nlp-core

September 8, 2023

1 Tokenizing sentences

[1]:
```
import nltk
# nltk.download()
```

```
showing info https://raw.githubusercontent.com/nltk/nltk_data/gh-pages/index.xml
```

[1]:
```
True
```

[2]:
```
paragraph = """Thank you all so very much. Thank you to the Academy.
               Thank you to all of you in this room. I have to congratulate
               the other incredible nominees this year. The Revenant was
               the product of the tireless efforts of an unbelievable cast
               and crew. First off, to my brother in this endeavor, Mr. Tom
               Hardy. Tom, your talent on screen can only be surpassed by
               your friendship off screen … thank you for creating a t
               ranscendent cinematic experience. Thank you to everybody at
               Fox and New Regency … my entire team. I have to thank
               everyone from the very onset of my career … To my parents;
               none of this would be possible without you. And to my
               friends, I love you dearly; you know who you are. And lastly,
               I just want to say this: Making The Revenant was about
               man's relationship to the natural world. A world that we
               collectively felt in 2015 as the hottest year in recorded
               history. Our production needed to move to the southern
               tip of this planet just to be able to find snow. Climate
               change is real, it is happening right now. It is the most
               urgent threat facing our entire species, and we need to work
               collectively together and stop procrastinating. We need to
               support leaders around the world who do not speak for the
               big polluters, but who speak for all of humanity, for the
               indigenous people of the world, for the billions and
               billions of underprivileged people out there who would be
               most affected by this. For our children's children, and
               for those people out there whose voices have been drowned
               out by the politics of greed. I thank you all for this
               amazing award tonight. Let us not take this planet for
```

```
granted. I do not take tonight for granted. Thank you so very␣
↪much."""
```

[3]:
```
# Tokenizing sentences
sentences = nltk.sent_tokenize(paragraph)
print(sentences)
```

```
['Thank you all so very much.', 'Thank you to the Academy.', 'Thank you to all
of you in this room.', 'I have to congratulate \n                the other
incredible nominees this year.', 'The Revenant was \n                the product
of the tireless efforts of an unbelievable cast\n                and crew.',
'First off, to my brother in this endeavor, Mr. Tom \n                Hardy.',
'Tom, your talent on screen can only be surpassed by \n                your
friendship off screen … thank you for creating a t\n                ranscendent
cinematic experience.', 'Thank you to everybody at \n                Fox and New
Regency … my entire team.', 'I have to thank \n                everyone from the
very onset of my career … To my parents; \n                none of this would be
possible without you.', 'And to my \n                friends, I love you dearly;
you know who you are.', "And lastly,\n                I just want to say this:
Making The Revenant was about\n                man's relationship to the natural
world.", 'A world that we\n                collectively felt in 2015 as the
hottest year in recorded\n                history.', 'Our production needed to
move to the southern\n                tip of this planet just to be able to find
snow.', 'Climate\n                change is real, it is happening right now.',
'It is the most\n                urgent threat facing our entire species, and we
need to work\n                collectively together and stop procrastinating.',
'We need to\n                support leaders around the world who do not speak
for the \n                big polluters, but who speak for all of humanity, for
the\n                indigenous people of the world, for the billions and \n
billions of underprivileged people out there who would be\n                most
affected by this.', 'For our children’s children, and \n                for those
people out there whose voices have been drowned\n                out by the
politics of greed.', 'I thank you all for this \n                amazing award
tonight.', 'Let us not take this planet for \n                granted.', 'I do
not take tonight for granted.', 'Thank you so very much.']
```

2 Tokenizing words

[4]:
```
# Tokenizing words
words = nltk.word_tokenize(paragraph)
print(words)
```

```
['Thank', 'you', 'all', 'so', 'very', 'much', '.', 'Thank', 'you', 'to', 'the',
'Academy', '.', 'Thank', 'you', 'to', 'all', 'of', 'you', 'in', 'this', 'room',
'.', 'I', 'have', 'to', 'congratulate', 'the', 'other', 'incredible',
'nominees', 'this', 'year', '.', 'The', 'Revenant', 'was', 'the', 'product',
'of', 'the', 'tireless', 'efforts', 'of', 'an', 'unbelievable', 'cast', 'and',
```

```
'crew', '.', 'First', 'off', ',', 'to', 'my', 'brother', 'in', 'this',
'endeavor', ',', 'Mr.', 'Tom', 'Hardy', '.', 'Tom', ',', 'your', 'talent', 'on',
'screen', 'can', 'only', 'be', 'surpassed', 'by', 'your', 'friendship', 'off',
'screen', '…', 'thank', 'you', 'for', 'creating', 'a', 't', 'ranscendent',
'cinematic', 'experience', '.', 'Thank', 'you', 'to', 'everybody', 'at', 'Fox',
'and', 'New', 'Regency', '…', 'my', 'entire', 'team', '.', 'I', 'have', 'to',
'thank', 'everyone', 'from', 'the', 'very', 'onset', 'of', 'my', 'career', '…',
'To', 'my', 'parents', ';', 'none', 'of', 'this', 'would', 'be', 'possible',
'without', 'you', '.', 'And', 'to', 'my', 'friends', ',', 'I', 'love', 'you',
'dearly', ';', 'you', 'know', 'who', 'you', 'are', '.', 'And', 'lastly', ',',
'I', 'just', 'want', 'to', 'say', 'this', ':', 'Making', 'The', 'Revenant',
'was', 'about', 'man', "'s", 'relationship', 'to', 'the', 'natural', 'world',
'.', 'A', 'world', 'that', 'we', 'collectively', 'felt', 'in', '2015', 'as',
'the', 'hottest', 'year', 'in', 'recorded', 'history', '.', 'Our', 'production',
'needed', 'to', 'move', 'to', 'the', 'southern', 'tip', 'of', 'this', 'planet',
'just', 'to', 'be', 'able', 'to', 'find', 'snow', '.', 'Climate', 'change',
'is', 'real', ',', 'it', 'is', 'happening', 'right', 'now', '.', 'It', 'is',
'the', 'most', 'urgent', 'threat', 'facing', 'our', 'entire', 'species', ',',
'and', 'we', 'need', 'to', 'work', 'collectively', 'together', 'and', 'stop',
'procrastinating', '.', 'We', 'need', 'to', 'support', 'leaders', 'around',
'the', 'world', 'who', 'do', 'not', 'speak', 'for', 'the', 'big', 'polluters',
',', 'but', 'who', 'speak', 'for', 'all', 'of', 'humanity', ',', 'for', 'the',
'indigenous', 'people', 'of', 'the', 'world', ',', 'for', 'the', 'billions',
'and', 'billions', 'of', 'underprivileged', 'people', 'out', 'there', 'who',
'would', 'be', 'most', 'affected', 'by', 'this', '.', 'For', 'our', 'children',
''', 's', 'children', ',', 'and', 'for', 'those', 'people', 'out', 'there',
'whose', 'voices', 'have', 'been', 'drowned', 'out', 'by', 'the', 'politics',
'of', 'greed', '.', 'I', 'thank', 'you', 'all', 'for', 'this', 'amazing',
'award', 'tonight', '.', 'Let', 'us', 'not', 'take', 'this', 'planet', 'for',
'granted', '.', 'I', 'do', 'not', 'take', 'tonight', 'for', 'granted', '.',
'Thank', 'you', 'so', 'very', 'much', '.']
```

3 Stemming

[5]:
```
from nltk.stem import PorterStemmer

sentences = nltk.sent_tokenize(paragraph)
stemmer = PorterStemmer()

# Stemming
for i in range(len(sentences)):
    words = nltk.word_tokenize(sentences[i])
    words = [stemmer.stem(word) for word in words]
    sentences[i] = ' '.join(words)


print(sentences)
```

```
['thank you all so veri much .', 'thank you to the academi .', 'thank you to all
of you in thi room .', 'i have to congratul the other incred nomine thi year .',
'the reven wa the product of the tireless effort of an unbeliev cast and crew
.', 'first off , to my brother in thi endeavor , mr. tom hardi .', 'tom , your
talent on screen can onli be surpass by your friendship off screen … thank you
for creat a t ranscend cinemat experi .', 'thank you to everybodi at fox and new
regenc … my entir team .', 'i have to thank everyon from the veri onset of my
career … to my parent ; none of thi would be possibl without you .', 'and to my
friend , i love you dearli ; you know who you are .', "and lastli , i just want
to say thi : make the reven wa about man 's relationship to the natur world .",
'a world that we collect felt in 2015 as the hottest year in record histori .',
'our product need to move to the southern tip of thi planet just to be abl to
find snow .', 'climat chang is real , it is happen right now .', 'it is the most
urgent threat face our entir speci , and we need to work collect togeth and stop
procrastin .', 'we need to support leader around the world who do not speak for
the big pollut , but who speak for all of human , for the indigen peopl of the
world , for the billion and billion of underprivileg peopl out there who would
be most affect by thi .', 'for our children ’ s children , and for those peopl
out there whose voic have been drown out by the polit of greed .', 'i thank you
all for thi amaz award tonight .', 'let us not take thi planet for grant .', 'i
do not take tonight for grant .', 'thank you so veri much .']
```

4 Lemmatization

[6]:
```
from nltk.stem import WordNetLemmatizer

sentences = nltk.sent_tokenize(paragraph)
lemmatizer = WordNetLemmatizer()

# Lemmatization
for i in range(len(sentences)):
    words = nltk.word_tokenize(sentences[i])
    words = [lemmatizer.lemmatize(word) for word in words]
```

```
    sentences[i] = ' '.join(words)

print(sentences)
```

```
['Thank you all so very much .', 'Thank you to the Academy .', 'Thank you to all
of you in this room .', 'I have to congratulate the other incredible nominee
this year .', 'The Revenant wa the product of the tireless effort of an
unbelievable cast and crew .', 'First off , to my brother in this endeavor , Mr.
Tom Hardy .', 'Tom , your talent on screen can only be surpassed by your
friendship off screen … thank you for creating a t ranscendent cinematic
experience .', 'Thank you to everybody at Fox and New Regency … my entire team
.', 'I have to thank everyone from the very onset of my career … To my parent ;
none of this would be possible without you .', 'And to my friend , I love you
dearly ; you know who you are .', "And lastly , I just want to say this : Making
The Revenant wa about man 's relationship to the natural world .", 'A world that
we collectively felt in 2015 a the hottest year in recorded history .', 'Our
production needed to move to the southern tip of this planet just to be able to
find snow .', 'Climate change is real , it is happening right now .', 'It is the
most urgent threat facing our entire specie , and we need to work collectively
together and stop procrastinating .', 'We need to support leader around the
world who do not speak for the big polluter , but who speak for all of humanity
, for the indigenous people of the world , for the billion and billion of
underprivileged people out there who would be most affected by this .', 'For our
child ’ s child , and for those people out there whose voice have been drowned
out by the politics of greed .', 'I thank you all for this amazing award tonight
.', 'Let u not take this planet for granted .', 'I do not take tonight for
granted .', 'Thank you so very much .']
```

5 Removing stopwords

[7]:
```
from nltk.corpus import stopwords

sentences = nltk.sent_tokenize(paragraph)

# Removing stopwords
for i in range(len(sentences)):
    words = nltk.word_tokenize(sentences[i])
    words = [word for word in words if word not in stopwords.words('english')]
    sentences[i] = ' '.join(words)

print(sentences)
```

```
['Thank much .', 'Thank Academy .', 'Thank room .', 'I congratulate incredible
nominees year .', 'The Revenant product tireless efforts unbelievable cast crew
.', 'First , brother endeavor , Mr. Tom Hardy .', 'Tom , talent screen surpassed
friendship screen … thank creating ranscendent cinematic experience .', 'Thank
everybody Fox New Regency … entire team .', 'I thank everyone onset career … To
```

```
parents ; none would possible without .', 'And friends , I love dearly ; know
.', "And lastly , I want say : Making The Revenant man 's relationship natural
world .", 'A world collectively felt 2015 hottest year recorded history .', 'Our
production needed move southern tip planet able find snow .', 'Climate change
real , happening right .', 'It urgent threat facing entire species , need work
collectively together stop procrastinating .', 'We need support leaders around
world speak big polluters , speak humanity , indigenous people world , billions
billions underprivileged people would affected .', 'For children ' children ,
people whose voices drowned politics greed .', 'I thank amazing award tonight
.', 'Let us take planet granted .', 'I take tonight granted .', 'Thank much .']
```

6 Tagged word paragraph

[8]:
```
# POS Tagging
words = nltk.word_tokenize(paragraph)

tagged_words = nltk.pos_tag(words)

# Tagged word paragraph
word_tags = []
for tw in tagged_words:
    word_tags.append(tw[0]+"_"+tw[1])

tagged_paragraph = ' '.join(word_tags)
print(tagged_paragraph)
```

```
Thank_NNP you_PRP all_DT so_RB very_RB much_JJ ._. Thank_VB you_PRP to_TO the_DT
Academy_NNP ._. Thank_NNP you_PRP to_TO all_DT of_IN you_PRP in_IN this_DT
room_NN ._. I_PRP have_VBP to_TO congratulate_VB the_DT other_JJ incredible_JJ
nominees_NNS this_DT year_NN ._. The_DT Revenant_NNP was_VBD the_DT product_NN
of_IN the_DT tireless_NN efforts_NNS of_IN an_DT unbelievable_JJ cast_NN and_CC
crew_NN ._. First_NNP off_RB ,_, to_TO my_PRP$ brother_NN in_IN this_DT
endeavor_NN ,_, Mr._NNP Tom_NNP Hardy_NNP ._. Tom_NNP ,_, your_PRP$ talent_NN
on_IN screen_NN can_MD only_RB be_VB surpassed_VBN by_IN your_PRP$ friendship_NN
off_IN screen_JJ …_NNP thank_NN you_PRP for_IN creating_VBG a_DT t_JJ
ranscendent_NN cinematic_JJ experience_NN ._. Thank_NNP you_PRP to_TO
everybody_VB at_IN Fox_NNP and_CC New_NNP Regency_NNP …_NNP my_PRP$ entire_JJ
team_NN ._. I_PRP have_VBP to_TO thank_VB everyone_NN from_IN the_DT very_RB
onset_NN of_IN my_PRP$ career_NN …_NN To_TO my_PRP$ parents_NNS ;_: none_NN
of_IN this_DT would_MD be_VB possible_JJ without_IN you_PRP ._. And_CC to_TO
my_PRP$ friends_NNS ,_, I_PRP love_VBP you_PRP dearly_RB ;_: you_PRP know_VBP
who_WP you_PRP are_VBP ._. And_CC lastly_RB ,_, I_PRP just_RB want_VBP to_TO
say_VB this_DT :_: Making_VBG The_DT Revenant_NNP was_VBD about_IN man_NN 's_POS
relationship_NN to_TO the_DT natural_JJ world_NN ._. A_DT world_NN that_IN
we_PRP collectively_RB felt_VBD in_IN 2015_CD as_IN the_DT hottest_JJS year_NN
in_IN recorded_JJ history_NN ._. Our_PRP$ production_NN needed_VBN to_TO move_VB
to_TO the_DT southern_JJ tip_NN of_IN this_DT planet_NN just_RB to_TO be_VB
```

```
able_JJ to_TO find_VB snow_JJ ._. Climate_NNP change_NN is_VBZ real_JJ ,_,
it_PRP is_VBZ happening_VBG right_RB now_RB ._. It_PRP is_VBZ the_DT most_RBS
urgent_JJ threat_NN facing_VBG our_PRP$ entire_JJ species_NNS ,_, and_CC we_PRP
need_VBP to_TO work_VB collectively_RB together_RB and_CC stop_VB
procrastinating_NN ._. We_PRP need_VBP to_TO support_VB leaders_NNS around_IN
the_DT world_NN who_WP do_VBP not_RB speak_VB for_IN the_DT big_JJ polluters_NNS
,_, but_CC who_WP speak_VBP for_IN all_DT of_IN humanity_NN ,_, for_IN the_DT
indigenous_JJ people_NNS of_IN the_DT world_NN ,_, for_IN the_DT billions_NNS
and_CC billions_NNS of_IN underprivileged_JJ people_NNS out_IN there_EX who_WP
would_MD be_VB most_RBS affected_VBN by_IN this_DT ._. For_IN our_PRP$
children_NNS '_VBP s_JJ children_NNS ,_, and_CC for_IN those_DT people_NNS
out_RP there_RB whose_WP$ voices_NNS have_VBP been_VBN drowned_VBN out_RP by_IN
the_DT politics_NNS of_IN greed_NN ._. I_PRP thank_VBP you_PRP all_DT for_IN
this_DT amazing_JJ award_NN tonight_NN ._. Let_VB us_PRP not_RB take_VB this_DT
planet_NN for_IN granted_VBN ._. I_PRP do_VBP not_RB take_VB tonight_NN for_IN
granted_VBN ._. Thank_NNP you_PRP so_RB very_RB much_JJ ._.
```

7 Named entity recognition

[9]:
```
paragraph = """The Taj Mahal was built by Emperor Shah Jahan"""

# POS Tagging
words = nltk.word_tokenize(paragraph)

tagged_words = nltk.pos_tag(words)

# Named entity recognition
namedEnt = nltk.ne_chunk(tagged_words)
namedEnt.draw()
```

- ORGANIZATION Georgia-Pacific Corp., WHO
- PERSON Eddy Bonte, President Obama
- LOCATION Murray River, Mount Everest
- DATE June, 2008-06-29
- TIME two fifty a m, 1:30 p.m.
- MONEY 175 million Canadian Dollars, GBP 10.40
- PERCENT twenty pct, 18.75 %
- FACILITY Washington Monument, Stonehenge
- GPE South East Asia, Midlothian

8 Bag Of Words Model

[10]:
```
import re

paragraph = """Thank you all so very much. Thank you to the Academy.
               Thank you to all of you in this room. I have to congratulate
```

```
                the other incredible nominees this year. The Revenant was
                the product of the tireless efforts of an unbelievable cast
                and crew. First off, to my brother in this endeavor, Mr. Tom
                Hardy. Tom, your talent on screen can only be surpassed by
                your friendship off screen … thank you for creating a t
                ranscendent cinematic experience. Thank you to everybody at
                Fox and New Regency … my entire team. I have to thank
                everyone from the very onset of my career … To my parents;
                none of this would be possible without you. And to my
                friends, I love you dearly; you know who you are. And lastly,
                I just want to say this: Making The Revenant was about
                man's relationship to the natural world. A world that we
                collectively felt in 2015 as the hottest year in recorded
                history. Our production needed to move to the southern
                tip of this planet just to be able to find snow. Climate
                change is real, it is happening right now. It is the most
                urgent threat facing our entire species, and we need to work
                collectively together and stop procrastinating. We need to
                support leaders around the world who do not speak for the
                big polluters, but who speak for all of humanity, for the
                indigenous people of the world, for the billions and
                billions of underprivileged people out there who would be
                most affected by this. For our children's children, and
                for those people out there whose voices have been drowned
                out by the politics of greed. I thank you all for this
                amazing award tonight. Let us not take this planet for
                granted. I do not take tonight for granted. Thank you so very␣
 ↪much."""
```

[11]:
```
dataset = nltk.sent_tokenize(paragraph)
for i in range(len(dataset)):
    dataset[i] = dataset[i].lower()
    dataset[i] = re.sub(r'\W',' ',dataset[i])
    dataset[i] = re.sub(r'\s+',' ',dataset[i])

print(dataset)
```

```
['thank you all so very much ', 'thank you to the academy ', 'thank you to all
of you in this room ', 'i have to congratulate the other incredible nominees
this year ', 'the revenant was the product of the tireless efforts of an
unbelievable cast and crew ', 'first off to my brother in this endeavor mr tom
hardy ', 'tom your talent on screen can only be surpassed by your friendship off
screen thank you for creating a t ranscendent cinematic experience ', 'thank you
to everybody at fox and new regency my entire team ', 'i have to thank everyone
from the very onset of my career to my parents none of this would be possible
without you ', 'and to my friends i love you dearly you know who you are ', 'and
lastly i just want to say this making the revenant was about man s relationship
```

```
to the natural world ', 'a world that we collectively felt in 2015 as the
hottest year in recorded history ', 'our production needed to move to the
southern tip of this planet just to be able to find snow ', 'climate change is
real it is happening right now ', 'it is the most urgent threat facing our
entire species and we need to work collectively together and stop
procrastinating ', 'we need to support leaders around the world who do not speak
for the big polluters but who speak for all of humanity for the indigenous
people of the world for the billions and billions of underprivileged people out
there who would be most affected by this ', 'for our children s children and for
those people out there whose voices have been drowned out by the politics of
greed ', 'i thank you all for this amazing award tonight ', 'let us not take
this planet for granted ', 'i do not take tonight for granted ', 'thank you so
very much ']
```

[12]:
```
# Creating word histogram
word2count = {}
for data in dataset:
    words = nltk.word_tokenize(data)
    for word in words:
        if word not in word2count.keys():
            word2count[word] = 1
        else:
            word2count[word] += 1

print(word2count)
```

```
{'thank': 8, 'you': 12, 'all': 4, 'so': 2, 'very': 3, 'much': 2, 'to': 16,
'the': 17, 'academy': 1, 'of': 10, 'in': 4, 'this': 9, 'room': 1, 'i': 6,
'have': 3, 'congratulate': 1, 'other': 1, 'incredible': 1, 'nominees': 1,
'year': 2, 'revenant': 2, 'was': 2, 'product': 1, 'tireless': 1, 'efforts': 1,
'an': 1, 'unbelievable': 1, 'cast': 1, 'and': 8, 'crew': 1, 'first': 1, 'off':
2, 'my': 5, 'brother': 1, 'endeavor': 1, 'mr': 1, 'tom': 2, 'hardy': 1, 'your':
2, 'talent': 1, 'on': 1, 'screen': 2, 'can': 1, 'only': 1, 'be': 4, 'surpassed':
1, 'by': 3, 'friendship': 1, 'for': 10, 'creating': 1, 'a': 2, 't': 1,
'ranscendent': 1, 'cinematic': 1, 'experience': 1, 'everybody': 1, 'at': 1,
'fox': 1, 'new': 1, 'regency': 1, 'entire': 2, 'team': 1, 'everyone': 1, 'from':
1, 'onset': 1, 'career': 1, 'parents': 1, 'none': 1, 'would': 2, 'possible': 1,
'without': 1, 'friends': 1, 'love': 1, 'dearly': 1, 'know': 1, 'who': 4, 'are':
1, 'lastly': 1, 'just': 2, 'want': 1, 'say': 1, 'making': 1, 'about': 1, 'man':
1, 's': 2, 'relationship': 1, 'natural': 1, 'world': 4, 'that': 1, 'we': 3,
'collectively': 2, 'felt': 1, '2015': 1, 'as': 1, 'hottest': 1, 'recorded': 1,
'history': 1, 'our': 3, 'production': 1, 'needed': 1, 'move': 1, 'southern': 1,
'tip': 1, 'planet': 2, 'able': 1, 'find': 1, 'snow': 1, 'climate': 1, 'change':
1, 'is': 3, 'real': 1, 'it': 2, 'happening': 1, 'right': 1, 'now': 1, 'most': 2,
'urgent': 1, 'threat': 1, 'facing': 1, 'species': 1, 'need': 2, 'work': 1,
'together': 1, 'stop': 1, 'procrastinating': 1, 'support': 1, 'leaders': 1,
'around': 1, 'do': 2, 'not': 3, 'speak': 2, 'big': 1, 'polluters': 1, 'but': 1,
'humanity': 1, 'indigenous': 1, 'people': 3, 'billions': 2, 'underprivileged':
```

```
1, 'out': 3, 'there': 2, 'affected': 1, 'children': 2, 'those': 1, 'whose': 1,
'voices': 1, 'been': 1, 'drowned': 1, 'politics': 1, 'greed': 1, 'amazing': 1,
'award': 1, 'tonight': 2, 'let': 1, 'us': 1, 'take': 2, 'granted': 2}
```

[13]:
```
import heapq

# Selecting best 100 features
freq_words = heapq.nlargest(100,word2count,key=word2count.get)
print(freq_words)
```

```
['the', 'to', 'you', 'of', 'for', 'this', 'thank', 'and', 'i', 'my', 'all',
'in', 'be', 'who', 'world', 'very', 'have', 'by', 'we', 'our', 'is', 'not',
'people', 'out', 'so', 'much', 'year', 'revenant', 'was', 'off', 'tom', 'your',
'screen', 'a', 'entire', 'would', 'just', 's', 'collectively', 'planet', 'it',
'most', 'need', 'do', 'speak', 'billions', 'there', 'children', 'tonight',
'take', 'granted', 'academy', 'room', 'congratulate', 'other', 'incredible',
'nominees', 'product', 'tireless', 'efforts', 'an', 'unbelievable', 'cast',
'crew', 'first', 'brother', 'endeavor', 'mr', 'hardy', 'talent', 'on', 'can',
'only', 'surpassed', 'friendship', 'creating', 't', 'ranscendent', 'cinematic',
'experience', 'everybody', 'at', 'fox', 'new', 'regency', 'team', 'everyone',
'from', 'onset', 'career', 'parents', 'none', 'possible', 'without', 'friends',
'love', 'dearly', 'know', 'are', 'lastly']
```

[14]:
```
import numpy as np

# Converting sentences to vectors
X = []
for data in dataset:
    vector = []
    for word in freq_words:
        if word in nltk.word_tokenize(data):
            vector.append(1)
        else:
            vector.append(0)
    X.append(vector)

X = np.asarray(X)

print(X)
```

```
[[0 0 1 … 0 0 0]
 [1 1 1 … 0 0 0]
 [0 1 1 … 0 0 0]
 …
 [0 0 0 … 0 0 0]
 [0 0 0 … 0 0 0]
 [0 0 1 … 0 0 0]]
```

9 IDF Dictionary

[15]:
```
# IDF Dictionary
word_idfs = {}
for word in freq_words:
    doc_count = 0
    for data in dataset:
        if word in nltk.word_tokenize(data):
            doc_count += 1
    word_idfs[word] = np.log(len(dataset)/(1+doc_count))

print(word_idfs)
```

```
{'the': 0.6466271649250525, 'to': 0.5596157879354227, 'you': 0.7419373447293773,
'of': 1.0986122886681098, 'for': 1.0986122886681098, 'this': 0.7419373447293773,
'thank': 0.8472978603872037, 'and': 0.965080896043587, 'i': 1.0986122886681098,
'my': 1.4350845252893227, 'all': 1.4350845252893227, 'in': 1.6582280766035324,
'be': 1.4350845252893227, 'who': 1.9459101490553132, 'world':
1.6582280766035324, 'very': 1.6582280766035324, 'have': 1.6582280766035324,
'by': 1.6582280766035324, 'we': 1.6582280766035324, 'our': 1.6582280766035324,
'is': 1.9459101490553132, 'not': 1.6582280766035324, 'people':
1.9459101490553132, 'out': 1.9459101490553132, 'so': 1.9459101490553132, 'much':
1.9459101490553132, 'year': 1.9459101490553132, 'revenant': 1.9459101490553132,
'was': 1.9459101490553132, 'off': 1.9459101490553132, 'tom': 1.9459101490553132,
'your': 2.3513752571634776, 'screen': 2.3513752571634776, 'a':
1.9459101490553132, 'entire': 1.9459101490553132, 'would': 1.9459101490553132,
'just': 1.9459101490553132, 's': 1.9459101490553132, 'collectively':
1.9459101490553132, 'planet': 1.9459101490553132, 'it': 1.9459101490553132,
'most': 1.9459101490553132, 'need': 1.9459101490553132, 'do':
1.9459101490553132, 'speak': 2.3513752571634776, 'billions': 2.3513752571634776,
'there': 1.9459101490553132, 'children': 2.3513752571634776, 'tonight':
1.9459101490553132, 'take': 1.9459101490553132, 'granted': 1.9459101490553132,
'academy': 2.3513752571634776, 'room': 2.3513752571634776, 'congratulate':
2.3513752571634776, 'other': 2.3513752571634776, 'incredible':
2.3513752571634776, 'nominees': 2.3513752571634776, 'product':
2.3513752571634776, 'tireless': 2.3513752571634776, 'efforts':
2.3513752571634776, 'an': 2.3513752571634776, 'unbelievable':
2.3513752571634776, 'cast': 2.3513752571634776, 'crew': 2.3513752571634776,
'first': 2.3513752571634776, 'brother': 2.3513752571634776, 'endeavor':
2.3513752571634776, 'mr': 2.3513752571634776, 'hardy': 2.3513752571634776,
'talent': 2.3513752571634776, 'on': 2.3513752571634776, 'can':
2.3513752571634776, 'only': 2.3513752571634776, 'surpassed': 2.3513752571634776,
'friendship': 2.3513752571634776, 'creating': 2.3513752571634776, 't':
2.3513752571634776, 'ranscendent': 2.3513752571634776, 'cinematic':
2.3513752571634776, 'experience': 2.3513752571634776, 'everybody':
2.3513752571634776, 'at': 2.3513752571634776, 'fox': 2.3513752571634776, 'new':
2.3513752571634776, 'regency': 2.3513752571634776, 'team': 2.3513752571634776,
```

```
'everyone': 2.3513752571634776, 'from': 2.3513752571634776, 'onset':
2.3513752571634776, 'career': 2.3513752571634776, 'parents': 2.3513752571634776,
'none': 2.3513752571634776, 'possible': 2.3513752571634776, 'without':
2.3513752571634776, 'friends': 2.3513752571634776, 'love': 2.3513752571634776,
'dearly': 2.3513752571634776, 'know': 2.3513752571634776, 'are':
2.3513752571634776, 'lastly': 2.3513752571634776}
```

