Nombre: Alexis Leal Mata Materia: PLN y vision Computacional Enero 2023

In [33]:
```
import nltk
import matplotlib.pyplot as plt
import re
import unicodedata
from nltk.corpus import stopwords
```

In [34]:
```
nltk.download('stopwords')
```

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

Out[34]: True

In [35]:
```
nltk.download('movie_reviews')
```

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

Out[35]: True

In [36]:
```
stop_words = sorted(stopwords.words('english'))
stop_words
```

Out[36]:
```
['a',
 'about',
 'above',
 'after',
 'again',
 'against',
 'ain',
 'all',
 'am',
 'an',
 'and',
 'any',
 'are',
 'aren',
 "aren't",
 'as',
 'at',
 'be',
 'because',
```

In [37]:
```
#Vemos los id de los documentos que contiene movies_reviews
nltk.corpus.movie_reviews.fileids()
#A simple vista parece que solo tenemos neg
```

Out[37]:
```
['neg/cv000_29416.txt',
 'neg/cv001_19502.txt',
 'neg/cv002_17424.txt',
 'neg/cv003_12683.txt',
 'neg/cv004_12641.txt',
 'neg/cv005_29357.txt',
 'neg/cv006_17022.txt',
 'neg/cv007_4992.txt',
 'neg/cv008_29326.txt',
 'neg/cv009_29417.txt',
 'neg/cv010_29063.txt',
 'neg/cv011_13044.txt',
 'neg/cv012_29411.txt',
 'neg/cv013_10494.txt',
 'neg/cv014_15600.txt',
 'neg/cv015_29356.txt',
 'neg/cv016_4348.txt',
 'neg/cv017_23487.txt',
 'neg/cv018_21672.txt',
```

In [38]:
```
#Vemos que tenemos dos categorias positiva y negativa
nltk.corpus.movie_reviews.categories()
```

Out[38]: ['neg', 'pos']

In [39]:
```
#Obtenemos la frecuencia de las palabras
frecuenciaGeneral =  nltk.FreqDist(nltk.corpus.movie_reviews.words())
frecuenciaGeneral
```

Out[39]:
```
FreqDist({',': 77717, 'the': 76529, '.': 65876, 'a': 38106, 'and': 35576, 'of': 34123, 'to': 31937, "'": 30585, 'is': 25195, 'in': 21822, ...})
```

In [40]:
```
#Esta funcion elimina caracteres especiales y acentos
#Fue creada por CHATGP
def remove_special_characters_and_accents(text):
    text = re.sub(r'[^a-zA-Z\s]', '', text)
    text = unicodedata.normalize('NFKD', text).encode('ASCII', 'ignore').decode('ASCII')
    return text
```

In [41]:
```
categoriaPositiva = nltk.corpus.movie_reviews.words(categories='pos')
print(f"Longitud de categoria positiva antes de la limpieza {len(categoriaPositiva)}")
categoriaNegativas = nltk.corpus.movie_reviews.words(categories='neg')
print(f"Longitud de categoria negativa antes de la limpieza {len(categoriaNegativas)}")
```

```
Longitud de categoria positiva antes de la limpieza 832564
Longitud de categoria negativa antes de la limpieza 751256
```

Limpieza de datos de categoria positiva

In [56]:
```
def limpiezaPalabras(lista,stopWords):
    sinStopWords = [w for w in lista if not w.lower() in stopWords]
    sincaracteresEspeciales = [remove_special_characters_and_accents(w) for w in sinStopWords]
    listaPalabras = [w for w in sincaracteresEspeciales if w != ""]
    return  listaPalabras
```

In [106]:
```
#Codigo de prueba
# Test = ["Alexis","alexis","Leal"]
# len(nltk.FreqDist([w.lower() for w in Test]))
```

Out[106]: 2

In [50]:
```
frecuenciaPositiva = nltk.FreqDist([w.lower() for w in limpiezaPalabras(categoriaPositiva,stop_words)])
print(f"Longitud de categoria positiva despues de la limpieza {len(frecuenciaPositiva)}")
```

```
Longitud de categoria positiva despues de la limpieza 29750
```

In [55]:
```
frecuenciaNegativa = nltk.FreqDist([w.lower() for w in limpiezaPalabras(categoriaNegativas,stop_words)])
print(f"Longitud de categoria negativa despues de la limpieza {len(frecuenciaNegativa)}")
```

```
Longitud de categoria negativa despues de la limpieza 27841
```

In [109]:
```
print(f"Numero de caracteres categoria positva {frecuenciaPositiva.N()}")
print(f"Numero de caracteres catageria negativa {frecuenciaNegativa.N()}")
```

```
Numero de caracteres categoria positva 372286
Numero de caracteres catageria negativa 330613
```

In [61]:
```
top20Positivas =frecuenciaPositiva.most_common(20)
```

In [100]:
```
top20Negativas =frecuenciaNegativa.most_common(20)
```

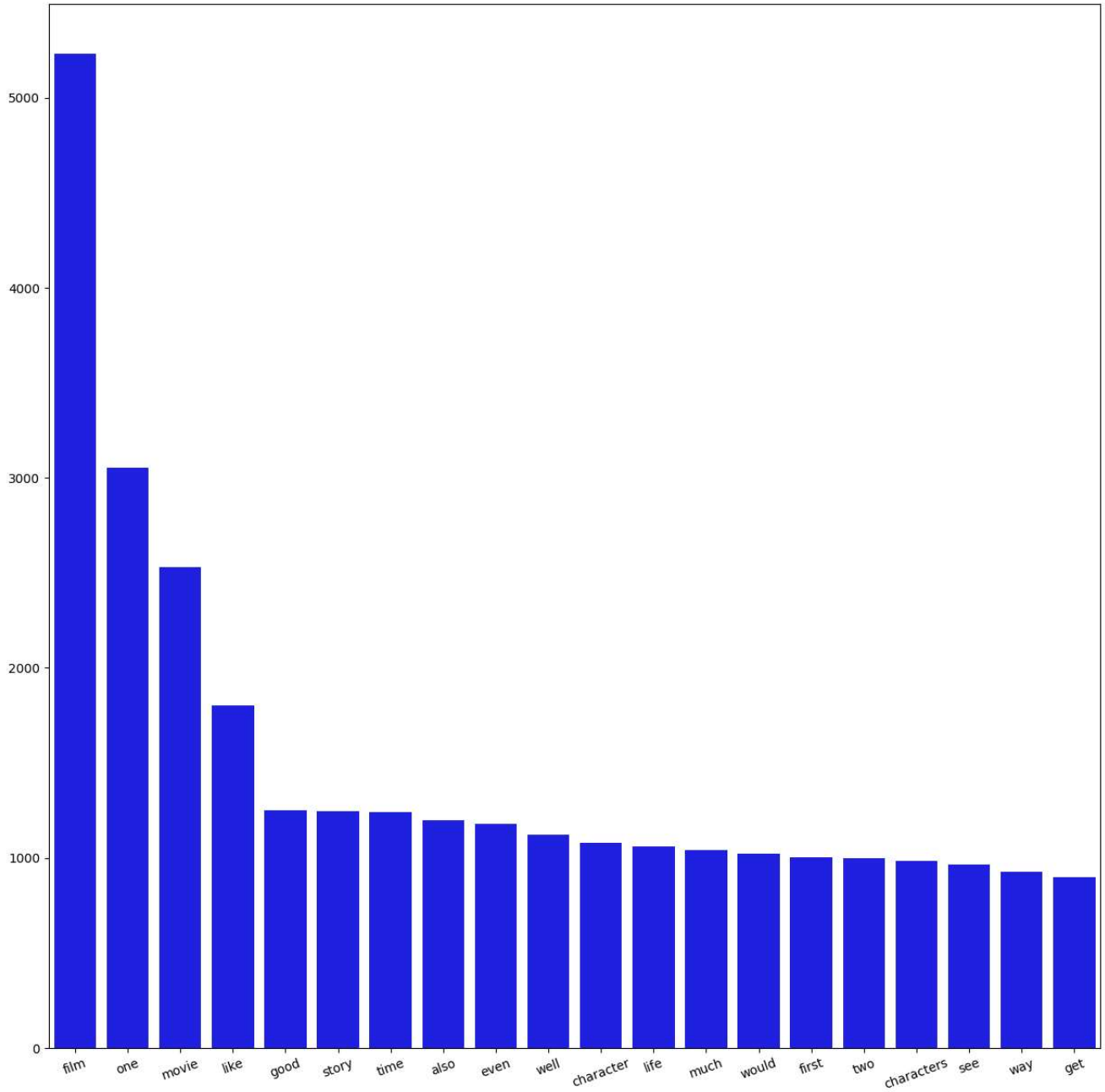
In [70]:
```
#Creamos un diccionario de las top 20 de la categoria por que este nos devuelve una tupla
dicPos = {}
for x,y in top20Positivas:
    dicPos[x] = y
```

In [101]:
```
dicNeg = {}
for x,y in top20Negativas:
    dicNeg[x] = y
```

In [93]:
```
import seaborn as sn
import  pandas as pd
```

Graficamos el top 20 de cada categoria

In [110]:
```
#Cremos una data frame con la libreria pandas
frec_dist = pd.Series(dicPos)
fig, ax = plt.subplots(figsize=(15,15))
grafica = sn.barplot(x=frec_dist.keys(), y=frec_dist.values, ax=ax,color='blue')
plt.xticks(rotation=20);
```



In [103]:
```
frec_distN = pd.Series(dicNeg)
fig, ax = plt.subplots(figsize=(15,15))
grafica = sn.barplot(x=frec_distN.keys(), y=frec_distN.values, ax=ax,color='blue')
plt.xticks(rotation=20);
```