In [1]:
```
import numpy as np
import pandas as pd
```

In [2]:
```
data = pd.read_csv("CFD.csv");
```

In [3]:
```
data = data.dropna()
data.head()
```

Out[3]:

	Text, Sentiment, Source, Date/Time, User ID, Location, Confidence Score
0	"I love this product!", Positive, Twitter, 202...
1	"The service was terrible.", Negative, Yelp Re...
2	"This movie is amazing!", Positive, IMDb, 2023...
3	"I'm so disappointed with their customer suppo...
4	"Just had the best meal of my life!", Positive...

In [5]:
```
dataord = pd.DataFrame(columns = ['Text', 'Sentiment', 'Source', 'Date/Time', 'User ID', 'Location', 'Confidence Score'])
for i in data.index:
    aux = data.iloc[i,0].split(",")
    dataord.loc[len(dataord)] = aux
dataord.head()
```

Out[5]:

	Text	Sentiment	Source	Date/Time	User ID	Location	Confidence Score
0	"I love this product!"	Positive	Twitter	2023-06-15 09:23:14	@user123	New York	0.85
1	"The service was terrible."	Negative	Yelp Reviews	2023-06-15 11:45:32	user456	Los Angeles	0.65
2	"This movie is amazing!"	Positive	IMDb	2023-06-15 14:10:22	moviefan789	London	0.92
3	"I'm so disappointed with their customer suppo...	Negative	Online Forum	2023-06-15 17:35:11	forumuser1	Toronto	0.78
4	"Just had the best meal of my life!"	Positive	TripAdvisor	2023-06-16 08:50:59	foodie22	Paris	0.88

In [6]:
```
dataord.shape
```

Out[6]:
```
(96, 7)
```

In [7]:
```
dataord.isnull().sum()
```

Out[7]:
```
Text               0
Sentiment          0
Source             0
Date/Time          0
User ID            0
Location           0
Confidence Score   0
dtype: int64
```

In [9]:
```
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
```

In [10]:
```
dataord.describe()
```

Out[10]:

	Text	Sentiment	Source	Date/Time	User ID	Location	Confidence Score
count	96	96	96	96	96	96	96
unique	75	2	27	75	73	11	25
top	"I love this product!"	Positive	Online Store	2023-06-15 09:23:14	user456	Sydney	0.92
freq	2	53	11	2	4	12	9

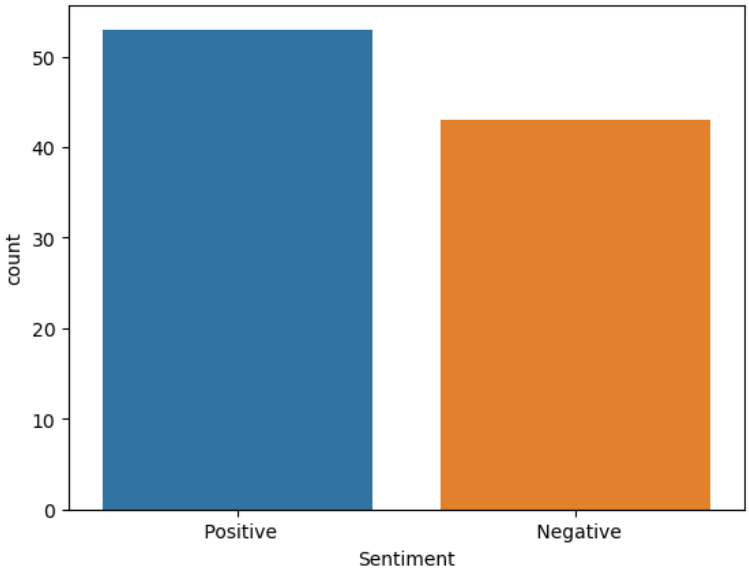
In [11]:
```
dataord.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 96 entries, 0 to 95
Data columns (total 7 columns):
 #   Column            Non-Null Count  Dtype 
---  ------            --------------  ----- 
 0   Text              96 non-null     object
 1   Sentiment         96 non-null     object
 2   Source            96 non-null     object
 3   Date/Time         96 non-null     object
 4   User ID           96 non-null     object
 5   Location          96 non-null     object
 6   Confidence Score  96 non-null     object
dtypes: object(7)
memory usage: 6.0+ KB
```

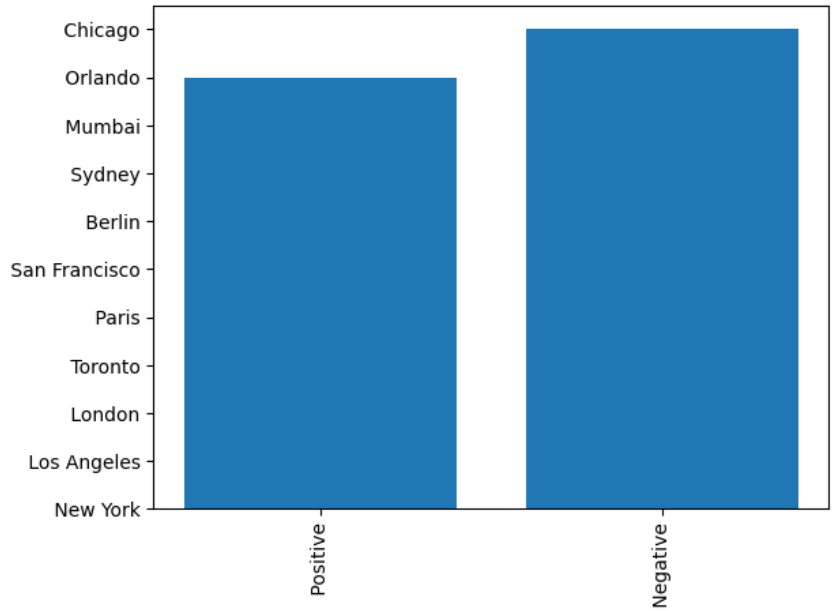
In [12]:
```
dataord.columns
```

Out[12]:
```
Index(['Text', 'Sentiment', 'Source', 'Date/Time', 'User ID', 'Location',
       'Confidence Score'],
      dtype='object')
```

In [13]:
```
sns.countplot(data = dataord , x= 'Sentiment' )
plt.show()
```



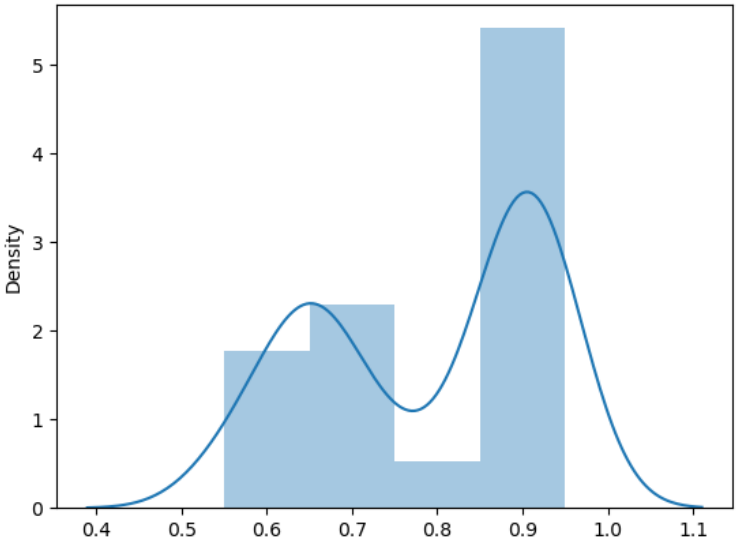
In [25]:
```
plt.bar(dataord['Sentiment'],dataord['Location'])
plt.xticks(rotation = 90)
plt.show()
```



In [27]:
```
sns.distplot(x= dataord['Confidence Score'])
```

```
C:\Users\Red Devil\AppData\Local\Temp\ipykernel_12384\882927438.py:1: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  sns.distplot(x= dataord['Confidence Score'])
```

Out[27]:
```
<Axes: ylabel='Density'>
```


Density
0
1
2
3
4
5
0.4
0.5
0.6
0.7
0.8
0.9
1.0
1.1