In [1]:

```
# install and import Numpy and pandas dependecy

import numpy as np
import pandas as pd
```

In [2]:

```
# access ".csv" file(this file store on same project folder) and assin in "df" by

df = pd.read_csv('sms_spam.csv')
```

In [3]:

```
print(df.columns)
```

```
Index(['type', 'text'], dtype='object')
```

In [4]:

```
# try to test open same row and colums from the .csv file

df.sample(10)
```

Out[4]:

	type	text
3146	ham	I.ll get there tomorrow and send it to you
1989	ham	Sorry, I'll call later
3325	ham	I don wake since. I checked that stuff and saw...
1874	spam	You have WON a guaranteed £1000 cash or a £200...
1403	ham	You have registered Sinco as Payee. Log in at ...
1843	ham	"Are you comingdown later?"
3821	ham	I got arrested for possession at, I shit you n...
5006	ham	Guess which pub im in? Im as happy as a pig in...
2555	ham	I'll reach in ard 20 mins ok...
4103	ham	Ok then i will come to ur home after half an hour

In [5]:

```
# Rows and Colums find

df.shape
```

Out[5]:

```
(5574, 2)
```

In [6]:

```
##########=============================================#########
```

output insite the current .csv file [11] the 5574(Rows) and 2(Colums)

1. Data Cleaning

2. EDA (Exploratory data analysis)

3. Text Preprocessing

4. Model building

5. Evalution

6. Improvement

7. Deploy

In [7]:

```
##########=============================================#########
```

1. Data Cleaning

In [8]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5574 entries, 0 to 5573
Data columns (total 2 columns):
 #   Column  Non-Null Count  Dtype 
---  ------  --------------  ----- 
 0   type    5574 non-null   object
 1   text    5574 non-null   object
dtypes: object(2)
memory usage: 87.2+ KB
```

In [9]:

```
# rename the cloums

df.rename(columns={'type':'target'},inplace=True)
df.sample(10)
```

Out[9]:

	target	text
5459	ham	Arun can u transfr me d amt
441	ham	Yes..he is really great..bhaji told kallis bes...
983	spam	Congrats! 2 mobile 3G Videophones R yours. cal...
3125	ham	My uncles in Atlanta. Wish you guys a great se...
5071	spam	5p 4 alfie Moon's Children in need song on ur ...
2517	ham	Yes.i'm in office da:)
1021	ham	Good afternoon on this glorious anniversary da...
4396	ham	Only just got this message, not ignoring you. ...
4042	spam	Please call our customer service representativ...
3897	spam	tells u 2 call 09066358152 to claim £5000 priz...

In [10]:

```
# cloumn target inside row name change name formate like ham->0 and spam-> 1
# for parpose of easy understaning

from sklearn.preprocessing import LabelEncoder
encoder = LabelEncoder()
```

In [11]:

```
# that error means not install "sklearn.preprocessing"

!pip install scikit-learn
```

```
Requirement already satisfied: scikit-learn in /home/cdac/.local/li
b/python3.8/site-packages (1.2.2)
Requirement already satisfied: scipy>=1.3.2 in /home/cdac/.local/li
b/python3.8/site-packages (from scikit-learn) (1.10.1)
Requirement already satisfied: joblib>=1.1.1 in /home/cdac/.local/li
b/python3.8/site-packages (from scikit-learn) (1.2.0)
Requirement already satisfied: numpy>=1.17.3 in /home/cdac/.local/li
b/python3.8/site-packages (from scikit-learn) (1.24.3)
Requirement already satisfied: threadpoolctl>=2.0.0 in /home/cdac/.l
ocal/lib/python3.8/site-packages (from scikit-learn) (3.1.0)
```

In [12]:

```
from sklearn.preprocessing import LabelEncoder
encoder = LabelEncoder()
```

In [13]:

```
encoder.fit_transform(df['target'])
```

Out[13]:

```
array([0, 0, 1, ..., 0, 0, 0])
```

In [14]:

```
df.sample(10)
```

Out[14]:

	target	text
3513	ham	Already one guy loving you:-.
1542	ham	Do u konw waht is rael FRIENDSHIP Im gving yuo...
3439	ham	Its good to hear from you
783	ham	Beerage?
1969	ham	2 laptop... I noe infra but too slow lar... I ...
2041	ham	You always make things bigger than they are
878	spam	Sunshine Quiz Wkly Q! Win a top Sony DVD playe...
3203	ham	Okay lor... Wah... like that def they wont let...
3145	ham	Haha I heard that, text me when you're around
1392	ham	Haha just kidding, papa needs drugs

In [15]:

```
# ham-> 0
# spam-> 1

df['target'] = encoder.fit_transform(df['target'])
```

In [16]:

```
df.head()
```

Out[16]:

	target	text
0	0	Go until jurong point, crazy.. Available only ...
1	0	Ok lar... Joking wif u oni...
2	1	Free entry in 2 a wkly comp to win FA Cup fina...
3	0	U dun say so early hor... U c already then say...
4	0	Nah I don't think he goes to usf, he lives aro...

In [17]:

```
# check "missing" value parsent or not
df.isnull().sum()
```

Out[17]:

```
target    0
text      0
dtype: int64
```

In [18]:

```
# check "duplicate" value parsent or not
df.duplicated().sum()
```

Out[18]:

```
414
```

In [19]:

```
# then "remove" duplicate value
df = df.drop_duplicates(keep='first')
```

In [20]:

```
# Now, recheck "duplicate" value parsent or not
df.duplicated().sum()
```

Out[20]:

```
0
```

In [21]:

```
# Review Rows and Colums parsent now
df.shape
```

Out[21]:

```
(5160, 2)
```

2. EDA (Exploratory data analysis)

In [22]:

```
print(df.columns)
```

```
Index(['target', 'text'], dtype='object')
```

In [23]:

```
# view parsent table

df.head()
```

Out[23]:

	target	text
0	0	Go until jurong point, crazy.. Available only ...
1	0	Ok lar... Joking wif u oni...
2	1	Free entry in 2 a wkly comp to win FA Cup fina...
3	0	U dun say so early hor... U c already then say...
4	0	Nah I don't think he goes to usf, he lives aro...

In [24]:

```
# filter or count parsent total number of "ham -> 0" and "spam -> 1"

df['target'].value_counts()
```

Out[24]:

```
target
0    4518
1     642
Name: count, dtype: int64
```

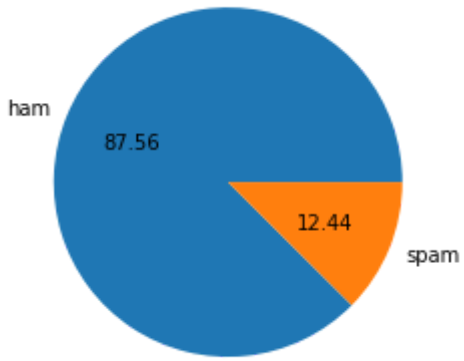
In [25]:

```
# then that data show on "Pie Chart format"

import matplotlib.pyplot as plt
plt.pie(df['target'].value_counts(), labels=['ham','spam'],autopct="%0.2f")
```

Out[25]:

```
([<matplotlib.patches.Wedge at 0x7f7a52ad9e20>,
  <matplotlib.patches.Wedge at 0x7f7a52ad9d00>],
 [Text(-1.0170346463201791, 0.4190948916228736, 'ham'),
  Text(1.0170346267009303, -0.4190949392337011, 'spam')],
 [Text(-0.5547461707200977, 0.22859721361247648, '87.56'),
  Text(0.5547461600186891, -0.22859723958201877, '12.44')])
```



In [26]:

```
# that errors means not install "matplotlib"

!pip install matplotlib
```

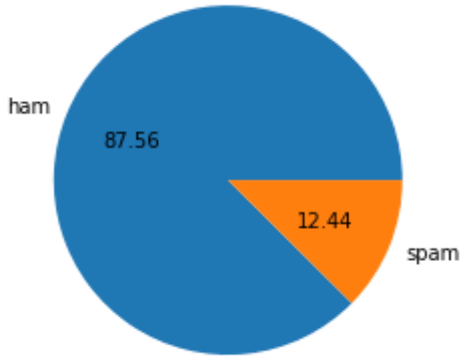
```
Requirement already satisfied: matplotlib in /home/cdac/.local/lib/p
ython3.8/site-packages (3.7.1)
Requirement already satisfied: pyparsing>=2.3.1 in /home/cdac/.loca
l/lib/python3.8/site-packages (from matplotlib) (3.0.9)
Requirement already satisfied: pillow>=6.2.0 in /usr/lib/python3/dis
t-packages (from matplotlib) (7.0.0)
Requirement already satisfied: python-dateutil>=2.7 in /usr/local/li
b/python3.8/dist-packages (from matplotlib) (2.8.2)
Requirement already satisfied: numpy>=1.20 in /home/cdac/.local/lib/
python3.8/site-packages (from matplotlib) (1.24.3)
Requirement already satisfied: fonttools>=4.22.0 in /home/cdac/.loca
l/lib/python3.8/site-packages (from matplotlib) (4.39.4)
Requirement already satisfied: importlib-resources>=3.2.0; python_ve
rsion < "3.10" in /home/cdac/.local/lib/python3.8/site-packages (fro
m matplotlib) (5.12.0)
Requirement already satisfied: kiwisolver>=1.0.1 in /home/cdac/.loca
l/lib/python3.8/site-packages (from matplotlib) (1.4.4)
Requirement already satisfied: cycler>=0.10 in /home/cdac/.local/li
b/python3.8/site-packages (from matplotlib) (0.11.0)
Requirement already satisfied: packaging>=20.0 in /home/cdac/.local/
lib/python3.8/site-packages (from matplotlib) (23.1)
Requirement already satisfied: contourpy>=1.0.1 in /home/cdac/.loca
l/lib/python3.8/site-packages (from matplotlib) (1.0.7)
Requirement already satisfied: six>=1.5 in /usr/lib/python3/dist-pac
kages (from python-dateutil>=2.7->matplotlib) (1.14.0)
Requirement already satisfied: zipp>=3.1.0; python_version < "3.10"
in /home/cdac/.local/lib/python3.8/site-packages (from importlib-res
ources>=3.2.0; python_version < "3.10"->matplotlib) (3.15.0)
```

In [27]:

```
import matplotlib.pyplot as plt
plt.pie(df['target'].value_counts(), labels=['ham','spam'],autopct="%0.2f")
```

Out[27]:

```
([<matplotlib.patches.Wedge at 0x7f7a52a39700>,
  <matplotlib.patches.Wedge at 0x7f7a52a39610>],
 [Text(-1.0170346463201791, 0.4190948916228736, 'ham'),
  Text(1.0170346267009303, -0.4190949392337011, 'spam')],
 [Text(-0.5547461707200977, 0.22859721361247648, '87.56'),
  Text(0.5547461600186891, -0.22859723958201877, '12.44')])
```

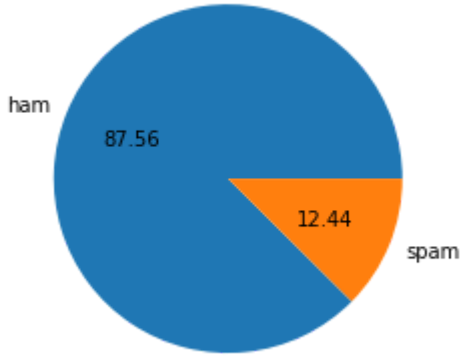


In [28]:

```
# remove extra code top of the "Pic Chart"
# by the using command this
# "plt.show()"
```

In [29]:

```
import matplotlib.pyplot as plt
plt.pie(df['target'].value_counts(), labels=['ham','spam'],autopct="%0.2f")
plt.show()
```



In [30]:

```
print(df.columns)
```

```
Index(['target', 'text'], dtype='object')
```

2.1 Data is imbalance So, Blance it

In [31]:

```
# get information from this Pic chart
#I see data "ham" and "spam" are not blanced
# sms ke ander kitne "No. of Alphabet, No. of Words, No of Santance" etc.
# use ho raha iska filtter karege
# iske liye "Three Cloumns" create karege

# so, i'm using "NLTK" Library
```

In [32]:

```
print(df.columns)
```

```
Index(['target', 'text'], dtype='object')
```

In [33]:

```
 import nltk
```

In [34]:

```
# that errors means not install "nltk"

!pip install nltk
```

```
Requirement already satisfied: nltk in /home/cdac/.local/lib/python
3.8/site-packages (3.8.1)
Requirement already satisfied: regex>=2021.8.3 in /home/cdac/.local/
lib/python3.8/site-packages (from nltk) (2023.5.5)
Requirement already satisfied: tqdm in /home/cdac/.local/lib/python
3.8/site-packages (from nltk) (4.65.0)
Requirement already satisfied: joblib in /home/cdac/.local/lib/pytho
n3.8/site-packages (from nltk) (1.2.0)
Requirement already satisfied: click in /usr/lib/python3/dist-packag
es (from nltk) (7.0)
```

In [35]:

```
# then after import nltk

import nltk
```

In [36]:

```
# then same importent "Dependency of NLTK download" so,

nltk.download('punkt')
```

```
[nltk_data] Downloading package punkt to /home/cdac/nltk_data...
[nltk_data]   Package punkt is already up-to-date!
```

Out[36]:

```
True
```

In [37]:

```
# "text" cloumn ke ander "character" ka lenthg find out then...
# har massage ka text charactor length count kar de raha hai

df['text'].apply(len)
```

Out[37]:

```
0       111
1        29
2       155
3        49
4        61
       ...
5569    160
5570     36
5571     57
5572    125
5573     26
Name: text, Length: 5160, dtype: int64
```

In [38]:

```
# ab "num_characters" cloums nam ke ander store kar dete hai "text length ko"
# create new cloumn of "num_characters"

df['num_characters'] = df['text'].apply(len)
```

In [39]:

```
# ab check karte hai parsent data table ko

df.head()
```

Out[39]:

	target	text	num_characters
0	0	Go until jurong point, crazy.. Available only ...	111
1	0	Ok lar... Joking wif u oni...	29
2	1	Free entry in 2 a wkly comp to win FA Cup fina...	155
3	0	U dun say so early hor... U c already then say...	49
4	0	Nah I don't think he goes to usf, he lives aro...	61

In [40]:

```
# "num of words count" karte hai ki "text" ke "row" ke santance me kitne words ha
#iske liye lambda santance run karega or NLTK library ke "word_tokennize" word co

df['text'].apply(lambda x:nltk.word_tokenize(x))
```

Out[40]:

