COMP40020 Assignment 2

Name: Bhavesh Kumar

Student Number: 16203173

In [52]:

```
import nltk
from nltk.corpus import stopwords
```

Task 1

In [53]:

```
# read data from file
def load_txt(file):
    with open(file, encoding="utf8") as f:
        return ' '.join(line.strip() for line in f.readlines())

# calculate lexical diversity
def lexical_diversity(text):
    return len(set(text)) / len(text)
```

Read lyrics from directory

In [54]:

```
moon_river       = load_txt('lyrics/Moon River.txt')
church_of        = load_txt('lyrics/Church of atrocity.txt')
evocation_under  = load_txt('lyrics/Evocation Under.txt')
blue_sky         = load_txt('lyrics/Blue Sky.txt')
```

Tokenize lyrics

In [55]:

```
tokens_moon_river       = nltk.word_tokenize(moon_river)
tokens_church_of        = nltk.word_tokenize(church_of)
tokens_evocation_under  = nltk.word_tokenize(evocation_under)
tokens_blue_sky         = nltk.word_tokenize(blue_sky)
```

Calculate lexical diversity

In [56]:

```
score_moon_river        = lexical_diversity(tokens_moon_river)
score_church_of         = lexical_diversity(tokens_church_of)
score_evocation_under   = lexical_diversity(tokens_evocation_under)
score_blue_sky          = lexical_diversity(tokens_blue_sky)

print("Diversity Score")
print("Classic songs")
print("Moon River:", score_moon_river)
print("Blue Sky:", score_blue_sky)
print('----------')
print("Modern songs")
print("Church Of Atrocity:", score_church_of)
print("Evocation under:", score_evocation_under)
```

```
Diversity Score
Classic songs
Moon River: 0.3561643835616438
Blue Sky: 0.3682539682539683
----------
Modern songs
Church Of Atrocity: 0.6762589928057554
Evocation under: 0.6683417085427136
```

Task 1 solution explanation:

Songs used this experiment: Moon River, Released 1961 Mr. Blue Sky, Released 1977 Church Of Atrocity, Released 2006 Evocation under starlit sky, Released

In this task, two modern and two classic songs have been selected after experimenting on various songs to select 4 among them to act as a counterexample as mentioned in the requirement of Task 1. As we can see the Diversity Score of classic songs Moon River Released year 1961 and Blue Sky Released year 1977, which is around 35% and 36% respectively, and for modern songs score has reached to 67% for one of the song (Church Of Atrocity Released year 2006).

It can be concluded that it would not be correct to assume modern songs will always have less diversity score than the classic song.

Task 2

Read stories from directory

In [57]:

```
aisle_six = load_txt('fiction/aisle.six.txt')
bestwish  = load_txt('fiction/bestwish.txt')
bluebrd   = load_txt('fiction/bluebrd.txt')
```

Filter data

In [58]:

```
print(aisle_six.find('ABCO -'))
print(aisle_six.find('Use the watermelon!'))

# parse only required content
aisle_six_final = aisle_six[546:4207]
```

```
539
4207
```

Tokenize stories

In [59]:

```
tokens_aisle_six = nltk.word_tokenize(aisle_six_final)
tokens_bestwish  = nltk.word_tokenize(bestwish)
tokens_bluebrd   = nltk.word_tokenize(bluebrd)
```

Part of speech frequency calculation (POS)

In [60]:

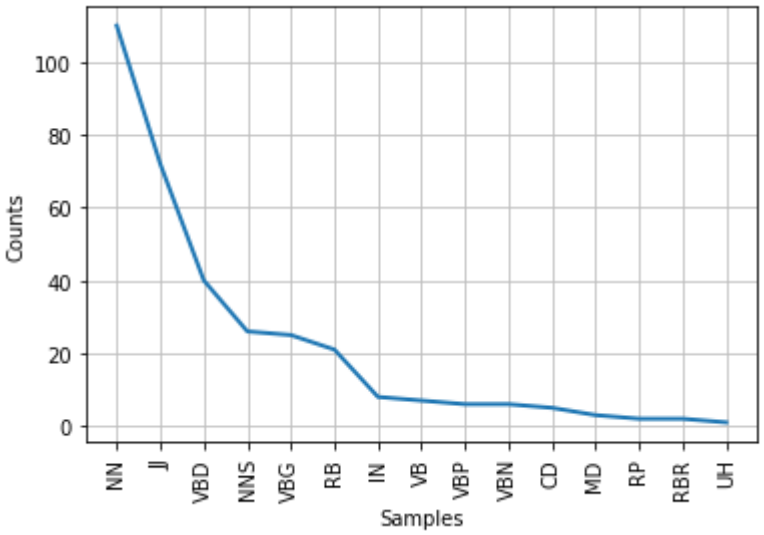
```
pos_freq = {"aisle_six":[],
            "bestwish" :[],
            "bluebrd"  :[]}

# remove punctuations
tokens_aisle_six = [t.lower() for t in tokens_aisle_six if t.isalnum()]
tokens_bestwish  = [t.lower() for t in tokens_bestwish if t.isalnum()]
tokens_bluebrd   = [t.lower() for t in tokens_bluebrd if t.isalnum()]

# remove stop words
filtered_aisle_six = [w.lower() for w in tokens_aisle_six if w.lower() not in stopwords.wor
filtered_bestwish  = [w.lower() for w in tokens_bestwish if w.lower() not in stopwords.word
filtered_bluebrd   = [w.lower() for w in tokens_bluebrd if w.lower() not in stopwords.words

''' Using nltk post_tag function to process filtered tokens to generate part of speech for
and associate with it and storing data in post_tagged dictionary so that it can be used to
a list for POS frequencies
'''
pos_tagged = {"aisle_six":nltk.pos_tag(filtered_aisle_six),
              "bestwish" :nltk.pos_tag(filtered_bestwish),
              "bluebrd"  :nltk.pos_tag(filtered_bluebrd)}

''' Iterating through every stories in the post_tagged list to get tag out of it and append
so that it can be used to plot in the later steps
'''
for key, value in pos_tagged.items():
    for word, tag in value:
         pos_freq[key].append(tag)
```

Task 2.a) Print frequencies of POS

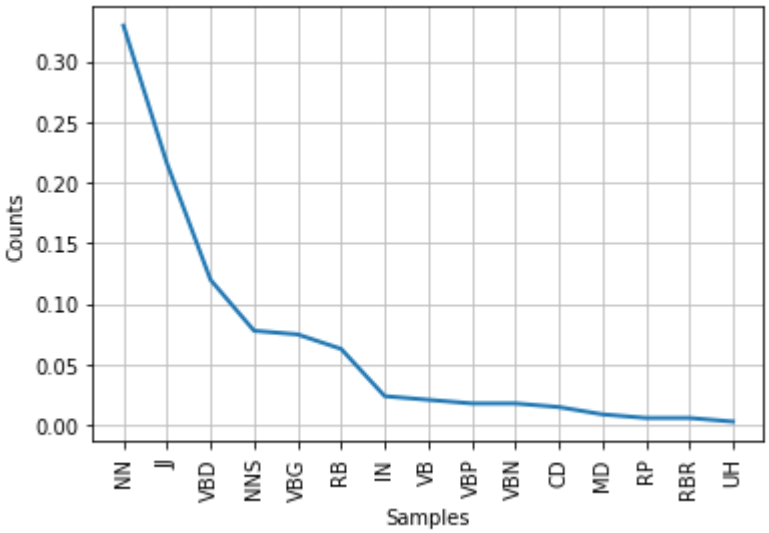
In [61]:

```
# print and plot the part of speech frequencies distribution of Story Aisle Six
fd_aisle_six = nltk.FreqDist(pos_freq['aisle_six'])
fd_aisle_six.tabulate()
fd_aisle_six.plot(20)

# calculate the percentage of POS used in Story Aisle Six
print('Plotting percentage of part of speech used in Aisle Six')
total_count = fd_aisle_six.N()
for word in fd_aisle_six:
    fd_aisle_six[word] /= float(total_count)
fd_aisle_six.plot(20)
```

```
 NN  JJ VBD NNS VBG  RB  IN  VB VBP VBN  CD  MD  RP RBR  UH
110  72  40  26  25  21   8   7   6   6   5   3   2   2   1
```



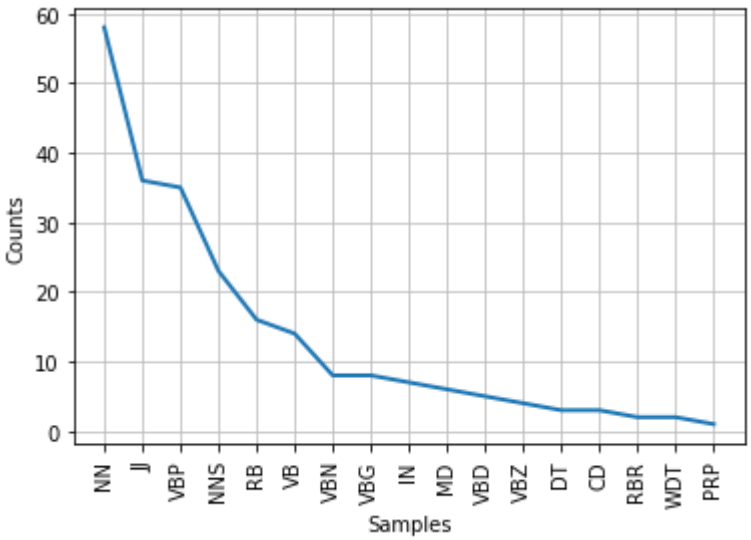
Plotting percentage of part of speech used in Aisle Six



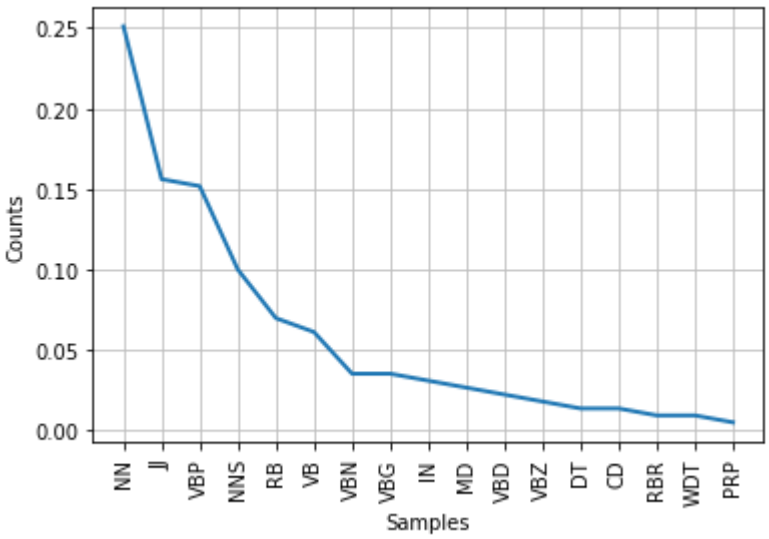
In [62]:

```
# print and plot the part of speech frequencies distribution of Story Best wish
fd_bestwish = nltk.FreqDist(pos_freq['bestwish'])

print('Quantitative details of each POS used in bestwish')
fd_bestwish.tabulate()

print('\nPlotting frequency distribution of POS in bestwish')
fd_bestwish.plot(20)

# calculate the percentage of POS used in Story Best wish
print('Plotting percentage of part of speech used in Best wish')
total_count = fd_bestwish.N()
for word in fd_bestwish:
    fd_bestwish[word] /= float(total_count)
fd_bestwish.plot(20)
```

```
Quantitative details of each POS used in bestwish
 NN  JJ VBP NNS  RB  VB VBN VBG  IN  MD VBD VBZ  DT  CD RBR WDT PRP
 58  36  35  23  16  14   8   8   7   6   5   4   3   3   2   2   1

Plotting frequency distribution of POS in bestwish
```



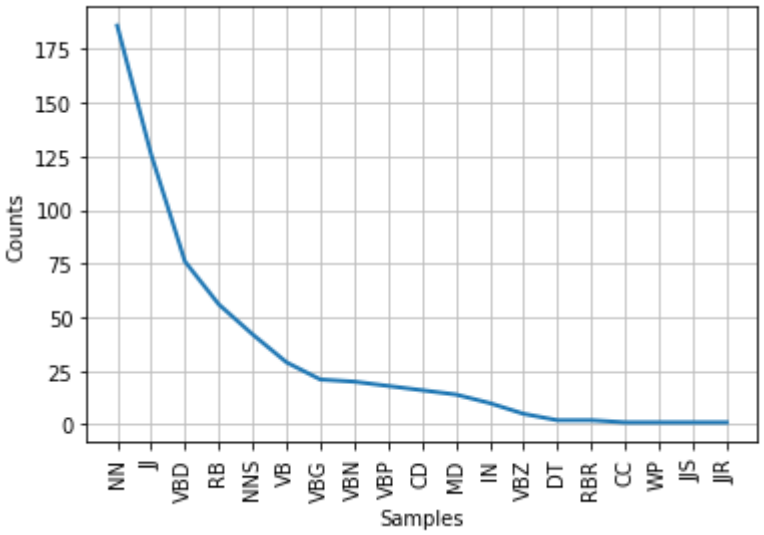

```
Plotting percentage of part of speech used in Best wish
```



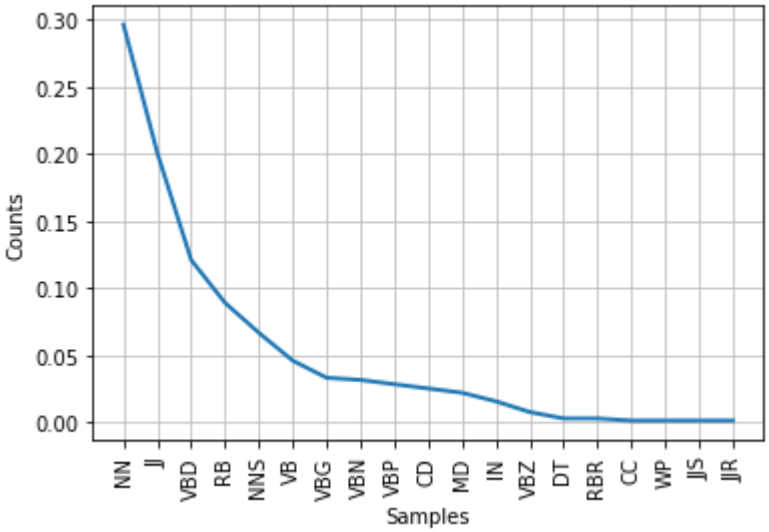
In [63]:

```
# print and plot the part of speech frequencies distribution of Story Bluebird
fd_bluebrd = nltk.FreqDist(pos_freq['bluebrd'])

print('Quantitative details of each POS used in Bluebird')
fd_bluebrd.tabulate()

print('\nPlotting frequency distribution of POS in Bluebird')
fd_bluebrd.plot(20)

# calculate the percentage of POS used in Story Bluebird
print('Plotting percentage of part of speech used in Bluebird')
total_count = fd_bluebrd.N()
for word in fd_bluebrd:
    fd_bluebrd[word] /= float(total_count)
fd_bluebrd.plot(20)
```

```
Quantitative details of each POS used in Bluebird
 NN  JJ VBD  RB NNS  VB VBG VBN VBP  CD  MD  IN VBZ  DT RBR  CC  WP JJS JJR
186 126  76  56  42  29  21  20  18  16  14  10   5   2   2   1   1   1   1

Plotting frequency distribution of POS in Bluebird
```




```
Plotting percentage of part of speech used in Bluebird
```



Summary

Task 2.b)

Table 1

	Percentage of top 6 part of speech used in 3 stories					
Story Name	NN (Noun, singular)	JJ (Adjective)	VBD (Verb, Past tense)	NNS (Noun, plural)	RB(Adverb)	VB(Verb, base form)
Aisle Six	34%	21%	12%	8%	6%	2%
Best wish	25%	16%	3%	10%	7%	6%
Bluebird	30%	20%	13%	7%	9%	5%

From above table (Table 1), it can be concluded that there are high use of Noun than any other POS in the stories which makes Aisle Six 34%, Best wish 25% and Bluebird with 30% in using Noun. And if we see the remaining 5 part of speech in the table, it kind of follow a pattern in terms of uses of the type of word. For example JJ(Adjective) uses is almost close in all of the stories around 20% except for Best wish with only 16%. This analysis also shows that it is not always consistent in terms of percentage uses of POS, as we can see VBD(Verb, past tense) is being uses very less in Best wish with only 3% which kind of states that this stories is not using sentence with past participle as compared to the other two stories such as Aisle Six and Bluebird with 12% and 13% respectively.

There is similarity in uses of the POS, it is because even though nouns and verbs are considered to be the most common word, we use high ratio of pronouns, verbs and adverbs while conversation and it doesn't follow the pattern in case of writing, here is is more of nouns and adjectives.

Task 2.c)

Writing often varies from one to another, it is just a way to use grammer to organize words into sentences and since there is not a specific rule to arrange the words and make sentences, it can be organised in various ways. According to Hudson, a good writing contains arounf 37% of word tokes as Nouns which kind of follows the pattern of part of speech in writing [1]. In general, a good writing uses combination of these POS nouns, verbs, adverbs and adjectives. For example, Table 1 summarizes the POS being used in three different stories and when we look into specifically nouns, adjectives and verbs, these are almost used in same ration as compared to other. Even popular database like WordNet uses only four part of speech which are generally found in writing.

References

[1] Hudson, R. (1994). About 37% of Word-Tokens are Nouns. Language, 70(2), 331-339. doi:10.2307/415831