10 TF Matrix

[16]:
```
# TF Matrix
tf_matrix = {}
for word in freq_words:
    doc_tf = []
    for data in dataset:
        frequency = 0
        for w in nltk.word_tokenize(data):
            if word == w:
                frequency += 1
        tf_word = frequency/len(nltk.word_tokenize(data))
        doc_tf.append(tf_word)
    tf_matrix[word] = doc_tf

print(tf_matrix)
```

```
{'the': [0.0, 0.2, 0.0, 0.1, 0.2, 0.0, 0.0, 0.0, 0.043478260869565216, 0.0, 0.1,
0.06666666666666667, 0.05263157894736842, 0.0, 0.05, 0.10638297872340426,
0.045454545454545456, 0.0, 0.0, 0.0, 0.0], 'to': [0.0, 0.2, 0.1111111111111111,
0.1, 0.0, 0.09090909090909091, 0.0, 0.08333333333333333, 0.08695652173913043,
0.07692307692307693, 0.1, 0.0, 0.21052631578947367, 0.0, 0.05,
0.02127659574468085, 0.0, 0.0, 0.0, 0.0, 0.0], 'you': [0.16666666666666666, 0.2,
0.2222222222222222, 0.0, 0.0, 0.0, 0.043478260869565216, 0.08333333333333333,
0.043478260869565216, 0.23076923076923078, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.1111111111111111, 0.0, 0.0, 0.2], 'of': [0.0, 0.0, 0.1111111111111111, 0.0,
0.13333333333333333, 0.0, 0.0, 0.0, 0.08695652173913043, 0.0, 0.0, 0.0,
0.05263157894736842, 0.0, 0.0, 0.06382978723404255, 0.045454545454545456, 0.0,
0.0, 0.0, 0.0], 'for': [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.043478260869565216, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0851063829787234, 0.09090909090909091,
0.1111111111111111, 0.125, 0.14285714285714285, 0.0], 'this': [0.0, 0.0,
0.1111111111111111, 0.1, 0.0, 0.09090909090909091, 0.0, 0.0,
0.043478260869565216, 0.0, 0.05, 0.0, 0.05263157894736842, 0.0, 0.0,
0.02127659574468085, 0.0, 0.1111111111111111, 0.125, 0.0, 0.0], 'thank':
[0.16666666666666666, 0.2, 0.1111111111111111, 0.0, 0.0, 0.0,
0.043478260869565216, 0.08333333333333333, 0.043478260869565216, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.1111111111111111, 0.0, 0.0, 0.2], 'and': [0.0, 0.0,
0.0, 0.0, 0.06666666666666667, 0.0, 0.0, 0.08333333333333333, 0.0,
0.07692307692307693, 0.05, 0.0, 0.0, 0.0, 0.1, 0.02127659574468085,
0.045454545454545456, 0.0, 0.0, 0.0, 0.0], 'i': [0.0, 0.0, 0.0, 0.1, 0.0, 0.0,
```

0.0, 0.0, 0.043478260869565216, 0.07692307692307693, 0.05, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.1111111111111111, 0.0, 0.14285714285714285, 0.0], 'my': [0.0, 0.0,
0.0, 0.0, 0.0, 0.09090909090909091, 0.0, 0.08333333333333333,
0.08695652173913043, 0.07692307692307693, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0], 'all': [0.16666666666666666, 0.0, 0.1111111111111111, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.02127659574468085, 0.0,
0.1111111111111111, 0.0, 0.0, 0.0], 'in': [0.0, 0.0, 0.1111111111111111, 0.0,
0.0, 0.09090909090909091, 0.0, 0.0, 0.0, 0.0, 0.0, 0.13333333333333333, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], 'be': [0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.043478260869565216, 0.0, 0.043478260869565216, 0.0, 0.0, 0.0,
0.05263157894736842, 0.0, 0.0, 0.02127659574468085, 0.0, 0.0, 0.0, 0.0, 0.0],
'who': [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.07692307692307693, 0.0,
0.0, 0.0, 0.0, 0.0, 0.06382978723404255, 0.0, 0.0, 0.0, 0.0, 0.0], 'world':
[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.05, 0.06666666666666667,
0.0, 0.0, 0.0, 0.0425531914893617, 0.0, 0.0, 0.0, 0.0, 0.0], 'very':
[0.16666666666666666, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.043478260869565216,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.2], 'have': [0.0, 0.0,
0.0, 0.1, 0.0, 0.0, 0.0, 0.0, 0.043478260869565216, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.045454545454545456, 0.0, 0.0, 0.0, 0.0], 'by': [0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.043478260869565216, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.02127659574468085, 0.045454545454545456, 0.0, 0.0, 0.0, 0.0], 'we': [0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.06666666666666667, 0.0, 0.0,
0.05, 0.02127659574468085, 0.0, 0.0, 0.0, 0.0, 0.0], 'our': [0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.05263157894736842, 0.0, 0.05, 0.0,
0.045454545454545456, 0.0, 0.0, 0.0, 0.0], 'is': [0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.2222222222222222, 0.05, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0], 'not': [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.02127659574468085, 0.0, 0.0, 0.125, 0.14285714285714285, 0.0],
'people': [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0425531914893617, 0.045454545454545456, 0.0, 0.0, 0.0, 0.0], 'out': [0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.02127659574468085, 0.09090909090909091, 0.0, 0.0, 0.0, 0.0], 'so':
[0.16666666666666666, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.2], 'much': [0.16666666666666666, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.2], 'year': [0.0, 0.0, 0.0, 0.1, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.06666666666666667, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], 'revenant':
[0.0, 0.0, 0.0, 0.0, 0.06666666666666667, 0.0, 0.0, 0.0, 0.0, 0.0, 0.05, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], 'was': [0.0, 0.0, 0.0, 0.0,
0.06666666666666667, 0.0, 0.0, 0.0, 0.0, 0.0, 0.05, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0], 'off': [0.0, 0.0, 0.0, 0.0, 0.0, 0.09090909090909091,
0.043478260869565216, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0], 'tom': [0.0, 0.0, 0.0, 0.0, 0.0, 0.09090909090909091,
0.043478260869565216, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0], 'your': [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.08695652173913043, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], 'screen':
[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.08695652173913043, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], 'a': [0.0, 0.0, 0.0, 0.0, 0.0,

0.0, 0.043478260869565216, 0.0, 0.0, 0.0, 0.0, 0.06666666666666667, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], 'entire': [0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.08333333333333333, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.05, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0], 'would': [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.043478260869565216, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.02127659574468085, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0], 'just': [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.05, 0.0, 0.05263157894736842, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], 's':
[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.05, 0.0, 0.0, 0.0, 0.0,
0.0, 0.045454545454545456, 0.0, 0.0, 0.0, 0.0], 'collectively': [0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.06666666666666667, 0.0, 0.0, 0.05,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0], 'planet': [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.05263157894736842, 0.0, 0.0, 0.0, 0.0, 0.0, 0.125,
0.0, 0.0], 'it': [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.1111111111111111, 0.05, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], 'most': [0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.05,
0.02127659574468085, 0.0, 0.0, 0.0, 0.0, 0.0], 'need': [0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.05, 0.02127659574468085, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0], 'do': [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.02127659574468085, 0.0, 0.0, 0.0,
0.14285714285714285, 0.0], 'speak': [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0425531914893617, 0.0, 0.0, 0.0, 0.0, 0.0],
'billions': [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0425531914893617, 0.0, 0.0, 0.0, 0.0, 0.0], 'there': [0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.02127659574468085,
0.045454545454545456, 0.0, 0.0, 0.0, 0.0], 'children': [0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.09090909090909091, 0.0,
0.0, 0.0, 0.0], 'tonight': [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.1111111111111111, 0.0, 0.14285714285714285,
0.0], 'take': [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.125, 0.14285714285714285, 0.0], 'granted': [0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.125, 0.14285714285714285, 0.0], 'academy': [0.0, 0.2, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], 'room':
[0.0, 0.0, 0.1111111111111111, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], 'congratulate': [0.0, 0.0, 0.0, 0.1,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0], 'other': [0.0, 0.0, 0.0, 0.1, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], 'incredible': [0.0, 0.0, 0.0, 0.1, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0],
'nominees': [0.0, 0.0, 0.0, 0.1, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], 'product': [0.0, 0.0, 0.0, 0.0,
0.06666666666666667, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0], 'tireless': [0.0, 0.0, 0.0, 0.0, 0.06666666666666667, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0],
'efforts': [0.0, 0.0, 0.0, 0.0, 0.06666666666666667, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], 'an': [0.0, 0.0, 0.0,
0.0, 0.06666666666666667, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0], 'unbelievable': [0.0, 0.0, 0.0, 0.0,

0.06666666666666667, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0], 'cast': [0.0, 0.0, 0.0, 0.0, 0.06666666666666667, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], 'crew':
[0.0, 0.0, 0.0, 0.0, 0.06666666666666667, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], 'first': [0.0, 0.0, 0.0, 0.0, 0.0,
0.09090909090909091, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0], 'brother': [0.0, 0.0, 0.0, 0.0, 0.0, 0.09090909090909091, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0],
'endeavor': [0.0, 0.0, 0.0, 0.0, 0.0, 0.09090909090909091, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], 'mr': [0.0, 0.0, 0.0,
0.0, 0.0, 0.09090909090909091, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0], 'hardy': [0.0, 0.0, 0.0, 0.0, 0.0,
0.09090909090909091, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0], 'talent': [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.043478260869565216,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], 'on':
[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.043478260869565216, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], 'can': [0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.043478260869565216, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0], 'only': [0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.043478260869565216, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0], 'surpassed': [0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.043478260869565216, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0], 'friendship': [0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.043478260869565216, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0], 'creating': [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.043478260869565216,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], 't':
[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.043478260869565216, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], 'ranscendent': [0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.043478260869565216, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], 'cinematic': [0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.043478260869565216, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0], 'experience': [0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.043478260869565216, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0], 'everybody': [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.08333333333333333, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0], 'at': [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.08333333333333333, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], 'fox': [0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.08333333333333333, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0], 'new': [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.08333333333333333, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0], 'regency': [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.08333333333333333, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], 'team': [0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.08333333333333333, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], 'everyone': [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.043478260869565216, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0], 'from': [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.043478260869565216, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0], 'onset': [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.043478260869565216,

```
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], 'career': [0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.043478260869565216, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], 'parents': [0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.043478260869565216, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0], 'none': [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.043478260869565216, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0], 'possible': [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.043478260869565216,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], 'without': [0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.043478260869565216, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], 'friends': [0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.07692307692307693, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0], 'love': [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.07692307692307693, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0],
'dearly': [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.07692307692307693,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], 'know': [0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.07692307692307693, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0], 'are': [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.07692307692307693, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0],
'lastly': [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.05, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0]}
```

11 Creating the Tf-Idf Model

[17]:
```
# Creating the Tf-Idf Model
tfidf_matrix = []
for word in tf_matrix.keys():
    tfidf = []
    for value in tf_matrix[word]:
        score = value * word_idfs[word]
        tfidf.append(score)
    tfidf_matrix.append(tfidf)

print(tfidf_matrix)
```

```
[[0.0, 0.1293254329850105, 0.0, 0.06466271649250525, 0.1293254329850105, 0.0,
0.0, 0.0, 0.028114224561958803, 0.0, 0.06466271649250525, 0.04310847766167016,
0.034033008680265917, 0.0, 0.03233135824625263, 0.06879012392819707,
0.029392143860229657, 0.0, 0.0, 0.0, 0.0], [0.0, 0.11192315758708454,
0.062179531992824735, 0.05596157879354227, 0.0, 0.05087416253958388, 0.0,
0.046634648994618555, 0.04866224242916719, 0.04304736830272482,
0.05596157879354227, 0.0, 0.11781385009166792, 0.0, 0.027980789396771136,
0.011906718892243035, 0.0, 0.0, 0.0, 0.0, 0.0], [0.12365622412156288,
0.14838746894587548, 0.16487496549541716, 0.0, 0.0, 0.0, 0.0322581454230164,
0.06182811206078144, 0.0322581454230164, 0.171216310322164, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.08243748274770858, 0.0, 0.0, 0.14838746894587548], [0.0, 0.0,
0.12206803207423442, 0.0, 0.1464816384890813, 0.0, 0.0, 0.0,
0.09553150336244433, 0.0, 0.0, 0.0, 0.057821699403584725, 0.0, 0.0,
```

0.07012418863838998, 0.04993692221218681, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.047765751681222164, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.09349891818451998, 0.09987384442437362, 0.12206803207423442,
0.13732653608351372, 0.15694461266687282, 0.0], [0.0, 0.0, 0.08243748274770858,
0.07419373447293774, 0.0, 0.06744884952085249, 0.0, 0.0, 0.0322581454230164,
0.0, 0.03709686723646887, 0.0, 0.03904933393312512, 0.0, 0.0,
0.01578590095168888, 0.0, 0.08243748274770858, 0.09274216809117217, 0.0, 0.0],
[0.14121631006453395, 0.16945957207744075, 0.09414420670968929, 0.0, 0.0, 0.0,
0.03683903740813929, 0.07060815503226697, 0.03683903740813929, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.09414420670968929, 0.0, 0.0, 0.16945957207744075],
[0.0, 0.0, 0.0, 0.0, 0.0643387264029058, 0.0, 0.0, 0.08042340800363225, 0.0,
0.07423699200335285, 0.048254044802179354, 0.0, 0.0, 0.0, 0.09650808960435871,
0.020533636086033768, 0.04386731345652668, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0,
0.10986122886681099, 0.0, 0.0, 0.0, 0.0, 0.047765751681222164,
0.08450863758985461, 0.054930614433405495, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.12206803207423442, 0.0, 0.15694461266687282, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0,
0.13046222957175663, 0.0, 0.11959037710744355, 0.12478995872081067,
0.11039111732994791, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0],
[0.2391807542148871, 0.0, 0.15945383614325806, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.030533713304028143, 0.0, 0.15945383614325806,
0.0, 0.0, 0.0], [0.0, 0.0, 0.18424756406705914, 0.0, 0.0, 0.1507480069639575,
0.0, 0.0, 0.0, 0.0, 0.0, 0.22109707688047098, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.062394979360405334, 0.0,
0.062394979360405334, 0.0, 0.0, 0.0, 0.07553076448891172, 0.0, 0.0,
0.030533713304028143, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.14968539608117795, 0.0, 0.0, 0.0, 0.0, 0.0,
0.12420703079076466, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.08291140383017663, 0.11054853844023549, 0.0, 0.0, 0.0,
0.07056289687674606, 0.0, 0.0, 0.0, 0.0, 0.0], [0.2763713461005887, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.07209687289580576, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.3316456153207065], [0.0, 0.0, 0.0, 0.16582280766035326,
0.0, 0.0, 0.0, 0.0, 0.07209687289580576, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.07537400348197874, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.07209687289580576, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.03528144843837303, 0.07537400348197874, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.11054853844023549, 0.0, 0.0,
0.08291140383017663, 0.03528144843837303, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0872751619265017, 0.0,
0.08291140383017663, 0.0, 0.07537400348197874, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.43242447756784735,
0.09729550745276566, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.03528144843837303, 0.0, 0.0,
0.20727850957544156, 0.23688972522907606, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0828046871938431l,
0.08845046132069606, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.041402343596921555,
0.17690092264139212, 0.0, 0.0, 0.0, 0.0], [0.3243183581758855, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,

0.38918202981106265], [0.3243183581758855, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.38918202981106265], [0.0, 0.0, 0.0, 0.19459101490553132, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.1297273432703542, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0],
[0.0, 0.0, 0.0, 0.0, 0.1297273432703542, 0.0, 0.0, 0.0, 0.0, 0.0,
0.09729550745276566, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0,
0.0, 0.0, 0.0, 0.1297273432703542, 0.0, 0.0, 0.0, 0.0, 0.0, 0.09729550745276566,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0,
0.17690092264139212, 0.08460478908936145, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0,
0.17690092264139212, 0.08460478908936145, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.20446741366638935, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.20446741366638935, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.08460478908936145, 0.0, 0.0, 0.0, 0.0, 0.1297273432703542, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.16215917908794275, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.09729550745276566, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.08460478908936145, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.041402343596921555, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.09729550745276566, 0.0, 0.10241632363449016, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.09729550745276566, 0.0, 0.0, 0.0, 0.0, 0.0, 0.08845046132069606, 0.0, 0.0,
0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.1297273432703542, 0.0, 0.0, 0.09729550745276566, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0], [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.10241632363449016, 0.0, 0.0, 0.0, 0.0, 0.0, 0.24323876863191415, 0.0, 0.0],
[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.21621223878392368, 0.09729550745276566, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.09729550745276566, 0.041402343596921555, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.09729550745276566,
0.041402343596921555, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.041402343596921555, 0.0, 0.0,
0.0, 0.277987164150759, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.10005852158142457, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.10005852158142457, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.041402343596921555,
0.08845046132069606, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.2137613870148616, 0.0, 0.0, 0.0,
0.0], [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.21621223878392368, 0.0, 0.277987164150759, 0.0], [0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.24323876863191415, 0.277987164150759, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.2432387686319141 5,
0.277987164150759, 0.0], [0.0, 0.47027505143269555, 0.0, 0.0, 0.0, 0.0, 0.0,

0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0,
0.0, 0.26126391746260863, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.23513752571634777, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0],
[0.0, 0.0, 0.0, 0.23513752571634777, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0,
0.23513752571634777, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.23513752571634777, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0,
0.0, 0.0, 0.0, 0.15675835047756517, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0.15675835047756517,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0],
[0.0, 0.0, 0.0, 0.0, 0.15675835047756517, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0,
0.15675835047756517, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0.15675835047756517, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0,
0.0, 0.0, 0.15675835047756517, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0.15675835047756517, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0],
[0.0, 0.0, 0.0, 0.0, 0.0, 0.2137613870148616, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0,
0.2137613870148616, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0, 0.2137613870148616, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0,
0.0, 0.0, 0.2137613870148616, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0, 0.2137613870148616, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.10223370683319467, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.10223370683319467, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.10223370683319467, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.10223370683319467, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.10223370683319467, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.10223370683319467, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.10223370683319467,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.10223370683319467, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.10223370683319467, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0], [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.19594793809695646, 0.0, 0.0, 0.0,

```
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.19594793809695646, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.19594793809695646, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.19594793809695646, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.19594793809695646, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0], [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.10223370683319467, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.10223370683319467, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.10223370683319467,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.10223370683319467, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.10223370683319467, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0], [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.10223370683319467, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.10223370683319467, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.10223370683319467,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.18087501978180598, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.18087501978180598, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0],
[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.18087501978180598, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.18087501978180598, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0], [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.18087501978180598,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.11756876285817389, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0]]
```

[18]:
```
# Finishing the Tf-Tdf model
X = np.asarray(tfidf_matrix)

X = np.transpose(X)

X
```

[18]:
```
array([[0.        , 0.        , 0.12365622, …, 0.        , 0.        ,
        0.        ],
       [0.12932543, 0.11192316, 0.14838747, …, 0.        , 0.        ,
        0.        ],
       [0.        , 0.06217953, 0.16487497, …, 0.        , 0.        ,
        0.        ],
       …,
       [0.        , 0.        , 0.        , …, 0.        , 0.        ,
        0.        ],
```

```
       [0.        , 0.        , 0.        , …, 0.        , 0.        ,
        0.        ],
       [0.        , 0.        , 0.14838747, …, 0.        , 0.        ,
        0.        ]])
```

12 N-Gram Modelling - Character Grams

[20]:
```
import random

# Sample data
text = """Global warming or climate change has become a worldwide concern. It␣
 ↪is gradually developing into an unprecedented environmental crisis evident␣
 ↪in melting glaciers, changing weather patterns, rising sea levels, floods,␣
 ↪cyclones and droughts. Global warming implies an increase in the average␣
 ↪temperature of the Earth due to entrapment of greenhouse gases in the␣
 ↪earth's atmosphere."""

# Order of the grams
n = 6

# Our N-Grams
ngrams = {}

# Creating the model
for i in range(len(text)-n):
    gram = text[i:i+n]
    if gram not in ngrams.keys():
        ngrams[gram] = []
    ngrams[gram].append(text[i+n])

# Testing our N-Gram Model
currentGram = text[0:n]
result = currentGram
for i in range(100):
    if currentGram not in ngrams.keys():
        break
    possibilities = ngrams[currentGram]
    nextItem = possibilities[random.randrange(len(possibilities))]
    result += nextItem
    currentGram = result[len(result)-n:len(result)]

print(result)
```

```
Global warming or climate changing weather patterns, rising sea levels, floods,
cyclones and droughts. Glo
```

13 N-Gram Modelling - Word Grams

[22]:
```
import random
import nltk

# Sample data
text = """Global warming or climate change has become a worldwide concern. It␣
 ↪is gradually developing into an unprecedented environmental crisis evident␣
 ↪in melting glaciers, changing weather patterns, rising sea levels, floods,␣
 ↪cyclones and droughts. Global warming implies an increase in the average␣
 ↪temperature of the Earth due to entrapment of greenhouse gases in the␣
 ↪earth's atmosphere."""

# Order of the grams
n = 3

# Our N-Grams
ngrams = {}

# Building the model
words = nltk.word_tokenize(text)
for i in range(len(words)-n):
    gram = ' '.join(words[i:i+n])
    if gram not in ngrams.keys():
        ngrams[gram] = []
    ngrams[gram].append(words[i+n])

# Testing the model
currentGram = ' '.join(words[0:n])
result = currentGram
for i in range(30):
    if currentGram not in ngrams.keys():
        break
    possibilities = ngrams[currentGram]
    nextItem = possibilities[random.randrange(len(possibilities))]
    result += ' '+nextItem
    rWords = nltk.word_tokenize(result)
    currentGram = ' '.join(rWords[len(rWords)-n:len(rWords)])

print(result)
```

```
Global warming or climate change has become a worldwide concern . It is
gradually developing into an unprecedented environmental crisis evident in
melting glaciers , changing weather patterns , rising sea levels ,
```

14 Latent Semantic Analysis

[24]:
```
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.decomposition import TruncatedSVD

# Sample Data
dataset = ["The amount of pollution is increasing day by day",
           "The concert was just great",
           "I love to see Gordon Ramsay cook",
           "Google is introducing a new technology",
           "AI Robots are examples of great technology present today",
           "All of us were singing in the concert",
           "We have launched campaigns to stop pollution and global warming"]

dataset = [line.lower() for line in dataset]

# Creating Tfidf Model
vectorizer = TfidfVectorizer()
X = vectorizer.fit_transform(dataset)

# Creating the SVD
lsa = TruncatedSVD(n_components=4, n_iter=100)
lsa.fit(X)

# Visualizing the concepts
terms = vectorizer.get_feature_names_out()
for i, comp in enumerate(lsa.components_):
    component_terms = zip(terms, comp)
    sorted_terms = sorted(component_terms, key=lambda x: x[1], reverse=True)
    sorted_terms = sorted_terms[:10]
    print("\nConcept", i, ":")
    for term in sorted_terms:
        print(term[0], term[1])
```

```
Concept 0 :
the 0.375550932899052
concert 0.3398647517392518
of 0.2966590935014907
great 0.29534386566053705
day 0.23972438835590226
just 0.2334940280471319
was 0.2334940280471319
technology 0.18242602421743695
is 0.18039816898768132
all 0.17593921279666488
```

```
Concept 1 :
to 0.35489414747820014
pollution 0.23747906096178792
cook 0.22373677054460808
gordon 0.22373677054460808
love 0.22373677054460808
ramsay 0.22373677054460808
see 0.22373677054460808
and 0.20380230261581173
campaigns 0.20380230261581173
global 0.20380230261581173

Concept 2 :
technology 0.36817515941203693
google 0.31828506931586914
introducing 0.31828506931586914
new 0.31828506931586914
is 0.2945962922839355
are 0.1252535675825629
examples 0.1252535675825629
present 0.1252535675825629
robots 0.1252535675825629
today 0.1252535675825629

Concept 3 :
day 0.39280592289723876
pollution 0.21620148753224416
amount 0.1964029614486194
by 0.19640296144861938
increasing 0.19640296144861938
is 0.1587942790036917
the 0.10316082823817758
and 0.06405380925965967
campaigns 0.06405380925965963
global 0.06405380925965963
```

[2]:
```
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.decomposition import TruncatedSVD
import nltk

# Sample Data
dataset = ["The amount of pollution is increasing day by day",
           "The concert was just great",
           "I love to see Gordon Ramsay cook",
           "Google is introducing a new technology",
           "AI Robots are examples of great technology present today",
           "All of us were singing in the concert",
```

```
          "We have launched campaigns to stop pollution and global warming"]

dataset = [line.lower() for line in dataset]

# Creating Tfidf Model
vectorizer = TfidfVectorizer()
X = vectorizer.fit_transform(dataset)

# Creating the SVD
lsa = TruncatedSVD(n_components=4, n_iter=100)
lsa.fit(X)

# First Column of V
row1 = lsa.components_[3]

# Word Concept Dictionary Creation
concept_words = {}

# Visualizing the concepts
terms = vectorizer.get_feature_names_out()
for i, comp in enumerate(lsa.components_):
    componentTerms = zip(terms, comp)
    sortedTerms = sorted(componentTerms, key=lambda x: x[1], reverse=True)
    sortedTerms = sortedTerms[:10]
    concept_words["Concept " + str(i)] = sortedTerms

# Sentence Concepts
for key in concept_words.keys():
    sentence_scores = []
    for sentence in dataset:
        words = nltk.word_tokenize(sentence)
        score = 0
        for word in words:
            for word_with_score in concept_words[key]:
                if word == word_with_score[0]:
                    score += word_with_score[1]
        sentence_scores.append(score)
    print("\n" + key + ":")
    for sentence_score in sentence_scores:
        print(sentence_score)
```

```
Concept 0:
1.3320569721000244
1.477747606393112
0
0.36282419320511305
```

```
0.7744289833794628
1.1880139909364644
0

Concept 1:
0.23747906096179003
0
1.4735780002012835
0
0
0
1.203780116287445

Concept 2:
0.2945962922839444
0
0
1.6176266596436142
0.9944429973248607
0
0

Concept 3:
1.8529773249144623
0.10316082823817779
0
0.15879427900369394
0
0.10316082823817779
0.4083629153112168
```

15 Finding synonyms and antonyms of words

[3]:
```
from nltk.corpus import wordnet

# Initializing the list of synnonyms and antonyms
synonyms = []
antonyms = []

for syn in wordnet.synsets("good"):
    for s in syn.lemmas():
        synonyms.append(s.name())
        for a in s.antonyms():
            antonyms.append(a.name())


# Displaying the synonyms and antonyms
```

```
print(set(synonyms))
print(set(antonyms))
```

```
{'unspoiled', 'in_force', 'soundly', 'unspoilt', 'commodity', 'full', 'dear',
'proficient', 'salutary', 'estimable', 'skillful', 'adept', 'honorable',
'goodness', 'sound', 'near', 'honest', 'thoroughly', 'well', 'expert', 'secure',
'ripe', 'in_effect', 'respectable', 'dependable', 'just', 'trade_good',
'practiced', 'skilful', 'upright', 'undecomposed', 'serious', 'effective',
'safe', 'right', 'good', 'beneficial'}
{'evilness', 'bad', 'evil', 'badness', 'ill'}
```

16 Word Negation Tracking

[5]:
```
import nltk
from nltk.corpus import wordnet

sentence = "I was not happy with the team's performance"

words = nltk.word_tokenize(sentence)

new_words = []

temp_word = ''
for word in words:
    antonyms = []
    if word == 'not':
        temp_word = 'not_'
    elif temp_word == 'not_':
        for syn in wordnet.synsets(word):
            for s in syn.lemmas():
                for a in s.antonyms():
                    antonyms.append(a.name())
        if len(antonyms) >= 1:
            word = antonyms[0]
        else:
            word = temp_word + word
        temp_word = ''
    if word != 'not':
        new_words.append(word)

sentence = ' '.join(new_words)

sentence
```

[5]:
```
"I was unhappy with the team 's performance"
```