

In [20]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from collections import defaultdict
```

In [21]:

```
# Load the Enron-Spam dataset
df = pd.read_csv("C:\\Users\\Abhishek Saini\\Desktop\\machin_learning_data\\enron_spam_d
df=df.dropna()
df = df[['v1', 'v2']] # Keep only the label ('v1') and message content ('v2')
print(df)
```

	v1	v2
14722	"" where i will	if given the chance
14820	"" where i will	if given the chance
15160	"" where i will	if given the chance

In [22]:

```
# Preprocess the data
df['v1'] = df['v1'].map({'spam': 1, 'ham': 0}) # Convert labels to binary (1 for spam,
```

In [23]:

```
# Split data into training and testing sets
train_size = int(len(df) * 0.8)
train_data = df[:train_size]
test_data = df[train_size:]
print("testing data=>",test_data)
print("training data",train_data)
```

testing data=>	v1	v2
15160 NaN	if given the chance	

training data	v1	v2
14722 NaN	if given the chance	
14820 NaN	if given the chance	

In [24]:

```
# Define a function to tokenize the text
def tokenize(text):
    tokens = text.split()
    return [token.lower() for token in tokens]
```

In [25]:

```
# Build a vocabulary
word_freq = defaultdict(int)
for _, message in train_data.iterrows():
    for word in tokenize(message['v2']):
        word_freq[word] += 1
```

In [26]:

```
# Define a function to extract features from messages
def extract_features(message):
    features = {}
    for word in tokenize(message):
        if word in word_freq:
            features[word] = 1
    return features
```

In [27]:

```
# Extract features for training
X_train = [extract_features(message) for message in train_data['v2']]
y_train = train_data['v1'].values
```

In [28]:

```
# Build a Naive Bayes classifier
class NaiveBayes:
    def __init__(self):
        self.class_prob = {}
        self.word_prob = {}

    def train(self, X, y):
        total_messages = len(y)
        spam_messages = sum(y)
        ham_messages = total_messages - spam_messages

        self.class_prob['spam'] = spam_messages / total_messages
        self.class_prob['ham'] = ham_messages / total_messages

        spam_word_count = defaultdict(int)
        ham_word_count = defaultdict(int)

        for i, message in enumerate(X):
            for word, _ in message.items():
                if y[i] == 1:
                    spam_word_count[word] += 1
                else:
                    ham_word_count[word] += 1

        total_spam_words = sum(spam_word_count.values())
        total_ham_words = sum(ham_word_count.values())

        self.word_prob['spam'] = {word: count / total_spam_words for word, count in spam_word_count.items()}
        self.word_prob['ham'] = {word: count / total_ham_words for word, count in ham_word_count.items()}

    def predict(self, X):
        predictions = []

        for message in X:
            spam_score = np.log(self.class_prob['spam'])
            ham_score = np.log(self.class_prob['ham'])

            for word in message:
                if word in self.word_prob['spam']:
                    spam_score += np.log(self.word_prob['spam'][word])
                if word in self.word_prob['ham']:
                    ham_score += np.log(self.word_prob['ham'][word])

            predictions.append(1 if spam_score > ham_score else 0)

        return np.array(predictions)
```

In [29]:

```
# Train the Naive Bayes classifier
nb_classifier = NaiveBayes()
nb_classifier.train(X_train, y_train)
```

In [30]:

```
# Define a function to predict if an email is spam or not
def predict_spam(message):
    features = extract_features(message)
    probabilities = nb_classifier.predict([features])[0]
    return "Probably a spam." if nb_classifier.predict([features])[0] == 1 else "Not a s
```

In [31]:

```
# Example usage
user_input = input("Insert an email: ")
result = predict_spam(user_input)
print(result)
```

Insert an email: it is my understanding the outages valero incurred at the above locations are force majeure situations during december and should not trigger the buyback feature of the contract . gary , i have not seen any documentation of this from valero . please follow up with your counterpart to secure the letter of notice or we will need to process the deal as if the buyback is active . thanks ken
Not a spam.

In []: