In [1]:
```
import pandas as pd
import numpy as np
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import MultinomialNB
data = pd.read_csv("language.csv")
print(data.head())
```

```
                                                Text  language
0  klement gottwaldi surnukeha palsameeriti ning ...  Estonian
1  sebes joseph pereira thomas  på eng the jesuit...   Swedish
2  ถนนเจริญกรุง อักษรโรมัน thanon charoen krung เ...      Thai
3  விசாகப்பட்டினம் தமிழ்ச்சங்கத்தை இந்துப் பத்திர...     Tamil
4  de spons behoort tot het geslacht haliclona en...     Dutch
```

from sklearn.feature_extraction.text import CountVectorizer ka matlab hai ki aap scikit-learn library ke feature_extraction.text module se CountVectorizer class ko import kar rahe ho

CountVectorizer kya karta hai? CountVectorizer ek tool hai jo text data ko numerical data mein convert karta hai. Yeh text ke har unique word ko count karta hai aur unhe ek matrix ke form mein store karta hai, jise bag-of-words (BoW) representation bhi kehte hain.

Bag of Words (BoW) ek fundamental technique hai jo text data ko numerical format mein convert karti hai, taaki usko machine learning models ke liye use kiya ja sake. BoW approach mein, ek text ko usme aaye unique words aur unke counts ke through represent kiya jata hai.

Is approach mein text ke grammar, word order ya context ka koi dhyaan nahi rakha jata, sirf yeh dekha jata hai ki kaunse words kitni baar aaye hain. Words ke count ko hi important feature mana jata hai.

Bag of Words ka Concept: Bag ka matlab hota hai ki words ko ek collection (ya bag) ke form mein dekha jata hai, bina kisi specific order ke. Words ka matlab hai text mein aaye words (terms).

In [15]:
```
from sklearn.feature_extraction.text import CountVectorizer

# Create a CountVectorizer object
vectorizer = CountVectorizer()

# Sample text data
data = ["love data science", "love machine learning"]

# Fit and transform the data
vectorized_data = vectorizer.fit_transform(data)

# Get the vocabulary (unique words)
print(vectorizer.get_feature_names_out())

# Convert the result to an array
print(vectorized_data.toarray())
```

```
['data' 'learning' 'love' 'machine' 'science']
[[1 0 1 0 1]
 [0 1 1 1 0]]
```

from sklearn.naive_bayes import MultinomialNB ka matlab hai ki aap scikit-learn library ke naive_bayes module se MultinomialNB class ko import kar rahe hain.

from sklearn.naive_bayes import MultinomialNB

Iska matlab hai ki aap scikit-learn library ke andar ek special algorithm Multinomial Naive Bayes (MultinomialNB) ko use karne ja rahe ho. Yeh ek tarah ka machine learning algorithm hai jo mainly text classification ke liye use hota hai, jaise:

Spam ya Not Spam emails ka classification. Positive ya Negative reviews ko alag karna.

In [16]:
```
from sklearn.naive_bayes import MultinomialNB
from sklearn.feature_extraction.text import CountVectorizer

# Sample emails
emails = ["I love learning", "Click here to win money", "Learning is fun", "Ge
labels = [0, 1, 0, 1]  # 0 means not spam, 1 means spam

# Step 1: Convert text into numbers (word counts)
vectorizer = CountVectorizer()
X = vectorizer.fit_transform(emails)

# Step 2: Train the MultinomialNB model
model = MultinomialNB()
model.fit(X, labels)

# Step 3: Predict if a new email is spam or not
new_email = ["Win money now!"]
new_email_vector = vectorizer.transform(new_email)
prediction = model.predict(new_email_vector)

print("Prediction:", prediction)  # 1 means spam
```

```
Prediction: [1]
```

In [ ]:
```

```

pandas as pd: Imports the pandas library, which is used for data manipulation and analysis, particularly for handling data in tables (DataFrames).

numpy as np: Imports numpy, which is a library for numerical computations. It's used for working with arrays and matrices.

CountVectorizer: This is a tool from the sklearn.feature_extraction.text module. It converts text data into a matrix of token counts (bag of words model).

train_test_split: A function from sklearn.model_selection that splits data into training and testing sets.

MultinomialNB: This is the Naive Bayes classifier for multinomially distributed data, often used for text classification tasks.

In [2]:
```
data.isnull().sum()
```

Out[2]:
```
Text        0
language    0
dtype: int64
```

In [3]:
```
data["language"].value_counts()
```

Out[3]:
```
Estonian      1000
Swedish       1000
English       1000
Russian       1000
Romanian      1000
Persian       1000
Pushto        1000
Spanish       1000
Hindi         1000
Korean        1000
Chinese       1000
French        1000
Portugese     1000
Indonesian    1000
Urdu          1000
Latin         1000
Turkish       1000
Japanese      1000
Dutch         1000
Tamil         1000
Thai          1000
Arabic        1000
Name: language, dtype: int64
```

In [4]:
```
x = np.array(data["Text"])
y = np.array(data["language"])

print(x)
```

```
['klement gottwaldi surnukeha palsameeriti ning paigutati mausoleumi surnukeh
a oli aga liiga hilja ja oskamatult palsameeritud ning hakkas ilmutama lagune
mise tundemärke  aastal viidi ta surnukeha mausoleumist ära ja kremeeriti zlí
ni linn kandis aastatel – nime gottwaldov ukrainas harkivi oblastis kandis zm
iivi linn aastatel – nime gotvald'
 'sebes joseph pereira thomas  på eng the jesuits and the sino-russian treaty
of nerchinsk  the diary of thomas pereira bibliotheca instituti historici s i
--   rome libris '
 'ถนนเจริญกรุง อักษรโรมัน thanon charoen krung เริ่มตั้งแต่ถนนสนามไชยถึงแม่น้ำเจ้าพระยาที่ถ
นนตก กรุงเทพมหานคร เป็นถนนรุ่นแรกที่ใช้เทคนิคการสร้างแบบตะวันตก ปัจจุบันผ่านพื้นที่เขตพระนคร
เขตป้อมปราบศัตรูพ่าย เขตสัมพันธวงศ์ เขตบางรัก เขตสาทร และเขตบางคอแหลม'
 ...
 'con motivo de la celebración del septuagésimoquinto º aniversario de la fun
dación del departamento en  guillermo ceballos espinosa presentó a la goberna
ción de caldas por encargo de su titular dilia estrada de gómez el himno que
fue adoptado para solemnizar dicha efemérides y que siguieron interpretando l
as bandas de música y los planteles de educación de esta sección del país en
retretas y actos oficiales con gran aceptación[]\u200b'
 '年月，當時還只有歲的她在美國出道，以mai-k名義推出首張英文《baby i like》，由美國
的獨立廠牌bip·record發行，以外國輸入盤的形式在日本發售，旋即被抢购一空。其後於月日
發行以倉木麻衣名義發行的首張日文單曲《love day after tomorrow》，正式於日本出道。
這張單曲初動銷量只得約萬張，可是其後每週銷量一直上升，並於年月正式突破百萬銷量，合计
万张。成為年最耀眼的新人歌手。'
 ' aprilie sonda spațială messenger a nasa și-a încheiat misiunea de studiu d
e  ani prăbușindu-se pe suprafața planetei mercur sonda a rămas fără combusti
bil fiind împinsă de gravitația solară din ce în ce mai aproape de mercur']
```

In [5]:
```
print(y)
```

```
['Estonian' 'Swedish' 'Thai' ... 'Spanish' 'Chinese' 'Romanian']
```

In [6]:
```
cv = CountVectorizer()
X = cv.fit_transform(x)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.33, rand
```

In [7]:
```
model = MultinomialNB()
model.fit(X_train,y_train)
model.score(X_test,y_test)
```

Out[7]:
```
0.953168044077135
```

In [11]:
```
user = input("Enter a Text: ")
data = cv.transform([user]).toarray()
output = model.predict(data)
print(output)
```

```
Enter a Text: 내 이름은 스와티입니다
['Korean']
```

In [ ]:

In [ ]:

In [ ]: