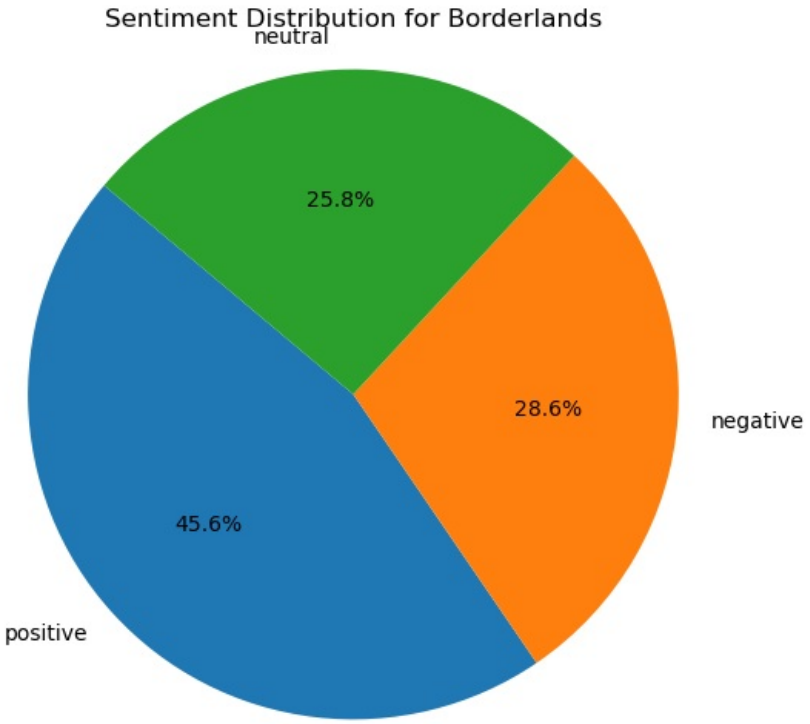
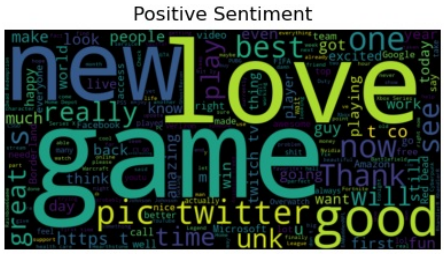
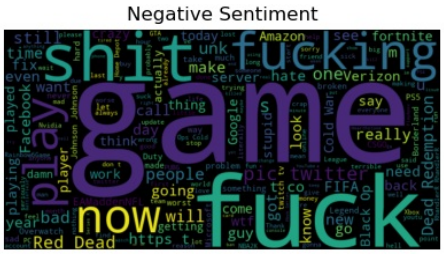
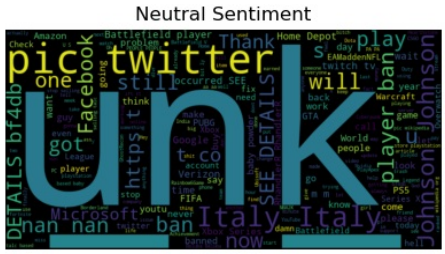
In [ ]:

In [18]:
```
import pandas as pd
from textblob import TextBlob
import matplotlib.pyplot as plt
from wordcloud import WordCloud

# Load the dataset
df = pd.read_csv("C:/Users/sathi/Downloads/twitter_training.csv", header=None, names=['ID', 'Topic', 'Sentiment

# Step 1: Convert 'Text' column to string
df['Text'] = df['Text'].astype(str)

# Step 2: Function to get sentiment polarity
def get_sentiment(text):
    analysis = TextBlob(text)
    return analysis.sentiment.polarity

# Step 3: Apply sentiment analysis to each tweet
df['Polarity'] = df['Text'].apply(get_sentiment)

# Step 4: Classify sentiment based on polarity
df['Sentiment'] = df['Polarity'].apply(lambda x: 'positive' if x > 0 else 'negative' if x < 0 else 'neutral')

# Step 5: Visualize sentiment distribution
sentiment_counts = df['Sentiment'].value_counts()
plt.figure(figsize=(6, 6))
plt.pie(sentiment_counts, labels=sentiment_counts.index, autopct='%1.1f%%', startangle=140)
plt.title('Sentiment Distribution for Borderlands')
plt.axis('equal')
plt.show()

# Step 6: Generate word clouds for each sentiment category
positive_text = ' '.join(df[df['Sentiment'] == 'positive']['Text'])
negative_text = ' '.join(df[df['Sentiment'] == 'negative']['Text'])
neutral_text = ' '.join(df[df['Sentiment'] == 'neutral']['Text'])

wordcloud_positive = WordCloud(width=800, height=400).generate(positive_text)
wordcloud_negative = WordCloud(width=800, height=400).generate(negative_text)
wordcloud_neutral = WordCloud(width=800, height=400).generate(neutral_text)

plt.figure(figsize=(12, 10))

plt.subplot(131)
plt.imshow(wordcloud_positive, interpolation='bilinear')
plt.title('Positive Sentiment')
plt.axis('off')

plt.subplot(132)
plt.imshow(wordcloud_negative, interpolation='bilinear')
plt.title('Negative Sentiment')
plt.axis('off')

plt.subplot(133)
plt.imshow(wordcloud_neutral, interpolation='bilinear')
plt.title('Neutral Sentiment')
plt.axis('off')

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









In [ ]:

In [ ]: