# SPAM NEWS DETECTION

Spam news detection refers to the process of identifying and filtering out false, misleading, or irrelevant information from legitimate news sources. The term "spam news" can encompass a range of content, including:

1. **Fake News**: Completely fabricated or highly distorted information presented as factual news.

2. **Clickbait**: Sensationalized or misleading headlines designed to attract clicks without providing meaningful content.

3. **Hoaxes**: False stories or claims intended to deceive and misinform readers.

4. **Misinformation**: Inaccurate or misleading information shared without malicious intent but still harmful.

5. **Propaganda**: Biased or one-sided information used to promote a particular agenda or ideology.

To detect spam news, various techniques and tools can be employed:

1. **Fact-Checking**: Fact-checking organizations and algorithms are used to verify the accuracy of claims and statements in news articles.

2. **Natural Language Processing (NLP)**: NLP models, like GPT-3, can be used to analyze the content of news articles for inconsistencies, biases, and misleading information.

3. **Source Verification**: Evaluating the credibility of the news source or website can help identify spam news. Reputable sources are more likely to provide accurate information.

4. **Cross-Referencing**: Comparing information with multiple reliable sources can help identify inconsistencies or misinformation.

5. **Crowdsourcing**: Involving the community in identifying and flagging suspicious news can be an effective method to detect spam news.

6. **Machine Learning Algorithms**: Algorithms can be trained to classify news articles as spam or legitimate based on various features, such as content, writing style, and source reputation.

7. **Semantic Analysis**: Analyzing the meaning and context of news articles to identify inconsistencies or false information.

It's essential to approach spam news detection with care, as it can be a delicate balance between preventing misinformation and protecting freedom of speech. Automated systems and human fact-checkers often work in tandem to address this issue effectively.

Several organizations and fact-checking platforms are dedicated to combatting spam news and providing accurate information to the public. Users should also be vigilant and critical consumers of news, verifying information from multiple reliable sources before accepting it as truth.

Detecting spam news using Machine Learning (ML) and Natural Language Processing (NLP) involves the application of algorithms and techniques to analyze and classify news articles as either legitimate or spam based on their content, style, and other features. Here's more about how ML and NLP are used in spam news detection:

1. **Data Collection:** To train ML models, you need a dataset of labeled news articles. This dataset should include both legitimate news articles and examples of spam, including fake news, clickbait, hoaxes, etc.

2**. Feature Extraction:**

- Text data from news articles is processed and transformed into numerical representations that can be used by ML algorithms.

- Features might include word frequencies, TF-IDF (Term Frequency-Inverse Document Frequency) values, n-grams, or word embeddings.

3. **Text Preprocessing:**

- Text data is cleaned by removing stopwords, punctuation, and special characters.

- Text is often tokenized into words or subword tokens, depending on the NLP model being used.

4. **Model Selection:**

- ML models commonly used for spam news detection include decision trees, random forests, support vector machines (SVM), and more advanced models like deep learning models (e.g., recurrent neural networks or transformers).

5. **Training the Model:**

- The ML model is trained on the labeled dataset, using features extracted from both legitimate and spam news articles.

- The model learns to distinguish between the two classes by identifying patterns, word usage, and other characteristics associated with spam news.

6. **Evaluation:**

- The trained model is evaluated on a separate test dataset to assess its performance, typically using metrics like accuracy, precision, recall, F1-score, and ROC-AUC.

7. **Fine-Tuning:**

- The model's hyperparameters may be fine-tuned to improve its performance.

8. **Real-Time Detection:**

- After training and evaluation, the model can be used for real-time spam news detection. It processes incoming news articles and classifies them as legitimate or spam.

9. **Ensemble Methods:**

- Multiple ML models can be combined using ensemble techniques to improve detection accuracy. Common ensemble methods include bagging and boosting.

10**. NLP Techniques:**

- NLP techniques can be applied to understand the semantic meaning of the text. This includes sentiment analysis, entity recognition, and topic modeling to identify patterns or inconsistencies.

11. **Continuous Learning:**

- To adapt to evolving spam tactics, models may undergo continuous learning and retraining to stay up-to-date.

12. **User Feedback:**

- User feedback can be valuable in improving the model's accuracy. Users can flag suspicious news, which is then used to enhance the training data.

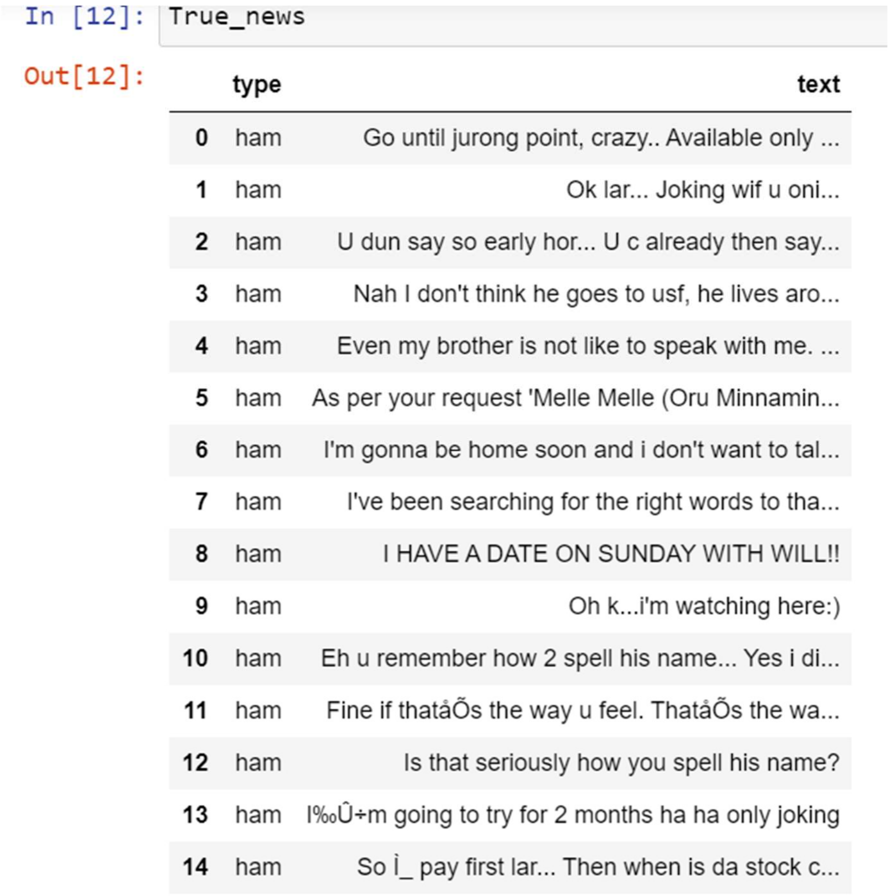
13. **Scalability:**

- The system should be scalable to handle a large volume of news articles efficiently.

It's important to note that spam news detection using ML and NLP is an evolving field, and the effectiveness of models depends on the quality of data, the choice of features, and the sophistication of the algorithms used. Moreover, it's essential to strike a balance between minimizing false positives (misclassifying legitimate news as spam) and false negatives (allowing spam news to go undetected).

## CODE

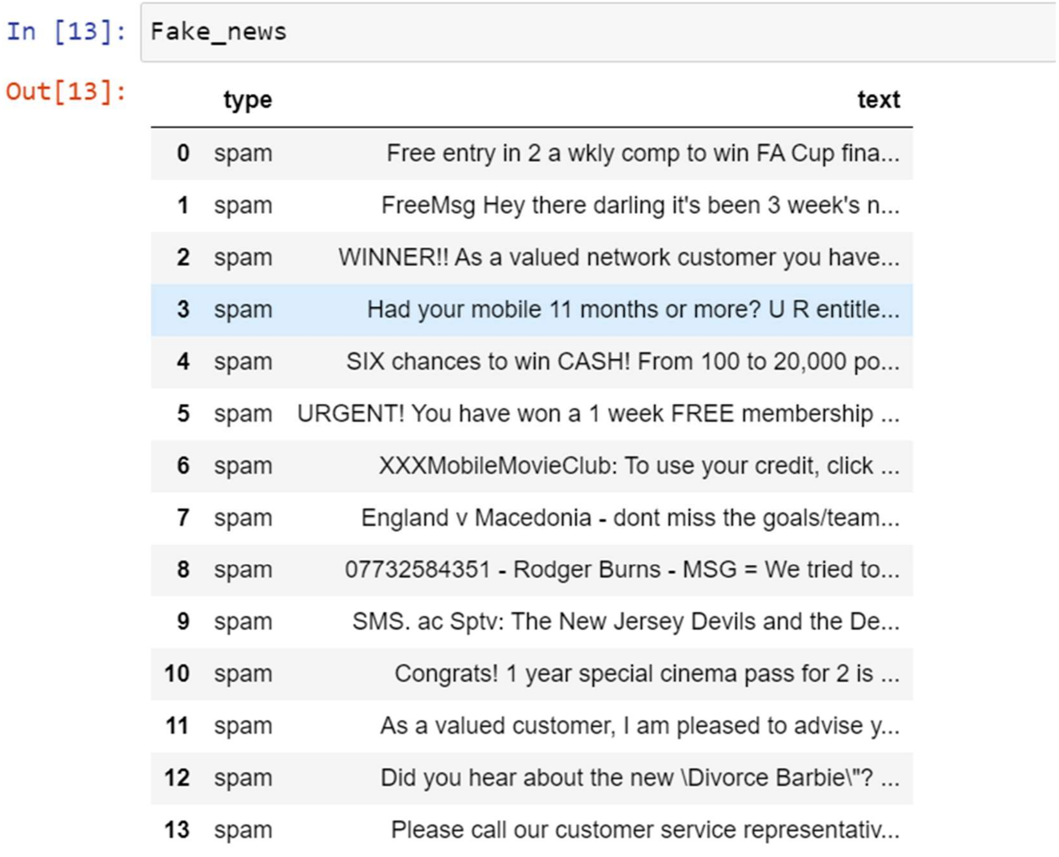
```
#IMPORTING LIBRARIES

import numpy as np

import pandas as pd

True_news = pd.read_excel('True.xlsx')

Fake_news = pd.read_excel('Fake.xlsx')
```



```
In [12]: True_news
```

Out[12]:

| | type | text |
|---|---|---|
| **0** | ham | Go until jurong point, crazy.. Available only ... |
| **1** | ham | Ok lar... Joking wif u oni... |
| **2** | ham | U dun say so early hor... U c already then say... |
| **3** | ham | Nah I don't think he goes to usf, he lives aro... |
| **4** | ham | Even my brother is not like to speak with me. ... |
| **5** | ham | As per your request 'Melle Melle (Oru Minnamin... |
| **6** | ham | I'm gonna be home soon and i don't want to tal... |
| **7** | ham | I've been searching for the right words to tha... |
| **8** | ham | I HAVE A DATE ON SUNDAY WITH WILL!! |
| **9** | ham | Oh k...i'm watching here:) |
| **10** | ham | Eh u remember how 2 spell his name... Yes i di... |
| **11** | ham | Fine if thatåÕs the way u feel. ThatåÕs the wa... |
| **12** | ham | Is that seriously how you spell his name? |
| **13** | ham | I‰Û÷m going to try for 2 months ha ha only joking |
| **14** | ham | So Ì_ pay first lar... Then when is da stock c... |



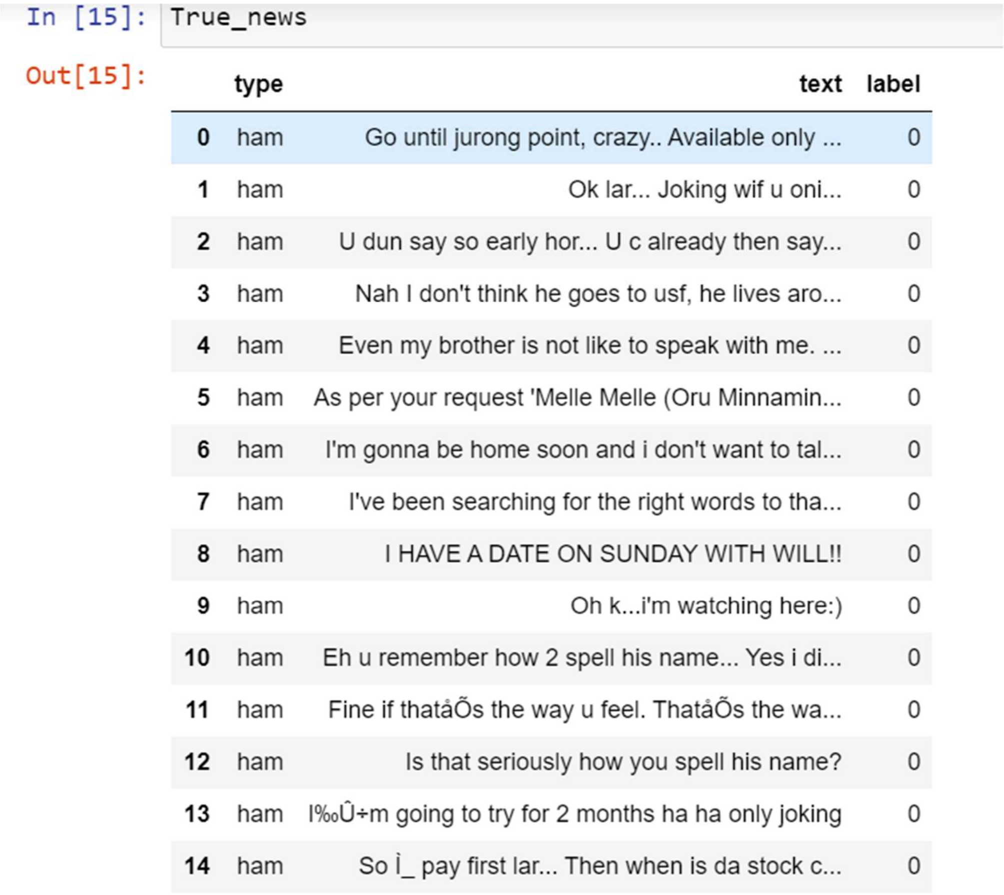
```
In [13]: Fake_news
```

Out[13]:

| | type | text |
|---|---|---|
| **0** | spam | Free entry in 2 a wkly comp to win FA Cup fina... |
| **1** | spam | FreeMsg Hey there darling it's been 3 week's n... |
| **2** | spam | WINNER!! As a valued network customer you have... |
| **3** | spam | Had your mobile 11 months or more? U R entitle... |
| **4** | spam | SIX chances to win CASH! From 100 to 20,000 po... |
| **5** | spam | URGENT! You have won a 1 week FREE membership ... |
| **6** | spam | XXXMobileMovieClub: To use your credit, click ... |
| **7** | spam | England v Macedonia - dont miss the goals/team... |
| **8** | spam | 07732584351 - Rodger Burns - MSG = We tried to... |
| **9** | spam | SMS. ac Sptv: The New Jersey Devils and the De... |
| **10** | spam | Congrats! 1 year special cinema pass for 2 is ... |
| **11** | spam | As a valued customer, I am pleased to advise y... |
| **12** | spam | Did you hear about the new \Divorce Barbie\"? ... |
| **13** | spam | Please call our customer service representativ... |

True_news['label'] = 0

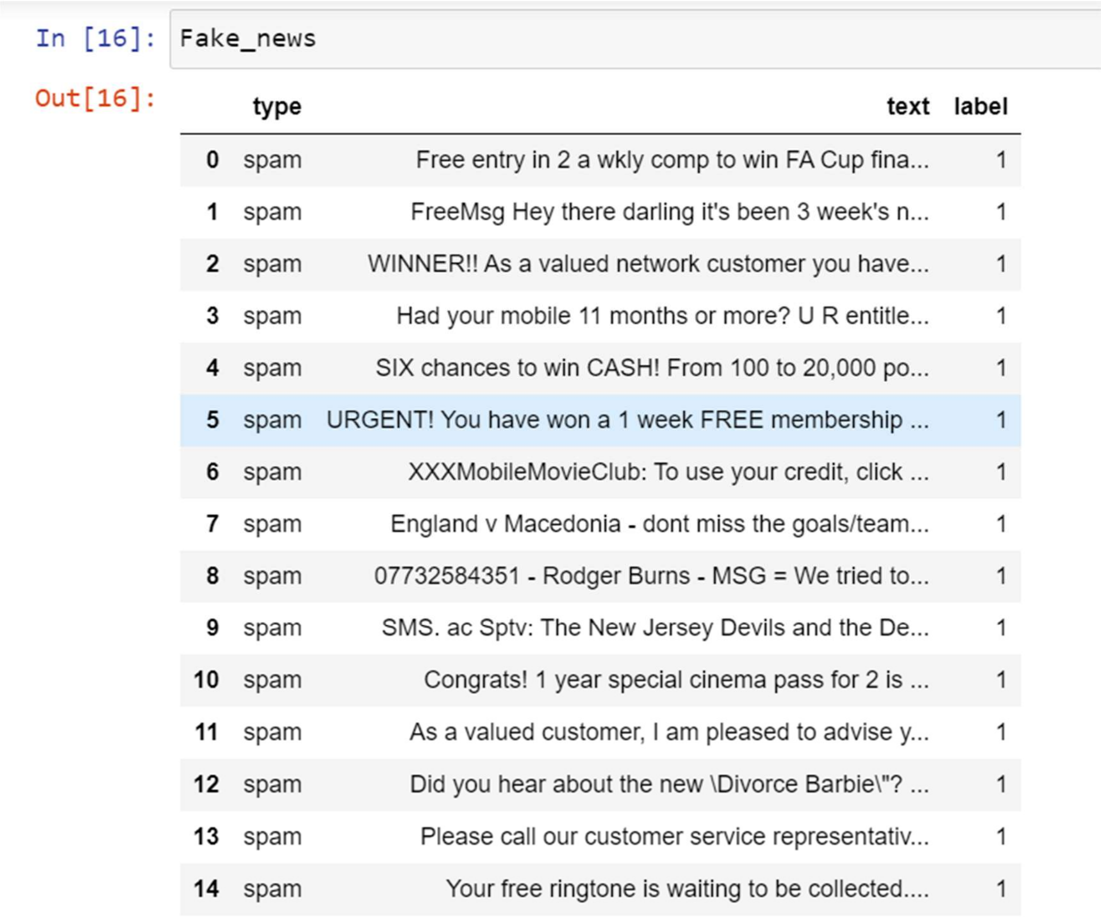
Fake_news['label'] = 1



```
In [15]: True_news
```

Out[15]:

| | type | text | label |
|---|---|---|---|
| **0** | ham | Go until jurong point, crazy.. Available only ... | 0 |
| **1** | ham | Ok lar... Joking wif u oni... | 0 |
| **2** | ham | U dun say so early hor... U c already then say... | 0 |
| **3** | ham | Nah I don't think he goes to usf, he lives aro... | 0 |
| **4** | ham | Even my brother is not like to speak with me. ... | 0 |
| **5** | ham | As per your request 'Melle Melle (Oru Minnamin... | 0 |
| **6** | ham | I'm gonna be home soon and i don't want to tal... | 0 |
| **7** | ham | I've been searching for the right words to tha... | 0 |
| **8** | ham | I HAVE A DATE ON SUNDAY WITH WILL!! | 0 |
| **9** | ham | Oh k...i'm watching here:) | 0 |
| **10** | ham | Eh u remember how 2 spell his name... Yes i di... | 0 |
| **11** | ham | Fine if thatåÕs the way u feel. ThatåÕs the wa... | 0 |
| **12** | ham | Is that seriously how you spell his name? | 0 |
| **13** | ham | I‰Û÷m going to try for 2 months ha ha only joking | 0 |
| **14** | ham | So Ì_ pay first lar... Then when is da stock c... | 0 |

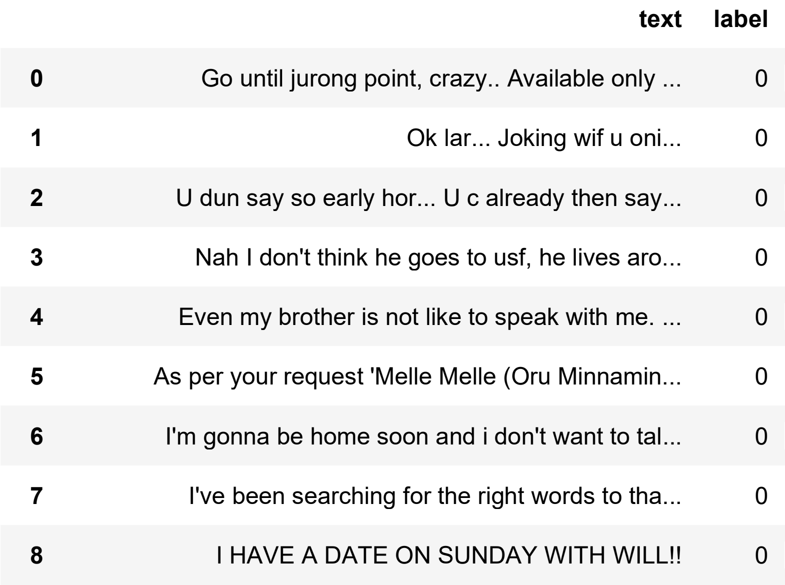


```
In [16]: Fake_news
```

Out[16]:

|  | type | text | label |
|---|---|---|---|
| **0** | spam | Free entry in 2 a wkly comp to win FA Cup fina... | 1 |
| **1** | spam | FreeMsg Hey there darling it's been 3 week's n... | 1 |
| **2** | spam | WINNER!! As a valued network customer you have... | 1 |
| **3** | spam | Had your mobile 11 months or more? U R entitle... | 1 |
| **4** | spam | SIX chances to win CASH! From 100 to 20,000 po... | 1 |
| **5** | spam | URGENT! You have won a 1 week FREE membership ... | 1 |
| **6** | spam | XXXMobileMovieClub: To use your credit, click ... | 1 |
| **7** | spam | England v Macedonia - dont miss the goals/team... | 1 |
| **8** | spam | 07732584351 - Rodger Burns - MSG = We tried to... | 1 |
| **9** | spam | SMS. ac Sptv: The New Jersey Devils and the De... | 1 |
| **10** | spam | Congrats! 1 year special cinema pass for 2 is ... | 1 |
| **11** | spam | As a valued customer, I am pleased to advise y... | 1 |
| **12** | spam | Did you hear about the new \Divorce Barbie\"? ... | 1 |
| **13** | spam | Please call our customer service representativ... | 1 |
| **14** | spam | Your free ringtone is waiting to be collected.... | 1 |

```
dataset1 = True_news[['text','label']]

dataset2 = Fake_news[['text','label']]

dataset = pd.concat([dataset1,dataset2])

dataset
```



|  | text | label |
|---|---|---|
| **0** | Go until jurong point, crazy.. Available only ... | 0 |
| **1** | Ok lar... Joking wif u oni... | 0 |
| **2** | U dun say so early hor... U c already then say... | 0 |
| **3** | Nah I don't think he goes to usf, he lives aro... | 0 |
| **4** | Even my brother is not like to speak with me. ... | 0 |
| **5** | As per your request 'Melle Melle (Oru Minnamin... | 0 |
| **6** | I'm gonna be home soon and i don't want to tal... | 0 |
| **7** | I've been searching for the right words to tha... | 0 |
| **8** | I HAVE A DATE ON SUNDAY WITH WILL!! | 0 |

| | text | label |
|---|---|---|
| **9** | Oh k...i'm watching here:) | 0 |
| **10** | Eh u remember how 2 spell his name... Yes i di... | 0 |
| **11** | Fine if thatåÕs the way u feel. ThatåÕs the wa... | 0 |
| **12** | Is that seriously how you spell his name? | 0 |
| **13** | I‰Û÷m going to try for 2 months ha ha only joking | 0 |
| **14** | So Ì_ pay first lar... Then when is da stock c... | 0 |
| **0** | Free entry in 2 a wkly comp to win FA Cup fina... | 1 |
| **1** | FreeMsg Hey there darling it's been 3 week's n... | 1 |
| **2** | WINNER!! As a valued network customer you have... | 1 |
| **3** | Had your mobile 11 months or more? U R entitle... | 1 |
| **4** | SIX chances to win CASH! From 100 to 20,000 po... | 1 |
| **5** | URGENT! You have won a 1 week FREE membership ... | 1 |
| **6** | XXXMobileMovieClub: To use your credit, click ... | 1 |
| **7** | England v Macedonia - dont miss the goals/team... | 1 |
| **8** | 07732584351 - Rodger Burns - MSG = We tried to... | 1 |
| **9** | SMS. ac Sptv: The New Jersey Devils and the De... | 1 |
| **10** | Congrats! 1 year special cinema pass for 2 is ... | 1 |
| **11** | As a valued customer, I am pleased to advise y... | 1 |
| **12** | Did you hear about the new \Divorce Barbie\"? ... | 1 |
| **13** | Please call our customer service representativ... | 1 |
| **14** | Your free ringtone is waiting to be collected.... | 1 |

```
dataset.shape
```

```
(30, 2)
```

```
## check for null values

dataset.isnull().sum()
```

```
text     0
label    0
dtype: int64
```

```
dataset['label'].value_counts()
```

```
0    15
1    15
Name: label, dtype: int64
```

```
dataset = dataset.sample(frac=1)

dataset
```

| | text | label |
|---|---|---|
| **9** | smsacsptvthenewjerseydevilsandthedetroitredwin... | 1 |
| **12** | isthatseriouslyhowyouspellhisname | 0 |
| **9** | ohkimwatchinghere | 0 |
| **6** | xxxmobilemovieclubtouseyourcreditclickthewapli... | 1 |
| **4** | sixchancestowincashfromtopoundstxtcshandsendto... | 1 |
| **12** | didyouhearaboutthenewdivorcebarbieitcomeswitha... | 1 |
| **4** | evenmybrotherisnotliketospeakwithmetheytreatme... | 0 |
| **7** | englandvmacedonia-dontmissthegoalsteamnewstxtu... | 1 |
| **0** | gountiljurongpointcrazyavailableonlyinbugisngr... | 0 |
| **10** | ehurememberhowspellhisnameyesididhevnaughtymak... | 0 |
| **2** | udunsaysoearlyhorucalreadythensay | 0 |
| **11** | fineifthatsthewayufeelthatsthewayitsgotab | 0 |
| **2** | winnerasavaluednetworkcustomeryouhavebeenselec... | 1 |
| **8** | ihaveadateonsundaywithwill | 0 |
| **1** | oklarjokingwifuoni | 0 |
| **3** | hadyourmobilemonthsormoreurentitledtoupdatetot... | 1 |
| **1** | freemsgheytheredarlingitsbeenweeksnowandnoword... | 1 |

|  | text | label |
|---|---|---|
| **14** | yourfreeringtoneiswaitingtobecollectedsimplyte... | 1 |
| **13** | pleasecallourcustomerservicerepresentativeonbe... | 1 |
| **8** | -rodgerburns-msgwetriedtocallyoureyourreplytoo... | 1 |
| **0** | freeentryinawklycomptowinfacupfinaltktsstmayte... | 1 |
| **5** | asperyourrequestmellemelleoruminnaminungintenu... | 0 |
| **14** | sopayfirstlarthenwhenisdastockcomin | 0 |
| **11** | asavaluedcustomeriampleasedtoadviseyouthatfoll... | 1 |
| **6** | imgonnabehomesoonandidontwanttotalkaboutthisst... | 0 |
| **3** | nahidontthinkhegoestousfhelivesaroundherethough | 0 |
| **7** | ivebeensearchingfortherightwordstothankyoufort... | 0 |
| **13** | imgoingtotryformonthshahaonlyjoking | 0 |
| **5** | urgentyouhavewonaweekfreemembershipinourprizej... | 1 |
| **10** | congratsyearspecialcinemapassforisyourscallnow... | 1 |

```
#NLP PART

import nltk

import re #rejects:works on fuzzy logic is used to clean and perform some operations on data

from nltk.corpus import stopwords

from nltk.stem import WordNetLemmatizer

import nltk

nltk.download('wordnet')
```

```
[nltk_data] Downloading package wordnet to C:\Users\BHAVYA
[nltk_data]     SRI\AppData\Roaming\nltk_data...
```

```
[nltk_data]   Package wordnet is already up-to-date!
```

Out[929]:

```
True
```

```
import nltk

nltk.download('stopwords')
```

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

Out[930]:

```
True
```

```
ps = WordNetLemmatizer()

nltk.download('wordnet')
```

```
[nltk_data] Downloading package wordnet to C:\Users\BHAVYA
[nltk_data]     SRI\AppData\Roaming\nltk_data...
[nltk_data]   Package wordnet is already up-to-date!
```

Out[938]:

```
True
```

```
from nltk.corpus import stopwords

stopwords = set(stopwords.words('english'))



def clean_row(row):

   row = re.sub('[^-a-zA-Z]', '', row)  # removes numbers and special symbols

   token = row.split()

   news = [pslemmatize(word) for word in token if not word in stopwords]

   cleanned_news = ' '.join(news)

   return cleanned_news

dataset['text']
```

```
9     smsacsptvthenewjerseydevilsandthedetroitredwin...
12                         isthatseriouslyhowyouspellhisname
9                                      ohkimwatchinghere
6     xxxmobilemovieclubtouseyourcreditclickthewapli...
4     sixchancestowincashfromtopoundstxtcshandsendto...
12    didyouhearaboutthenewdivorcebarbieitcomeswitha...
4     evenmybrotherisnotliketospeakwithmetheytreatme...
7     englandvmacedonia-dontmissthegoalsteamnewstxtu...
0     gountiljurongpointcrazyavailableonlyinbugisngr...
10    ehurememberhowspellhisnameyesididhevnaughtymak...
2                      udunsaysoearlyhorucalreadythensay
11             fineifthatsthewayufeelthatsthewayitsgotab
2     winnerasavaluednetworkcustomeryouhavebeenselec...
8                             ihaveadateonsundaywithwill
```

```
1                                    oklarjokingwifuoni
3     hadyourmobilemonthsormoreurentitledtoupdatetot...
1     freemsgheytheredarlingitsbeenweeksnowandnoword...
14    yourfreeringtoneiswaitingtobecollectedsimplyte...
13    pleasecallourcustomerservicerepresentativeonbe...
8     -rodgerburns-msgwetriedtocallyoureyourreplytoo...
0     freeentryinawklycomptowinfacupfinaltktsstmayte...
5     asperyourrequestmellemelleoruminnaminungintenu...
14                  sopayfirstlarthenwhenisdastockcomin
11    asavaluedcustomeriampleasedtoadviseyouthatfoll...
6     imgonnabehomesoonandidontwanttotalkaboutthisst...
3       nahidontthinkhegoestousfhelivesaroundherethough
7     ivebeensearchingfortherightwordstothankyoufort...
13                  imgoingtotryformonthshahaonlyjoking
5     urgentyouhavewonaweekfreemembershipinourprizej...
10    congratsyearspecialcinemapassforisyourscallnow...
Name: text, dtype: object
```

```
import re

from nltk.corpus import stopwords

from nltk.stem import WordNetLemmatizer


# Specify the NLTK data directory where you have downloaded the English stopwords

nltk.data.path.append(r"C:\Users\BHAVYA SRI\AppData\Roaming\nltk_data")


# Download stopwords if you haven't already

nltk.download('stopwords')


stopwords = set(stopwords.words('english'))


def clean_row(row):


   row = re.sub('[^-a-zA-Z]', '', row)  # removes numbers and special symbols

   token = row.split()

   # Initialize a simple lemmatizer

   simple_lemmatizer = WordNetLemmatizer()

   news = [simple_lemmatizer.lemmatize(word) for word in token if not word in stopwords]

   cleanned_news = ' '.join(news)

   return cleanned_news
```

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

```
import nltk

nltk.download('omw-1.4')
```

```
 [nltk_data] Downloading package omw-1.4 to C:\Users\BHAVYA
[nltk_data]     SRI\AppData\Roaming\nltk_data...
[nltk_data]   Package omw-1.4 is already up-to-date!
                                                              Out[1004]:True
```

```
dataset['text'] = dataset['text'].apply(lambda x : clean_row(x))
dataset['text']
```

```
9     smsacsptvthenewjerseydevilsandthedetroitredwin...
12                        isthatseriouslyhowyouspellhisname
9                                        ohkimwatchinghere
6     xxxmobilemovieclubtouseyourcreditclickthewapli...
4     sixchancestowincashfromtopoundstxtcshandsendto...
12    didyouhearaboutthenewdivorcebarbieitcomeswitha...
4     evenmybrotherisnotliketospeakwithmetheytreatme...
7     englandvmacedonia-dontmissthegoalsteamnewstxtu...
0     gountiljurongpointcrazyavailableonlyinbugisngr...
10    ehurememberhowspellhisnameyesididhevnaughtymak...
2                         udunsaysoearlyhorucalreadythensay
11              fineifthatsthewayufeelthatsthewayitsgotab
2     winnerasavaluednetworkcustomeryouhavebeenselec...
8                                ihaveadateonsundaywithwill
1                                        oklarjokingwifuoni
3     hadyourmobilemonthsormoreurentitledtoupdatetot...
1     freemsgheytheredarlingitsbeenweeksnowandnoword...
14    yourfreeringtoneiswaitingtobecollectedsimplyte...
13    pleasecallourcustomerservicerepresentativeonbe...
8     -rodgerburns-msgwetriedtocallyoureyourreplytoo...
0     freeentryinawklycomptowinfacupfinaltktsstmayte...
5     asperyourrequestmellemelleoruminnaminungintenu...
14                      sopayfirstlarthenwhenisdastockcomin
11    asavaluedcustomeriampleasedtoadviseyouthatfoll...
6     imgonnabehomesoonandidontwanttotalkaboutthisst...
3       nahidontthinkhegoestousfhelivesaroundherethough
7     ivebeensearchingfortherightwordstothankyoufort...
13                      imgoingtotryformonthshahaonlyjoking
5     urgentyouhavewonaweekfreemembershipinourprizej...
10    congratsyearspecialcinemapassforisyourscallnow...
Name: text, dtype: object
```

```
from sklearn.feature_extraction.text import TfidfVectorizer

vectorizer = TfidfVectorizer(max_features = 50,lowercase = False, ngram_range=(1,2))

X = dataset.iloc[:30,0]

Y = dataset.iloc[:30,1]

X
```

```
9     smsacsptvthenewjerseydevilsandthedetroitredwin...
12                        isthatseriouslyhowyouspellhisname
9                                        ohkimwatchinghere
```

```
6     xxxmobilemovieclubtouseyourcreditclickthewapli...
4     sixchancestowincashfromtopoundstxtcshandsendto...
12    didyouhearaboutthenewdivorcebarbieitcomeswitha...
4     evenmybrotherisnotliketospeakwithmetheytreatme...
7     englandvmacedonia-dontmissthegoalsteamnewstxtu...
0     gountiljurongpointcrazyavailableonlyinbugisngr...
10    ehurememberhowspellhisnameyesididhevnaughtymak...
2                      udunsaysoearlyhorucalreadythensay
11            fineifthatsthewayufeelthatsthewayitsgotab
2     winnerasavaluednetworkcustomeryouhavebeenselec...
8                            ihaveadateonsundaywithwill
1                                    oklarjokingwifuoni
3     hadyourmobilemonthsormoreurentitledtoupdatetot...
1     freemsgheytheredarlingitsbeenweeksnowandnoword...
14    yourfreeringtoneiswaitingtobecollectedsimplyte...
13    pleasecallourcustomerservicerepresentativeonbe...
8     -rodgerburns-msgwetriedtocallyoureyourreplytoo...
0     freeentryinawklycomptowinfacupfinaltktsstmayte...
5     asperyourrequestmellemelleoruminnaminungintenu...
14                  sopayfirstlarthenwhenisdastockcomin
11    asavaluedcustomeriampleasedtoadviseyouthatfoll...
6     imgonnabehomesoonandidontwanttotalkaboutthisst...
3       nahidontthinkhegoestousfhelivesaroundherethough
7     ivebeensearchingfortherightwordstothankyoufort...
13                  imgoingtotryformonthshahaonlyjoking
5     urgentyouhavewonaweekfreemembershipinourprizej...
10    congratsyearspecialcinemapassforisyourscallnow...
Name: text, dtype: object
```

Y

```
9     1
12    0
9     0
6     1
4     1
12    1
4     0
7     1
0     0
10    0
2     0
11    0
2     1
8     0
1     0
3     1
1     1
14    1
13    1
8     1
0     1
5     0
14    0
11    1
6     0
3     0
7     0
13    0
```

```
5     1
10    1
Name: label, dtype: int64
```

```
from sklearn.model_selection import train_test_split

train_data,test_data,train_label,test_label = train_test_split(X,Y,test_size=0.45,random_state=3)

train_data
```

```
4     sixchancestowincashfromtopoundstxtcshandsendto...
11    asavaluedcustomeriampleasedtoadviseyouthatfoll...
4     evenmybrotherisnotliketospeakwithmetheytreatme...
0     freeentryinawklycomptowinfacupfinaltktsstmayte...
10    ehurememberhowspellhisnameyesididhevnaughtymak...
11            fineifthatsthewayufeelthatsthewayitsgotab
10    congratsyearspecialcinemapassforisyourscallnow...
8     -rodgerburns-msgwetriedtocallyoureyourreplytoo...
5     asperyourrequestmellemelleoruminnaminungintenu...
9     smsacsptvthenewjerseydevilsandthedetroitredwin...
0     gountiljurongpointcrazyavailableonlyinbugisngr...
5     urgentyouhavewonaweekfreemembershipinourprizej...
6     xxxmobilemovieclubtouseyourcreditclickthewapli...
3       nahidontthinkhegoestousfhelivesaroundherethough
6     imgonnabehomesoonandidontwanttotalkaboutthisst...
2                     udunsaysoearlyhorucalreadythensay
Name: text, dtype: object
```

```
vec_train_data = vectorizer.fit_transform(train_data)

train_data
```

```
4     sixchancestowincashfromtopoundstxtcshandsendto...
11    asavaluedcustomeriampleasedtoadviseyouthatfoll...
4     evenmybrotherisnotliketospeakwithmetheytreatme...
0     freeentryinawklycomptowinfacupfinaltktsstmayte...
10    ehurememberhowspellhisnameyesididhevnaughtymak...
11            fineifthatsthewayufeelthatsthewayitsgotab
10    congratsyearspecialcinemapassforisyourscallnow...
8     -rodgerburns-msgwetriedtocallyoureyourreplytoo...
5     asperyourrequestmellemelleoruminnaminungintenu...
9     smsacsptvthenewjerseydevilsandthedetroitredwin...
0     gountiljurongpointcrazyavailableonlyinbugisngr...
5     urgentyouhavewonaweekfreemembershipinourprizej...
6     xxxmobilemovieclubtouseyourcreditclickthewapli...
3       nahidontthinkhegoestousfhelivesaroundherethough
6     imgonnabehomesoonandidontwanttotalkaboutthisst...
2                     udunsaysoearlyhorucalreadythensay
Name: text, dtype: object
```

```
vec_train_data = vectorizer.fit_transform(train_data)

vec_train_data = vec_train_data.toarray()

type(vec_train_data)
```

```
numpy.ndarray
```

```
vec_test_data = vectorizer.fit_transform(test_data)

vec_test_data = vec_test_data.toarray()
```

vec_train_data.shape,vec_test_data.shape

```
((16, 19), (14, 16))
```

vec_train_data

```
array([[0.        , 0.        , 0.        , 0.        , 0.        ,
         0.        , 0.        , 0.        , 0.        , 0.        ,
         0.        , 0.        , 0.        , 1.        , 0.        ,
         0.        , 0.        , 0.        , 0.        ],
        [1.        , 0.        , 0.        , 0.        , 0.        ,
         0.        , 0.        , 0.        , 0.        , 0.        ,
         0.        , 0.        , 0.        , 0.        , 0.        ,
         0.        , 0.        , 0.        , 0.        ],
        [0.        , 0.        , 0.        , 0.        , 1.        ,
         0.        , 0.        , 0.        , 0.        , 0.        ,
         0.        , 0.        , 0.        , 0.        , 0.        ,
         0.        , 0.        , 0.        , 0.        ],
        [0.        , 0.        , 0.        , 0.        , 0.        ,
         0.        , 1.        , 0.        , 0.        , 0.        ,
         0.        , 0.        , 0.        , 0.        , 0.        ,
         0.        , 0.        , 0.        , 0.        ],
        [0.        , 0.        , 0.        , 1.        , 0.        ,
         0.        , 0.        , 0.        , 0.        , 0.        ,
         0.        , 0.        , 0.        , 0.        , 0.        ,
         0.        , 0.        , 0.        , 0.        ],
        [0.        , 0.        , 0.        , 0.        , 0.        ,
         1.        , 0.        , 0.        , 0.        , 0.        ,
         0.        , 0.        , 0.        , 0.        , 0.        ,
         0.        , 0.        , 0.        , 0.        ],
        [0.        , 0.        , 0.57735027, 0.        , 0.        ,
         0.        , 0.        , 0.        , 0.        , 0.57735027,
         0.        , 0.        , 0.        , 0.        , 0.        ,
         0.        , 0.        , 0.57735027, 0.        ],
        [0.        , 0.        , 0.        , 0.        , 0.        ,
         0.        , 0.        , 0.        , 0.        , 0.        ,
         0.70710678, 0.        , 0.70710678, 0.        , 0.        ,
         0.        , 0.        , 0.        , 0.        ],
        [0.        , 1.        , 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.        , 1.        ,
         0.        , 0.        , 0.        , 0.        ],
        [0.        , 0.        , 0.        , 0.        , 0.        ,
         0.        , 0.        , 1.        , 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.        , 1.        , 0.        , 0.        ],
        [0.        , 0.        , 0.        , 0.        , 0.        ,
         0.        , 0.        , 0.        , 0.        , 0.        ,
         0.        , 0.        , 0.        , 0.        , 0.        ,
         0.        , 0.        , 0.        , 1.        ],
        [0.        , 0.        , 0.        , 0.        , 0.        ,
         0.        , 0.        , 0.        , 0.        , 0.        ,
         0.        , 1.        , 0.        , 0.        , 0.        ,
         0.        , 0.        , 0.        , 0.        ],
        [0.        , 0.        , 0.        , 0.        , 0.        ,
         0.        , 0.        , 0.        , 1.        , 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.        ,
         1.        , 0.        , 0.        , 0.        ]])
```

vec_test_data

```
array([[0.        , 0.        , 0.        , 0.        , 1.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        ],
       [1.        , 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.        , 1.        , 0.        ,
        0.        ],
       [0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 1.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        ],
       [0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.70710678, 0.70710678, 0.        , 0.        ,
        0.        ],
       [0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        1.        , 0.        , 0.        , 0.        , 0.        ,
        0.        ],
       [0.        , 0.        , 0.        , 0.        , 0.        ,
        1.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        ],
       [0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 1.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        ],
       [0.        , 0.        , 0.        , 1.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        ],
       [0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 1.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        ],
       [0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        1.        ],
       [0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 1.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        ],
       [0.        , 0.70710678, 0.70710678, 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.        , 1.        ,
        0.        ]])
```

training_data = pd.DataFrame(vec_train_data)

testing_data = pd.DataFrame(vec_test_data)

training_data

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 0.0 | 0.0 | 0.00000 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.00000 | 0.000000 | 0.0 | 0.000000 | 1.0 | 0.0 | 0.0 | 0.0 | 0.00000 | 0.0 |
| **1** | 1.0 | 0.0 | 0.00000 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.00000 | 0.000000 | 0.0 | 0.000000 | 0.0 | 0.0 | 0.0 | 0.0 | 0.00000 | 0.0 |
| **2** | 0.0 | 0.0 | 0.00000 | 0.0 | 1.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.00000 | 0.000000 | 0.0 | 0.000000 | 0.0 | 0.0 | 0.0 | 0.0 | 0.00000 | 0.0 |
| **3** | 0.0 | 0.0 | 0.00000 | 0.0 | 0.0 | 0.0 | 1.0 | 0.0 | 0.0 | 0.00000 | 0.000000 | 0.0 | 0.000000 | 0.0 | 0.0 | 0.0 | 0.0 | 0.00000 | 0.0 |
| **4** | 0.0 | 0.0 | 0.00000 | 1.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.00000 | 0.000000 | 0.0 | 0.000000 | 0.0 | 0.0 | 0.0 | 0.0 | 0.00000 | 0.0 |
| **5** | 0.0 | 0.0 | 0.00000 | 0.0 | 0.0 | 1.0 | 0.0 | 0.0 | 0.0 | 0.00000 | 0.000000 | 0.0 | 0.000000 | 0.0 | 0.0 | 0.0 | 0.0 | 0.00000 | 0.0 |
| **6** | 0.0 | 0.0 | 0.57735 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.57735 | 0.000000 | 0.0 | 0.000000 | 0.0 | 0.0 | 0.0 | 0.0 | 0.57735 | 0.0 |
| **7** | 0.0 | 0.0 | 0.00000 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.00000 | 0.707107 | 0.0 | 0.707107 | 0.0 | 0.0 | 0.0 | 0.0 | 0.00000 | 0.0 |

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **8** | 0.0 | 1.0 | 0.00000 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.00000 | 0.000000 | 0.0 | 0.000000 | 0.0 | 0.0 | 0.0 | 0.0 | 0.00000 | 0.0 |
| **9** | 0.0 | 0.0 | 0.00000 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.00000 | 0.000000 | 0.0 | 0.000000 | 0.0 | 1.0 | 0.0 | 0.0 | 0.00000 | 0.0 |
| **10** | 0.0 | 0.0 | 0.00000 | 0.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.0 | 0.00000 | 0.000000 | 0.0 | 0.000000 | 0.0 | 0.0 | 0.0 | 0.0 | 0.00000 | 0.0 |
| **11** | 0.0 | 0.0 | 0.00000 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.00000 | 0.000000 | 0.0 | 0.000000 | 0.0 | 0.0 | 0.0 | 1.0 | 0.00000 | 0.0 |
| **12** | 0.0 | 0.0 | 0.00000 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.00000 | 0.000000 | 0.0 | 0.000000 | 0.0 | 0.0 | 0.0 | 0.0 | 0.00000 | 1.0 |
| **13** | 0.0 | 0.0 | 0.00000 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.00000 | 0.000000 | 1.0 | 0.000000 | 0.0 | 0.0 | 0.0 | 0.0 | 0.00000 | 0.0 |
| **14** | 0.0 | 0.0 | 0.00000 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.00000 | 0.000000 | 0.0 | 0.000000 | 0.0 | 0.0 | 0.0 | 0.0 | 0.00000 | 0.0 |
| **15** | 0.0 | 0.0 | 0.00000 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.00000 | 0.000000 | 0.0 | 0.000000 | 0.0 | 0.0 | 1.0 | 0.0 | 0.00000 | 0.0 |

testing_data

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 0.0 | 0.000000 | 0.000000 | 0.0 | 1.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.000000 | 0.000000 | 0.0 | 0.0 | 0.0 |
| **1** | 1.0 | 0.000000 | 0.000000 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.000000 | 0.000000 | 0.0 | 0.0 | 0.0 |
| **2** | 0.0 | 0.000000 | 0.000000 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.000000 | 0.000000 | 1.0 | 0.0 | 0.0 |

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 3 | 0.0 | 0.000000 | 0.000000 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.0 | 0.0 | 0.000000 | 0.000000 | 0.0 | 0.0 | 0.0 |
| 4 | 0.0 | 0.000000 | 0.000000 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.707107 | 0.707107 | 0.0 | 0.0 | 0.0 |
| 5 | 0.0 | 0.000000 | 0.000000 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.000000 | 0.000000 | 0.0 | 0.0 | 0.0 |
| 6 | 0.0 | 0.000000 | 0.000000 | 0.0 | 0.0 | 1.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.000000 | 0.000000 | 0.0 | 0.0 | 0.0 |
| 7 | 0.0 | 0.000000 | 0.000000 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.0 | 0.000000 | 0.000000 | 0.0 | 0.0 | 0.0 |
| 8 | 0.0 | 0.000000 | 0.000000 | 1.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.000000 | 0.000000 | 0.0 | 0.0 | 0.0 |
| 9 | 0.0 | 0.000000 | 0.000000 | 0.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.0 | 0.0 | 0.0 | 0.000000 | 0.000000 | 0.0 | 0.0 | 0.0 |
| 10 | 0.0 | 0.000000 | 0.000000 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.000000 | 0.000000 | 0.0 | 0.0 | 1.0 |
| 11 | 0.0 | 0.000000 | 0.000000 | 0.0 | 0.0 | 0.0 | 1.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.000000 | 0.000000 | 0.0 | 0.0 | 0.0 |
| 12 | 0.0 | 0.707107 | 0.707107 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.000000 | 0.000000 | 0.0 | 0.0 | 0.0 |
| 13 | 0.0 | 0.000000 | 0.000000 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.000000 | 0.000000 | 0.0 | 1.0 | 0.0 |

```
#MODEL

from sklearn.naive_bayes import MultinomialNB

clf = MultinomialNB

import numpy as np

from sklearn.preprocessing import OneHotEncoder
```

```
one_hot_encoder = OneHotEncoder()
train_label_array = np.array(train_label).reshape(1,-1)
train_label_encoded = one_hot_encoder.fit_transform(train_label_array)
```

```
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.naive_bayes import MultinomialNB
# Create a TF-IDF vectorizer
vectorizer = TfidfVectorizer()
# Vectorize the text data
train_data_vectorized = vectorizer.fit_transform(training_data)
# Create the Multinomial Naive Bayes classifier
clf = MultinomialNB()
# Fit the classifier to the vectorized data
clf.fit(train_data_vectorized, train_label)
```

```
test_label
```

```
3      1
12     1
14     0
7      0
13     1
1      0
8      0
9      0
1      1
12     0
14     1
13     0
7      1
2      1
Name: label, dtype: int64
```

```
y
```

```
9      1
12     0
9      0
6      1
4      1
12     1
4      0
7      1
0      0
```

```
10    0
2     0
11    0
2     1
8     0
Name: label, dtype: int64
```

```
from sklearn.metrics import accuracy_score


# Ensure y has the same length as test_label

y = Y[:len(test_label)]

print("Length of y:", len(y))

print("Length of test_label:", len(test_label))
```

```
Length of y: 14
Length of test_label: 14
```

```
accuracy_score(test_label, y)
```

```
0.5
```

```
y1 = Y[:len(train_label)]

print("Length of y1:", len(y1))

print("Length of train_label:", len(train_label))
```

```
Length of y1: 16
Length of train_label: 16
```

```
accuracy_score(train_label,y1)
```

```
0.5625
```

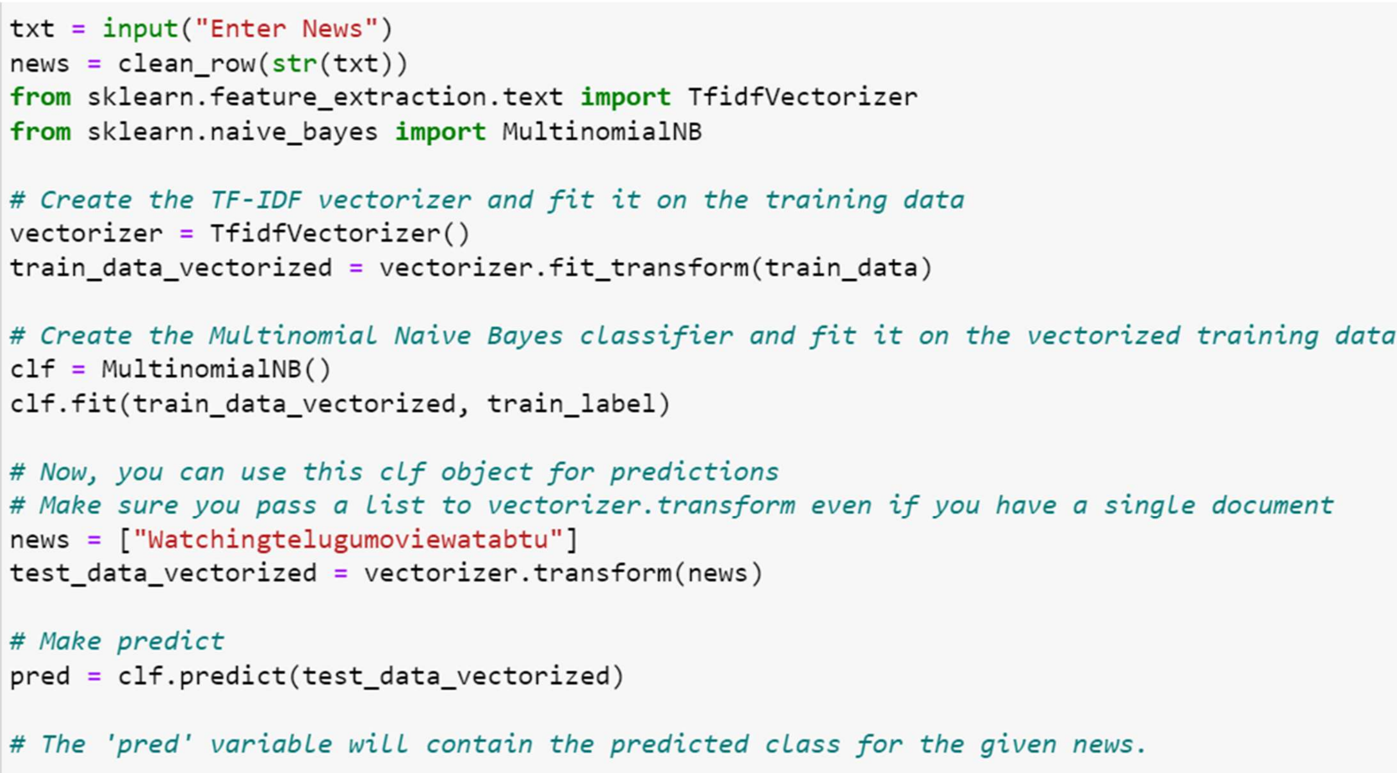
```
txt ='Watching telugu movie..wat abt u?'

news = clean_row(txt)

news
```

```
'Watchingtelugumoviewatabtu'
```

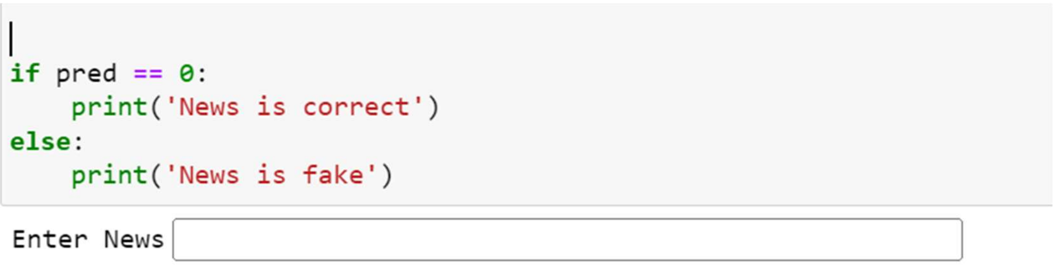
```
from sklearn.feature_extraction.text import TfidfVectorizer

from sklearn.naive_bayes import MultinomialNB

# Create the TF-IDF vectorizer and fit it on the training data

vectorizer = TfidfVectorizer()

train_data_vectorized = vectorizer.fit_transform(train_data)

# Create the Multinomial Naive Bayes classifier and fit it on the vectorized training data
```

```
clf = MultinomialNB()

clf.fit(train_data_vectorized, train_label)

# Now, you can use this clf object for predictions

# Make sure you pass a list to vectorizer.transform even if you have a single document

news = ["Watchingtelugumoviewatabtu"]

test_data_vectorized = vectorizer.transform(news)

# Make predictions

pred = clf.predict(test_data_vectorized)

# The 'pred' variable will contain the predicted class for the given news.

pred
```

```
array([0], dtype=int64)
```



```
txt = input("Enter News")
news = clean_row(str(txt))
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.naive_bayes import MultinomialNB

# Create the TF-IDF vectorizer and fit it on the training data
vectorizer = TfidfVectorizer()
train_data_vectorized = vectorizer.fit_transform(train_data)

# Create the Multinomial Naive Bayes classifier and fit it on the vectorized training data
clf = MultinomialNB()
clf.fit(train_data_vectorized, train_label)

# Now, you can use this clf object for predictions
# Make sure you pass a list to vectorizer.transform even if you have a single document
news = ["Watchingtelugumoviewatabtu"]
test_data_vectorized = vectorizer.transform(news)

# Make predict
pred = clf.predict(test_data_vectorized)

# The 'pred' variable will contain the predicted class for the given news.
```



```
if pred == 0:
    print('News is correct')
else:
    print('News is fake')
```

Enter News