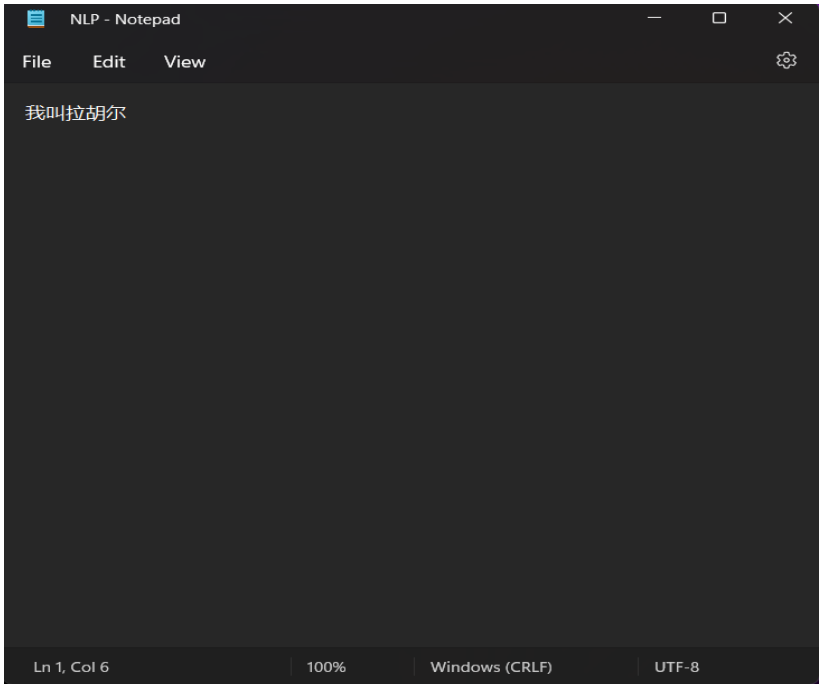
NATURAL LANGUAGE PROCESSESING

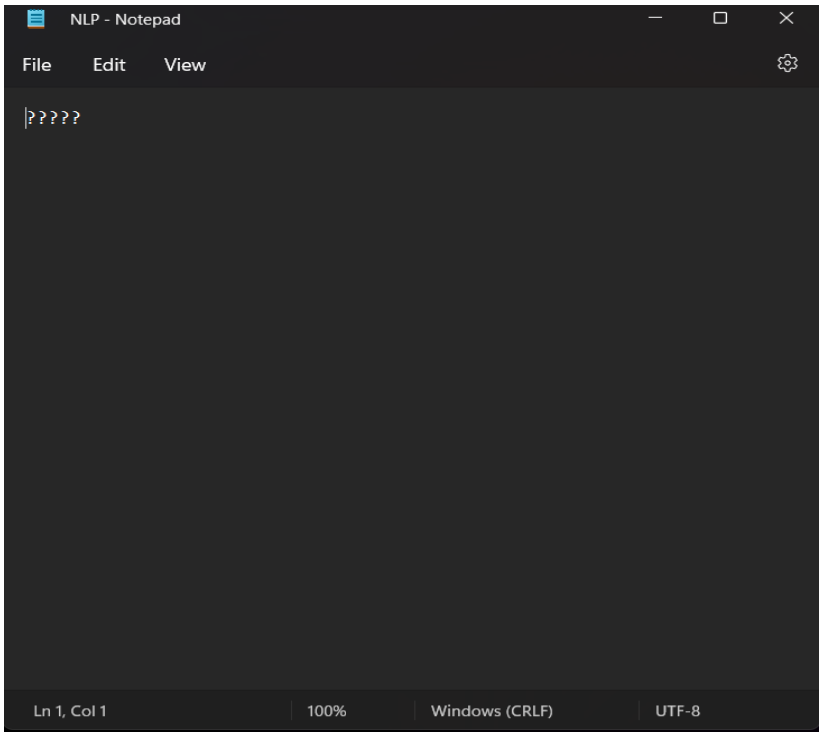
NAME: RAHUL GUDIVADA REG NO: 19BCE2469

DATE: 15/07/2022

Task 1:

Encoding:

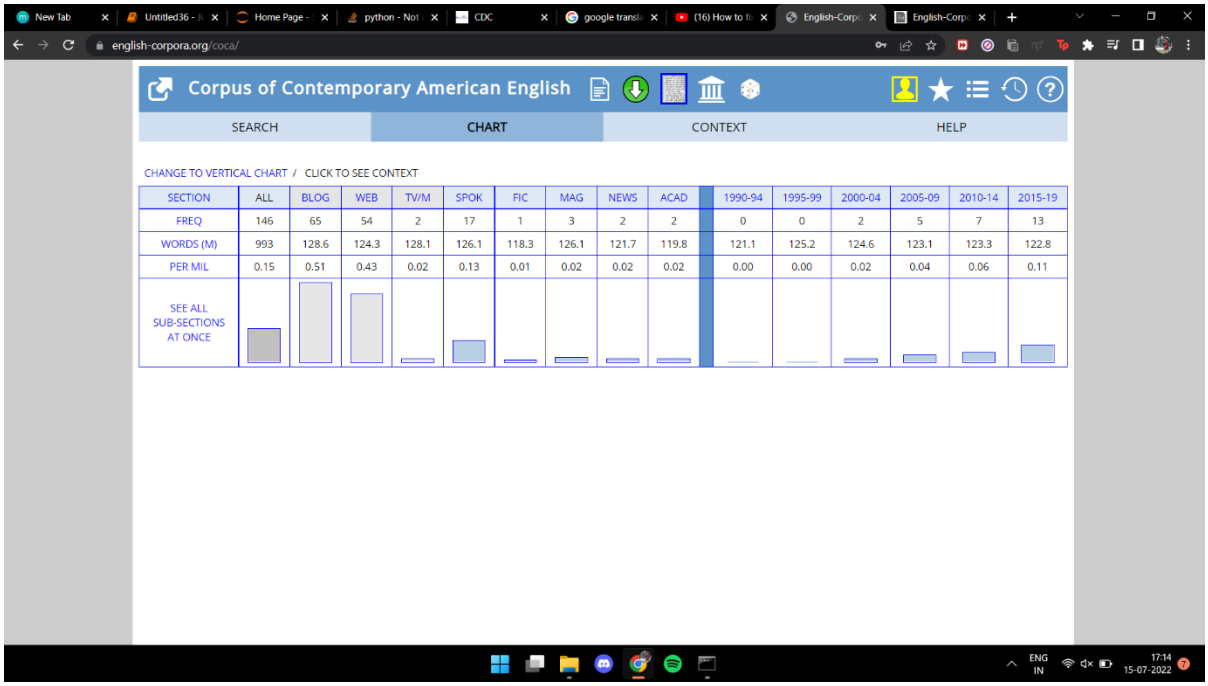




Task 2:

2.1

COCA, Story from Data



SECTION	ALL	BLOG	WEB	TV/M	SPOK	FIC	MAG	NEWS	ACAD	1990-94	1995-99	2000-04	2005-09	2010-14	2015-19
FREQ	146	65	54	2	17	1	3	2	2	0	0	2	5	7	13
WORDS (M)	993	128.6	124.3	128.1	126.1	118.3	126.1	121.7	119.8	121.1	125.2	124.6	123.1	123.3	122.8
PER MIL	0.15	0.51	0.43	0.02	0.13	0.01	0.02	0.02	0.02	0.00	0.00	0.02	0.04	0.06	0.11

