Machine Learning 8 Assignment

Problem Statement In this assignment students have to find the frequency of words in a webpage. User can use urllib and BeautifulSoup to extract text from webpage.

In [1]:

```
from bs4 import BeautifulSoup
import urllib.request
import nltk
response = urllib.request.urlopen('http://php.net/')
html = response.read()
soup = BeautifulSoup(html,"html5lib")
```

In [2]:

```
sentences = soup.get_text(strip=True)
print (sentences)
```

```
PHP: Hypertext PreprocessorDownloadsDocumentationGet InvolvedHelpGetting S
tartedIntroductionA simple tutorialLanguage ReferenceBasic syntaxTypesVari
ablesConstantsExpressionsOperatorsControl StructuresFunctionsClasses and O
bjectsNamespacesErrorsExceptionsGeneratorsReferences ExplainedPredefined V
ariablesPredefined ExceptionsPredefined Interfaces and ClassesContext opti
ons and parametersSupported Protocols and WrappersSecurityIntroductionGene
ral considerationsInstalled as CGI binaryInstalled as an Apache moduleSess
ion SecurityFilesystem SecurityDatabase SecurityError ReportingUsing Regis
ter GlobalsUser Submitted DataMagic QuotesHiding PHPKeeping CurrentFeature
sHTTP authentication with PHPCookiesSessionsDealing with XFormsHandling fi
le uploadsUsing remote filesConnection handlingPersistent Database Connect
ionsSafe ModeCommand line usageGarbage CollectionDTrace Dynamic TracingFun
ction ReferenceAffecting PHP's BehaviourAudio Formats ManipulationAuthenti
cation ServicesCommand Line Specific ExtensionsCompression and Archive Ext
ensionsCredit Card ProcessingCryptography ExtensionsDatabase ExtensionsDat
e and Time Related ExtensionsFile System Related ExtensionsHuman Language
and Character Encoding SupportImage Processing and GenerationMail Related
ExtensionsMathematical ExtensionsNon-Text MIME OutputProcess Control Exten
sionsOther Basic ExtensionsOther ServicesSearch Engine ExtensionsServer Sp
ecific ExtensionsSession ExtensionsText ProcessingVariable and Type Relate
```

Tokenizing

In [3]:

```
words = [i for i in sentences.split()]
```

In [4]:

```
len(words)
```

Out[4]:

```
2981
```

Word Frequency Counting

In [5]:

```
wordfreq = nltk.FreqDist(words)
for key,val in wordfreq.items():

    print (str(key) + ':' + str(val))
```

```
PHP::1
Hypertext:1
PreprocessorDownloadsDocumentationGet:1
InvolvedHelpGetting:1
StartedIntroductionA:1
simple:1
tutorialLanguage:1
ReferenceBasic:1
syntaxTypesVariablesConstantsExpressionsOperatorsControl:1
StructuresFunctionsClasses:1
and:74
ObjectsNamespacesErrorsExceptionsGeneratorsReferences:1
ExplainedPredefined:1
VariablesPredefined:1
ExceptionsPredefined:1
Interfaces:1
ClassesContext:1
options:1
parametersSupported:1
```

Frequency Distibution Plot

In [6]:

```
wordfreq.plot(30, cumulative=False)
```

```
<Figure size 640x480 with 1 Axes>
```

In [7]:

```
from nltk.corpus import stopwords
import string
words = [i for i in sentences.split() if (i not in stopwords.words('english')) & (i not in
```

In [8]:

```
len(words)
```

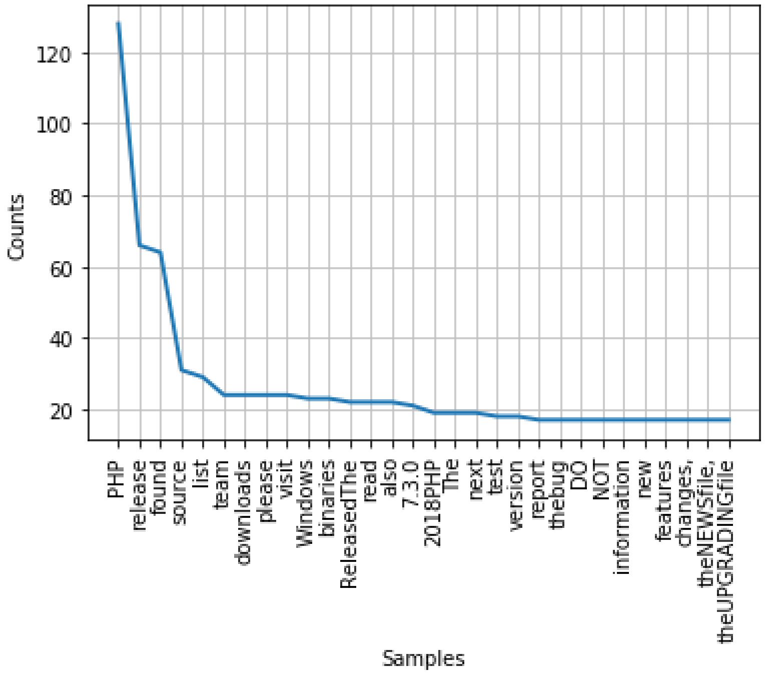
Out[8]:

```
2121
```

Frequency Distibution Plot for the most commonly ocuuring 30 words

In [9]:

```
wordfreq = nltk.FreqDist(words)
wordfreq.plot(30, cumulative=False)
```



Note

Stopwords has been removed using nltk.corpus.stopwords library.

Tokenizing using NLTK

In [10]:

```
sentences = nltk.sent_tokenize(sentences)
words = []
for i in range(len(sentences)):
    word = nltk.word_tokenize(sentences[i])
    for j in word:
        if j not in string.punctuation:#remove punctuations as a part of being considered a
            words.append(j)
```

In [11]:

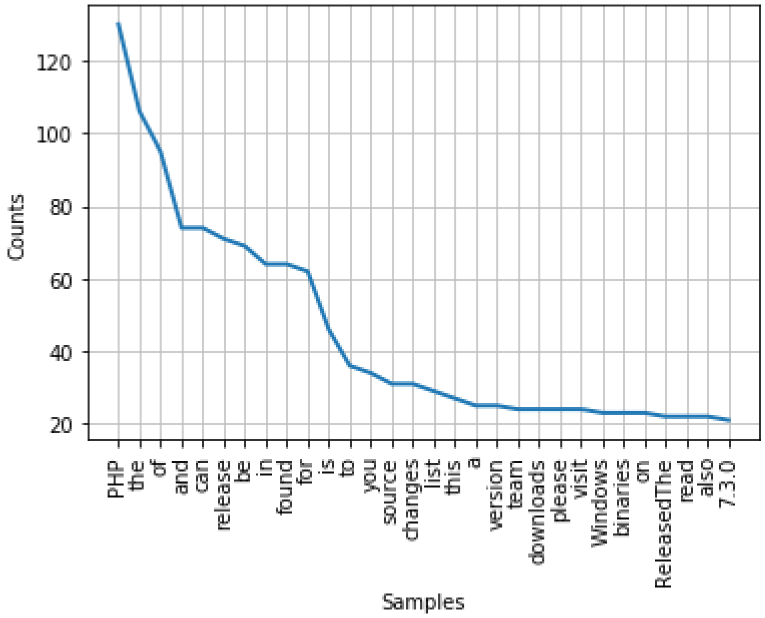
```
len(words)
```

Out[11]:

```
2987
```

In [12]:

```
freq = nltk.FreqDist(words)
freq.plot(30,cumulative=False)
```



Genearting tokens without stopwords

In [13]:

```
words_no_stopwords =[]
for i in range(len(sentences)):
    word = nltk.word_tokenize(sentences[i])

    for j in word:

        if (j not in stopwords.words('english'))  & (j not in string.punctuation):

            #print(j)
            words_no_stopwords.append(j)
```

In [14]:

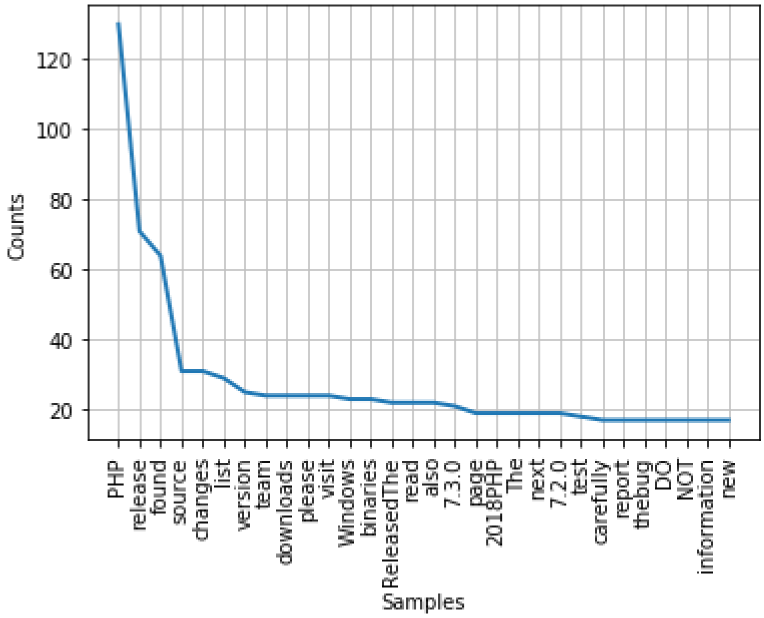
```
len(words_no_stopwords)
```

Out[14]:

```
2144
```

In [15]:

```
freq = nltk.FreqDist(words_no_stopwords)
freq.plot(30,cumulative=False)
```



In [ ]:

```
```