[65]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

[7]:
```
df = pd.read_csv("twitter_training.csv")
df.head()
```

[7]:

	2401	Borderlands	Positive	im getting on borderlands and i will murder you all ,
0	2401	Borderlands	Positive	I am coming to the borders and I will kill you...
1	2401	Borderlands	Positive	im getting on borderlands and i will kill you ...
2	2401	Borderlands	Positive	im coming on borderlands and i will murder you...
3	2401	Borderlands	Positive	im getting on borderlands 2 and i will murder ...
4	2401	Borderlands	Positive	im getting into borderlands and i can murder y...

*[9]:
```
df.isnull().sum()
df = df.dropna()
df.head()
```

[9]:

	2401	Borderlands	Positive	im getting on borderlands and i will murder you all ,
0	2401	Borderlands	Positive	I am coming to the borders and I will kill you...
1	2401	Borderlands	Positive	im getting on borderlands and i will kill you ...
2	2401	Borderlands	Positive	im coming on borderlands and i will murder you...
3	2401	Borderlands	Positive	im getting on borderlands 2 and i will murder ...
4	2401	Borderlands	Positive	im getting into borderlands and i can murder y...

[21]:
```
df.columns = ['id', 'game', 'sentiment', 'text']
```

[23]:
```
df.head()
```

[23]:

	id	game	sentiment	text
0	2401	Borderlands	Positive	I am coming to the borders and I will kill you...
1	2401	Borderlands	Positive	im getting on borderlands and i will kill you ...
2	2401	Borderlands	Positive	im coming on borderlands and i will murder you...
3	2401	Borderlands	Positive	im getting on borderlands 2 and i will murder ...
4	2401	Borderlands	Positive	im getting into borderlands and i can murder y...

[33]:
```
df.columns
```

[33]:
```
Index(['id', 'game', 'sentiment', 'text'], dtype='object')
```

[35]:
```
df.describe(include='all')
```

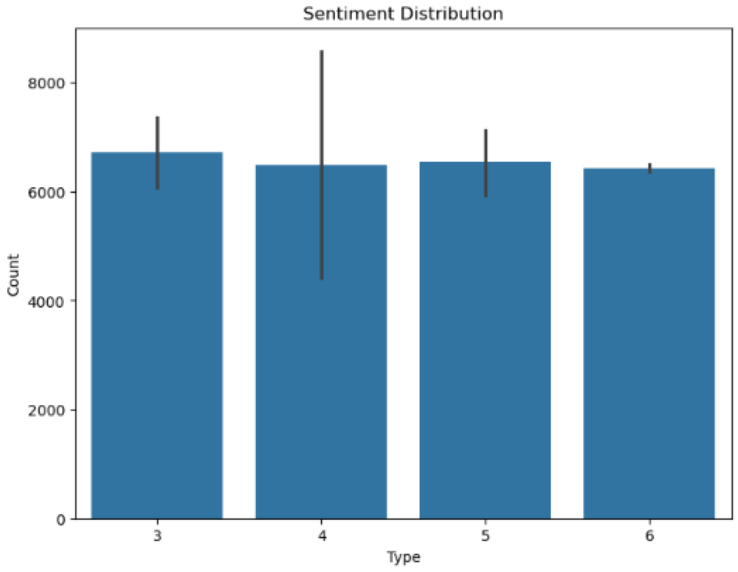
[35]:

	id	game	sentiment	text
count	73995.000000	73995	73995	73995
unique	NaN	32	4	69488
top	NaN	MaddenNFL	Negative	At the same time, despite the fact that there ...
freq	NaN	2377	22358	172
mean	6430.333685	NaN	NaN	NaN
std	3737.655932	NaN	NaN	NaN
min	1.000000	NaN	NaN	NaN
25%	3194.000000	NaN	NaN	NaN
50%	6418.000000	NaN	NaN	NaN
75%	9595.000000	NaN	NaN	NaN
max	13200.000000	NaN	NaN	NaN

[39]:
```
id_types = df['id'].value_counts()
id_types
```

[39]:
```
id
5203     6
6104     6
6093     6
6095     6
6097     6
        ..
9341     3
11668    3
8164     3
9490     3
12825    3
Name: count, Length: 12447, dtype: int64
```

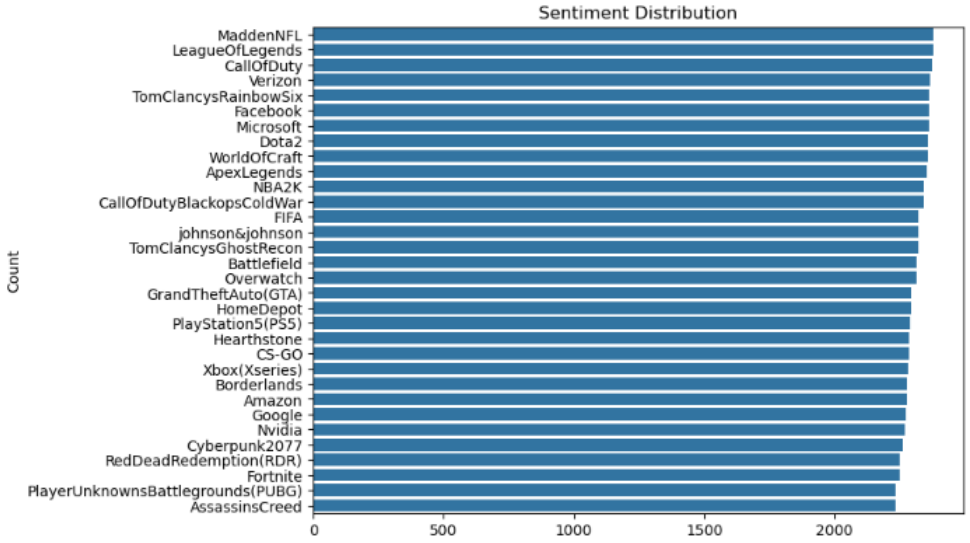
[41]:
```
plt.figure(figsize=(8, 6))
sns.barplot(y=id_types.index, x=id_types.values)
plt.xlabel('Type')
plt.ylabel('Count')
plt.title('Sentiment Distribution')
plt.show()
```



[43]:
```
game_types = df['game'].value_counts()
game_types
```

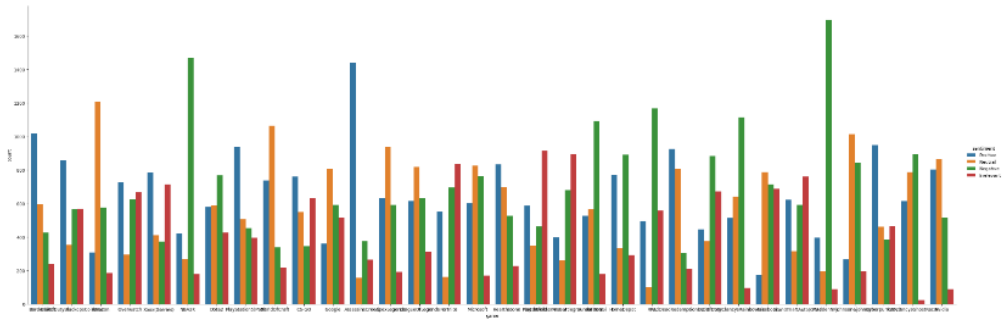
[43]:
```
game
MaddenNFL                          2377
LeagueOfLegends                    2377
CallOfDuty                         2376
Verizon                            2365
TomClancysRainbowSix               2364
Facebook                           2362
Microsoft                          2361
Dota2                              2359
WorldOfCraft                       2357
ApexLegends                        2353
NBA2K                              2343
CallOfDutyBlackopsColdWar          2343
FIFA                               2324
johnson&johnson                    2324
TomClancysGhostRecon               2321
Battlefield                        2316
Overwatch                          2316
GrandTheftAuto(GTA)                2293
HomeDepot                          2292
PlayStation5(PS5)                  2291
Hearthstone                        2286
CS-GO                              2284
Xbox(Xseries)                      2283
Borderlands                        2279
Amazon                             2276
Google                             2274
Nvidia                             2271
Cyberpunk2077                      2262
RedDeadRedemption(RDR)             2249
Fortnite                           2249
PlayerUnknownsBattlegrounds(PUBG)  2234
AssassinsCreed                     2234
Name: count, dtype: int64
```

[47]:
```
plt.figure(figsize=(8, 6))
sns.barplot(y=game_types.index, x=game_types.values)
plt.xlabel('Type')
plt.ylabel('Count')
plt.title('Sentiment Distribution')
plt.show()
```



[49]:
```
sns.catplot(x="game",hue="sentiment", kind="count",height=10,aspect=3, data=df)
```

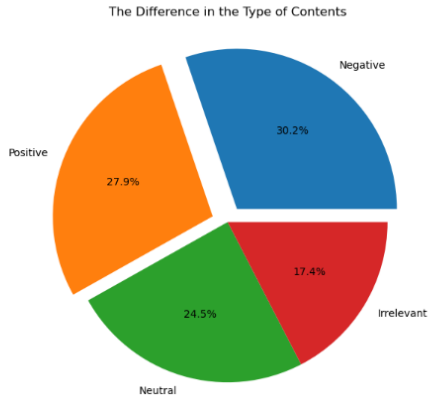
[49]:
```
<seaborn.axisgrid.FacetGrid at 0x2363e4c3020>
```



[51]:
```
sentiment_types = df['sentiment'].value_counts()
sentiment_types
```

[51]:
```
sentiment
Negative      22358
Positive      20654
Neutral       18108
Irrelevant    12875
Name: count, dtype: int64
```

[59]:
```
plt.figure(figsize=(12,7))
plt.pie(x=sentiment_types.values, labels=sentiment_types.index, autopct='%.1f%%', explode=[0.1, 0.1,0,0])
plt.title('The Difference in the Type of Contents')
plt.show()
```



[67]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

[69]:
```
df = pd.read_csv("twitter_validation.csv")
df.head()
```

[69]:

	3364	Facebook	Irrelevant	I mentioned on Facebook that I was struggling for motivation to go for a run the other day, which has been translated by Tom's great auntie as 'Hayley can't get out of bed' and told to his grandma, who now thinks I'm a lazy, terrible person 😩
0	352	Amazon	Neutral	BBC News - Amazon boss Jeff Bezos rejects clai...
1	8312	Microsoft	Negative	@Microsoft Why do I pay for WORD when it funct...
2	4371	CS-GO	Negative	CSGO matchmaking is so full of closet hacking,...
3	4433	Google	Neutral	Now the President is slapping Americans in the...
4	6273	FIFA	Negative	Hi @EAHelp I've had Madeleine McCann in my cel...

[71]:
```
df.columns = ['id', 'social', 'sentiment', 'text']
```

[73]:
```
df.head()
```

[73]:

	id	social	sentiment	text
0	352	Amazon	Neutral	BBC News - Amazon boss Jeff Bezos rejects clai...
1	8312	Microsoft	Negative	@Microsoft Why do I pay for WORD when it funct...
2	4371	CS-GO	Negative	CSGO matchmaking is so full of closet hacking,...
3	4433	Google	Neutral	Now the President is slapping Americans in the...
4	6273	FIFA	Negative	Hi @EAHelp I've had Madeleine McCann in my cel...

[75]:
```
df.columns
```

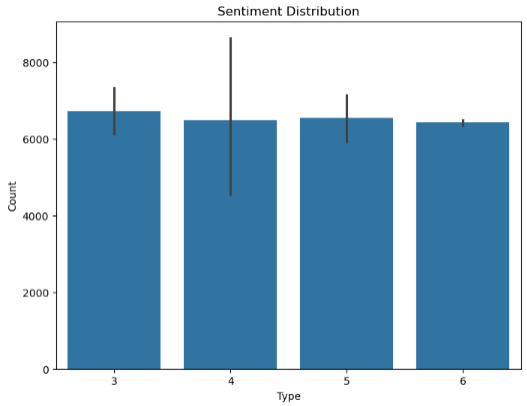
[75]:
```
Index(['id', 'social', 'sentiment', 'text'], dtype='object')
```

[77]:
```
df.describe(include='all')
```

[77]:

	id	social	sentiment	text
count	999.000000	999	999	999
unique	NaN	32	4	998
top	NaN	RedDeadRedemption(RDR)	Neutral	Wow
freq	NaN	40	285	2
mean	6435.159159	NaN	NaN	NaN
std	3728.912226	NaN	NaN	NaN
min	6.000000	NaN	NaN	NaN
25%	3241.500000	NaN	NaN	NaN
50%	6560.000000	NaN	NaN	NaN
75%	9662.500000	NaN	NaN	NaN
max	13197.000000	NaN	NaN	NaN

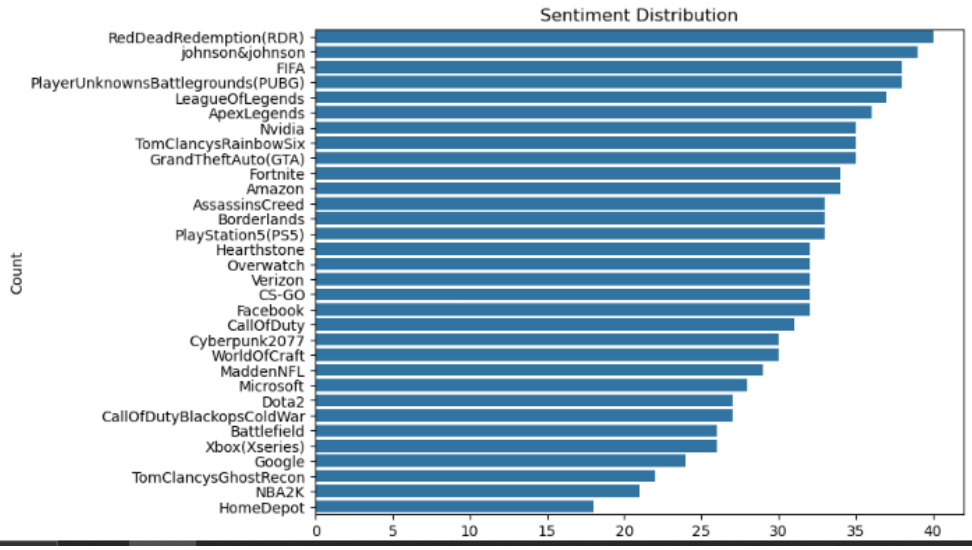
[79]:
```
plt.figure(figsize=(8, 6))
sns.barplot(y=id_types.index, x=id_types.values)
plt.xlabel('Type')
plt.ylabel('Count')
plt.title('Sentiment Distribution')
plt.show()
```



[81]:
```
social_types = df['social'].value_counts()
social_types
```

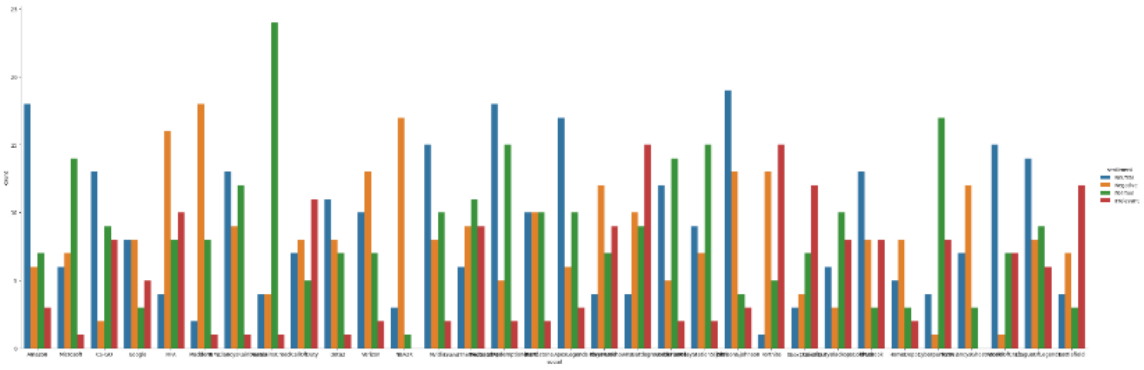
[81]:
```
social
RedDeadRedemption(RDR)              40
johnson&johnson                     39
FIFA                                38
PlayerUnknownsBattlegrounds(PUBG)   38
LeagueOfLegends                     37
ApexLegends                         36
Nvidia                              35
TomClancysRainbowSix                35
GrandTheftAuto(GTA)                 35
Fortnite                            34
Amazon                              34
AssassinsCreed                      33
Borderlands                         33
PlayStation5(PS5)                   33
Hearthstone                         32
Overwatch                           32
Verizon                             32
CS-GO                               32
Facebook                            32
CallOfDuty                          31
Cyberpunk2077                       30
WorldOfCraft                        30
MaddenNFL                           29
Microsoft                           28
Dota2                               27
CallOfDutyBlackopsColdWar           27
Battlefield                         26
Xbox(Xseries)                       26
Google                              24
TomClancysGhostRecon                22
NBA2K                               21
HomeDepot                           18
Name: count, dtype: int64
```

[83]:
```
plt.figure(figsize=(8, 6))
sns.barplot(y=social_types.index, x=social_types.values)
plt.xlabel('Type')
plt.ylabel('Count')
plt.title('Sentiment Distribution')
plt.show()
```



[85]:
```
sns.catplot(x="social",hue="sentiment", kind="count",height=10,aspect=3, data=df)
```

[85]:
```
<seaborn.axisgrid.FacetGrid at 0x236418417f0>
```



[87]:
```
sentiment_types = df['sentiment'].value_counts()
sentiment_types
```

[87]:
```
sentiment
Neutral       285
Positive      277
Negative      266
Irrelevant    171
Name: count, dtype: int64
```

[89]:
```
plt.figure(figsize=(12,7))
plt.pie(x=sentiment_types.values, labels=sentiment_types.index, autopct='%.1f%%', explode=[0.1, 0.1,0,0])
plt.title('The Difference in the Type of Contents')
plt.show()
```