

CIS 668 Assignment 1 Report

Lichen Liang

Methods

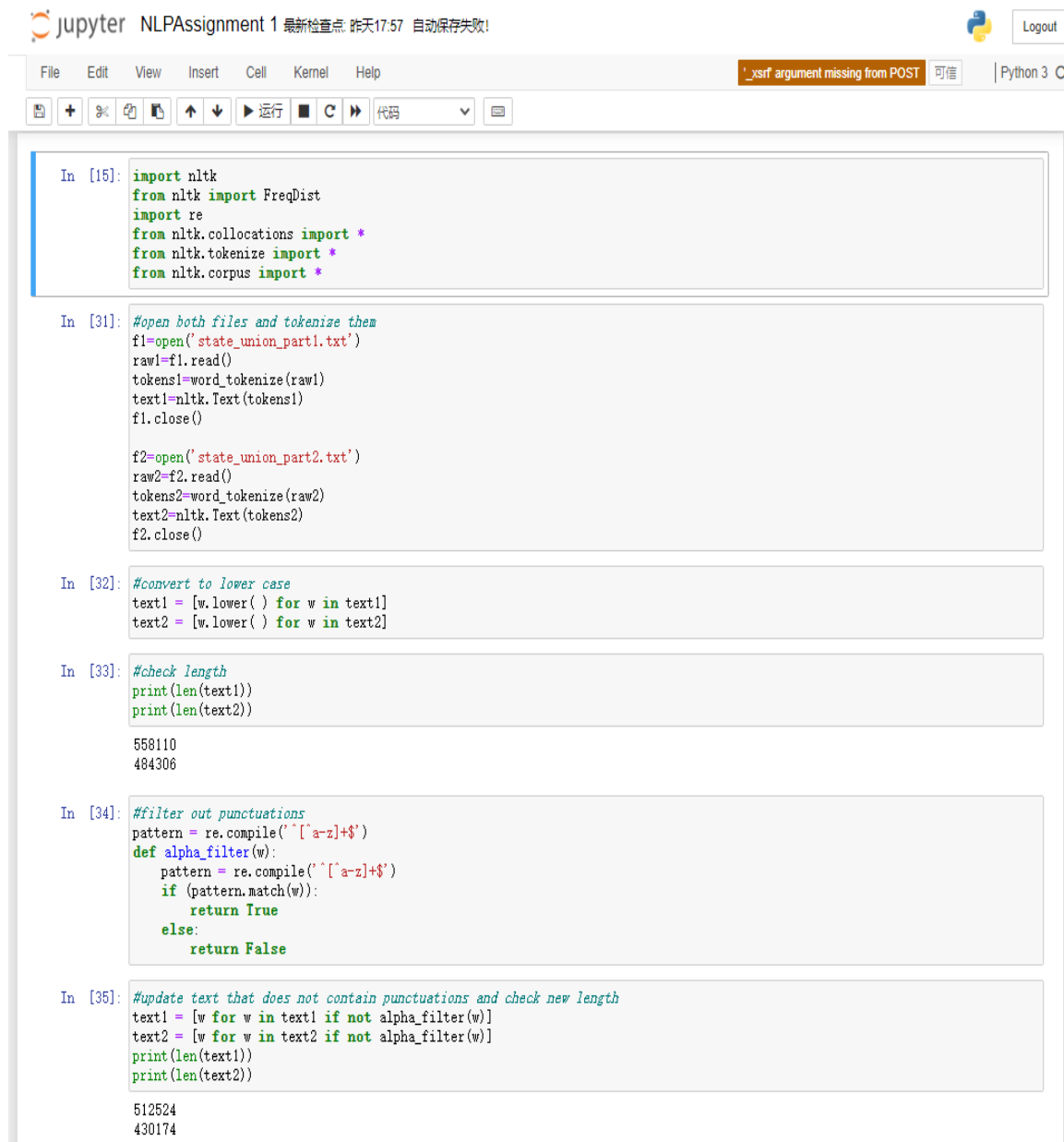
Firstly, open the two different files into raw and tokenize them into tokens. To simple things up, change all words into lower case. Using the filtering method discussed in class to filter out punctuations since we do not want any punctuations to show up in most frequent words or bigrams (figure 1). In addition, imported stop-words and added a few of our own. Filter again the texts for stop-words so that those irrelevant or undesired words will not show up in the result (figure 2). After two filters, apply the same methods used in lab2 on two different texts to find the 50 most frequent words, 50 most frequent bigrams using raw frequency and 50 most frequent bigrams using pmi and filter of 5 (figure 3-8 shows the methods and results).

Comparison

- A) Both texts are about presidential speech, so we can expect a lot of similarities in words such as: united, states, nation, people, government, congress, war, peace, economy, etc. However, there are lots of speeches in the two different text and many of them have topics that others do not mention. For example, a speech during war period will be much different from speech during peaceful period. Moreover, as time progress, the use of words in speeches are changing. Therefore, as expected, not all words can be matched. (figure 2-3, 5-6)
- B) Since we already filtered out stop-words and punctuations, we can focus on meaningful words. Again, we can expect some similarities such as: united states, american people, federal government, etc. At the same time, we also can expect differences due to different speech topics for the same reason as part A). In addition, as a bigram rather than single word, it is even harder to find more similarities between the two texts (figure 3-4, 6-7)
- C) Comparing with raw frequency and mutual information, we do not see to much similarities. When we use mutual information with filter, the results contains a lot of human and location names such as thomas jefferson, santa fe, rocky mountain, etc. This is because many human names are mentioned at the very beginning of the text and have appeared more than five times. Also, for location names, these do not change over time so they are the same in every speech. (figure 4-5, 7-8)

Potential Improvements: Using methods such as stemming and lemmatization to get root words can further help to get more specific results

Code and Results



```
In [15]: import nltk
from nltk import FreqDist
import re
from nltk.collocations import *
from nltk.tokenize import *
from nltk.corpus import *

In [31]: #open both files and tokenize them
f1=open('state_union_part1.txt')
raw1=f1.read()
tokens1=word_tokenize(raw1)
text1=nltk.Text(tokens1)
f1.close()

f2=open('state_union_part2.txt')
raw2=f2.read()
tokens2=word_tokenize(raw2)
text2=nltk.Text(tokens2)
f2.close()

In [32]: #convert to lower case
text1 = [w.lower() for w in text1]
text2 = [w.lower() for w in text2]

In [33]: #check length
print(len(text1))
print(len(text2))

558110
484306

In [34]: #filter out punctuations
pattern = re.compile('[^a-z]+$')
def alpha_filter(w):
    pattern = re.compile('[^a-z]+$')
    if (pattern.match(w)):
        return True
    else:
        return False

In [35]: #update text that does not contain punctuations and check new length
text1 = [w for w in text1 if not alpha_filter(w)]
text2 = [w for w in text2 if not alpha_filter(w)]
print(len(text1))
print(len(text2))

512524
430174
```

Figure 1.

```
In [36]: #import stopwords and add more stopwords
nltk.download('stopwords')
nltkstopwords = nltk.corpus.stopwords.words('english')
morestopwords = ['could', 'would', 'might', 'must', 'need', 'sha', 'wo', 'y', 's', 'd', 'll', 't', 'm', 're', 've']
stopwords = nltkstopwords + morestopwords
```

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

```
In [37]: ##update text that does not contain stopwords and check new length
text1 = [w for w in text1 if not w in stopwords]
text2 = [w for w in text2 if not w in stopwords]
print(len(text1))
print(len(text2))
```

```
237467
220789
```

```
In [38]: #find most common 50 words in state_union_part1.txt
text1dist = FreqDist(text1)
tlmc = text1dist.most_common(50)
tlmc
```

```
Out[38]: [('states', 2725),
('government', 2220),
('united', 1864),
('may', 1562),
('congress', 1500),
('upon', 1455),
('public', 1375),
('country', 1163),
('great', 1073),
('made', 1061),
('state', 1045),
('last', 911),
('war', 834),
('present', 812),
('time', 808),
('people', 786),
('year', 785),
('power', 744),
('citizens', 723),
('subject', 711),
('shall', 694),
('without', 663),
('union', 643),
('act', 627),
('treaty', 624),
('one', 620),
('part', 618),
('mexico', 605),
('general', 601),
('every', 590),
.]
```

Figure 2.

```
( 'general', 601),
( 'every', 590),
( 'treasury', 590),
( 'necessary', 575),
( 'constitution', 557),
( 'new', 548),
( 'duty', 529),
( 'foreign', 519),
( 'two', 510),
( 'commerce', 506),
( 'nations', 502),
( 'peace', 501),
( 'system', 494),
( 'laws', 492),
( 'duties', 488),
( 'within', 479),
( 'law', 477),
( 'us', 463),
( 'interests', 451),
( 'interest', 444),
( 'amount', 443),
( 'also', 438)]
```

In [39]: *#find most common 50 bigram in state_union_part1.txt by raw frequency*

```
bigram_measures = nltk.collocations.BigramAssocMeasures()
finder1 = BigramCollocationFinder.from_words(text1)
scored1 = finder1.score_ngrams(bigram_measures.raw_freq)
scored1[0:50]
```

```
Out[39]: [(('united', 'states'), 0.007672645041205725),
 (('great', 'britain'), 0.001153844534187908),
 (('last', 'session'), 0.0010190889681513642),
 (('public', 'debt'), 0.0007537889475169181),
 (('state', 'union'), 0.0007285222788850661),
 (('house', 'representatives'), 0.0006232444929190162),
 (('fiscal', 'year'), 0.0006064000471644481),
 (('union', 'address'), 0.0006064000471644481),
 (('report', 'secretary'), 0.0005853444899712381),
 (('public', 'lands'), 0.0005474444870234601),
 (('two', 'countries'), 0.0005137555955143241),
 (('present', 'year'), 0.00044637781249605207),
 (('within', 'limits'), 0.00042111114386420007),
 (('secretary', 'treasury'), 0.00041690003242555806),
 (('fellow', 'citizens'), 0.0004084778095482741),
 (('session', 'congress'), 0.0004042666981096321),
 (('act', 'congress'), 0.00039163336379370605),
 (('general', 'government'), 0.00039163336379370605),
 (('year', 'ending'), 0.00039163336379370605),
 (('british', 'government'), 0.0003874222523550641),
 (('two', 'governments'), 0.00037478891803913807),
 (('citizens', 'united'), 0.0003621555837232121),
 (('federal', 'government'), 0.0003579444722845701),
 (('secretary', 'war'), 0.00035373336084592807),
 (('annual', 'message'), 0.00034110002653000204),
 (('public', 'service'), 0.0003368889150913601),
 (('senate', 'house'), 0.0003326778036527181)]
```

Figure 3.

```
(('senate', 'house'), 0.0003326176036021181),
(('consideration', 'congress'), 0.00032425558077543406),
(('last', 'annual'), 0.00031583335789815004),
(('attention', 'congress'), 0.0003116222464595081),
(('government', 'united'), 0.00030741113502086607),
(('public', 'money'), 0.00029056668926629803),
(('indian', 'tribes'), 0.00027793335495037206),
(('mexican', 'government'), 0.00027372224351173005),
(('part', 'united'), 0.00027372224351173005),
(('treasury', 'notes'), 0.00027372224351173005),
(('upon', 'subject'), 0.00026951113207308804),
(('commercial', 'intercourse'), 0.000265300020634446),
(('several', 'states'), 0.000265300020634446),
(('secretary', 'state'), 0.00026108890919580407),
(('provision', 'made'), 0.00025687779775716206),
(('article', 'treaty'), 0.00024003335200259405),
(('claims', 'citizens'), 0.00024003335200259405),
(('address', 'december'), 0.00023582224056395204),
(('ending', '30th'), 0.00023582224056395204),
(('new', 'mexico'), 0.00023582224056395204),
(('favorable', 'consideration'), 0.00023161112912531006),
(('naval', 'force'), 0.00023161112912531006),
(('30th', 'june'), 0.00022740001768666805),
(('bank', 'united'), 0.00022740001768666805)]
```

```
In [40]: #find most common 50 bigram in state_union_part1.txt by pmi and min frequency = 5
finder2 = BigramCollocationFinder.from_words(text1)
finder2.apply_freq_filter(5)
scored2 = finder2.score_ngrams(bigram_measures.pmi)
scored2[:50]
```

```
Out[40]: [('bona', 'fide'), 15.535439420385778),
(('posse', 'comitatus'), 15.535439420385778),
(('punta', 'arenas'), 15.535439420385778),
(('ballot', 'box'), 15.272405014551982),
(('del', 'norte'), 15.272405014551982),
(('millard', 'fillmore'), 15.272405014551982),
(('clayton', 'bulwer'), 14.85736751527314),
(('guadalupe', 'hidalgo'), 14.687442513830828),
(('porto', 'rico'), 14.687442513830828),
(('writ', 'mandamus'), 14.594333109439344),
(('franklin', 'pierce'), 14.535439420385778),
(('la', 'plata'), 14.397935896635842),
(('vera', 'cruz'), 14.272405014551982),
(('entangling', 'alliances'), 14.20201568660584),
(('seminaries', 'learning'), 14.00937060871819),
(('gun', 'boats'), 13.880087591773222),
(('nucleus', 'around'), 13.85736751527314),
(('ruler', 'universe'), 13.85736751527314),
(('costa', 'rica'), 13.857367515273136),
(('santa', 'anna'), 13.7699046740228),
(('santa', 'fe'), 13.7699046740228),
(('van', 'buren'), 13.7699046740228),
(('project', 'gutenberg'), 13.769904674022799),
(('sublime', 'porte'), 13.728084498328172),
(('tea', 'coffee'), 13.609440001829554).
```

Figure 4.

```
(('sublime', 'poise'), 13.120004490320112),
(('tea', 'coffee'), 13.609440001829554),
(('martin', 'van'), 13.599979672580488),
(('ad', 'valorem'), 13.535439420385776),
(('beacons', 'buoys'), 13.397935896635842),
(('water', 'witch'), 13.397935896635842),
(('quincy', 'adams'), 13.39793589663584),
(('statute', 'book'), 13.333805559216128),
(('buenos', 'ayres'), 13.27240501455198),
(('indiana', 'illinois'), 13.134901490802047),
(('de', 'facto'), 13.12401317465931),
(('franking', 'privilege'), 13.10248001310967),
(('rocky', 'mountains'), 13.050012593215534),
(('andrew', 'jackson'), 12.967550433023561),
(('retired', 'list'), 12.9125090694656),
(('sooner', 'later'), 12.872474407663347),
(('circulating', 'medium'), 12.812973395914685),
(('intent', 'meaning'), 12.79435771774734),
(('th', 'jefferson'), 12.7699046740228),
(('john', 'quincy'), 12.769904674022799),
(('precious', 'metals'), 12.71101098496923),
(('thomas', 'jefferson'), 12.68244183277246),
(('lake', 'erie'), 12.62854882477726),
(('almighty', 'god'), 12.599979672580488),
(('john', 'tyler'), 12.599979672580488),
(('san', 'jacinto'), 12.571965296410891),
(('san', 'juan'), 12.571965296410891)]
```

```
In [41]: #find most common 50 words in state_union_part2.txt
text2dist = FreqDist(text2)
t2mc = text2dist.most_common(50)
t2mc
```

```
Out[41]: [('people', 1506),
('world', 1490),
('new', 1441),
('america', 1271),
('year', 1265),
('congress', 1230),
('us', 1216),
('government', 1111),
('years', 1111),
('american', 950),
('nation', 861),
('one', 804),
('every', 780),
('make', 778),
('work', 754),
('federal', 744),
('time', 741),
('states', 711),
('americans', 688),
('help', 686),
('security', 685),
('war', 674),
('economic', 671),
```

Figure 5.

```

('economic', 671),
('peace', 668),
('united', 651),
('nations', 645),
('also', 639),
('program', 638),
('country', 630),
('national', 609),
('economy', 588),
('great', 583),
('last', 572),
('many', 564),
('free', 558),
('first', 553),
('let', 549),
('state', 520),
('tax', 514),
('know', 507),
('million', 507),
('freedom', 503),
('budget', 501),
('health', 489),
('n't', 479),
('future', 475),
('system', 463),
('programs', 462),
('tonight', 461),
('union', 460)]

In [42]: #find most common 50 bigram in state_union_part2.txt by raw frequency
finder3 = BigramCollocationFinder.from_words(text2)
scored3 = finder3.score_ngrams(bigram_measures.raw_freq)
scored3[0:50]

Out[42]: [('united', 'states'), 0.002092495550050048],
          [('state', 'union'), 0.001209299376327625],
          [('american', 'people'), 0.0010824814642033797],
          [('last', 'year'), 0.0010190725081412571],
          [('fiscal', 'year'), 0.0008424332733967725],
          [('federal', 'government'), 0.0008333748511021835],
          [('social', 'security'), 0.0008243164288075945],
          [('health', 'care'), 0.0008061995842184167],
          [('let', 'us'), 0.0007971411619238278],
          [('years', 'ago'), 0.000733732205861705],
          [('union', 'address'), 0.0006250311383266376],
          [('united', 'nations'), 0.0006114435048847543],
          [('billion', 'dollars'), 0.0005887974491482819],
          [('million', 'dollars'), 0.0005752098157063984],
          [('soviet', 'union'), 0.0005661513934118095],
          [('men', 'women'), 0.0005118008596442758],
          [('free', 'world'), 0.0004936840150550979],
          [('ca', 'n't'), 0.0004619795370240365],
          [('every', 'american'), 0.0004483919035821531],
          [('economic', 'growth'), 0.0004257458478456807],
          [('middle', 'east'), 0.00041215821440379727],
          [('make', 'sure'), 0.0003985705809619139].

```

Figure 6.

```

(('make', 'sure'), 0.0003985705809619139),
(('free', 'nations'), 0.00038498294752003045),
(('first', 'time'), 0.0003668661029308525),
(('four', 'years'), 0.0003668661029308525),
(('state', 'local'), 0.00036233689178355807),
(('ask', 'congress'), 0.00035327846948896913),
(('members', 'congress'), 0.00034422004719438014),
(('armed', 'forces'), 0.0003396908360470857),
(('world', 'war'), 0.0003396908360470857),
(('next', 'years'), 0.0003351616248997912),
(('work', 'together'), 0.0003351616248997912),
(('21st', 'century'), 0.0003306324137524967),
(('foreign', 'policy'), 0.0003170447803106133),
(('mr.', 'speaker'), 0.0003170447803106133),
(('new', 'jobs'), 0.0003170447803106133),
(('two', 'years'), 0.0003034571468687299),
(('vice', 'president'), 0.0003034571468687299),
(('around', 'world'), 0.00028986951342684645),
(('national', 'security'), 0.00028534030227955195),
(('address', 'january'), 0.00027175266883766857),
(('human', 'rights'), 0.00026722345769037407),
(('health', 'insurance'), 0.0002626942465430796),
(('fellow', 'americans'), 0.00025363582424849063),
(('fellow', 'citizens'), 0.00025363582424849063),
(('past', 'year'), 0.00025363582424849063),
(('past', 'years'), 0.00025363582424849063),
(('states', 'america'), 0.00025363582424849063),
(('civil', 'rights'), 0.0002445774019539017),
(('young', 'people'), 0.0002445774019539017)]

```

In [43]: `#find most common 50 bigram in state_union_part2.txt by pmi and min frequency = 5`

```

finder4 = BigramCollocationFinder.from_words(text2)
finder4.apply_freq_filter(5)
scored4 = finder4.score_ngrams(bigram_measures.pmi)
scored4[.50]

```

Out[43]:

```

[('el', 'salvador'), 15.167346270647108),
 ('ladies', 'gentlemen'), 15.167346270647108),
 ('bin', 'laden'), 14.94495384931066),
 ('saudi', 'arabia'), 14.944953849310657),
 ('sam', 'rayburn'), 14.752308771368263),
 ('gerald', 'r.'), 14.529916350031815),
 ('jimmy', 'carter'), 14.430380676480901),
 ('endowed', 'creator'), 14.319349364092155),
 ('vol', 'p.'), 14.292877152730966),
 ('northern', 'ireland'), 14.167346270647108),
 ('o'neill', 'jr.'), 14.096956942755707),
 ('r.', 'ford'), 14.070484731394519),
 ('lyndon', 'b.'), 14.051869053227172),
 ('floor', 'appears'), 14.015343177202059),
 ('iron', 'curtain'), 13.944953849310657),
 ('grass', 'roots'), 13.904311864813312),
 ('200th', 'anniversary'), 13.845418175759747),
 ('william', 'j.'), 13.845418175759747),
 ('thomas', 'jefferson'), 13.788834647393376),
 ('', ''), 13.752308771368265)

```

Figure 7.

```

(('thomas', 'jefferson'), 13.788834647393376),
(('red', 'tape'), 13.752308771368265),
(('sons', 'daughters'), 13.752308771368263),
(('jill', 'biden'), 13.681919443476865),
(('b.', 'johnson'), 13.664845930117925),
(('barack', 'obama'), 13.664845930117922),
(('teen', 'pregnancy'), 13.58238376992595),
(('abraham', 'lincoln'), 13.494920928675612),
(('mom', 'dad'), 13.459527022140415),
(('p.', 'o'neill'), 13.444880246176016),
(('j.', 'clinton'), 13.430380676480903),
(('empowerment', 'zones'), 13.359991348589503),
(('ronald', 'reagan'), 13.292877152730965),
(('synthetic', 'fuels'), 13.278377583035851),
(('small-business', 'owner'), 13.26688194419802),
(('old-age', 'survivors'), 13.2173869531467),
(('greece', 'turkey'), 13.207988255144453),
(('elementary', 'secondary'), 13.12586963467095),
(('harry', 's.'), 13.089343758645832),
(('dwight', 'd.'), 13.029842746897174),
(('intercontinental', 'ballistic'), 13.006354393974803),
(('h.w.', 'bush'), 12.997421269204796),
(('w.', 'bush'), 12.997421269204796),
(('feeding', 'hungry'), 12.970949057843605),
(('small-business', 'owners'), 12.94495384931066),
(('thomas', 'p.'), 12.914365529477235),
(('river', 'basins'), 12.894327776240692),
(('status', 'quo'), 12.89432777624069),
(('commander', 'chief'), 12.859223975284776),
(('prime', 'minister'), 12.845418175759743),
(('nationwide', 'radio'), 12.804776191262398),
(('spoke', 'p.m.'), 12.79811246098139)]

```

Figure 8.