In [1]:
```
from bs4 import BeautifulSoup
from nltk.corpus import stopwords
import nltk#natural language Toolkit
from nltk.tokenize import word_tokenize
from nltk.stem import WordNetLemmatizer
from nltk.corpus import cmudict
from textblob import TextBlob
from nltk.sentiment import SentimentIntensityAnalyzer
import requests
import pandas as pd
import numpy as np
from readability import Readability
```

In [2]:
```
import pandas as pd
df=pd.read_csv("Downloads/a.csv")
df
```

Out[2]:

	URL_ID	URL	POSITIVE SCORE	NEGATIVE SCORE	POLARITY SCORE	SUBJECTIVITY SCORE	AVG SENTENCE LENGTH	PERCENTAGE OF COMPLEX WORDS	FOG INDEX	AVG NUMBER OF WORDS PER SENTENCE	...
0	37	https://insights.blackcoffer.com/ai-in-healthc...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...
1	38	https://insights.blackcoffer.com/what-if-the-c...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...

2 rows × 26 columns

In [3]:
```
df1=pd.read_csv("Downloads/i/Input.csv")
```

In [4]:
```
df1
```

Out[4]:

	URL_ID	URL
0	37	https://insights.blackcoffer.com/ai-in-healthc...
1	38	https://insights.blackcoffer.com/what-if-the-c...
2	39	https://insights.blackcoffer.com/what-jobs-wil...
3	40	https://insights.blackcoffer.com/will-machine-...
4	41	https://insights.blackcoffer.com/will-ai-repla...
...	...	...
109	146	https://insights.blackcoffer.com/blockchain-fo...
110	147	https://insights.blackcoffer.com/the-future-of...
111	148	https://insights.blackcoffer.com/big-data-anal...
112	149	https://insights.blackcoffer.com/business-anal...
113	150	https://insights.blackcoffer.com/challenges-an...

114 rows × 2 columns

In [5]:
```
df1["Positive Score"]=np.nan
df1["Negative Score"]=np.nan
df1["Polarity Score"]=np.nan
df1["Subjectivity Score"]=np.nan
df1["AVG SENTENCE LENGTH"]=np.nan
df1["PERCENTAGE OF COMPLEX WORDS"]=np.nan
df1["FOG INDEX"]=np.nan
df1["AVG NUMBER OF WORDS PER SENTENCE"]=np.nan
df1["COMPLEX WORD COUNT"]=np.nan
df1["SYLLABLE PER WORD"]=np.nan
df1["PERSONAL PRONOUNS"]=np.nan
df1["AVG WORD LENGTH"]=np.nan
```

In [6]:
```
df1
```

Out[6]:

	URL_ID	URL	Positive Score	Negative Score	Polarity Score	Subjectivity Score	AVG SENTENCE LENGTH	PERCENTAGE OF COMPLEX WORDS	FOG INDEX	AVG NUMBER OF WORDS PER SENTENCE	CO
0	37	https://insights.blackcoffer.com/ai-in-healthc...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	38	https://insights.blackcoffer.com/what-if-the-c...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	39	https://insights.blackcoffer.com/what-jobs-wil...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
3	40	https://insights.blackcoffer.com/will-machine-...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
4	41	https://insights.blackcoffer.com/will-ai-repla...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
...	...	...	...	...	...	...	...	...	...	...	
109	146	https://insights.blackcoffer.com/blockchain-fo...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
110	147	https://insights.blackcoffer.com/the-future-of...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
111	148	https://insights.blackcoffer.com/big-data-anal...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
112	149	https://insights.blackcoffer.com/business-anal...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
113	150	https://insights.blackcoffer.com/challenges-an...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	

114 rows × 14 columns

In [7]:
```
l=[]
for i in df1["URL"]:
    url = i
    response = requests.get(url)
    html_content = response.text
    soup = BeautifulSoup(html_content, 'html.parser')
    text = soup.get_text()
    l.append(str(text))
```