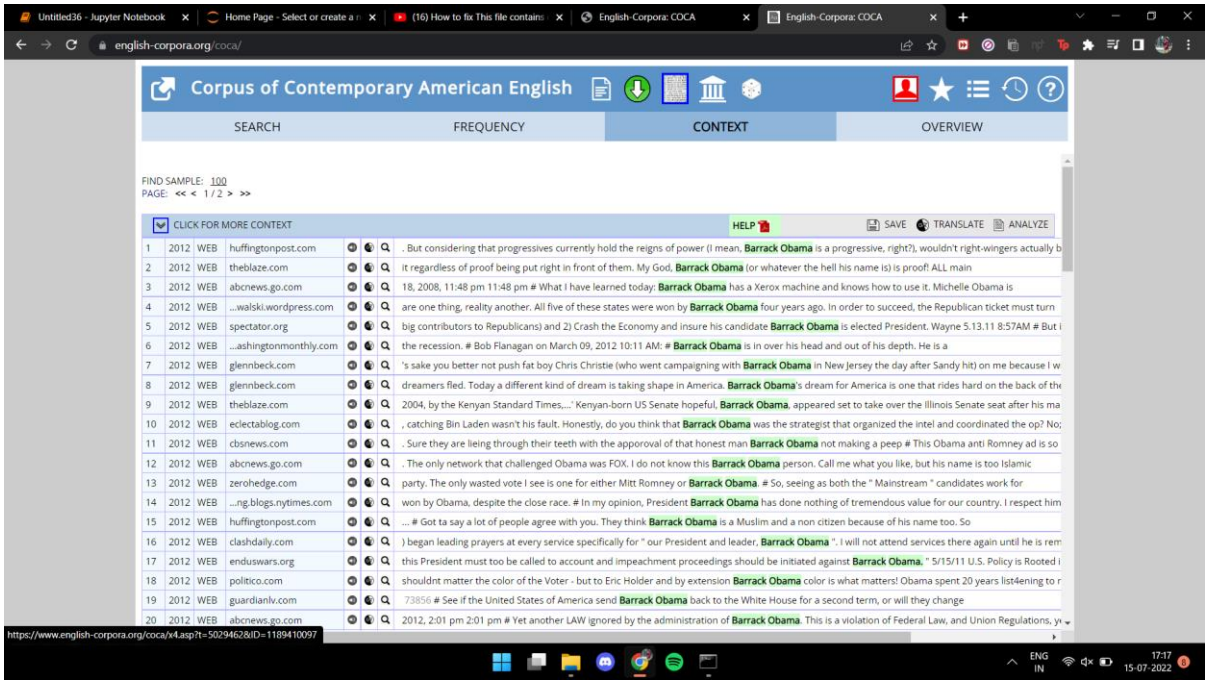
From the graph,we can clearly see that Barrack Obama wasn't famous during the time period of 1990-94 till 2000-2004. But slowly slowly he started gaining popularity as he started entering politics. From these graphs we can see that from 2005 onwards, people starting looking for him more in internet and how famous he became.

We can also see the frequency that is the frequency of him being in blogs is 65, whereas in WEB it is 54.

We can also see how many times his name as a WORD(M) is used in Blogs which is 65,wheres in WEB it is 54.

2.2

Concordance



Task 3:

Accessing books, brown corpus, inaugural corpus

jupyter NLP Task 3 and 4 Last Checkpoint: 3 hours ago (autosaved)

File Edit View Insert Cell Kernel Widgets Help

In [1]:
```
#Task 3: Accessing books,brown corpus,inaugral corpus.
```

In [2]:
```
import nltk
```

In [3]:
```
from nltk.book import *
```

```
*** Introductory Examples for the NLTK Book ***
Loading text1, ..., text9 and sent1, ..., sent9
Type the name of the text or sentence to view it.
Type: 'texts()' or 'sents()' to list the materials.
text1: Moby Dick by Herman Melville 1851
text2: Sense and Sensibility by Jane Austen 1811
text3: The Book of Genesis
text4: Inaugural Address Corpus
text5: Chat Corpus
text6: Monty Python and the Holy Grail
text7: Wall Street Journal
text8: Personals Corpus
text9: The Man Who Was Thursday by G . K . Chesterton 1908
```

In [4]:
```
from nltk.corpus import brown
brown.categories()
brown.words(categories='adventure')[:20]
```

Out[4]:
```
['Dan',
 'Morgan',
 'told',
 'himself',
 'he',
 'would',
 'forget',
 'Ann',
 'Turner',
 '.',
 'He',
 'was',
 'well',
 'rid',
 'of',
 'her',
 '.',
 'He',
 'certainly',
 "didn't"]
```

In [5]:
```
from nltk.corpus import inaugural
inaugural.fileids()
```

Out[5]:
```
['1789-Washington.txt',
 '1793-Washington.txt',
 '1797-Adams.txt',
 '1801-Jefferson.txt',
 '1805-Jefferson.txt',
 '1809-Madison.txt',
```

jupyter NLP Task 3 and 4 Last Checkpoint: 3 hours ago (autosaved)

File Edit View Insert Cell Kernel Widgets Help

```
 '1853-Pierce.txt',
 '1857-Buchanan.txt',
 '1861-Lincoln.txt',
 '1865-Lincoln.txt',
 '1869-Grant.txt',
 '1873-Grant.txt',
 '1877-Hayes.txt',
 '1881-Garfield.txt',
 '1885-Cleveland.txt',
 '1889-Harrison.txt',
 '1893-Cleveland.txt',
 '1897-McKinley.txt',
 '1901-McKinley.txt',
 '1905-Roosevelt.txt',
 '1909-Taft.txt',
 '1913-Wilson.txt',
 '1917-Wilson.txt',
 '1921-Harding.txt',
 '1925-Coolidge.txt',
 '1929-Hoover.txt',
 '1933-Roosevelt.txt',
 '1937-Roosevelt.txt',
 '1941-Roosevelt.txt',
 '1945-Roosevelt.txt',
 '1949-Truman.txt',
 '1953-Eisenhower.txt',
 '1957-Eisenhower.txt',
 '1961-Kennedy.txt',
 '1965-Johnson.txt',
 '1969-Nixon.txt',
 '1973-Nixon.txt',
 '1977-Carter.txt',
 '1981-Reagan.txt',
 '1985-Reagan.txt',
 '1989-Bush.txt',
 '1993-Clinton.txt',
 '1997-Clinton.txt',
 '2001-Bush.txt',
 '2005-Bush.txt',
 '2009-Obama.txt',
 '2013-Obama.txt',
 '2017-Trump.txt',
 '2021-Biden.txt']
```

In [6]:
```
inaugural.words(fileids='2009-Obama.txt')
#find for lincoln,trump
#compare first 5 words of lincoln and obama and then lincoln and trump
```

Out[6]:
```
['My', 'fellow', 'citizens', ':', 'I', 'stand', 'here', ...]
```

In [7]:
```
inaugural.words(fileids='1861-Lincoln.txt')
```

Out[7]:
```
['Fellow', '-', 'Citizens', 'of', 'the', 'United', ...]
```

In [8]:
```
inaugural.words(fileids='2017-Trump.txt')
```

Out[8]:
```
['Chief', 'Justice', 'Roberts', ',', 'President', ...]
```

Task 4:

Experiencing frequency distribution and conditional frequency distribution.

```
In [9]: #Task 4: Experiencing frequency distribution and conditional freuency distribution.

In [10]: text1='Thor: Love and Thunder is a 2022 American superhero film based on Marvel Comics featuring the character Thor, produced by
         fd=nltk.FreqDist(text1.split())
         fd

Out[10]: FreqDist({'the': 8, 'and': 7, 'is': 4, 'film': 3, 'Marvel': 3, 'by': 3, 'to': 3, 'Thor:': 2, 'Studios': 2, 'Waititi,': 2, ...})

In [ ]:

In [11]: from nltk.probability import ConditionalFreqDist
         cfd=ConditionalFreqDist((len(word),word)for word in text1.split())
         cfd[5]

Out[11]: FreqDist({'Thor:': 2, 'based': 1, 'Thor,': 1, 'Taika': 1, 'stars': 1, 'Chris': 1, 'Bale,': 1, 'Tessa': 1, 'film,': 1, 'inner':
         1})

In [ ]:
```

Task 5:

Suggest a Corpus Application

Corpus Approaches to Social-Media

The language of online communities, linguistic variety in brief media texts, and the use of images in computer-mediated communication are all examined from the perspective of corpora. The collection's in-depth descriptions of the methodological aspects of working with social media corpora are one of its strongest features. The collection includes research using novel and creative research approaches for the analysis of multimodal material and atypical corpus texts, as well as research using conventional corpus linguistic methods to social media data.

Even if social media has become increasingly popular as a news source, there are still concerns about how easily it may disseminate rumors and false information. It has been challenging to systematically examine this phenomenon, however, because it is necessary to gather extensive, objective data and make in-situ assessments of its accuracy. A corpus called CREDBANK was created to fill this gap by methodically fusing computer and human computing. CREDBANK is specifically a corpus of tweets, topics, events, and related human credibility assessments.