Text Indexing Code Report

March 21, 2019

0.0.1 Name : Kang Yeongeun

0.0.2 StudentNo. 20151532

github : https://github.com/yeonun/NLP_Assignment

1 Import the pakages

1.1 re for text data, panda for data analysis, matplot for drawing graph and numpy for calculating

```
In [1]: from collections import Counter
        import re
        import matplotlib.pyplot as plt
        import pandas as pd
        import numpy as np
```

2 File import

```
In [2]: f=open("./500 DAYS OF SUMMER.txt")
        data = f.read()
```

3 Convert all characters to lowercase and parse them word by word

```
In [3]: parse = re.sub("[^0-9a-zA-Z\\s]","",data)
        parse = parse.lower().split()
```

3.1 Save the names of the characters written in capital letters in the script

```
In [4]: charnamelist = re.sub("[^0-9a-zA-Z\\s]","",data)
        charnamelist = charnamelist.split()

        charname = []
        lowercharname =[]
        for s in charnamelist:
            if s.isupper():
```

```
            charname.append(s)
            lowercharname.append(s.lower())
```

4 Use 'Counter' to count by word

```
In [5]: counts = Counter(parse)
        counts = counts.most_common()
```

5 Made a except word list for meaningful statistics.

5.1 article, conjunctions, pronoun, charactrer name, etc...

```
In [6]: articles = ["a","the","an"]
        conjunctions = ["and","or","as","but","nor","so","while","although","however"\
                        ,"instead","moreover","furthermore","likewise","specifically"\
                        ,"way","yet","for","because","since","actually","that","though",\
                        "admittedly","thus","therefor","after","before","when",\
                        "while","until","whenever","next","first","second","finally",\
                        "meanwile","until","unless","seen","also","beside","then",\
                       "just","by","no","why","about","here","there","where","how"\
                       ,"theres"]
        pronouns = ["i","my","me","you","he","she","it","we","they","mine","yours",\
                    "this","these","thats","those","who","what","which","one","none"\
                    ,"any","some","each","every","other","others","another","anbody",\
                    "its","her","his","him","was","were","dont","youre"\
                   ,"their","your","shes","hes","them"]
        etc = ["to","of","in","at","is","be","are","am","if","with","will","on",\
               "has","had","im","do","not","from","now","into","up"\
              ,"can","like","have","know","well","cant","been"]


        exceptword = articles + conjunctions + pronouns + etc + charname + lowercharname
```

6 Create a list of filtered words

```
In [7]: length = len(counts)
        newcount = []
        for i in range(length):
            if counts[i][0] not in exceptword:
                newcount.append(counts[i])
```

7 Create a DataFrame with filtered top 20 word

```
In [8]: newcount_to_frame = pd.DataFrame(newcount[:20], columns=["word","counts"])
        countsum = sum(newcount_to_frame["counts"])
        print("""filtered top 20 are:
```

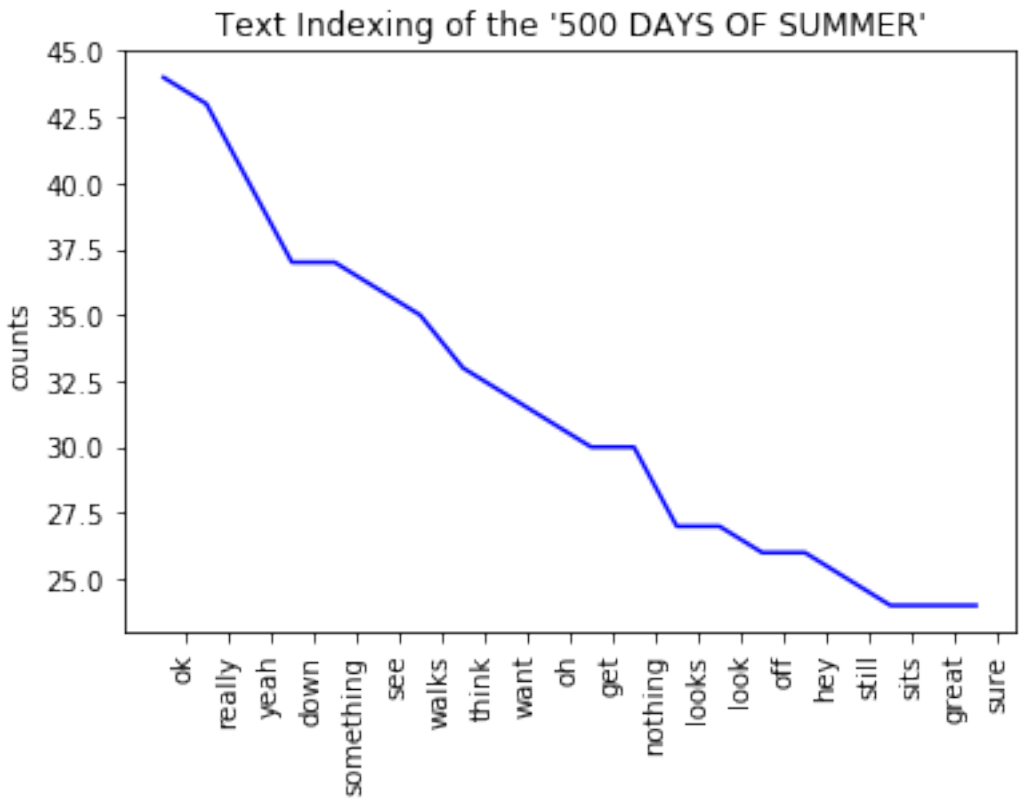
```
        """, newcount_to_frame[:20])
```

```
filtered top 20 are:

          word  counts
0           ok      44
1       really      43
2         yeah      40
3         down      37
4    something      37
5          see      36
6        walks      35
7        think      33
8         want      32
9           oh      31
10         get      30
11     nothing      30
12       looks      27
13        look      27
14         off      26
15         hey      26
16       still      25
17        sits      24
18       great      24
19        sure      24
```

8 Create a graph

```
In [9]: fword = [newcount[i][0] for i in range(len(newcount))][:20]
        fnumber = [newcount[i][1] for i in range(len(newcount))][:20]
        fxs = [i for i, _ in enumerate(fword)]
        plt.plot(fxs, fnumber,'b')
        plt.ylabel("counts")
        plt.xticks([i+0.5 for i, _ in enumerate(fword)],fword,rotation = 90)
        plt.title("Text Indexing of the '500 DAYS OF SUMMER'")
        plt.show()
```


Text Indexing of the '500 DAYS OF SUMMER'
counts
45.0
42.5
40.0
37.5
35.0
32.5
30.0
27.5
25.0
ok
really
yeah
down
something
see
walks
think
want
oh
get
nothing
looks
look
off
hey
still
sits
great
sure

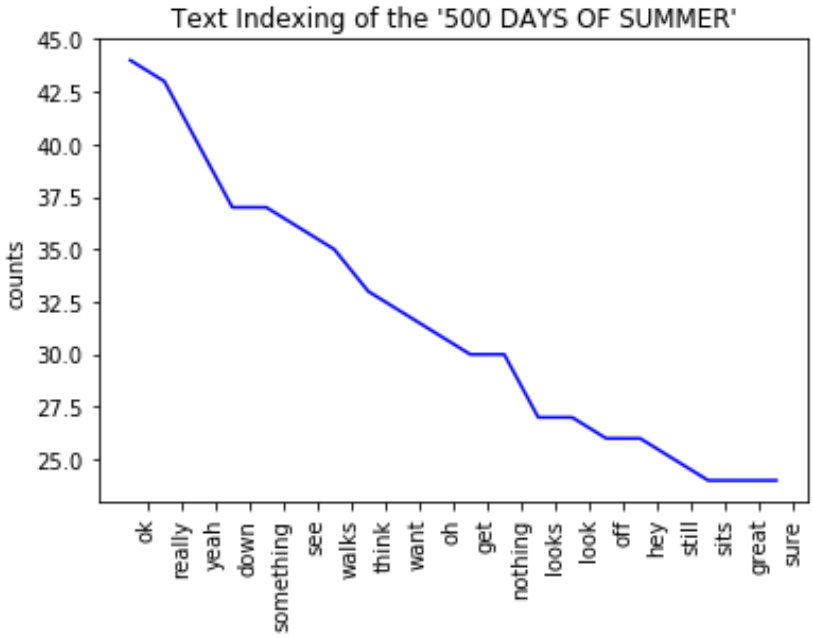
Data set (10 movie scripts)

- 500 DAYS OF SUMMER
- INCEPTION
- INTERSTELLAR
- IT
- LA LA LAND
- LES MISERABLES
- THE AVENGERS
- THE CURIOUS CASE OF BENJAMIN BUTTON
- V FOR VENDETTA
- ZOOTOPIA

- **500 DAYS OF SUMMER**

filtered top 20 are:

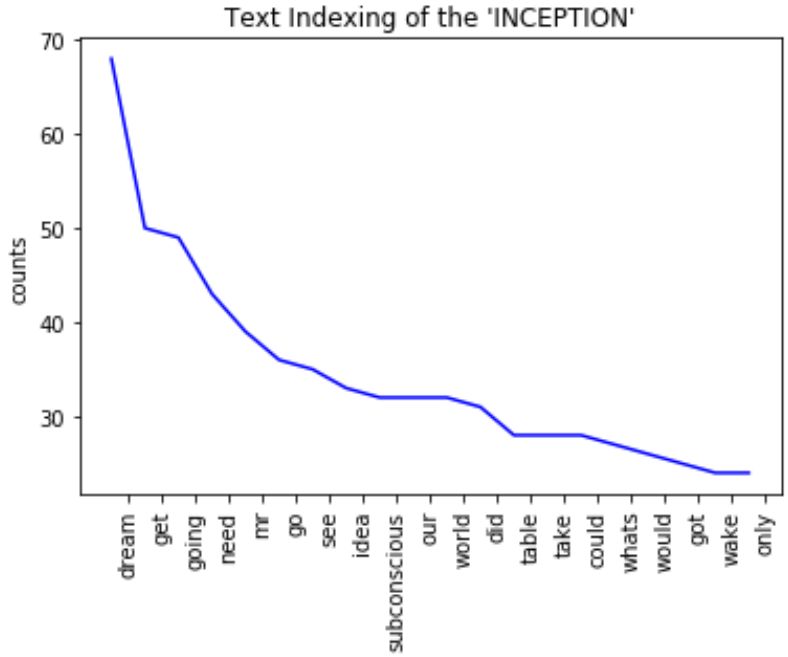
	word	counts
0	ok	44
1	really	43
2	yeah	40
3	down	37
4	something	37
5	see	36
6	walks	35
7	think	33
8	want	32
9	oh	31
10	get	30
11	nothing	30
12	looks	27
13	look	27
14	off	26
15	hey	26
16	still	25
17	sits	24
18	great	24
19	sure	24



- **INCEPTION**

filtered top 20 are:

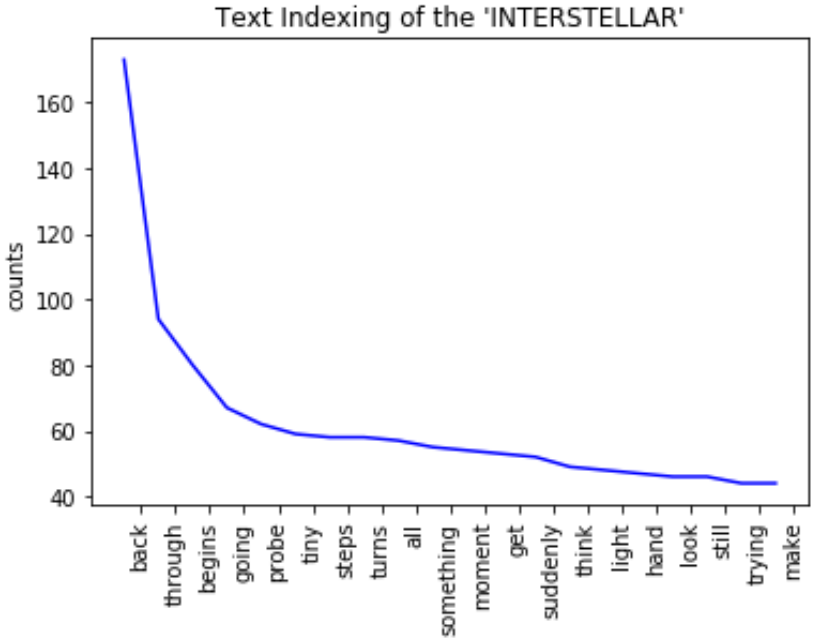
	word	counts
0	dream	68
1	get	50
2	going	49
3	need	43
4	mr	39
5	go	36
6	see	35
7	idea	33
8	subconscious	32
9	our	32
10	world	32
11	did	31
12	table	28
13	take	28
14	could	28
15	whats	27
16	would	26
17	got	25
18	wake	24
19	only	24



- **INTERSTELLAR**

filtered top 20 are:

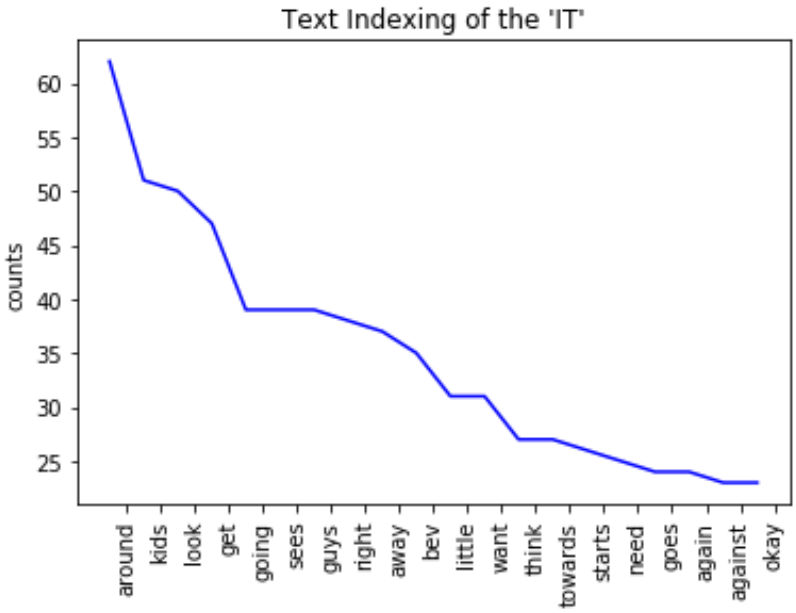
	word	counts
0	back	173
1	through	94
2	begins	80
3	going	67
4	probe	62
5	tiny	59
6	steps	58
7	turns	58
8	all	57
9	something	55
10	moment	54
11	get	53
12	suddenly	52
13	think	49
14	light	48
15	hand	47
16	look	46
17	still	46
18	trying	44
19	make	44



- **IT**

filtered top 20 are:

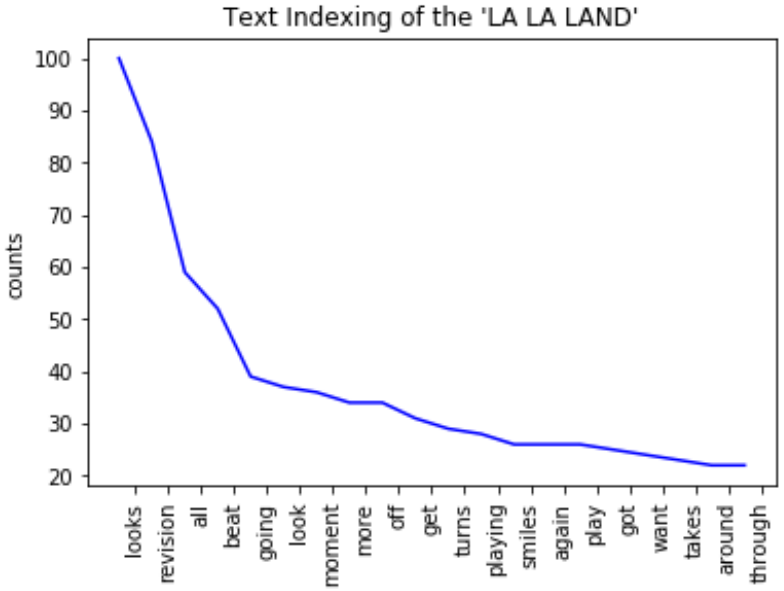
	word	counts
0	around	62
1	kids	51
2	look	50
3	get	47
4	going	39
5	sees	39
6	guys	39
7	right	38
8	away	37
9	bev	35
10	little	31
11	want	31
12	think	27
13	towards	27
14	starts	26
15	need	25
16	goes	24
17	again	24
18	against	23
19	okay	23



- **LA LA LAND**

filtered top 20 are:

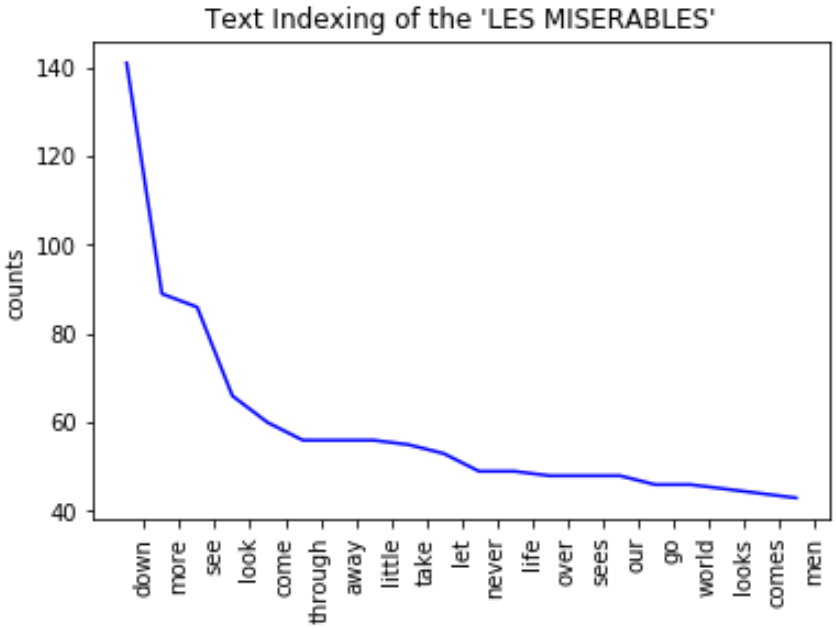
	word	counts
0	looks	100
1	revision	84
2	all	59
3	beat	52
4	going	39
5	look	37
6	moment	36
7	more	34
8	off	34
9	get	31
10	turns	29
11	playing	28
12	smiles	26
13	again	26
14	play	26
15	got	25
16	want	24
17	takes	23
18	around	22
19	through	22



- **LES MISERABLES**

filtered top 20 are:

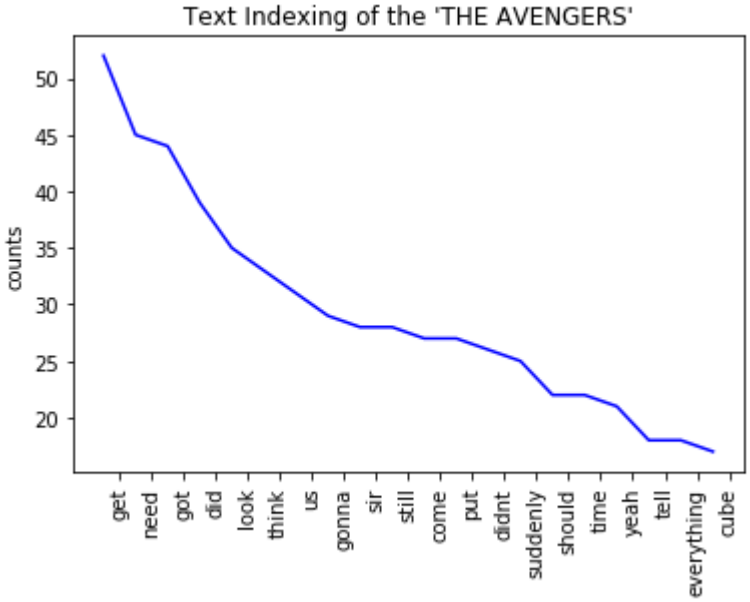
	word	counts
0	down	141
1	more	89
2	see	86
3	look	66
4	come	60
5	through	56
6	away	56
7	little	56
8	take	55
9	let	53
10	never	49
11	life	49
12	over	48
13	sees	48
14	our	48
15	go	46
16	world	46
17	looks	45
18	comes	44
19	men	43



- **THE AVENGERS**

filtered top 20 are:

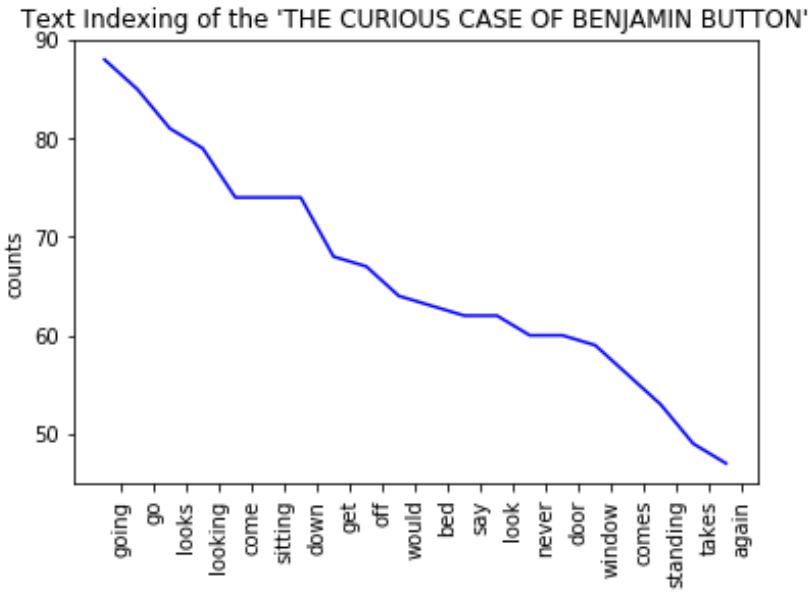
	word	counts
0	get	52
1	need	45
2	got	44
3	did	39
4	look	35
5	think	33
6	us	31
7	gonna	29
8	sir	28
9	still	28
10	come	27
11	put	27
12	didnt	26
13	suddenly	25
14	should	22
15	time	22
16	yeah	21
17	tell	18
18	everything	18
19	cube	17



- **THE CURIOUS CASE OF BENJAMIN BUTTON**

filtered top 20 are:

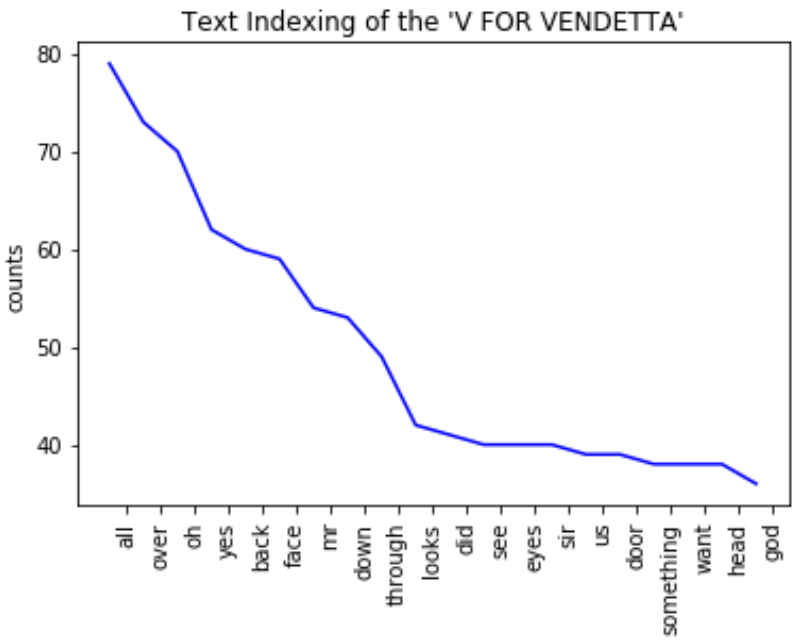
	word	counts
0	going	88
1	go	85
2	looks	81
3	looking	79
4	come	74
5	sitting	74
6	down	74
7	get	68
8	off	67
9	would	64
10	bed	63
11	say	62
12	look	62
13	never	60
14	door	60
15	window	59
16	comes	56
17	standing	53
18	takes	49
19	again	47



- **V FOR VENDETTA**

filtered top 20 are:

	word	counts
0	all	79
1	over	73
2	oh	70
3	yes	62
4	back	60
5	face	59
6	mr	54
7	down	53
8	through	49
9	looks	42
10	did	41
11	see	40
12	eyes	40
13	sir	40
14	us	39
15	door	39
16	something	38
17	want	38
18	head	38
19	god	36



- **ZOOTOPIA**

filtered top 20 are:

	word	counts
0	looks	59
1	all	58
2	off	54
3	go	47
4	gonna	40
5	see	40
6	hey	40
7	did	37
8	okay	36
9	find	35
10	get	34
11	yeah	34
12	look	31
13	down	30
14	got	30
15	sir	30
16	think	29
17	going	28
18	our	25
19	thank	23