```
0       [Go, until, jurong, point, ,, crazy, .., Avail...
1                 [Ok, lar, ..., Joking, wif, u, oni, ...]
2       [Free, entry, in, 2, a, wkly, comp, to, win, F...
3       [U, dun, say, so, early, hor, ..., U, c, alrea...
4       [Nah, I, do, n't, think, he, goes, to, usf, ,,...
                              ...
5569    [This, is, the, 2nd, time, we, have, tried, 2,...
5570      [Will, ü, b, going, to, esplanade, fr, home, ?]
5571    [Pity, ,, *, was, in, mood, for, that, ., So, ...
5572    [The, guy, did, some, bitching, but, I, acted,...
5573                  [Rofl, ., Its, true, to, its, name]
Name: text, Length: 5160, dtype: object
```

In [41]:

```
# "text" ke sabhi santance "words" me divide ho kar "Array list store" ho gaya
# ab Array me store words ka "length" count kar lenge
# So, use "len()"

df['text'].apply(lambda x:len(nltk.word_tokenize(x)))
```

Out[41]:

```
0       24
1        8
2       37
3       13
4       15
        ..
5569    35
5570     9
5571    15
5572    27
5573     7
Name: text, Length: 5160, dtype: int64
```

In [42]:

```
# ab "num_words" cloums nam ke ander store kar dete hai "inside Arrry words lengt
# create new cloumn of "num_words"

df ['num_words'] = df['text'].apply(lambda x:len(nltk.word_tokenize(x)))
```

In [43]:

```
# ab check karte hai parsent data table ko

df.head()
```

Out[43]:

	target	text	num_characters	num_words
0	0	Go until jurong point, crazy.. Available only ...	111	24
1	0	Ok lar... Joking wif u oni...	29	8
2	1	Free entry in 2 a wkly comp to win FA Cup fina...	155	37
3	0	U dun say so early hor... U c already then say...	49	13
4	0	Nah I don't think he goes to usf, he lives aro...	61	15

In [44]:

```
# "No. of Santance count" "text" ke ak "row" me kitne santance hai
#iske liye lambda santance run karega or NLTK library ke "sent_tokenize" word cou

df['text'].apply(lambda x:nltk.sent_tokenize(x))
```

Out[44]:

```
0       [Go until jurong point, crazy.., Available onl...
1                         [Ok lar..., Joking wif u oni...]
2       [Free entry in 2 a wkly comp to win FA Cup fin...
3       [U dun say so early hor... U c already then sa...
4       [Nah I don't think he goes to usf, he lives ar...
                              ...
5569    [This is the 2nd time we have tried 2 contact ...
5570                [Will ü b going to esplanade fr home?]
5571    [Pity, * was in mood for that., So...any other...
5572    [The guy did some bitching but I acted like i'...
5573                          [Rofl., Its true to its name]
Name: text, Length: 5160, dtype: object
```

In [45]:

```
# "text" ke sabhi rows me "santance" me divide ho kar "Array list store" ho gaya
# ab Array me store santance ka "length" count kar lenge
# So, use "len()"

df['text'].apply(lambda x:len(nltk.sent_tokenize(x)))
```

Out[45]:

```
0       2
1       2
2       2
3       1
4       1
       ..
5569    4
5570    1
5571    2
5572    1
5573    2
Name: text, Length: 5160, dtype: int64
```

In [46]:

```
# ab "num_sentences" cloums nam ke ander store kar dete hai "inside Arrry sentanc
# create new cloumn of "num_sentences"

df ['num_sentences'] = df['text'].apply(lambda x:len(nltk.sent_tokenize(x)))
```

In [47]:

```
# ab check karte hai parsent data table ko

df.head()
```

Out[47]:

	target	text	num_characters	num_words	num_sentences
0	0	Go until jurong point, crazy.. Available only ...	111	24	2
1	0	Ok lar... Joking wif u oni...	29	8	2
2	1	Free entry in 2 a wkly comp to win FA Cup fina...	155	37	2
3	0	U dun say so early hor... U c already then say...	49	13	1
4	0	Nah I don't think he goes to usf, he lives aro...	61	15	1

In [48]:

```
print(df.columns)
```

```
Index(['target', 'text', 'num_characters', 'num_words', 'num_sentenc
es'], dtype='object')
```

In [49]:

```
# ab check karte hai ki pure table ka data analysis karte hai...
# like maximum or minmum length, total %, etc...
# So, use this funtion ".describe()"

df[['num_characters','num_words','num_sentences']].describe()
```

Out[49]:

	num_characters	num_words	num_sentences
count	5160.000000	5160.000000	5160.000000
mean	79.141085	18.588178	1.970543
std	58.289153	13.396252	1.455918
min	2.000000	1.000000	1.000000
25%	36.000000	9.000000	1.000000
50%	61.000000	15.000000	1.000000
75%	118.000000	26.000000	2.000000
max	910.000000	220.000000	38.000000

In [50]:

```
# yaha jese ki num_characters ke column ke ander
# maximum No of character use 910.000.. (describe both data ham and spam)
# so seprate data analysis for ham and spam
```

In [51]:

```
# for "ham ->0 message" data analysis

df[df['target'] == 0][['num_characters','num_words','num_sentences']].describe()
```

Out[51]:

	num_characters	num_words	num_sentences
count	4518.000000	4518.000000	4518.000000
mean	70.860558	17.289951	1.827579
std	56.584422	13.579652	1.394245
min	2.000000	1.000000	1.000000
25%	34.000000	8.000000	1.000000
50%	53.000000	13.000000	1.000000
75%	91.000000	22.000000	2.000000
max	910.000000	220.000000	38.000000

In [52]:

```
print(df.columns)
```

```
Index(['target', 'text', 'num_characters', 'num_words', 'num_sentenc
es'], dtype='object')
```

In [53]:

```
# for "spam ->1 message" data analysis

df[df['target'] == 1][['num_characters','num_words','num_sentences']].describe()
```

Out[53]:

	num_characters	num_words	num_sentences
count	642.000000	642.000000	642.000000
mean	137.414330	27.724299	2.976636
std	29.975596	7.028380	1.484527
min	13.000000	2.000000	1.000000
25%	131.000000	25.000000	2.000000
50%	148.000000	29.000000	3.000000
75%	157.000000	32.000000	4.000000
max	223.000000	46.000000	9.000000

In [54]:

```
print(df.columns)
```

```
Index(['target', 'text', 'num_characters', 'num_words', 'num_sentenc
es'], dtype='object')
```

In [ ]:

```

```

In [55]:

```
# ab check karte hai Histogram (Bar Graph)ham or spam message ko
# iske liye "seaborn" Library ki jarurate hogi

import seaborn as sns
```

In [56]:

```
# that errors means not install "seaborn"

!pip install seaborn
```

```
Requirement already satisfied: seaborn in /home/cdac/.local/lib/pyth
on3.8/site-packages (0.12.2)
Requirement already satisfied: numpy!=1.24.0,>=1.17 in /home/cdac/.l
ocal/lib/python3.8/site-packages (from seaborn) (1.24.3)
Requirement already satisfied: pandas>=0.25 in /usr/local/lib/python
3.8/dist-packages (from seaborn) (2.0.1)
Requirement already satisfied: matplotlib!=3.6.1,>=3.1 in /home/cda
c/.local/lib/python3.8/site-packages (from seaborn) (3.7.1)
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/
lib/python3.8/dist-packages (from pandas>=0.25->seaborn) (2.8.2)
Requirement already satisfied: tzdata>=2022.1 in /usr/local/lib/pyth
on3.8/dist-packages (from pandas>=0.25->seaborn) (2023.3)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python
3.8/dist-packages (from pandas>=0.25->seaborn) (2023.3)
Requirement already satisfied: contourpy>=1.0.1 in /home/cdac/.loca
l/lib/python3.8/site-packages (from matplotlib!=3.6.1,>=3.1->seabor
n) (1.0.7)
Requirement already satisfied: importlib-resources>=3.2.0; python_ve
rsion < "3.10" in /home/cdac/.local/lib/python3.8/site-packages (fro
m matplotlib!=3.6.1,>=3.1->seaborn) (5.12.0)
Requirement already satisfied: kiwisolver>=1.0.1 in /home/cdac/.loca
l/lib/python3.8/site-packages (from matplotlib!=3.6.1,>=3.1->seabor
n) (1.4.4)
Requirement already satisfied: cycler>=0.10 in /home/cdac/.local/li
b/python3.8/site-packages (from matplotlib!=3.6.1,>=3.1->seaborn)
(0.11.0)
Requirement already satisfied: packaging>=20.0 in /home/cdac/.local/
lib/python3.8/site-packages (from matplotlib!=3.6.1,>=3.1->seaborn)
(23.1)
Requirement already satisfied: pyparsing>=2.3.1 in /home/cdac/.loca
l/lib/python3.8/site-packages (from matplotlib!=3.6.1,>=3.1->seabor
n) (3.0.9)
Requirement already satisfied: pillow>=6.2.0 in /usr/lib/python3/dis
t-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (7.0.0)
Requirement already satisfied: fonttools>=4.22.0 in /home/cdac/.loca
l/lib/python3.8/site-packages (from matplotlib!=3.6.1,>=3.1->seabor
n) (4.39.4)
Requirement already satisfied: six>=1.5 in /usr/lib/python3/dist-pac
kages (from python-dateutil>=2.8.2->pandas>=0.25->seaborn) (1.14.0)
Requirement already satisfied: zipp>=3.1.0; python_version < "3.10"
in /home/cdac/.local/lib/python3.8/site-packages (from importlib-res
ources>=3.2.0; python_version < "3.10"->matplotlib!=3.6.1,>=3.1->sea
born) (3.15.0)
```

In [57]:

```
# run/import again

import seaborn as sns
```

In [58]:

```
# traget column ke num_characters row me kitne charactor used ho rahe hai

df[df['target'] == 0]['num_characters']
```

Out[58]:

```
0       111
1        29
3        49
4        61
6        77
       ...
5567     12
5570     36
5571     57
5572    125
5573     26
Name: num_characters, Length: 4518, dtype: int64
```

In [59]:

```
print(df.columns)
```

```
Index(['target', 'text', 'num_characters', 'num_words', 'num_sentenc
es'], dtype='object')
```

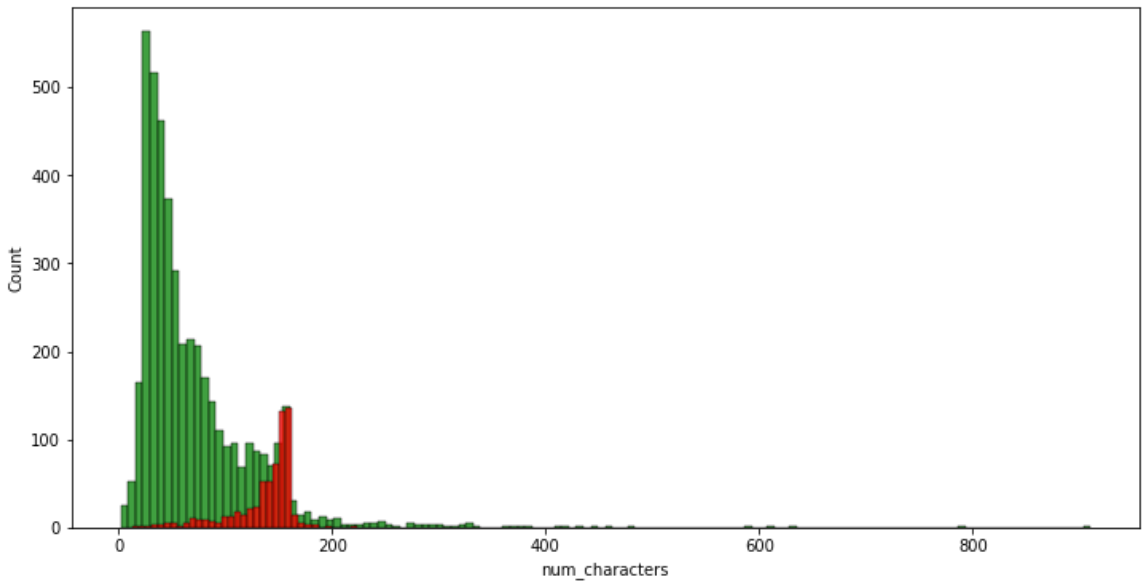
In [60]:

```
# library seaborn ka "histplot" function used kar show karte hai
# ham-> 0 message ko color "green"
# spam-> 1 message ko color "red"
# or figure ka size bada kar dekhte hai

plt.figure(figsize=(12,6))
sns.histplot(df[df['target'] == 0]['num_characters'],color='green')
sns.histplot(df[df['target'] == 1]['num_characters'],color='red')
```

Out[60]:

```
<Axes: xlabel='num_characters', ylabel='Count'>
```



In [61]:

```
# owhi ab "num_words" or "num_sentances" ke sath check karte hai
```

In [62]:

```
print(df.columns)
```

```
Index(['target', 'text', 'num_characters', 'num_words', 'num_sentenc
es'], dtype='object')
```

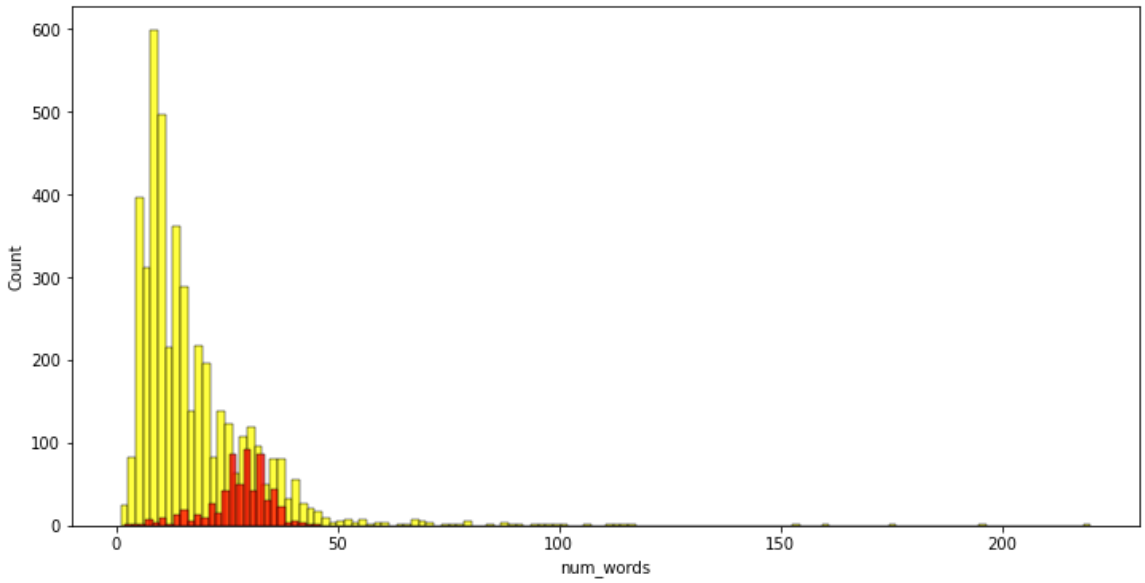
In [63]:

```
# num_words
# ham-> 0 message ko color "yellow"
# spam-> 1 message ko color "red"

plt.figure(figsize=(12,6))
sns.histplot(df[df['target'] == 0]['num_words'],color='yellow')
sns.histplot(df[df['target'] == 1]['num_words'],color='red')
```

Out[63]:

```
<Axes: xlabel='num_words', ylabel='Count'>
```



In [64]:

```
print(df.columns)
```

```
Index(['target', 'text', 'num_characters', 'num_words', 'num_sentenc
es'], dtype='object')
```

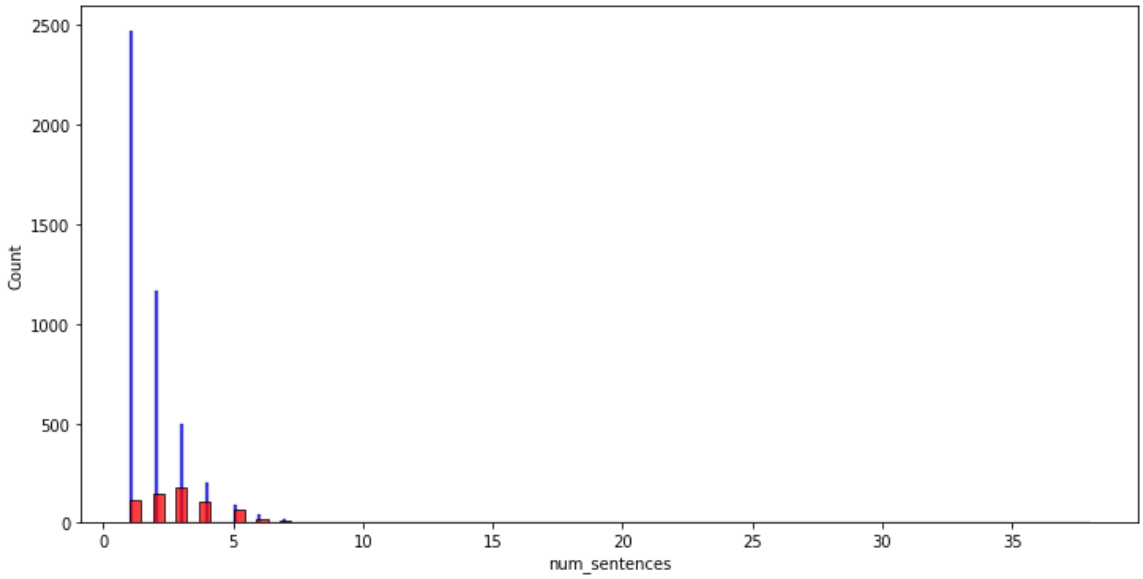
In [65]:

```
# num_sentences
# ham-> 0 message ko color "blue"
# spam-> 1 message ko color "red"

plt.figure(figsize=(12,6))
sns.histplot(df[df['target'] == 0]['num_sentences'],color='blue')
sns.histplot(df[df['target'] == 1]['num_sentences'],color='red')
```

Out[65]:

```
<Axes: xlabel='num_sentences', ylabel='Count'>
```



In [66]:

```
print(df.columns)
```

```
Index(['target', 'text', 'num_characters', 'num_words', 'num_sentenc
es'], dtype='object')
```

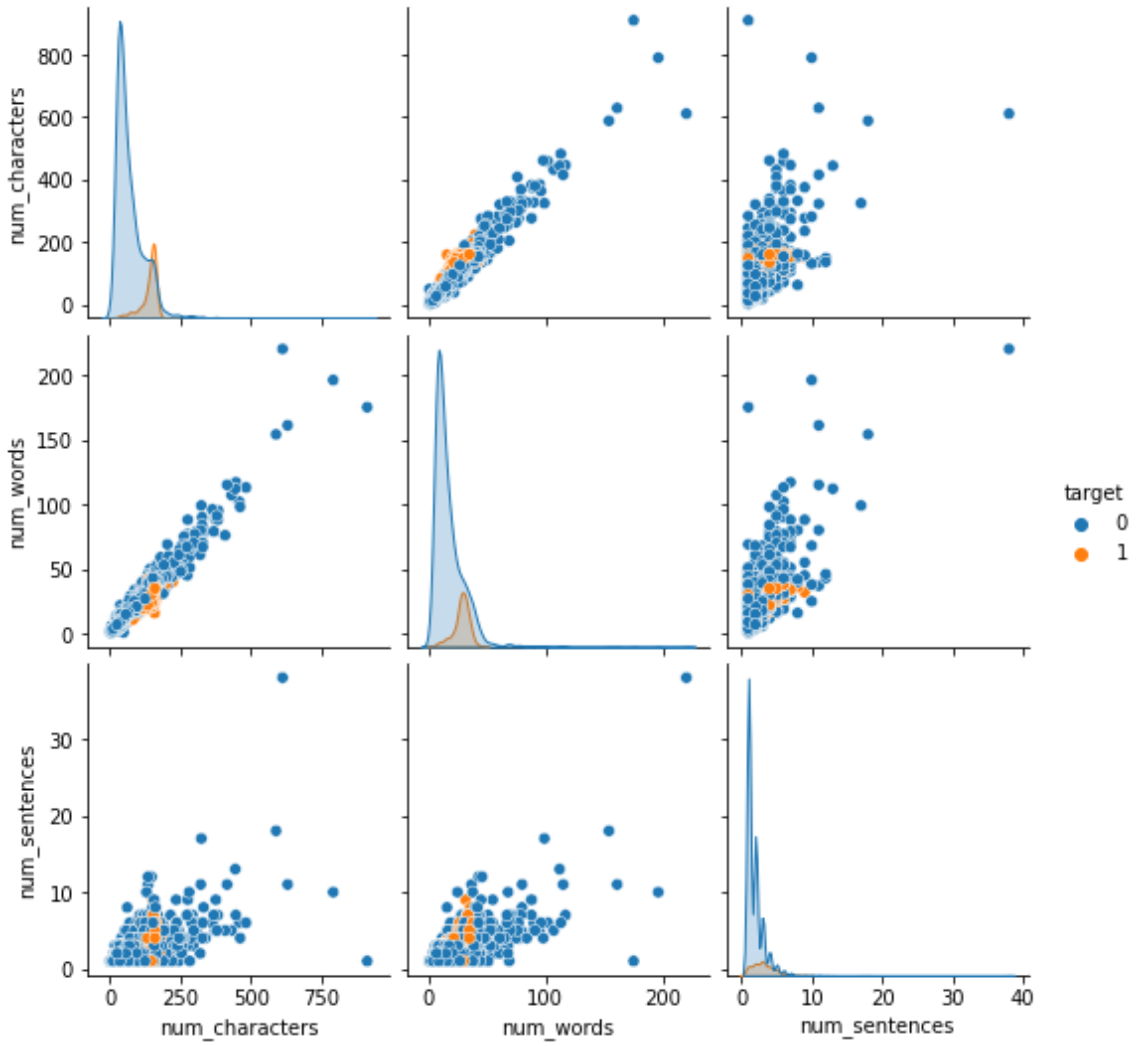
In [67]:

```
#ab check karte hai ki "ham or spam ke message" ka tino
#Three (charactor, words, or sentance) apas me kaya relation hai

sns.pairplot(df,hue='target')
```

Out[67]:

```
<seaborn.axisgrid.PairGrid at 0x7f7a4b3e0b50>
```



In [68]:

```
print(df.columns)
```

```
Index(['target', 'text', 'num_characters', 'num_words', 'num_sentenc
es'], dtype='object')
```

In [59]:

```
# ab sabhi ko "heatmap" me chaeck karte hai apas me relation hai

sns.heatmap(df.corr(),annot=True)
```

```
---------------------------------------------------------------------------
-------
ValueError                                Traceback (most recent cal
l last)
<ipython-input-59-bfded30e3083> in <module>
      1 # ab sabhi ko "heatmap" me chaeck karte hai apas me relation
hai
      2
----> 3 sns.heatmap(df.corr(),annot=True)

/usr/local/lib/python3.8/dist-packages/pandas/core/frame.py in corr
(self, method, min_periods, numeric_only)
  10057         cols = data.columns
  10058         idx = cols.copy()
> 10059         mat = data.to_numpy(dtype=float, na_value=np.nan, co
py=False)
  10060
  10061         if method == "pearson":

/usr/local/lib/python3.8/dist-packages/pandas/core/frame.py in to_nu
mpy(self, dtype, copy, na_value)
   1836         if dtype is not None:
   1837             dtype = np.dtype(dtype)
-> 1838         result = self._mgr.as_array(dtype=dtype, copy=copy,
na_value=na_value)
   1839         if result.dtype is not dtype:
   1840             result = np.array(result, dtype=dtype, copy=Fals
e)

/usr/local/lib/python3.8/dist-packages/pandas/core/internals/manager
s.py in as_array(self, dtype, copy, na_value)
   1730                 arr.flags.writeable = False
   1731         else:
-> 1732             arr = self._interleave(dtype=dtype, na_value=na_
value)
   1733             # The underlying data was copied within _interle
ave, so no need
   1734             # to further copy if copy=True or setting na_val
ue

/usr/local/lib/python3.8/dist-packages/pandas/core/internals/manager
s.py in _interleave(self, dtype, na_value)
   1792             else:
   1793                 arr = blk.get_values(dtype)
-> 1794             result[rl.indexer] = arr
   1795             itemmask[rl.indexer] = 1
   1796

ValueError: could not convert string to float: 'Go until jurong poin
t, crazy.. Available only in bugis n great world la e buffet... Cine
there got amore wat...'
```

In [67]:

```
# this errors means "string se float" me convert "nahi" kar pa raha hai
# to "matplotlib.pyplot" Library import karte hai
```

In [68]:

```
import matplotlib.pyplot as plt
```

In [69]:

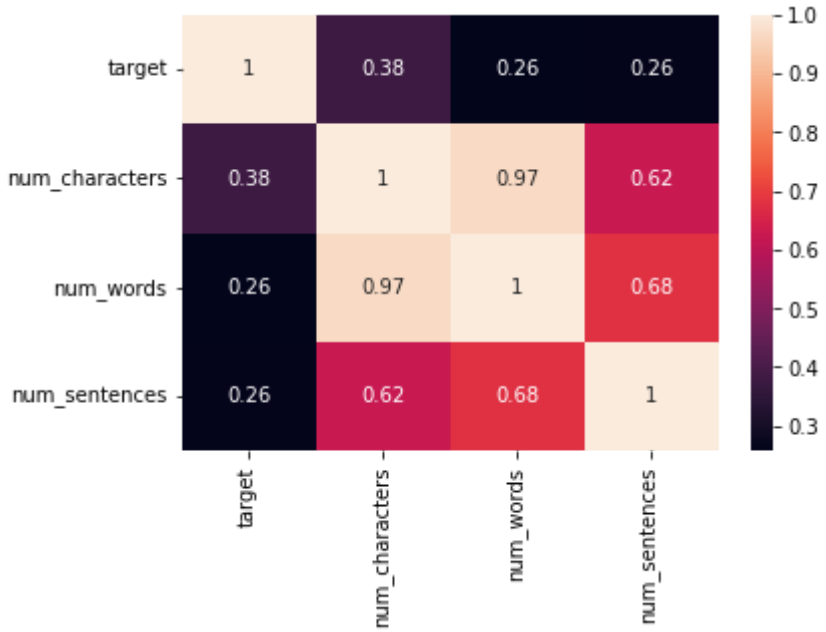
```
df = df.select_dtypes(include=[float, int])
```

In [70]:

```
sns.heatmap(df.corr(), annot=True)
```

Out[70]:

<Axes: >



In [71]:

```
#yaha appas me relation

# num_character <-> num_character => 1
# num_character <-> num_words => 0.97
# num_character <-> num_sentance => 0.62

# num_words <-> num_character => 0.97
# num_words <-> num_words => 1
# soon on.....

#strong relation
# num_character <-> num_character => 1
# num_words <-> num_words => 1
# num_sentance <-> num_sentance => 1

# "Model banane ke liye kisi ak ko lege jese "num_character" ko"
# ham tino (three) ko nahi lege kunki jayada strong ho jayega
```

3. Data Preprocessing

3.1 Lower case

3.2 Tokenization

3.3 Removing special characters

3.4 Removing stop words and punctuation

3.5 Stemming

In [72]:

```
# sabse pehale ham ye sabhi ka ak-ak "example" ke rup me dekh lete hai
# phir apne Project par apply karge
```

In [70]:

```
print(df.columns)
```

```
Index(['target', 'text', 'num_characters', 'num_words', 'num_sentenc
es'], dtype='object')
```

In [ ]:

```

```

In [71]:

```
#3.3 Example of ""Remove Special character"in the "sentence words""
# iske liye loop bana kar "isalnum()" call karte hai,
# "isalnum()" ye Alphabetic and Number ko select karega
# ".append()" ye yaha "y" me assin kar dega value ko

# def transform_text(text2):
#     text2 = text2.lower()
#     text2 = nltk.word_tokenize(text2)

#     y = []
#     for i in text2:
#         if i.isalnum():
#             y.append(i)
#     return y

# transform_text('Hi how Are You? e.g 20%')
```

In [ ]:

```

```

In [72]:

```
# 3.4.1 Example "StopWords"
# StopWords => yese "words" so sentence ke "meaning" me koi contribution "nahi" h
# kewal iska kam sentance "formation" hota hai
# e.g..
```

In [73]:

```
# iske liye "NLTK Library" se "stopwords" find out karege
import nltk
```

In [74]:

```
# download karte hai "stopwords"

nltk.download('stopwords')
```

```
[nltk_data] Downloading package stopwords to /home/cdac/nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
```

Out[74]:

True

In [75]:

```
# ab filter karte hai "stopwords" se "english words" ka list out karte hai

stopwords.words('english')
```

```
---------------------------------------------------------------------------
-------
NameError                                 Traceback (most recent cal
l last)
<ipython-input-75-b93a77273381> in <module>
      1 # ab filter karte hai "stopwords" se "english words" ka list
out karte hai
      2
----> 3 stopwords.words('english')

NameError: name 'stopwords' is not defined
```

In [76]:

```
# ab aage future Direct use kar sakte hai iss command ke through

from nltk.corpus import stopwords
stopwords.words('english')
```

```
 'isn',
 "isn't",
 'ma',
 'mightn',
 "mightn't",
 'mustn',
 "mustn't",
 'needn',
 "needn't",
 'shan',
 "shan't",
 'shouldn',
 "shouldn't",
 'wasn',
 "wasn't",
 'weren',
 "weren't",
 'won',
 "won't",
 'wouldn',
```

In [77]:

```
# 3.4.1 Example "Punctuation" list sort list karte hai
# iske liye "import string library"

import string
string.punctuation
```

Out[77]:

```
'!"#$%&\'()*+,-./:;<=>?@[\\]^_`{|}~'
```

In [78]:

```
# 3.5 Example "Stemming"
# ye "varb words" ko "original word" me la deta hai
# e.g.. Loving -> Love, Dancing -> Dance, Played -> Play etc..
# iske liye NLTK ka PorterStemmer module import karna hoga

from nltk.stem.porter import PorterStemmer
ps = PorterStemmer()
ps.stem('loving')
```

Out[78]:

```
'love'
```

In [79]:

```
print(df.columns)
```

```
Index(['target', 'text', 'num_characters', 'num_words', 'num_sentenc
es'], dtype='object')
```

In [80]:

```
# ab isse uppr wale transform_text par apply karte hai
# yaha "text = y[:]" ye "cloning" hai jo "y" ke value ko text me store kar raha h

def transform_text(text):
    #Ex.3.1
    text = text.lower()

    #Ex.3.2
    text = nltk.word_tokenize(text)

    #Ex.3.3
    y = []
    for i in text:
        if i.isalnum():
            y.append(i)

    #Ex.3.4
    text = y[:]
    y.clear()

    for i in text:
        if i not in stopwords.words('english') and i not in string.punctuation:
            y.append(i)
    #Ex.3.5
    text = y[:]
    y.clear()

    for i in text:
        y.append(ps.stem(i))

    return " ".join(y)

transform_text('I loved the CDAC leactures on Machine Learning. How about you? ')
```

Out[80]:

```
'love cdac leactur machin learn'
```

In [81]:

```
print(df.columns)
```

```
Index(['target', 'text', 'num_characters', 'num_words', 'num_sentenc
es'], dtype='object')
```

In [82]:

```
df.head()
```

Out[82]:

	target	text	num_characters	num_words	num_sentences
0	0	Go until jurong point, crazy.. Available only ...	111	24	2
1	0	Ok lar... Joking wif u oni...	29	8	2
2	1	Free entry in 2 a wkly comp to win FA Cup fina...	155	37	2
3	0	U dun say so early hor... U c already then say...	49	13	1
4	0	Nah I don't think he goes to usf, he lives aro...	61	15	1

In [83]:

```
df['text'].apply(transform_text)
```

Out[83]:

```
0       go jurong point crazi avail bugi n great world...
1                                   ok lar joke wif u oni
2       free entri 2 wkli comp win fa cup final tkt 21...
3                     u dun say earli hor u c alreadi say
4                    nah think goe usf live around though
                              ...
5569    2nd time tri 2 contact u pound prize 2 claim e...
5570                              ü b go esplanad fr home
5571                                   piti mood suggest
5572    guy bitch act like interest buy someth els nex...
5573                                      rofl true name
Name: text, Length: 5160, dtype: object
```

In [84]:

```
# mere liye only "traget" and "tranformed_text" cloumns imported hai

df['transformed_text'] = df['text'].apply(transform_text)
```

In [85]:

```
df.head()
```

Out[85]:

	target	text	num_characters	num_words	num_sentences	transformed_text
0	0	Go until jurong point, crazy.. Available only ...	111	24	2	go jurong point crazi avail bugi n great world...
1	0	Ok lar... Joking wif u oni...	29	8	2	ok lar joke wif u oni
2	1	Free entry in 2 a wkly comp to win FA Cup fina...	155	37	2	free entri 2 wkli comp win fa cup final tkt 21...
3	0	U dun say so early hor... U c already then say...	49	13	1	u dun say earli hor u c alreadi say
4	0	Nah I don't think he goes to usf, he lives aro...	61	15	1	nah think goe usf live around though

In [96]:

```
print(df.columns)
```

```
Index(['target', 'text', 'num_characters', 'num_words', 'num_sentenc
es',
       'transformed_text'],
      dtype='object')
```

In [97]:

```
# ab "ham->0 spam->1" ye demo message "kya-kya words used" huwa hai
#usee a "Image me dekhe" ge, jo jyda use hoga owh sabse bada dikhega
# iske liye hame "WordCloud Library" ka use karna hoga
```

In [98]:

```
# from wordcloud import WordCloud
# wc = WordCloud(width=500,height=500,min_font_size=10,backgroud_color='white')
# spam_wc = wc.generate(df[df['target'] == 1]['transformed_text'].str.cat(sep=" "
# plt.imshow(spam_wc)
```

In [ ]:

```

```

In [87]:

```
# ab most "top 30",50 etc. "common used" words in the "ham and spam mesaage"
#ko sortlist karte hai
```

In [88]:

```
#hame only "target" and "transformed_text" cloumns ke data par parfom karege
#sabse pehale spam->1 message ko table se alag karte hai

df[df['target'] == 1]
```

Out[88]:

	target	text	num_characters	num_words	num_sentences	transformed_text
2	1	Free entry in 2 a wkly comp to win FA Cup fina...	155	37	2	free entri 2 wkli comp win fa cup final tkt 21...
5	1	FreeMsg Hey there darling it's been 3 week's n...	147	39	4	freemsg hey darl 3 week word back like fun sti...
8	1	WINNER!! As a valued network customer you have...	157	32	5	winner valu network custom select receivea pri...
9	1	Had your mobile 11 months or more? U R entitle...	154	31	3	mobil 11 month u r entitl updat latest colour ...
11	1	SIX chances to win CASH! From 100 to 20,000 po...	136	31	3	six chanc win cash 100 pound txt csh11 send co...
...	...	...	...	...	...	...
5539	1	Want explicit SEX in 30 secs? Ring 02073162414...	90	18	3	want explicit sex 30 sec ring 02073162414 cost...
5542	1	ASKED 3MOBILE IF 0870 CHATLINES INCLU IN FREE ...	158	38	6	ask 3mobil 0870 chatlin inclu free min india c...
5549	1	Had your contract mobile 11 Mnths? Latest Moto...	160	35	5	contract mobil 11 mnth latest motorola nokia e...
5568	1	REMINDER FROM O2: To get 2.50 pounds free call...	147	30	1	remind o2 get pound free call credit detail gr...
5569	1	This is the 2nd time we have tried 2 contact u...	160	35	4	2nd time tri 2 contact u pound prize 2 claim e...

642 rows × 6 columns

In [89]:

```
# ab isme se "transformed_text" me used words ko ak "List" me dal dete hai
# yaha har message ak "item hai"

df[df['target'] == 1]['transformed_text'].tolist()
```

```
ree camcord pleas call 08000930705 deliveri tomorrow',
 'sm ac sptv new jersey devil detroit red wing play ice hockey cor
rect incorrect end repli end sptv',
 'congrat 1 year special cinema pass 2 call 09061209465 c suprman
v matrix3 starwars3 etc 4 free 150pm dont miss',
 'valu custom pleas advis follow recent review mob award bonu priz
e call 09066364589',
 'urgent ur award complimentari trip eurodisinc trav aco entry41 c
laim txt di 87121 morefrmmob shracomorsglsuplt 10 ls1 3aj',
 'hear new divorc barbi come ken stuff',
 'pleas call custom servic repres 0800 169 6031 guarante cash priz
e',
 'free rington wait collect simpli text password mix 85069 verifi
get usher britney fml po box 5249 mk17 92h 450ppw 16',
 'gent tri contact last weekend draw show prize guarante call clai
m code k52 valid 12hr 150ppm',
 'winner u special select 2 receiv 4 holiday flight inc speak live
oper 2 claim',
 'privat 2004 account statement 07742676969 show 786 unredeem bonu
point claim call 08719180248 identifi code 45239 expir',
```

In [90]:

```
# ab sabhi message ko ak-ak kar "print" karte hai by the help of "for loop"

for msg in df[df['target'] == 1]['transformed_text'].tolist():
    print(msg)
```

```
cash prize claim call09050000327 c rstm sw7 3ss 150ppm
88800 89034 premium phone servic call 08718711108
sm ac sun0819 post hello seem cool want say hi hi stop send stop 6
2468
get ur 1st rington free repli msg tone gr8 top 20 tone phone everi
week per wk 2 opt send stop 08452810071 16
hi sue 20 year old work lapdanc love sex text live bedroom text su
e textoper g2 1da 150ppmsg
forward 448712404000 pleas call 08712404000 immedi urgent messag w
ait
review keep fantast nokia game deck club nokia go 2 unsubscrib ale
rt repli word
4mth half price orang line rental latest camera phone 4 free phone
call mobilesdirect free 08000938767 updat or2stoptxt cs
08714712388 cost 10p
urgent 2nd attempt contact u u call 09071512433 b4 050703 csbcm423
5wc1n3xx callcost 150ppm mobilesvari 50
guarante cash prize claim yr prize call custom servic repres 08714
712394
email alertfrom jeri stewarts 2kbsubject prescripiton drvgsto list
```

In [91]:

```
# ab isse "msg" se sabhi "words" ko ak-ak "alag" kar "list" me append(assin) kar

spam_corpus = []
for msg in df[df['target'] == 1]['transformed_text'].tolist():
    for word in msg.split():
        spam_corpus.append(word)
```

In [92]:

```
# check(view) list

spam_corpus
```

Out[92]:

```
['free',
 'entri',
 '2',
 'wkli',
 'comp',
 'win',
 'fa',
 'cup',
 'final',
 'tkt',
 '21st',
 'may',
 'text',
 'fa',
 '87121',
 'receiv',
 'entri',
 'question',
```

In [93]:

```
# count list length(spam_corpus)
# ki kitene words hai isse list me

len(spam_corpus)
```

Out[93]:

```
9808
```

In [94]:

```
# ab check karte hai ki isse list used words ka information nikalte hai
#like kitini par used huwa hai,"most_common" word, "least_common" word used....
# yaha "most_common" used nikal rahe hai "Top 30" words me se
# iske liye "Collections Library" ka used karte hai

from collections import Counter
Counter(spam_corpus).most_common(30)
```

Out[94]:

```
[('call', 313),
 ('free', 186),
 ('2', 154),
 ('txt', 139),
 ('text', 122),
 ('ur', 119),
 ('u', 118),
 ('mobil', 110),
 ('stop', 108),
 ('repli', 103),
 ('claim', 97),
 ('4', 95),
 ('prize', 79),
 ('get', 73),
 ('new', 64),
 ('servic', 64),
 ('send', 60),
 ('tone', 59),
 ('urgent', 57),
 ('award', 55),
 ('nokia', 54),
 ('contact', 54),
 ('phone', 52),
 ('cash', 50),
 ('pleas', 50),
 ('week', 49),
 ('win', 46),
 ('min', 45),
 ('c', 43),
 ('guarante', 42)]
```

In [106]:

```
# ab ak DataFrame me add kar dete hai sabhi ko

from collections import Counter
pd.DataFrame(Counter(spam_corpus).most_common(30))
```

Out[106]:

	0	1
0	call	313
1	free	186
2	2	154
3	txt	139
4	text	122
5	ur	119
6	u	118
7	mobil	110
8	stop	108
9	repli	103
10	claim	97
11	4	95
12	prize	79
13	get	73
14	new	64
15	servic	64
16	send	60
17	tone	59
18	urgent	57
19	award	55
20	nokia	54
21	contact	54
22	phone	52
23	cash	50
24	pleas	50
25	week	49
26	win	46
27	min	45
28	c	43
29	guarante	42

In [124]:

```
print(df.columns)
```

```
Index(['target', 'text', 'num_characters', 'num_words', 'num_sentenc
es',
       'transformed_text'],
      dtype='object')
```

In [127]:

```
# ab isse "DataFrame" ko ak "Bar Chart" me "show" karte hai

# from collections import Counter
# sns.barplot(pd.DataFrame(Counter(spam_corpus).most_common(30))[0],pd.DataFrame(
# plt.xticks(rotation='vertical')
# plt.show()
```

```
---------------------------------------------------------------------
-------
TypeError                                 Traceback (most recent cal
l last)
<ipython-input-127-2cc61ab93d29> in <module>
      3 from collections import Counter
      4 get_ipython().run_line_magic('matplotlib', 'inline')
----> 5 sns.barplot(pd.DataFrame(Counter(spam_corpus).most_common(3
0))[0],pd.DataFrame(Counter(spam_corpus).most_common(30))[1])
      6 plt.xticks(rotation='vertical')
      7 plt.show()

TypeError: barplot() takes from 0 to 1 positional arguments but 2 we
re given
```

4 Model Building

In [131]:

```
#ab hamare liye two cloumns importent hai
# 1st "target" (column jo ki 0-> ham, 1-> spam) ye hame liye "Output" ka kam karege
# 2nd "tranformed_text" (column jo sare filter karne ke bad mila hai) ye hamre li

# lekin, dono cloumns ka "data(Row ka)" hame "interger(0 ya 1)" chahiye jo ki "ve
# lekin,, yaha hamare pass 'target' column ka data "interger" hai. But "transform
# to isse("tranformed_text" ka sabhi text ko) "integer"(yani vector) bana hoga..

# iske liye "CountVectorizer Library" ka used karege
```

In [132]:

```
from sklearn.feature_extraction.text import CountVectorizer
cv = CountVectorizer()
```

In [133]:

```
# "tranform_text" columns ke data(text)) ko interger(0 ya 1)convert kar dete hai
#or usse array ke rup me "X" me assin(store) kar dete hai

X = cv.fit_transform(df['transformed_text']).toarray()
```

In [134]:

```
# yaha X hame mil gaya or X me sabhi 0 ke rup me assin(store) hoga jo ki sahi hai
X
```

Out[134]:

```
array([[0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       ...,
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0]])
```

In [136]:

```
X.shape
```

Out[136]:

```
(5160, 6784)
```

In [137]:

```
# yaha sms => 5160 and word => 6784

print(df.columns)
```

```
Index(['target', 'text', 'num_characters', 'num_words', 'num_sentenc
es',
       'transformed_text'],
      dtype='object')
```

In [138]:

```
# ab hame Y bhi nikalna hoga to..

y = df['target'].values
```

In [139]:

```
y
```

Out[139]:

```
array([0, 0, 1, ..., 0, 0, 0])
```

4.1 Model Building (apply diff. Algo.)

check for best Accuracy

then select one Algo. after build Model

In [ ]: