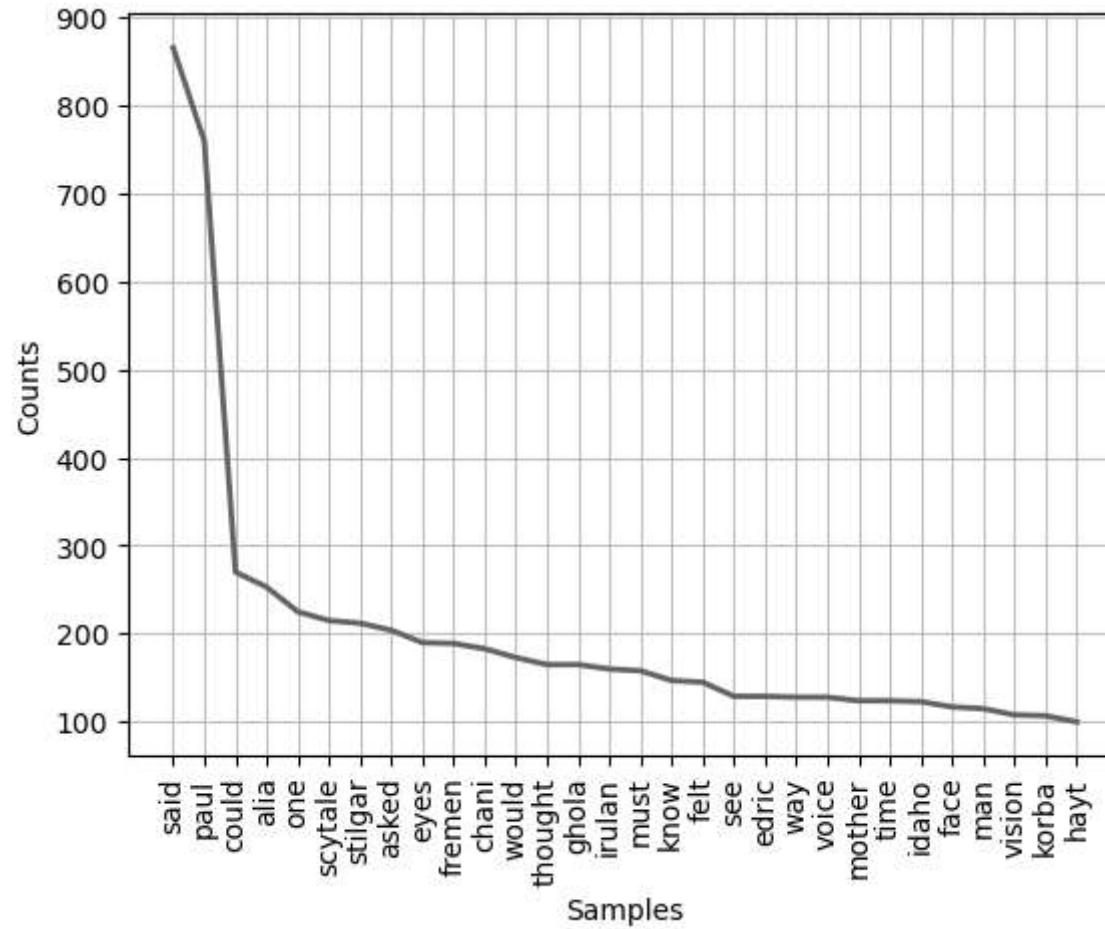
find the 30 most common words

```
In [29]: fdist.most_common(30)
```

```
Out[29]: [('said', 865),  
          ('paul', 759),  
          ('could', 270),  
          ('alia', 253),  
          ('one', 225),  
          ('scytale', 215),  
          ('stilgar', 212),  
          ('asked', 204),  
          ('eyes', 190),  
          ('fremen', 189),  
          ('chani', 183),  
          ('would', 173),  
          ('thought', 165),  
          ('ghola', 165),  
          ('irulan', 160),  
          ('must', 158),  
          ('know', 147),  
          ('felt', 145),  
          ('see', 129),  
          ('edric', 129),  
          ('way', 128),  
          ('voice', 128),  
          ('mother', 124),  
          ('time', 124),  
          ('idaho', 123),  
          ('face', 117),  
          ('man', 115),  
          ('vision', 108),  
          ('korba', 107),  
          ('hayt', 100)]
```

Plot the 30 most common words on grpah:

```
In [30]: #Plot the most common words on grpah:  
fdist.plot(30)
```



```
Out[30]: <Axes: xlabel='Samples', ylabel='Counts'>
```

Named Entity Recognition

In [83]: # Perform standard imports

```
import spacy
nlp = spacy.load('en_core_web_sm')
# Import the displaCy library
from spacy import displacy
doc = nlp(text)
displacy.render(doc, style='ent', jupyter=True)
```

A: You Priests do well to make common cause with the Bene Gesserit PERSON

Sisterhood ORG .

They, too, survive by concealing what they do. But they cannot conceal the

fact that the Lady Jessica PRODUCT was a Bene Gesserit PERSON -trained adept. You know she trained her son in the sisterhood's ways. My crime was to discuss this as a phenomenon, to expound upon their mental arts and their genetic program.

You don't want attention called to the fact that Muad'dib CARDINAL was the

Sisterhood ORG 's hoped for captive messiah, that he was their kwisatz haderach before he was your prophet.

Q: If I had any doubts about your death sentence, you have dispelled them.

Because Dune Messiah consists of several in-universe terminologies, the displaCy library cannot accurately recognize the fictional entities in the text string. "Jessica" is the name of Paul's mother but is recognized as "Product." "Bene Gesserit" is the organization in the story, but is recognized as a singular "Person." "Muad'dib" is a fictional name for a species of desert mouse that the main character Paul adopts as the title for his Messiah status, but is recognized as a "Cardinal." To correctly label the fictional entities, I will need to individually add the correct label of each word to the displaCy library.

Generate the wordcloud and call it: wordcloud

hint: wordcloud = WordCloud().generate(text)

```
In [32]: #Generating the wordCloud :  
wordcloud = WordCloud().generate(text)
```

Plot the wordcloud :

```
In [33]: #Plot the wordcloud :  
        plt.figure(figsize = (12, 12))  
        plt.imshow(wordcloud)
```

Out[33]: <matplotlib.image.AxesImage at 0x27022c54cd0>



```
In [34]: #To remove the axis value  
plt.axis("off")  
plt.imshow(wordcloud)  
plt.show()
```



Assign the top 30 words to a variable called: `most_common_words`

```
In [35]: most_common_words = fdist.most_common(30)
```

```
In [58]: ##### print the most common words
```

```
In [36]: print (most_common_words[:])
```

```
[('said', 865), ('paul', 759), ('could', 270), ('alicia', 253), ('one', 225), ('scytale', 215), ('stilgar', 212), ('asked', 204), ('eyes', 190), ('fremen', 189), ('chaney', 183), ('would', 173), ('thought', 165), ('ghola', 165), ('irulan', 160), ('must', 158), ('know', 147), ('felt', 145), ('see', 129), ('edric', 129), ('way', 128), ('voice', 128), ('mother', 124), ('time', 124), ('idaho', 123), ('face', 117), ('man', 115), ('vision', 108), ('korba', 107), ('hayt', 100)]
```

```
In [59]: # the most common words list's elements are in the form of tuple  
# get only the first element of each tuple of the word list
```

```
In [37]: word_features = [item[0] for item in most_common_words]  
print (word_features[:20])
```

```
['said', 'paul', 'could', 'alia', 'one', 'scytale', 'stilgar', 'asked', 'eyes', 'fremen', 'chani', 'would', 'thought', 'ghola', 'irulan', 'must', 'know', 'felt', 'see', 'edric']
```

```
In [38]: word_features = [item[0] for item in most_common_words]  
print (word_features[:15])
```

```
['said', 'paul', 'could', 'alia', 'one', 'scytale', 'stilgar', 'asked', 'eyes', 'fremen', 'chani', 'would', 'thought', 'ghola', 'irulan']
```

Get the tag of those words using nlp.pos_tag()

```
In [39]: nltk.pos_tag(word_features[:])
```

```
Out[39]: [('said', 'VBD'),  
          ('paul', 'NN'),  
          ('could', 'MD'),  
          ('alia', 'VB'),  
          ('one', 'CD'),  
          ('scytale', 'NN'),  
          ('stilgar', 'NN'),  
          ('asked', 'VBD'),  
          ('eyes', 'NNS'),  
          ('fremen', 'NNS'),  
          ('chani', 'NNS'),  
          ('would', 'MD'),  
          ('thought', 'VB'),  
          ('ghola', 'NN'),  
          ('irulan', 'NN'),  
          ('must', 'MD'),  
          ('know', 'VB'),  
          ('felt', 'VBD'),  
          ('see', 'VB'),  
          ('edric', 'JJ'),  
          ('way', 'NN'),  
          ('voice', 'NN'),  
          ('mother', 'NN'),  
          ('time', 'NN'),  
          ('idaho', 'JJ'),  
          ('face', 'NN'),  
          ('man', 'NN'),  
          ('vision', 'NN'),  
          ('korba', 'VBZ'),  
          ('hayt', 'NN')]
```

Text Summarization

```
In [40]: import nltk
from nltk.corpus import stopwords
from nltk.tokenize import sent_tokenize, word_tokenize
from sklearn.feature_extraction.text import TfidfVectorizer
import numpy as np
import re
```

```
In [41]: #import txt file into the text_file variable and print the file type
text_file = open("DuneMessiah.txt")
print (type(text_file))

<class '_io.TextIOWrapper'>
```

```
In [42]: #read the text_file into the variable text and print the file type
text = text_file.read()
print(type(text))

<class 'str'>
```

```
In [43]: #print the text  
text
```

```
Out[43]: '\x0cConverted to "PDF" by\n" by ->MKM<\n>MKM<--\nDune Messiah\n\x0cFrank Herbert\nCopyright 1969\nExcerpts from  
the Death Cell Interview with Bronso of IX\nQ: What led you to take your particular approach to a history of Muad  
\'dib?\nA: Why should I answer your questions?\nQ: Because I will preserve your words.\nA: Ahhh! The ultimate app  
eal to a historian!\nQ: Will you cooperate then?\nA: Why not? But you\'ll never understand what inspired my Analy  
sis of\nHistory.\nNever. You Priests have too much at stake to . . .\nQ: Try me.\nA: Try you? Well, Again . . . w  
hy not? I was caught by the shallowness of\nthe common view of this planet which arises from its popular name: Du  
ne.\nNot Arrakis, notice, but Dune. History is obsessed by Dune as desert, as\nbirthplace of the Fremen. Such his  
tory concentrates on the customs which\ngrew out of water scarcity and the fact that Fremen led semi-nomadic live  
s\nin stillsuits which recovered most of their body\'s moisture.\nQ: Are these things not true, then?\nA: They ar  
e surface truth. As well ignore what lies beneath that surface as .\n.\n. as try to understand my birthplanet, I  
x, without exploring how we derived\nour name from the fact that we are the ninth planet of our sun. No . . . no.  
It\nis not enough to see Dune as a place of savage storms. It is not enough to\ntalk about the threat posed by th  
e gigantic sandworms.\n\x0cQ: But such things are crucial to the Arrakeen character!\nA: Crucial? Of course. But  
they produce a one-view planet in the same way\nthat Dune is a one-crop planet because it is the sole and exclusi  
ve source of\nthe spice, melange.\nQ: Yes. Let us hear you expand on the sacred spice.\nA: Sacred! As with all th  
ings sacred, it gives with one hand and takes with\nthe other. It extends life and allows the adept to foresee hi  
s future, but it ties\nhim to a cruel addiction and marks his eyes as yours are marked: total blue\nwithout any w  
hite. Your eyes, your organs of sight, become one thing\nwithout contrast, a single view.\nQ: Such heresy brought  
you to this cell!\nA: I was brought to this cell by your Priests. As with all priests, you learned\nnearly to call  
it home.
```

```
In [44]: # Split the text into sentences keeping the original format  
original_sentences = text.strip().split('.')
```

```
# Now 'original_sentences' is an array of text  
print(original_sentences)
```

["Converted to “PDF” by\n” by ->MKM<\n>MKM<--\nDune Messiah\n\x0cFrank Herbert\nCopyright 1969\nExcerpts from the Death Cell Interview with Bronso of IX\nQ: What led you to take your particular approach to a history of Muad'dib?\nA: Why should I answer your questions?\nQ: Because I will preserve your words”, ”\nA: Ahhh! The ultimate appeal to a historian!\nQ: Will you cooperate then?\nA: Why not? But you'll never understand what inspired my Analysis of\nHistory”, '\nNever', ' You Priests have too much at stake to ', ' ', ' ', '\nQ: Try me', '\nA: Try you? Well, Again ', ' ', ' ', ' why not? I was caught by the shallowness of\nthe common view of this planet which arises from its popular name: Dune', '\nNot Arrakis, notice, but Dune', ' History is obsessed by Dune as desert, as\nbirthplace of the Fremen', " Such history concentrates on the customs which\nngrew out of water scarcity and the fact that Fremen led semi-nomadic lives\nin stillsuits which recovered most of their body's moisture", '\nQ: Are these things not true, then?\nA: They are surface truth', ' As well ignore what lies beneath that surface as ', '\n', '\n', ' as try to understand my birthplanet, Ix, without exploring how we derived\nour name from the fact that we are the ninth planet of our sun', ' No ', ' ', ' ', ' no', ' It\nis not enough to see Dune as a place of savage storms', ' It is not enough to\ntalk about the threat posed by the gigantic sandworms', '\n\nQ: But such things are crucial to the Arrakeen character!\nA: Crucial? Of course', ' But they produce a one-view planet in the same way\nthat Dune is a one-crop planet because it is the sole and exclusive source of\nthe spice, melange', '\nQ: Yes', ' Let us hear you expand on the sacred spice', '\nA: Sacred! As with all things sacred, it gives with one hand and takes with\nthe other', ' It extends life and allows the adept to foresee his future, but it ties\nhim to a cruel addiction and marks his eyes as yours are marked: total blue\nwithout any white', ' Your eyes, your organs of sight, become one thing\nwithout contrast, a single view', '\nQ: Such heresy brought you to this cell!\nA: I w

```
In [47]: original_sentences[101]
```

```
Out[47]: '\nThink on the paradox, the failure implicit in this moment, for you surely\nhave read other histories and know the surface facts'
```

```
In [48]: # Tokenize the text into sentences  
sentences = sent_tokenize(text)
```

```
In [49]: print(sentences)
```

```
e, me range. , Q. Yes. , Let us hear you expand on the sacred spice. , A. Sacred: , As with all things sacred, it gives with one hand and takes with\nthe other.', 'It extends life and allows the adept to foresee his future, but it ties\nhim to a cruel addiction and marks his eyes as yours are marked: total blue\nwithout any white.', 'Your eyes, your organs of sight, become one thing\nwithout contrast, a single view.', 'Q: Such heresy brought you to this cell!', 'A: I was brought to this cell by your Priests.', 'As with all priests, you learned\nnearly to call the truth heresy.', 'Q: You are here because you dared to say that Paul Atreides lost something\nessential to his humanity before he could become Muad'dib.', 'A: Not to speak of his losing his father here in the Harkonnen war.', 'Nor the\ndeath of Duncan Idaho, who sacrificed himself that Paul and the Lady\nJessica could escape.', 'Q: Your cynicism is duly noted.', 'A: Cynicism!', 'That, no doubt is a greater crime than heresy.', "But, you see,\nI'm not really a cynic.", "I'm just an observer and commentator.", 'I saw true\nnobility in Paul as he fled into the desert with his pregnant mother.', 'Of\ncourse, she was a great asset as well as a burden.', "Q: The flaw in your historians is that you'll never leave well enough alone.", "You see true nobility in the Holy Muad'dib, but you must append a cynical\nfootnote.", "It's no wonder that the Bene Gesserit also denounce you.", 'A: You Priests do well to make common cause with the Bene Gesserit\nSisterhood.', 'They, too, survive by concealing what they do.', 'But they cannot conceal the\nfact that the Lady Jessica was a Bene Gesserit-trained adept.', "You know she\ntrained her son in the sisterhood's ways.", 'My crime was to discuss this as a\nphenomenon, to expound upon their mental arts and their genetic program.', "You don't want attention called to the fact that Muad'dib was the\nSisterhood's hoped for captive messiah, that he was their kwisatz haderach\nbefore he was your prophet.", 'Q: If I had any doubts about your death sentence, you have dispelled them.', 'A: I can only die once.', 'Q: There are deaths and there are deaths.', 'A: Beware lest you make a martyr of me.', "I do not think Muad'dib .", '.', '.', 'Te
```

```
In [50]: len(sentences)
```

```
Out[50]: 8667
```

```
In [51]: # Preprocess the text (remove punctuation and stopwords)
```

```
nltk.download('punkt')
nltk.download('stopwords')
stop_words = set(stopwords.words('english'))
```

```
[nltk_data] Downloading package punkt to
[nltk_data]      C:\Users\Jerry\AppData\Roaming\nltk_data...
[nltk_data]      Package punkt is already up-to-date!
[nltk_data] Downloading package stopwords to
[nltk_data]      C:\Users\Jerry\AppData\Roaming\nltk_data...
[nltk_data]      Package stopwords is already up-to-date!
```

The function takes a string of text as input and performs several text preprocessing steps on that input text

words = word_tokenize(text)

- This step tokenizes the text, separating it into words based on spaces and punctuation.

```
word.lower():
```

- It converts each word to lowercase. This helps ensure that the text is case-insensitive, so "Word" and "word" are treated as the same.

```
for word in words if word.isalnum():
```

- It filters out words that are not composed entirely of alphabetical characters. The word.isalnum() method checks if a word contains only alphanumeric characters (letters and numbers) and no special characters or punctuation. This step removes non-alphabetic characters from the words.

```
words = [word for word in words if word not in stop_words]
```

- This line removes common stop words from the list of words. It appears that there is a pre-defined list of stop words stored in the variable stop_words. Stop words are words like "the," "a," "an," "in," etc., which are often filtered out in natural language processing tasks because they are very common and often do not provide significant meaning to the text.

```
return ' '.join(words)
```

- The preprocessed words are joined back into a single string with spaces between them, effectively turning the list of words into a preprocessed text. This string is then returned as the output of the preprocess_text function.

```
In [52]: def preprocess_text(text):
```

```
    words = word_tokenize(text)
    words = [word.lower() for word in words if word.isalnum()]
    words = [word for word in words if word not in stop_words]
    return ' '.join(words)
```

```
In [53]: #cleaned_sentence = re.sub(r'[^a-zA-Z]', ' ', sentence)

def preprocess_text(text):
    # Tokenize the text into words
    words = word_tokenize(text)

    # Replace non-alphabetical characters with spaces using regular expression
    cleaned_sentence = re.sub(r'[^a-zA-Z]', ' ', text)
    # Tokenize the cleaned sentence into words
    cleaned_words = word_tokenize(cleaned_sentence)

    # Convert words to lowercase, filter out non-alphabetical characters, and remove stop words
    cleaned_words = [word.lower() for word in cleaned_words if word.isalnum() and word not in stop_words]

    # Join the cleaned words back into a single string
    return ' '.join(cleaned_words)
```

```
In [54]: preprocessed_sentences = []
for sentence in sentences:
    preprocessed_sentence = preprocess_text(sentence)
    preprocessed_sentences.append(preprocessed_sentence)
```

```
In [55]: print(sentences)
```

"\x0cConverted to “PDF” by\n” by ->MKM<\n>MKM<--\nDune Messiah\n\x0cFrank Herbert\nCopyright 1969\nExcerpts from the Death Cell Interview with Bronso of IX\nQ: What led you to take your particular approach to a history of Muad'dib?", 'A: Why should I answer your questions?', 'Q: Because I will preserve your words.', 'A: Ahhh!', 'The ultimate appeal to a historian!', 'Q: Will you cooperate then?', 'A: Why not?', "But you'll never understand what inspired my Analysis of\nHistory.", 'Never.', 'You Priests have too much at stake to .', '.', '.', 'Q: Try me.', 'A: Try you?', 'Well, Again .', '.', '.', 'why not?', 'I was caught by the shallowness of\nthe common view of this planet which arises from its popular name: Dune.', 'Not Arrakis, notice, but Dune.', 'History is obsessed by Dune as desert, as\nbirthplace of the Fremen.', "Such history concentrates on the customs which\nngrew out of water scarcity and the fact that Fremen led semi-nomadic lives\nin stillsuits which recovered most of their body's moisture.", 'Q: Are these things not true, then?', 'A: They are surface truth.', 'As well ignore what lies beneath that surface as .', '.', '.', 'as try to understand my birthplanet, Ix, without exploring how we derived\nour name from the fact that we are the ninth planet of our sun.', 'No .', '.', '.', 'no.', 'It\nnis not enough to see Dune as a place of savage storms.', 'It is not enough to\nntalk about the threat posed by the gigantic sandworms.', 'Q: But such things are crucial to the Arrakeen character!', 'A: Crucial?', 'Of course.', 'But they produce a one-vie w planet in the same way\nthat Dune is a one-crop planet because it is the sole and exclusive source of\nthe spice, melange.', 'Q: Yes.', 'Let us hear you expand on the sacred spice.', 'A: Sacred!', 'As with all things sacred, it gives with one hand and takes with\nthe other.', 'It extends life and allows the adept to foresee his future, but it ties\nhim to a cruel addiction and marks his eyes as yours are marked: total blue\nwithout any white.', 'Y our eyes, your organs of sight, become one thing\nwithout contrast, a single view.', 'Q: Such heresy brought you to this cell! I was brought to this cell by your Priests! ' As with all Priests you learned\nnearly to see

```
In [56]: # Calculate TF-IDF scores  
tfidf = TfidfVectorizer()  
tfidf_matrix = tfidf.fit_transform(sentences)  
tfidf_scores = tfidf_matrix.sum(axis=1)
```

Since this is summary of an entire novel, I want to find the top 20 most important sentences instead of top 3.

```
In [57]: # Find the most important sentences based on TF-IDF scores  
top_sentence_indices = np.argsort(tfidf_scores, axis=0)[-20:]
```

```
In [58]: #print the the indices of the sentences chosen as the top 20 most important sentences  
top_sentence_indices
```

```
Out[58]: matrix([[5100],  
                 [2489],  
                 [1926],  
                 [2971],  
                 [1820],  
                 [2539],  
                 [1289],  
                 [ 126],  
                 [4675],  
                 [8455],  
                 [6764],  
                 [2803],  
                 [5148],  
                 [5399],  
                 [2293],  
                 [2560],  
                 [3890],  
                 [4124],  
                 [4091],  
                 [7403]], dtype=int64)
```

Print some of the top 20 sentences

```
In [63]: sentences[5100]
```

```
Out[63]: '= = = = =\nNo matter how exotic human civilization becomes, no matter the\ndevelopments of life and society nor th  
e complexity of the machine / human\ninterface, there always come interludes of lonely power when the course of\nhumankind, the very future of humankind, depends upon the relatively\nsimple actions of single individuals.'
```

```
In [59]: sentences[126]
```

```
Out[59]: 'The kwisatz haderach, then, the one who could be "many places at once,"\nthis prophet, this man through whom the Ben  
e Gesserit hoped to control\n\x0chuman destiny\n-- this man became Emperor Muad\'dib and executed a marriage of\nconven  
ience with a daughter of the Padishah Emperor he had defeated.'
```

```
In [60]: sentences[2803]
```

```
Out[60]: "Hoping he didn't betray too much\nof the terror which small enclosures such as this room imposed upon him,\n\x0cEdri  
c rolled over in the orange gas, directed his gaze at Paul, who sat alone\non a long divan of striped gray and bla  
k."
```

```
In [61]: sentences[7403]
```

```
Out[61]: "He had summoned odd companions\nfor this journey, she thought -- Bijaz, the Tleilaxu dwarf; the ghola, Hayt,\nwho mi  
ght be Duncan Idaho's revenant; Edric, the Guild Steersman-\nAmbassador; Gaius Helen Mohiam, the Bene Gesserit Revere  
nd Mother he\nso obviously hated; Lichna, Otheym's strange daughter, who seemed unable\nto move beyond the watchful e  
yes of guards; Stilgar, her uncle of the Naibs,\nand his favorite wife, Harah ."
```

```
In [68]: len(top_sentence_indices)
```

```
Out[68]: 20
```

```
In [80]: top_sentence_indices = np.argsort(tfidf_scores, axis=0)[-20:]

# Flatten the nested list of indices and convert to integers
flat_indices = [i for sublist in top_sentence_indices.tolist() for i in sublist]

# Extract the top 20 sentences
top_sentences = [sentences[i] for i in flat_indices]

# Join the top sentences to create the summary
summary = " ".join(top_sentences)

# Print the summary
print(summary)
```

= = = = =

No matter how exotic human civilization becomes, no matter the developments of life and society nor the complexity of the machine / human interface, there always come interludes of lonely power when the course of humankind, the very future of humankind, depends upon the relatively simple actions of single individuals. -St. Alia of the Knife as taken from The Irulan Report Alia stood like a black-robed sentinel figure on the south platform of her temple, the Fane of the Oracle which Paul's Fremen cohorts had built for her against a wall of his stronghold. The sharply silver light of noon poured through clerestory windows onto a floor worked in green, blue and eggshell tiles to simulate a bayou with water plants and, here and there, a splash of exotic color to indicate bird or animal. Damnable clever words, heavy with manipulation leverages -- that undertone of comfortable humor, the unspoken air of shared Secrets: his manner said he and Paul were two sophisticates, men of a wider universe who understood things not granted common folk. He felt

suddenly fearful that in reaching for any new thing he might let fall what was most precious, that even the slightest noise from him might send the universe crashing back, receding until he never could recapture any piece of it. When they had gone, muttering and objecting, but more fearful of her than they were of Stilgar, she stripped off her robe, leaving only the sheathed crysknife on its thong around her neck, strewed garments behind as she made for the bath. His words oddly in rhythm to the melody of his son's baliset, Farok said: "I

owned a crysknife, water rings to ten liters, my own lance which had been my father's, a coffee service, a bottle made of red glass older than any memory in my sietch. The kwisatz haderach, then, the one who could be "many places at once," this prophet, this man through whom the Bene Gesserit hoped to control human destiny

