

```

!pip install pandas
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.naive_bayes import MultinomialNB
!pip install scikit-learn
from sklearn.model_selection import GridSearchCV

```

```

Requirement already satisfied: pandas in /usr/local/lib/python3.10/dist-packages (1.5.3)
Requirement already satisfied: python-dateutil>=2.8.1 in /usr/local/lib/python3.10/dist-
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.10/dist-packages (
Requirement already satisfied: numpy>=1.21.0 in /usr/local/lib/python3.10/dist-packages
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.10/dist-packages (from
Requirement already satisfied: scikit-learn in /usr/local/lib/python3.10/dist-packages (
Requirement already satisfied: numpy>=1.17.3 in /usr/local/lib/python3.10/dist-packages
Requirement already satisfied: scipy>=1.3.2 in /usr/local/lib/python3.10/dist-packages (
Requirement already satisfied: joblib>=1.1.1 in /usr/local/lib/python3.10/dist-packages
Requirement already satisfied: threadpoolctl>=2.0.0 in /usr/local/lib/python3.10/dist-pa

```

```
from google.colab import files
```

```
uploaded=files.upload()
```



imdb reviews.csv

- **imdb reviews.csv**(text/csv) - 52724248 bytes, last modified: 1/19/2022 - 100% done
Saving imdb reviews.csv to imdb reviews (1).csv

```

path="./imdb reviews (1).csv"
reviews = pd.read_csv(path)
reviews.head()

```

	text	label	
0	I grew up (b. 1965) watching and loving the Th...	0	
1	When I put this movie in my DVD player, and sa...	0	
2	Why do people who do not know what a particula...	0	
3	Even though I have great interest in Biblical ...	0	
4	Im a die hard Dads Army fan and nothing will e...	1	

Next steps:

[Generate code with reviews](#)

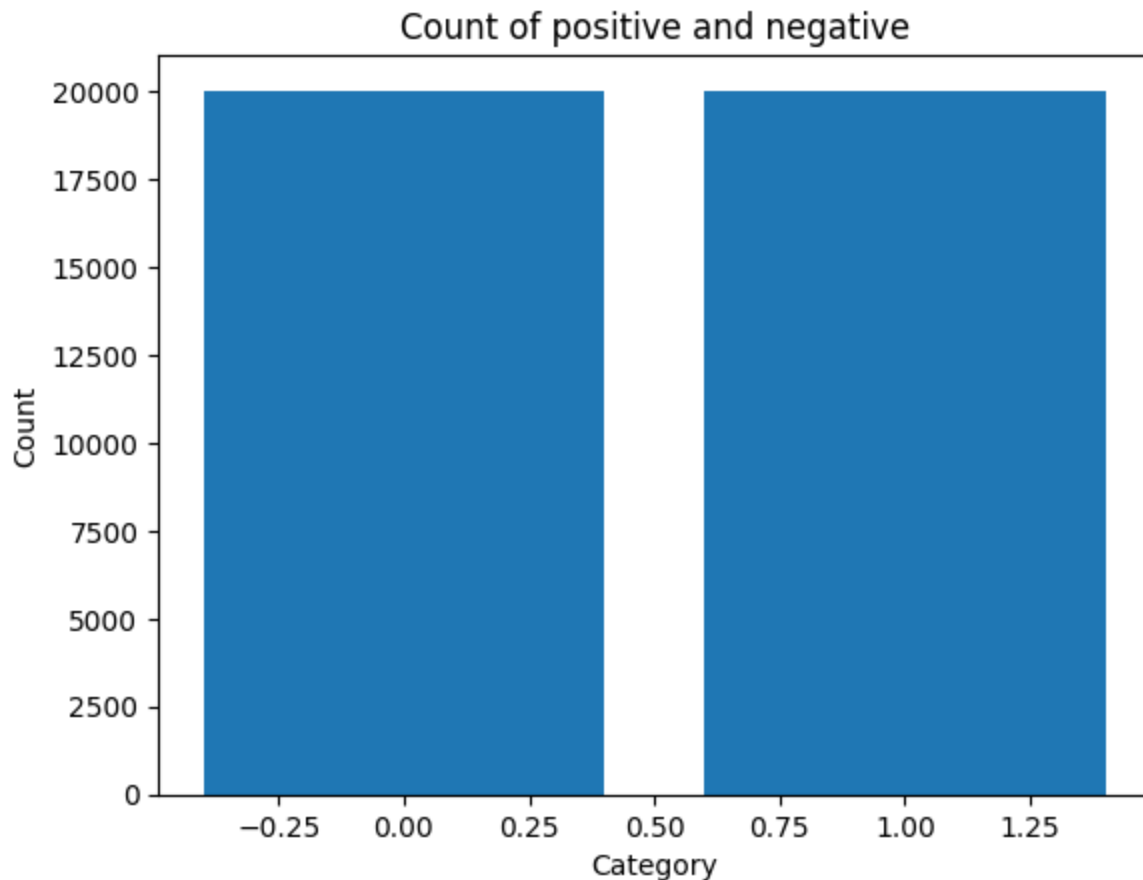
 [View recommended plots](#)

```
category_counts = reviews['label'].value_counts()

plt.bar(category_counts.index, category_counts.values)

plt.xlabel('Category')
plt.ylabel('Count')
plt.title('Count of positive and negative')

plt.show()
```



```
xtrain,xtest,ytrain,ytest=train_test_split(reviews.text,reviews.label,test_size=0.3,random_s
```

```
v=CountVectorizer()
xtrain_count=v.fit_transform(xtrain.values)
xtrain_count.toarray()[ :3]
```

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

```
model=MultinomialNB()
model.fit(xtrain_count,ytrain)
```

▼ MultinomialNB

MultinomialNB()

```
import pickle
with open("movie_rev.pkl", "wb") as file:
    pickle.dump(model, file)

# Input message
new = "it is an amazing movie"

# Transform the input message using CountVectorizer
new_v = v.transform([new])

# Predict sentiment
prediction = model.predict(new_v)

# Print the predicted sentiment
if prediction == 1:
    print("Positive sentiment")
else:
    print("Negative sentiment")
```

Positive sentiment