In [8]:
```
l
```

Out[8]:
```
['\n\n\n  \n\nAI in healthcare to Improve Patient Outcomes | Blackcoffer Insights\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n
\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\nSign in\n\n\n
\n\nOur Success Stories\n\nBanking, Financials, Securities, and Insurance\nEnergy\nEntertainment\nFast Moving Con
sumer Goods\nGovernment & Think Tanks\nHealthcare\nInfrastructure & Real Estate\nIT\nLifestyle, eCommerce & Onlin
e Market Place\nProduction & Manufacturing\nResearch & Academia\nRetail & Supply Chain\nTelecom\n\n\nWhat We Do\n
\nBanking, Financials, Securities, and Insurance\nEnergy\nEntertainment\nFast Moving Consumer Goods\nGovernment &
Think Tanks\nHealthcare\nHospitality\nInfrastructure & Real Estate\nIT Services\nLifestyle, eCommerce & Online Ma
rket Place\nNews & Media\nProduction & Manufacturing\nResearch & Academia\nRetail & Supply Chain\n\n\nWhat We Thi
nk\n\nAutomobiles & Components\nBFSI\nAsset and Portfolio\nBanks\nCapital Markets\nDerivatives and Securities\nDi
versified Financials\nFinance & Accounting\nInsurance\nSecurities and Capital Markets\nCapital Goods\nCommercial 
& Professional Services\nConsumer Discretionary\nConsumer Durables & Apparel\nConsumer Services\nConsumer Staples
\nFood & Staples Retailing\nFood, Beverage & Tobacco\nHousehold & Personal Products\nData Science\nAnalytics\nArt
ificial Intelligence\nBig Data\nBusiness Analytics\nData Visualization\nInternet of Things\nMachine Learning\nSta
tistics\nEnergy\nDataOil\n\n\nHow To\n\nAnalytics\nApplication Development\nArtificial Intelligence\nBusiness Ana
lytics\nExample\nOptimization\nProjects\nSoftware Development\nSource Code Audit\nStatistics\nWeb & Mobile App De
velopment\n\n\nSchedule Demo\nContact\n \n\n\n\n\n\n\n\nSign in\n\n\n\n\n\n\nWelcome!Log into your account\n\nyou
r username\nyour password\n\n\nForgot your password?\n\n\n\n\n\n\n\n\nPassword recovery\n\n\nRecover your passw
ord\n\nyour email\n\n\n\n\n \n\n\n\n\n\n\n\nSearch\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n \n\n\n \n\n\n  \n\n\n\n\n \n
\n\n\n\n\n\nSign in\nWelcome! Log into your account\n\n\nyour username\nyour password\n\n\nForgot your password?
```

In [9]:
```
for index,rows in df1.iterrows():
    url=i
    response=requests.get(url)
    html_content = response.text
    soup = BeautifulSoup(html_content, 'html.parser')
    text = soup.get_text()
    df1.loc[index, 'Text_Soup'] = text
df1
```

Out[9]:

	URL_ID	URL	Positive Score	Negative Score	Polarity Score	Subjectivity Score	AVG SENTENCE LENGTH	PERCENTAGE OF COMPLEX WORDS	FOG INDEX	AVG NUMBER OF WORDS PER SENTENCE	CO
0	37	https://insights.blackcoffer.com/ai-in-healthc...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	38	https://insights.blackcoffer.com/what-if-the-c...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	39	https://insights.blackcoffer.com/what-jobs-wil...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
3	40	https://insights.blackcoffer.com/will-machine-...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
4	41	https://insights.blackcoffer.com/will-ai-repla...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
...	...	...	...	...	...	...	...	...	...	...	
109	146	https://insights.blackcoffer.com/blockchain-fo...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
110	147	https://insights.blackcoffer.com/the-future-of...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
111	148	https://insights.blackcoffer.com/big-data-anal...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
112	149	https://insights.blackcoffer.com/business-anal...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
113	150	https://insights.blackcoffer.com/challenges-an...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	

114 rows × 15 columns

In [11]:
```
lemma = WordNetLemmatizer()
stop_words = stopwords.words('english')
def text_prep(x):
     corp = str(x).lower()
     corp = re.sub('[^a-zA-Z]+',' ', corp).strip()
     tokens = word_tokenize(corp)
     words = [t for t in tokens if t not in stop_words]
     lemmatize = [lemma.lemmatize(w) for w in words]

     return lemmatize
```

In [12]:
```
def get_subjectivity_score(text):
    blob = TextBlob(text)
    return blob.sentiment.subjectivity
```

In [13]:
```
def get_avg_sentence_length(text):
    sentences = nltk.sent_tokenize(text)
    total_words = sum([len(nltk.word_tokenize(sentence)) for sentence in sentences])
    return total_words / len(sentences)
```

In [14]:
```
def get_percentage_complex_words(text):
    words = nltk.word_tokenize(text)
    total_words = len(words)
    complex_words = [word for word in words if len(word) > 2 and nltk.pos_tag([word])[0][1] in ['JJ', 'JJR', 'JJS', 'F
    num_complex_words = len(complex_words)
    return (num_complex_words / total_words) * 100
```

In [15]:
```
def fog_index(text):
    sentences = re.split('(?<!\w\.\w.)(?<![A-Z][a-z]\.)(?<=\.|\?)\s', text)
    total_words = 0
    polysyllabic_words = 0
    for sentence in sentences:
        words = sentence.split()
        total_words += len(words)
        polysyllabic_words += sum(1 for word in words if len(re.findall(r'[aeiouy]{2,}', word.lower())) >= 2)
    avg_sentence_length = total_words / len(sentences)
    polysyllabic_percentage = (polysyllabic_words / total_words) * 100
    fog_index = 0.4 * (avg_sentence_length + polysyllabic_percentage)
    return fog_index
```

In [16]:
```
def get_avg_words_per_sentence(text):
    sentences = nltk.sent_tokenize(text)
    total_words = sum([len(nltk.word_tokenize(sentence)) for sentence in sentences])
    return total_words / len(sentences)
```

In [17]:
```
def get_complex_word_count(text):
    words = nltk.word_tokenize(text)
    complex_words = [word for word in words if len(word) > 2 and nltk.pos_tag([word])[0][1] in ['JJ', 'JJR', 'JJS', 'F
    return len(complex_words)
```

In [18]:
```
def get_word_count(text):
    words = nltk.word_tokenize(text)
    return len(words)
```

In [19]:
```
def get_avg_syllables_per_word(text):
    words = nltk.word_tokenize(text)
    d = cmudict.dict()
    total_syllables = 0
    for word in words:
        if word.lower() in d:
            total_syllables += len(list(y for y in d[word.lower()][0] if y[-1].isdigit()))
    return total_syllables / len(words)
```

In [20]:
```
def get_personal_pronoun_count(text):
    words = nltk.word_tokenize(text)
    personal_pronouns = [word for word in words if word.lower() in ['i', 'me', 'my', 'mine', 'we', 'us', 'our', 'ours'
    return len(personal_pronouns)
```

In [21]:
```
def get_avg_word_length(text):
    words = nltk.word_tokenize(text)
    total_length = sum([len(word) for word in words])
    return total_length / len(words)
```

In [22]:
```
analyzer = SentimentIntensityAnalyzer()
positive_words = open("Downloads/i/positive-words.txt").read().splitlines()
negative_words = open("Downloads/i/negative-words.txt").read().splitlines()

opinion_lexicon = {}
for word in positive_words:
    opinion_lexicon[word] = 1.0

for word in negative_words:
    opinion_lexicon[word] = -1.0

analyzer.lexicon.update(opinion_lexicon)
```

In [25]:
```
import re
for index, row in df1.iterrows():
    scores = analyzer.polarity_scores(row['Text_Soup'])
    df1.loc[index, 'POSITIVE SCORE'] = scores['pos']
    df1.loc[index, 'NEGATIVE SCORE'] = scores['neg']
    df1.loc[index, 'POLARITY SCORE'] = scores['compound']
    df1.loc[index, 'SUBJECTIVITY SCORE'] = get_subjectivity_score(row['Text_Soup'])
    df1.loc[index, 'AVG SENTENCE LENGTH'] = get_avg_sentence_length(row['Text_Soup'])
    df1.loc[index, 'PERCENTAGE OF COMPLEX WORDS'] = get_percentage_complex_words(row['Text_Soup'])
    df1.loc[index, 'FOG INDEX'] = fog_index(row['Text_Soup'])
    df1.loc[index, 'AVG NUMBER OF WORDS PER SENTENCE'] = get_avg_words_per_sentence(row['Text_Soup'])
    df1.loc[index, 'COMPLEX WORD COUNT'] = get_complex_word_count(row['Text_Soup'])
    df1.loc[index, 'WORD COUNT'] = get_word_count(row['Text_Soup'])
    df1.loc[index, 'SYLLABLE PER WORD'] = get_avg_syllables_per_word(row['Text_Soup'])
    df1.loc[index, 'PERSONAL PRONOUNS'] = get_personal_pronoun_count(row['Text_Soup'])
    df1.loc[index, 'AVG WORD LENGTH'] = get_avg_word_length(row['Text_Soup'])
df1
```

Out[25]:

	URL_ID	URL	Positive Score	Negative Score	Polarity Score	Subjectivity Score	AVG SENTENCE LENGTH	PERCENTAGE OF COMPLEX WORDS	FOG INDEX	AVG NUMBER OF WORDS PER SENTENCE
0	37	https://insights.blackcoffer.com/ai-in-healthc...	NaN	NaN	NaN	NaN	28.539474	8.852006	10.451631	28.539474
1	38	https://insights.blackcoffer.com/what-if-the-c...	NaN	NaN	NaN	NaN	28.539474	8.852006	10.451631	28.539474
2	39	https://insights.blackcoffer.com/what-jobs-wil...	NaN	NaN	NaN	NaN	28.539474	8.852006	10.451631	28.539474
3	40	https://insights.blackcoffer.com/will-machine-...	NaN	NaN	NaN	NaN	28.539474	8.852006	10.451631	28.539474
4	41	https://insights.blackcoffer.com/will-ai-repla...	NaN	NaN	NaN	NaN	28.539474	8.852006	10.451631	28.539474
...	...	...	...	...	...	...	...	...	...	...
109	146	https://insights.blackcoffer.com/blockchain-fo...	NaN	NaN	NaN	NaN	29.108108	8.867224	10.923151	29.108108
110	147	https://insights.blackcoffer.com/the-future-of...	NaN	NaN	NaN	NaN	29.108108	8.867224	10.923151	29.108108
111	148	https://insights.blackcoffer.com/big-data-anal...	NaN	NaN	NaN	NaN	29.108108	8.867224	10.923151	29.108108
112	149	https://insights.blackcoffer.com/business-anal...	NaN	NaN	NaN	NaN	29.108108	8.867224	10.923151	29.108108
113	150	https://insights.blackcoffer.com/challenges-an...	NaN	NaN	NaN	NaN	29.108108	8.867224	10.923151	29.108108

114 rows × 20 columns

In [26]:
```
df1.drop(['Positive Score','Negative Score','Polarity Score','Subjectivity Score'],axis=1,inplace=True)
```

In [27]:
```
df1
```

Out[27]:

	URL_ID	URL	AVG SENTENCE LENGTH	PERCENTAGE OF COMPLEX WORDS	FOG INDEX	AVG NUMBER OF WORDS PER SENTENCE	COMPLEX WORD COUNT	SYLLABLE PER WORD	PERSONAL PRONOUNS	W LEN
0	37	https://insights.blackcoffer.com/ai-in-healthc...	28.539474	8.852006	10.451631	28.539474	192.0	1.447672	42.0	5.118
1	38	https://insights.blackcoffer.com/what-if-the-c...	28.539474	8.852006	10.451631	28.539474	192.0	1.447672	42.0	5.118
2	39	https://insights.blackcoffer.com/what-jobs-wil...	28.539474	8.852006	10.451631	28.539474	192.0	1.447672	42.0	5.118
3	40	https://insights.blackcoffer.com/will-machine-...	28.539474	8.852006	10.451631	28.539474	192.0	1.447672	42.0	5.118
4	41	https://insights.blackcoffer.com/will-ai-repla...	28.539474	8.852006	10.451631	28.539474	192.0	1.447672	42.0	5.118
...	...	...	...	...	...	...	...	...	...	
109	146	https://insights.blackcoffer.com/blockchain-fo...	29.108108	8.867224	10.923151	29.108108	191.0	1.448932	42.0	5.112
110	147	https://insights.blackcoffer.com/the-future-of...	29.108108	8.867224	10.923151	29.108108	191.0	1.448932	42.0	5.112
111	148	https://insights.blackcoffer.com/big-data-anal...	29.108108	8.867224	10.923151	29.108108	191.0	1.448932	42.0	5.112
112	149	https://insights.blackcoffer.com/business-anal...	29.108108	8.867224	10.923151	29.108108	191.0	1.448932	42.0	5.112
113	150	https://insights.blackcoffer.com/challenges-an...	29.108108	8.867224	10.923151	29.108108	191.0	1.448932	42.0	5.112

114 rows × 16 columns

In [28]:
```
df1.describe()
```

Out[28]:

	URL_ID	AVG SENTENCE LENGTH	PERCENTAGE OF COMPLEX WORDS	FOG INDEX	AVG NUMBER OF WORDS PER SENTENCE	COMPLEX WORD COUNT	SYLLABLE PER WORD	PERSONAL PRONOUNS	AVG WORD LENGTH	POSITIVE SCORE	NEGATIVE SCORE
count	114.000000	114.000000	114.000000	114.000000	114.000000	114.000000	114.000000	114.0	114.000000	114.000000	114.000000
mean	93.500000	28.848731	8.860282	10.708072	28.848731	191.456140	1.448357	42.0	5.115191	0.168456	0.058912
std	33.052988	0.284472	0.007613	0.235888	0.284472	0.500272	0.000631	0.0	0.002608	0.000500	0.001001
min	37.000000	28.539474	8.852006	10.451631	28.539474	191.000000	1.447672	42.0	5.112813	0.168000	0.058000
25%	65.250000	28.539474	8.852006	10.451631	28.539474	191.000000	1.447672	42.0	5.112813	0.168000	0.058000
50%	93.500000	29.108108	8.867224	10.923151	29.108108	191.000000	1.448932	42.0	5.112813	0.168000	0.058000
75%	121.750000	29.108108	8.867224	10.923151	29.108108	192.000000	1.448932	42.0	5.118027	0.169000	0.060000
max	150.000000	29.108108	8.867224	10.923151	29.108108	192.000000	1.448932	42.0	5.118027	0.169000	0.060000