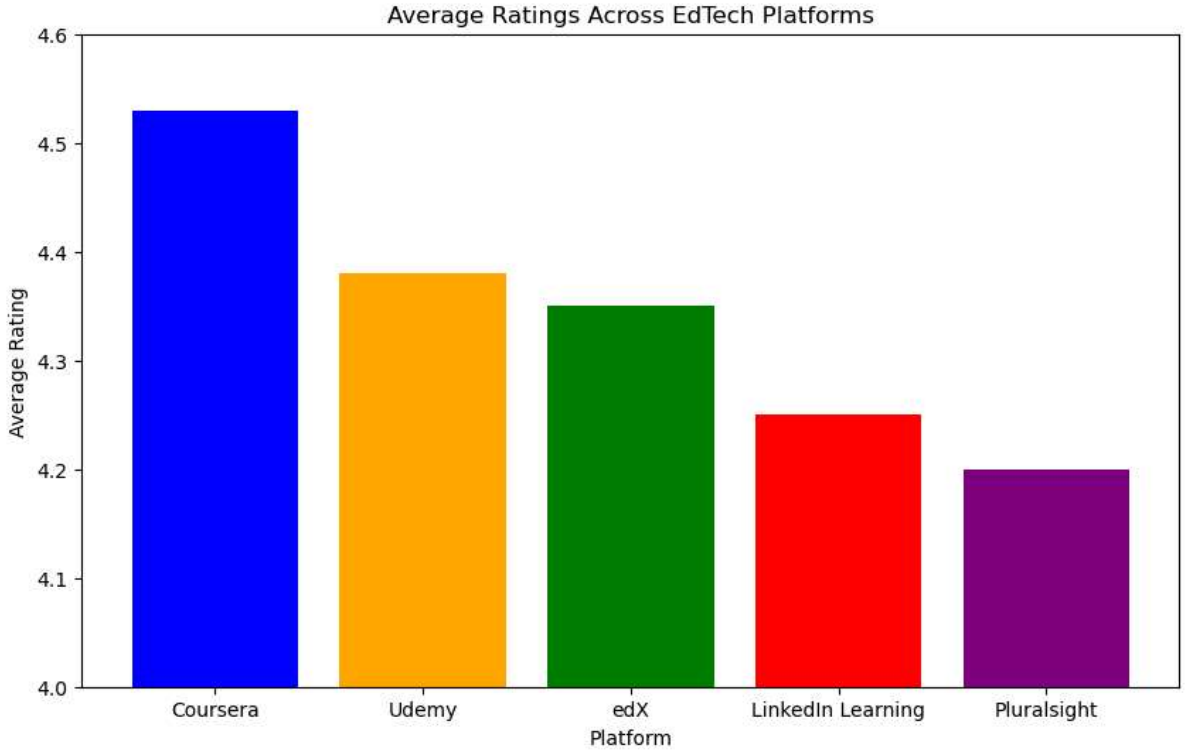
Report topic : Customer Sentiment and User Experience Analysis of EdTech platforms like Coursera, Udemy, edX, LinkedIn Learning, and Pluralsight. (Data Visualization and analysis porocess) Repoprt by : Anumoy Modak, MR Analyst. Organization : Quantuva Technology. Date: 10/08/2024.

1. Bar Chart: Average Ratings Across Platforms:

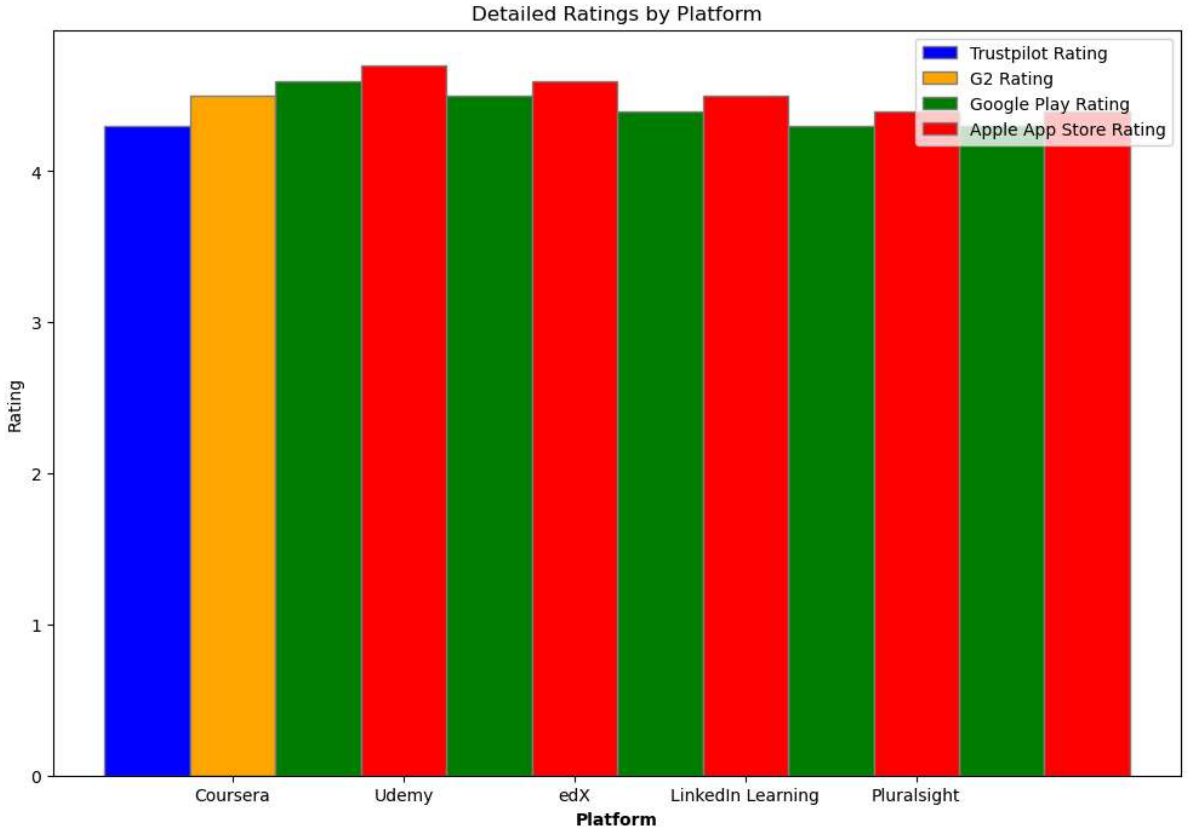
In [ ]:
```
import matplotlib.pyplot as plt

# Data from the table
platforms = ['Coursera', 'Udemy', 'edX', 'LinkedIn Learning', 'Pluralsight']
average_ratings = [4.53, 4.38, 4.35, 4.25, 4.20]

# Plotting the bar chart
plt.figure(figsize=(10, 6))
plt.bar(platforms, average_ratings, color=['blue', 'orange', 'green', 'red', 'purpl
plt.title('Average Ratings Across EdTech Platforms')
plt.xlabel('Platform')
plt.ylabel('Average Rating')
plt.ylim(4.0, 4.6)
plt.show()
```



1. Stacked Bar Chart: Detailed Ratings by Platform

In [ ]:
```
import numpy as np

# Detailed ratings from the table
trustpilot_ratings = [4.3, 4.0, 4.2, 4.1, 4.0]
g2_ratings = [4.5, 4.4, 4.3, 4.2, 4.1]
google_play_ratings = [4.6, 4.5, 4.4, 4.3, 4.3]
apple_app_store_ratings = [4.7, 4.6, 4.5, 4.4, 4.4]

# Bar positions
```

```
bar_width = 0.5
r1 = np.arange(len(platforms))
r2 = [x + bar_width for x in r1]
r3 = [x + bar_width for x in r2]
r4 = [x + bar_width for x in r3]

# Plotting the stacked bar chart
plt.figure(figsize=(12, 8))
plt.bar(r1, trustpilot_ratings, color='blue', width=bar_width, edgecolor='grey', la
plt.bar(r2, g2_ratings, color='orange', width=bar_width, edgecolor='grey', label='G
plt.bar(r3, google_play_ratings, color='green', width=bar_width, edgecolor='grey',
plt.bar(r4, apple_app_store_ratings, color='red', width=bar_width, edgecolor='grey'

plt.xlabel('Platform', fontweight='bold')
plt.xticks([r + bar_width for r in range(len(platforms))], platforms)
plt.title('Detailed Ratings by Platform')
plt.ylabel('Rating')
plt.legend()
plt.show()
```

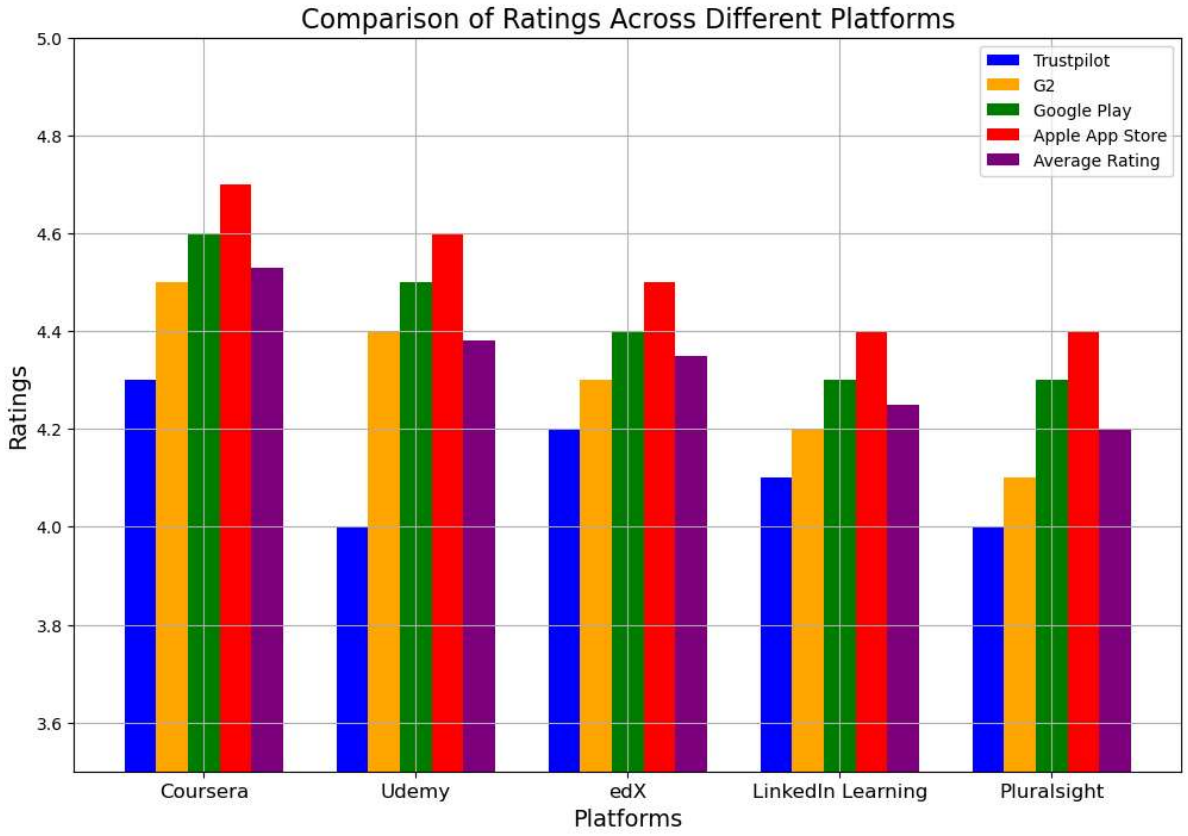


1. Bar Chart: Ratings Comparison Across Platforms:

In [ ]:

```
import matplotlib.pyplot as plt
import numpy as np

# Data for visualization
platforms = ['Coursera', 'Udemy', 'edX', 'LinkedIn Learning', 'Pluralsight']
trustpilot_ratings = [4.3, 4.0, 4.2, 4.1, 4.0]
g2_ratings = [4.5, 4.4, 4.3, 4.2, 4.1]
google_play_ratings = [4.6, 4.5, 4.4, 4.3, 4.3]
apple_app_store_ratings = [4.7, 4.6, 4.5, 4.4, 4.4]
average_ratings = [4.53, 4.38, 4.35, 4.25, 4.2]

# Bar width
bar_width = 0.15
index = np.arange(len(platforms))
```

```
# Plotting the bar chart
plt.figure(figsize=(12, 8))
plt.bar(index, trustpilot_ratings, bar_width, label='Trustpilot', color='blue')
plt.bar(index + bar_width, g2_ratings, bar_width, label='G2', color='orange')
plt.bar(index + 2 * bar_width, google_play_ratings, bar_width, label='Google Play',
plt.bar(index + 3 * bar_width, apple_app_store_ratings, bar_width, label='Apple App
plt.bar(index + 4 * bar_width, average_ratings, bar_width, label='Average Rating',

# Adding labels and title
plt.xlabel('Platforms', fontsize=14)
plt.ylabel('Ratings', fontsize=14)
plt.title('Comparison of Ratings Across Different Platforms', fontsize=16)
plt.xticks(index + 2 * bar_width, platforms, fontsize=12)
plt.ylim(3.5, 5)
plt.legend()
plt.grid(True)
plt.show()
```

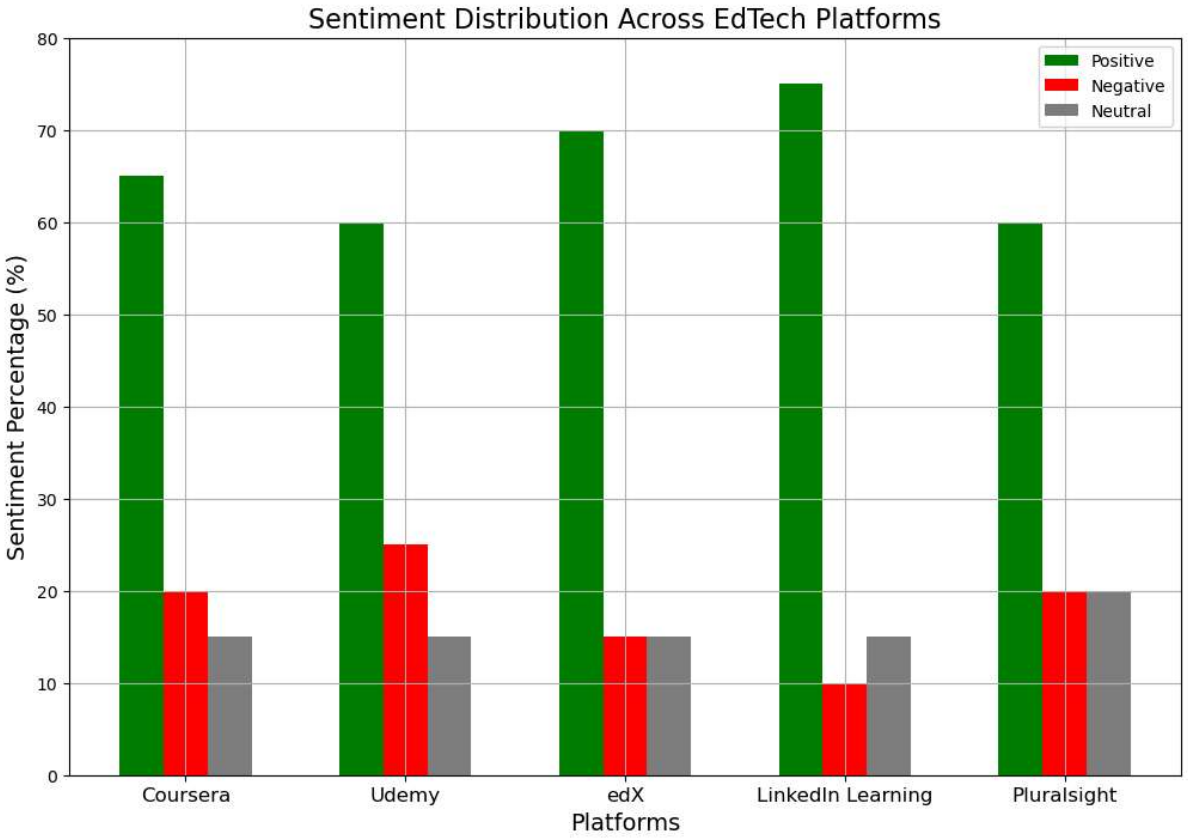


1. Bar Chart: Sentiment Distribution Across Platforms:

In [ ]:

```
import matplotlib.pyplot as plt
import numpy as np

# Data for visualization
platforms = ['Coursera', 'Udemy', 'edX', 'LinkedIn Learning', 'Pluralsight']
positive = [65, 60, 70, 75, 60]
negative = [20, 25, 15, 10, 20]
neutral = [15, 15, 15, 15, 20]

# Bar width
bar_width = 0.2
index = np.arange(len(platforms))

# Plotting the bar chart
plt.figure(figsize=(12, 8))
```

```
plt.bar(index, positive, bar_width, label='Positive', color='green')
plt.bar(index + bar_width, negative, bar_width, label='Negative', color='red')
plt.bar(index + 2 * bar_width, neutral, bar_width, label='Neutral', color='gray')

# Adding labels and title
plt.xlabel('Platforms', fontsize=14)
plt.ylabel('Sentiment Percentage (%)', fontsize=14)
plt.title('Sentiment Distribution Across EdTech Platforms', fontsize=16)
plt.xticks(index + bar_width, platforms, fontsize=12)
plt.ylim(0, 80)
plt.legend()
plt.grid(True)
plt.show()
```

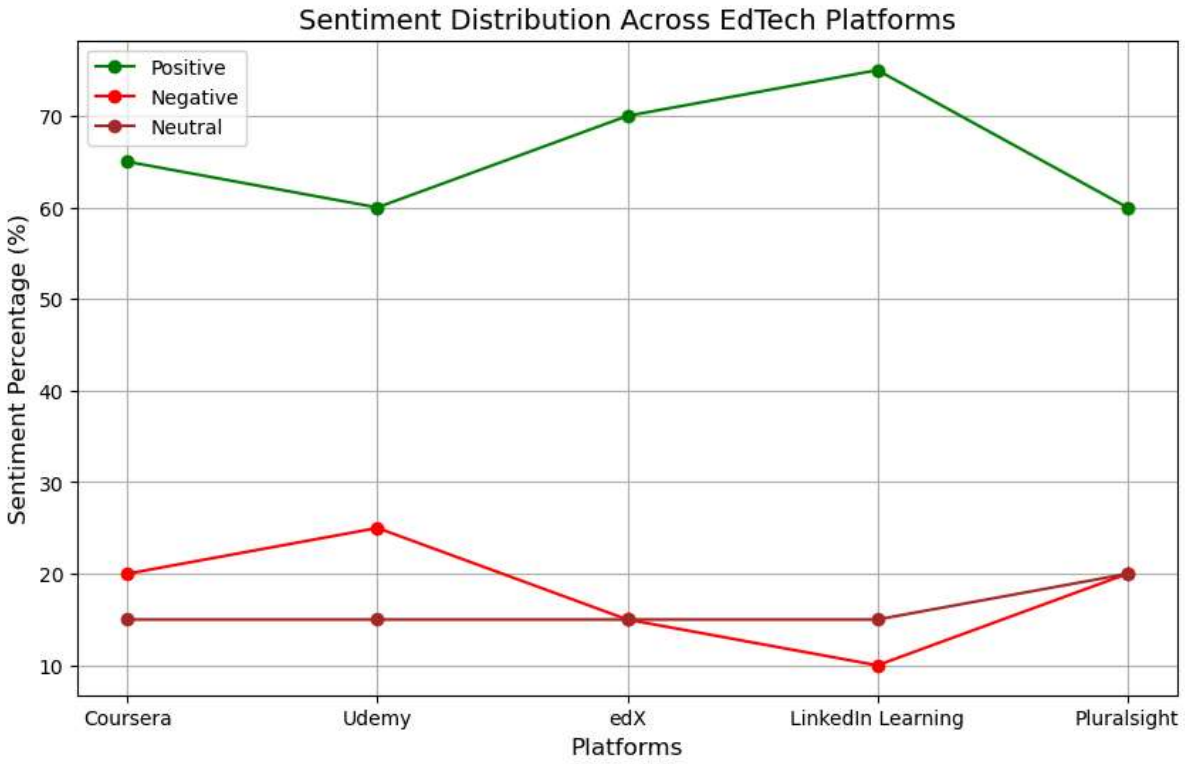


1. Line Chart: Sentiment Distribution Across Platforms:

In [ ]:

```
import matplotlib.pyplot as plt

# Data for visualization
platforms = ['Coursera', 'Udemy', 'edX', 'LinkedIn Learning', 'Pluralsight']
positive = [65, 60, 70, 75, 60]
negative = [20, 25, 15, 10, 20]
neutral = [15, 15, 15, 15, 20]

# Plotting the line graph
plt.figure(figsize=(10, 6))
plt.plot(platforms, positive, marker='o', linestyle='-', color='green', label='Posi
plt.plot(platforms, negative, marker='o', linestyle='-', color='red', label='Negati
plt.plot(platforms, neutral, marker='o', linestyle='-', color='brown', label='Neutr

# Adding labels and title
plt.xlabel('Platforms', fontsize=12)
plt.ylabel('Sentiment Percentage (%)', fontsize=12)
plt.title('Sentiment Distribution Across EdTech Platforms', fontsize=14)
plt.legend()
```

```
plt.grid(True)
plt.show()
```

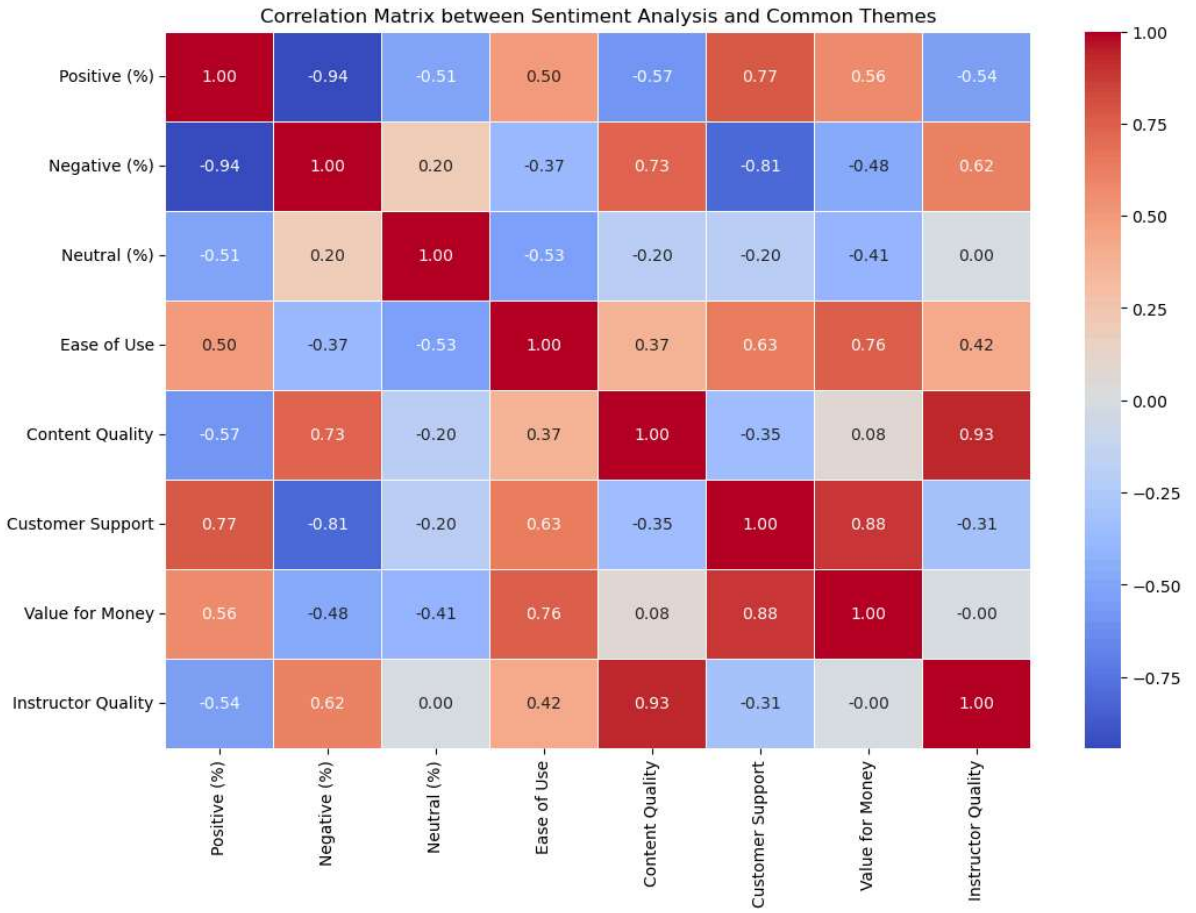


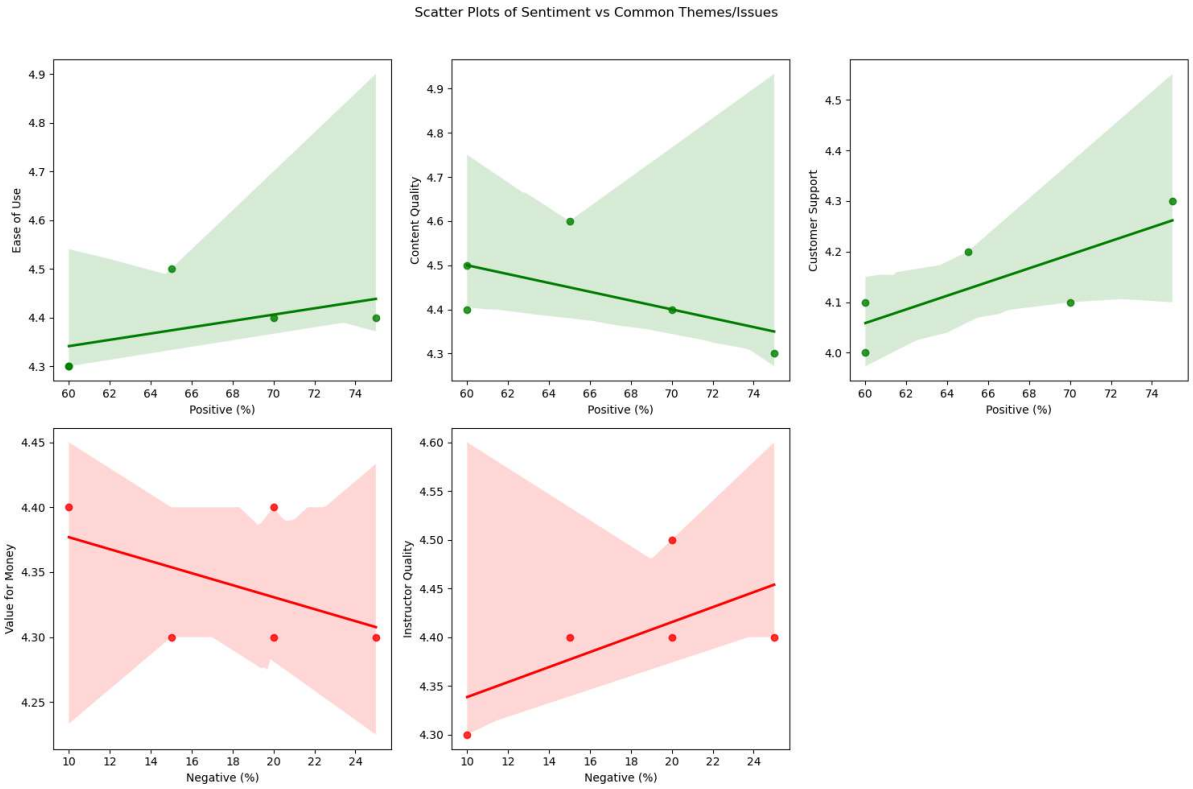
1. Correlational analysis between Sentiment distributions and common themes and issues.

In [ ]:

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt

# Data for sentiment analysis
sentiment_data = {
    'Platform': ['Coursera', 'Udemy', 'edX', 'LinkedIn Learning', 'Pluralsight'],
    'Positive (%)': [65, 60, 70, 75, 60],
    'Negative (%)': [20, 25, 15, 10, 20],
    'Neutral (%)': [15, 15, 15, 15, 20]
}

# Data for common themes/issues
themes_data = {
    'Platform': ['Coursera', 'Udemy', 'edX', 'LinkedIn Learning', 'Pluralsight'],
    'Ease of Use': [4.5, 4.3, 4.4, 4.4, 4.3],
    'Content Quality': [4.6, 4.5, 4.4, 4.3, 4.4],
    'Customer Support': [4.2, 4.0, 4.1, 4.3, 4.1],
    'Value for Money': [4.4, 4.3, 4.3, 4.4, 4.3],
    'Instructor Quality': [4.5, 4.4, 4.4, 4.3, 4.4]
}

# Convert to DataFrame
sentiment_df = pd.DataFrame(sentiment_data)
themes_df = pd.DataFrame(themes_data)

# Merge the two DataFrames on Platform
merged_df = pd.merge(sentiment_df, themes_df, on='Platform')

# Calculate the correlation matrix
correlation_matrix = merged_df.drop(columns=['Platform']).corr()
```

```
# Display the correlation matrix
correlation_matrix

# Visualize the correlation matrix using a heatmap
plt.figure(figsize=(12, 8))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', fmt=".2f", linewidths=
plt.title('Correlation Matrix between Sentiment Analysis and Common Themes')
plt.show()

# Scatter plot for each sentiment and common theme
fig, axes = plt.subplots(2, 3, figsize=(15, 10))
fig.suptitle('Scatter Plots of Sentiment vs Common Themes/Issues')

# Positive Sentiment vs Common Themes
sns.regplot(x='Positive (%)', y='Ease of Use', data=merged_df, ax=axes[0, 0], color
sns.regplot(x='Positive (%)', y='Content Quality', data=merged_df, ax=axes[0, 1], c
sns.regplot(x='Positive (%)', y='Customer Support', data=merged_df, ax=axes[0, 2],

# Negative Sentiment vs Common Themes
sns.regplot(x='Negative (%)', y='Value for Money', data=merged_df, ax=axes[1, 0], c
sns.regplot(x='Negative (%)', y='Instructor Quality', data=merged_df, ax=axes[1, 1]

# Remove the empty subplot
axes[1, 2].axis('off')

# Adjust layout
plt.tight_layout(rect=[0, 0, 1, 0.96])
plt.show()
```

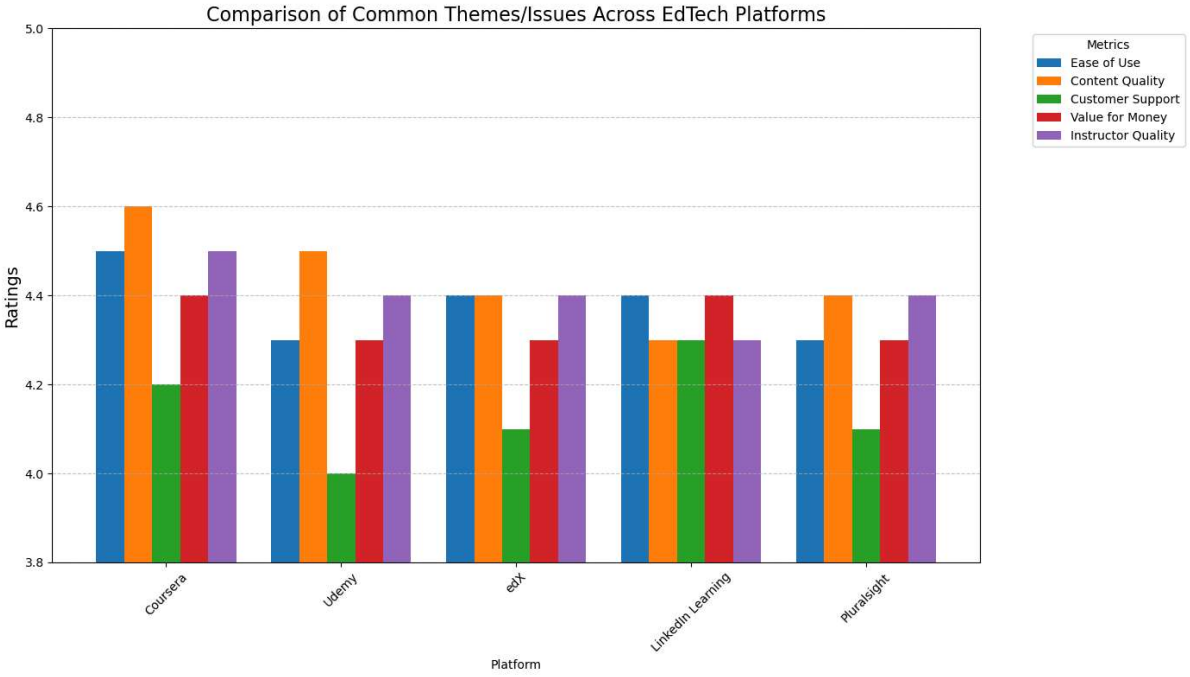




1. Bar chart: Common Theme Data (Rating out of 5):

In [ ]:
```
import pandas as pd
import matplotlib.pyplot as plt

# Data for common themes/issues
data = {
    'Platform': ['Coursera', 'Udemy', 'edX', 'LinkedIn Learning', 'Pluralsight'],
    'Ease of Use': [4.5, 4.3, 4.4, 4.4, 4.3],
    'Content Quality': [4.6, 4.5, 4.4, 4.3, 4.4],
    'Customer Support': [4.2, 4.0, 4.1, 4.3, 4.1],
    'Value for Money': [4.4, 4.3, 4.3, 4.4, 4.3],
    'Instructor Quality': [4.5, 4.4, 4.4, 4.3, 4.4]
}

# Convert to DataFrame
df = pd.DataFrame(data)

# Set the Platform as the index
df.set_index('Platform', inplace=True)

# Plotting the data
plt.figure(figsize=(12, 8))
df.plot(kind='bar', figsize=(14, 8), width=0.8)

plt.title('Comparison of Common Themes/Issues Across EdTech Platforms', fontsize=16
plt.ylabel('Ratings', fontsize=14)
plt.xticks(rotation=45)
plt.ylim(3.8, 5)
plt.legend(title='Metrics', bbox_to_anchor=(1.05, 1), loc='upper left')
plt.grid(axis='y', linestyle='--', alpha=0.7)

plt.tight_layout()
plt.show()
```

```
<Figure size 1200x800 with 0 Axes>
```


Comparison of Common Themes/Issues Across EdTech Platforms
Metrics
Ease of Use
Content Quality
Customer Support
Value for Money
Instructor Quality
Ratings
5.0
4.8
4.6
4.4
4.2
4.0
3.8
Coursera
Udemy
edX
LinkedIn Learning
Pluralsight
Platform