-- this man became Emperor Muad'dib and executed a marriage of convenience with a daughter of the Padishah Emperor he had defeated. The wide Fremen belt which gathered the waist of her golden robe, the water rings worn as a necklace, one hand at her hip (never far from the knife), the trenchant stare which was her first inspection of any room -- everything about her stood now only as a background for violence. Some of the Fremen back there in the

sietch had said Muad'dib would never die, that he had entered the ruh-world where all possible futures existed, that he would be present henceforth in the alam al-mythal, wandering there endlessly even after his flesh had ceased to be. Hayt felt that his eyes cut through layers of the dwarf, seeing the little man as a hired intellect, and beneath that, a creature imprisoned by hungers and cravings which lay huddled in the eyes -- layer after layer, until finally, there was only an entity-aspect being manipulated by symbols. Hoping he didn't betray too much of the terror which small enclosures such as this room imposed upon him,

Edric rolled over in the orange gas, directed his gaze at Paul, who sat alone on a long divan of striped gray and black. The temple across the thronged square was new, its rituals of recent devising, but there was something about this setting in a desert sink at the edge of Arrakeen -- something in the way wind-driven sand had begun to

weather stones and plastene, something in the haphazard way buildings had gone up around the Fane. At the core of the non-being, there he existed, allowing himself to be led through the streets of his city, following a track so familiar to his visions that it froze his heart with grief,
I should know what Alia saw, bethought, I have seen it enough times myself. For furnishings,
it contained one flimsy basket chair occupied now by the Reverend Mother Gaius Helen Mohiam, a pallet with a brown cover upon which had been spread a deck of the new Dune Tarot cards, a metered water tap above a reclamation basin, a Fremen privy with moisture seals. There were mnemonic amplifiers, digit mills from 1x to strengthen and sensitize fingers and toes, odor synthesizers, tactility sensitizers, temperature gradient fields, pattern betrayers to prevent her falling into detectable habits, alpha-wave-response trainers, blink-synchronizers to tone abilities in light / dark / spectrum analysis . Far within his fortress citadel, so deep within that the sound sometimes lost itself in the flow of city noises, a ten-string rebaba tinkled with a song of the Jihad, a lament for a woman left behind on Arrakis: Her hips are dunes curved by the wind,
Her eyes shine like summer heat. Aides and attendants stood around the Emperor in a curiously ordered sequence -- attentive household guardsmen along the draped back wall, that abomination, Alia, two steps below Paul and on his left hand; Stilgar, the Imperial lackey, on the step directly below Alia; and on the right, one step up from the floor of the hall, a solitary figure: the fleshly revenant of Duncan Idaho, the ghol. A small clue, but it joined other subtle indications -- the relative suppression and selection of words by her escort, the traces of primitive shyness in their eyes when they called her Reverend Mother, the cold and bland, essentially odorless nature of these halls -- all combined to reveal much that a Bene Gesserit could interpret. He had summoned odd companions for this journey, she thought -- Bijaz, the Tleilaxu dwarf; the ghol, Hayt, who might be Duncan Idaho's revenant; Edric, the Guild Steersman-Ambassador; Gaius Helen Mohiam, the Bene Gesserit Reverend Mother he so obviously hated; Lichna, Otheym's strange daughter, who seemed unable to move beyond the watchful eyes of guards; Stilgar, her uncle of the Naibs, and his favorite wife, Harah .

Sort the indices in numerical order to print the important sentences in the chronological order they appeared in the story

```
In [81]: # Flatten the nested list of indices and convert to integers
flat_indices = [i for sublist in top_sentence_indices.tolist() for i in sublist]

# Sort the indices in numeric order
sorted_indices = sorted(flat_indices)

# Extract and print the sentences in numeric order
sorted_sentences = [sentences[i] for i in sorted_indices]

# Join the sorted sentences to create the summary
summary = " ".join(sorted_sentences)

# Print or use the summary as needed
print(summary)
```

The kwisatz haderach, then, the one who could be "many places at once," this prophet, this man through whom the Bene Gesserit hoped to control human destiny

-- this man became Emperor Muad'dib and executed a marriage of convenience with a daughter of the Padishah Emperor he had defeated. His words oddly in rhythm to the melody of his son's baliset, Farok said: "I owned a crysknife, water rings to ten liters, my own lance which had been my father's, a coffee service, a bottle made of red glass older than any memory in my sietch. He felt suddenly fearful that in reaching for any new thing he might let fall what was most precious, that even the slightest noise from him might send the universe crashing back, receding until he never could recapture any piece of it. The sharply silver light of noon poured through clerestory windows onto a floor worked in green, blue and eggshell tiles to simulate a bayou with water plants and, here and there, a splash of exotic color to indicate bird or animal. For furnishings,

it contained one flimsy basket chair occupied now by the Reverend Mother Gaius Helen Mohiam, a pallet with a brown cover upon which had been spread a deck of the new Dune Tarot cards, a metered water tap above a reclamation basin, a Fremen privy with moisture seals. -St. Alia of the Knife as taken from The Irulan Report Alia stood like a

black-robed sentinel figure on the south platform of her temple, the Fane of the Oracle which Paul's Fremen cohorts had built for her against a wall of his stronghold. When they had gone, muttering and objecting, but more fearful of her than they were of Stilgar, she stripped off her robe, leaving only the sheathed crysknife on its thong around her neck, strewed garments behind as she made for the bath. There were mnemonic amplifiers, digit mills from lx to strengthen and sensitize fingers and toes, odor synthesizers, tactility sensitizers, temperature gradient fields, pattern betrayers to prevent her falling into detectable habits, alpha-wave-response trainers, blink-

synchronizers to tone abilities in light / dark / spectrum analysis . Hoping he didn't betray too much of the terror which small enclosures such as this room imposed upon him,

Edric rolled over in the orange gas, directed his gaze at Paul, who sat alone

on a long divan of striped gray and black. Damnable clever words, heavy with manipulation leverages -- that undertone of comfortable humor, the unspoken air of shared Secrets: his manner said he and Paul were two

sophisticates, men of a wider universe who understood things not granted

common folk. Far within his fortress citadel, so deep within that the sound sometimes lost

itself in the flow of city noises, a ten-string rebaba tinkled with a song of

the Jihad, a lament for a woman left behind on Arrakis: Her hips are dunes

curved by the wind,

Her eyes shine like summer heat. A small clue, but it joined other subtle indications -- the relative suppression and selection of words by her escort, the traces of primitive shyness in their eyes when they called her Reverend Mother, the cold and bland, essentially odorless nature of these halls -- all combined to reveal

much that a Bene Gesserit could interpret. Aides and attendants stood around the Emperor in a curiously ordered sequence -- attentive household guardsmen along the draped back wall, that

abomination, Alia, two steps below Paul and on his left hand; Stilgar, the Imperial lackey, on the step directly below Alia; and on the right, one step up from the floor of the hall, a solitary figure: the fleshly revenant of Duncan Idaho, the ghola. The wide Fremen belt which gathered the waist of her golden robe, the water rings worn as a necklace, one hand at her hip (never far from the knife), the trenchant stare which was her first inspection of any room -- everything about her stood now only as a background for violence. = = = = =

No matter how exotic human civilization becomes, no matter the developments of life and society nor the complexity of the machine / human interface, there always come interludes of lonely power when the course of humankind, the very future of humankind, depends upon the relatively simple actions of single individuals. The temple across the thronged square was new, its rituals of recent devising, but there was something about this setting in a desert sink at the edge of Arrakeen -- something in the way wind-driven sand had begun to weather stones and plastene, something in the haphazard way buildings had gone up around the Fane. At the core of the non-being, there he existed, allowing himself to be led through the streets of his city, following a track so familiar to his visions that it froze his heart with grief,

I should know what Alia saw, bethought, I have seen it enough times myself. Hayt felt that his eyes cut through layers of the dwarf, seeing the little man as a hired intellect, and beneath that, a creature imprisoned by hungers and cravings which lay huddled in the eyes -- layer after layer, until finally, there was only an entity-aspect being manipulated by symbols. He had summoned odd companions for this journey, she thought -- Bijaz, the Tleilaxu dwarf; the ghola, Hayt, who might be Duncan Idaho's revenant; Edric, the Guild Steersman-Ambassador; Gaius Helen Mohiam, the Bene Gesserit Reverend Mother he so obviously hated; Lichna, Otheym's strange daughter, who seemed unable to move beyond the watchful eyes of guards; Stilgar, her uncle of the Naibs, and his favorite wife, Harah . Some of the Fremen back there in the sietch had said Muad'dib would never die, that he had entered the ruh-world where all possible futures existed, that he would be present henceforth in the alam al-mythal, wandering there endlessly even after his flesh had ceased to be.

Conclusion

```
In [82]: print(type(sentences))
```

```
<class 'list'>
```

tf-idf score is able to automatically label the top N most important sentences in the transformed matrix of the list of all the sentences of Frank Herbert's Dune Messiah novel. Displaying the summary in the order of tf-idf importance ranking does not make chronological sense in the story. Flattening the list of indices to a list of integer and sorting that list will yield the indexes of the top 20 most important sentences sorted in numeric

order. After sorted index sentences and joining them to summary. The chronological summary makes sense on the beginning and the end. The automatic ranking of sentences does not convey the middle of the story well. Some of the sentences are components of a greater event, instead of summarizing the whole event of that important scene.

In []: