```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

dataset = pd.read_csv('SMSSpamCollection', sep='\t', header=None)
dataset
```

```
         0                                                   1
0      ham  Go until jurong point, crazy.. Available only ...
1      ham                      Ok lar... Joking wif u oni...
2     spam  Free entry in 2 a wkly comp to win FA Cup fina...
3      ham  U dun say so early hor... U c already then say...
4      ham  Nah I don't think he goes to usf, he lives aro...
...    ...                                                 ...
5567  spam  This is the 2nd time we have tried 2 contact u...
5568   ham              Will ü b going to esplanade fr home?
5569   ham  Pity, * was in mood for that. So...any other s...
5570   ham  The guy did some bitching but I acted like i'd...
5571   ham                        Rofl. Its true to its name

[5572 rows x 2 columns]
```

```
dataset.columns
```

```
Index([0, 1], dtype='int64')
```

Changing the column name

```
dataset.rename(columns={0: 'result', 1: 'message'}, inplace=True)
dataset
```

```
     result                                            message
0       ham  Go until jurong point, crazy.. Available only ...
1       ham                      Ok lar... Joking wif u oni...
2      spam  Free entry in 2 a wkly comp to win FA Cup fina...
3       ham  U dun say so early hor... U c already then say...
4       ham  Nah I don't think he goes to usf, he lives aro...
...     ...                                               ...
5567   spam  This is the 2nd time we have tried 2 contact u...
5568    ham              Will ü b going to esplanade fr home?
5569    ham  Pity, * was in mood for that. So...any other s...
5570    ham  The guy did some bitching but I acted like i'd...
5571    ham                        Rofl. Its true to its name

[5572 rows x 2 columns]
```

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

```
result
ham     4825
spam     747
Name: count, dtype: int64
```

Feature Extraction

```
import re
from sklearn.feature_extraction.text import CountVectorizer

X = dataset['message']
y = dataset['result']

def remove_punct(X):
    for i in range(0, len(X)):
        X[i] = re.sub('[^a-zA-Z]', ' ', X[i])
        X[i] = X[i].lower()
    return X

X = remove_punct(X)
vectorizer = CountVectorizer(stop_words='english')
X = vectorizer.fit_transform(X)

from sklearn.preprocessing import LabelEncoder
le = LabelEncoder()

y = le.fit_transform(y)
y
```

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

```
from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.25)
```

Training the model

```
from sklearn.naive_bayes import MultinomialNB

model = MultinomialNB()

model.fit(X_train, y_train)
```

```
MultinomialNB()
```

```
y_preds = model.predict(X_test)
y_preds
```

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

Checking Performance

```
from sklearn.metrics import confusion_matrix, accuracy_score,
precision_score, recall_score

cm = confusion_matrix(y_test, y_preds)
print(cm)
```

```
[[1166   27]
 [  11  189]]
```

```
pre = precision_score(y_test, y_preds)
pre
```

```
0.875
```

```
recall = recall_score(y_test, y_preds)
recall
```

```
0.945
```

```
acc = accuracy_score(y_test, y_preds)
print(f'Accuracy: {acc}')
```

```
Accuracy: 0.9727207465900933
```

Inputs

Ham: Heyyy... Nice to see you

Spam: Free entry in 2 a wkly comp to win FA Cup

```
def predict(user_message, model, vectorizer):
    user_message = re.sub('^[a-zA-Z]', ' ', user_message)
    user_message = user_message.lower()
    user_message = [user_message]
    message = vectorizer.transform(user_message)
    pred = model.predict(message)
    if pred == 0:
        return "Ham"
    else:
        return"Spam"

user_message = input()
print(predict(user_message, model, vectorizer))
```

```
 Heyyy... Nice to see you
```

```
Ham
```