SMS SPAM DETECTION USING NLP

Import Necessary Libraries:

```
import pandas as pd
import numpy as np
import nltk
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.naive_bayes import MultinomialNB
from sklearn.metrics import accuracy_score
import matplotlib.pyplot as plt
```

Loading the dataset:

```
dataset = pd.read_csv("spam.csv", encoding = "latin-1")
```

Extract the labels i.e, (ham or spam):

```
dataset_labels = dataset['v1']
```

Dropping the unnecesary columns:

```
dataset = dataset.drop(columns = ['Unnamed: 2','Unnamed: 3', 'Unnamed: 4'])
```

Data exploration:

```
dataset.head(10)
```

	v1	v2
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...
5	spam	FreeMsg Hey there darling it's been 3 week's n...
6	ham	Even my brother is not like to speak with me. ...
7	ham	As per your request 'Melle Melle (Oru Minnamin...
8	spam	WINNER!! As a valued network customer you have...
9	spam	Had your mobile 11 months or more? U R entitle...

Next steps: Generate code with dataset View recommended plots

Spam and ham declarations:

```
df_ham = dataset[dataset.v1 == "ham"]
df_spam = dataset[dataset.v1 == 'spam']
```

Displaying Ham and Spam:

```
df_ham.head(10)
```

	v1	v2
0	ham	Go until jurong point, crazy.. Available only ...
1	ham	Ok lar... Joking wif u oni...
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...
6	ham	Even my brother is not like to speak with me. ...
7	ham	As per your request 'Melle Melle (Oru Minnamin...
10	ham	I'm gonna be home soon and i don't want to tal...
13	ham	I've been searching for the right words to tha...
14	ham	I HAVE A DATE ON SUNDAY WITH WILL!!
16	ham	Oh k...i'm watching here:)

Next steps: Generate code with `df_ham` View recommended plots

```
df_spam.head(10)
```

	v1	v2
2	spam	Free entry in 2 a wkly comp to win FA Cup fina...
5	spam	FreeMsg Hey there darling it's been 3 week's n...
8	spam	WINNER!! As a valued network customer you have...
9	spam	Had your mobile 11 months or more? U R entitle...
11	spam	SIX chances to win CASH! From 100 to 20,000 po...
12	spam	URGENT! You have won a 1 week FREE membership ...
15	spam	XXXMobileMovieClub: To use your credit, click ...
19	spam	England v Macedonia - dont miss the goals/team...
34	spam	Thanks for your subscription to Ringtone UK yo...
42	spam	07732584351 - Rodger Burns - MSG = We tried to...

Next steps: Generate code with `df_spam` View recommended plots

Split Data into Training and Testing Sets:

```
X_train, X_test, y_train, y_test = train_test_split(dataset, dataset_labels)
```

Feature Extraction:

extracting n-grams from the text data(bigrams)

```
vect = TfidfVectorizer(min_df=1, stop_words='english' , lowercase=True)
x_counts = vect.fit(X_train.v2)
```

preparing for training set

```
x_train_df = vect.transform(X_train.v2)
```

preparing for test set

```
x_test_df = vect.transform(X_test.v2) #creates a new feature representation
```

now applying machine learning model

```
clf = MultinomialNB()
```

Train the model:

```
clf.fit(x_train_df, y_train)
```

```
▾ MultinomialNB
MultinomialNB()
```

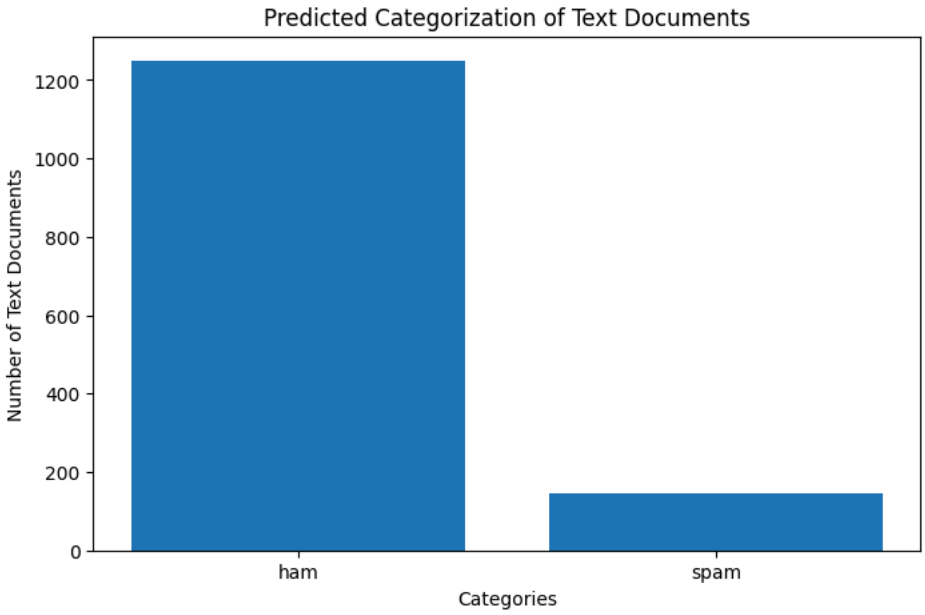
```
def visualize_predictions(text_data, predicted_labels):
    category_counts = predicted_labels.value_counts()
    plt.figure(figsize=(8, 5))
    plt.bar(category_counts.index, category_counts.values)
    plt.xlabel("Categories")
    plt.ylabel("Number of Text Documents")
    plt.title("Predicted Categorization of Text Documents")
    plt.show()
```

Predicting

```
predict = clf.predict(x_test_df)
predict = pd.Series(predict)
```

visualization

```
visualize_predictions(X_test.v2, predict)
```



Performance Evaluation

```
acc = accuracy_score(y_test, predict)
print(acc)
```

```
0.9676956209619526
```