



Taller: Análisis de sentimientos en Twitter

En diversas plataformas, como Kaggle, se encuentran bases de datos y exploraciones de análisis disponibles para su uso. Kaggle, en particular, ofrece una amplia variedad de conjuntos de datos creados por desarrolladores. En este contexto, se introduce la plataforma Kaggle como una fuente para la obtención de datos, lo que permite extraer información y llevar a cabo el análisis exploratorio de datos utilizando conjuntos de datos relacionados con Twitter.

A continuación, se presenta un script base que muestra el conjunto de datos, de sentimientos de Twitter, y grafica las cantidades de Tweets clasificando si dicho tweets es neutral, positivo o negativo, introduciendo al análisis de sentimientos.

```
import pandas as pd
import matplotlib.pyplot as plt
from textblob import TextBlob
```

```
# Leer el archivo CSV
df = pd.read_csv('twitter_sentiment_data.csv')
```



```
# Función para realizar el análisis de sentimientos
def analizar_sentimientos(texto):
    # Utilizar TextBlob para obtener la polaridad del
    sentimiento
    blob = TextBlob(texto)
    polaridad = blob.sentiment.polarity

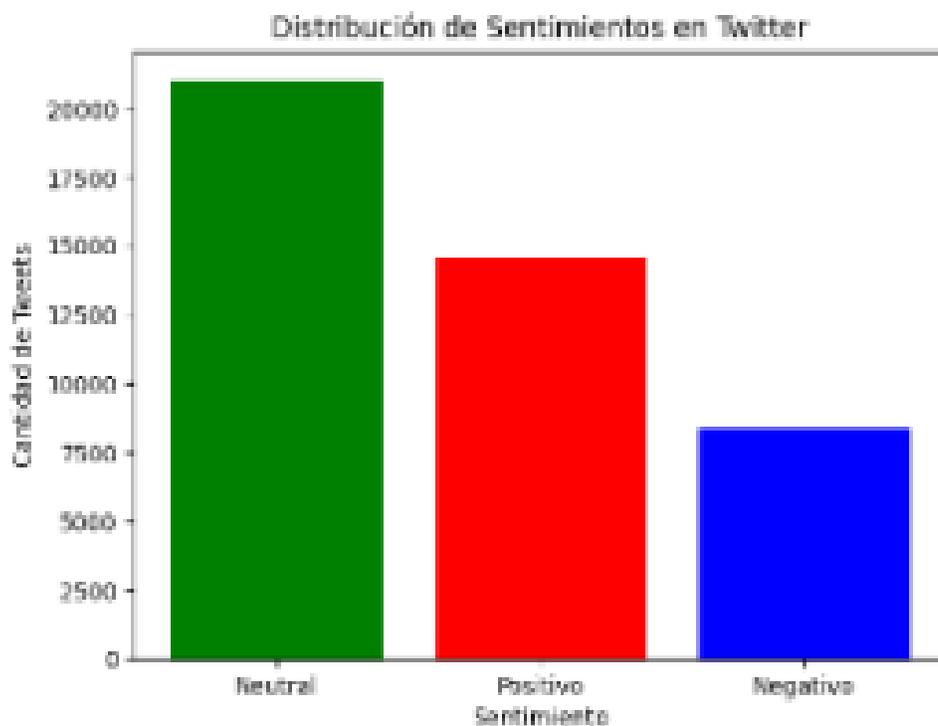
    # Asignar etiquetas de sentimiento en base a la
    polaridad
    if polaridad > 0:
        return 'Positivo'
    elif polaridad < 0:
        return 'Negativo'
    else:
        return 'Neutral'

# Aplicar la función de análisis de sentimientos a cada
tweet
df['sentimiento'] =
df['message'].apply(analizar_sentimientos)

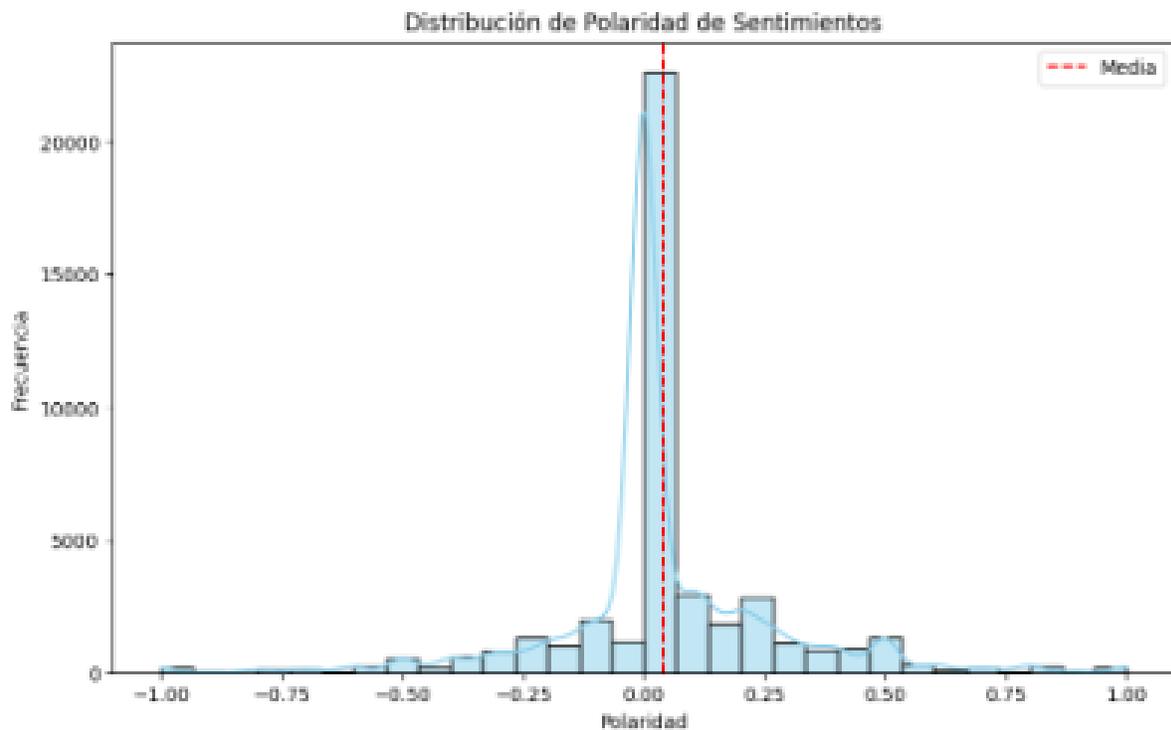
# Mostrar la distribución de los sentimientos en un
gráfico de barras
sentimiento_counts = df['sentimiento'].value_counts()
plt.bar(sentimiento_counts.index,
sentimiento_counts.values,
```

```
color=['green', 'red', 'blue']  
plt.xlabel('Sentimiento')  
plt.ylabel('Cantidad de Tweets')  
plt.title('Distribución de Sentimientos en Twitter')  
plt.show()
```

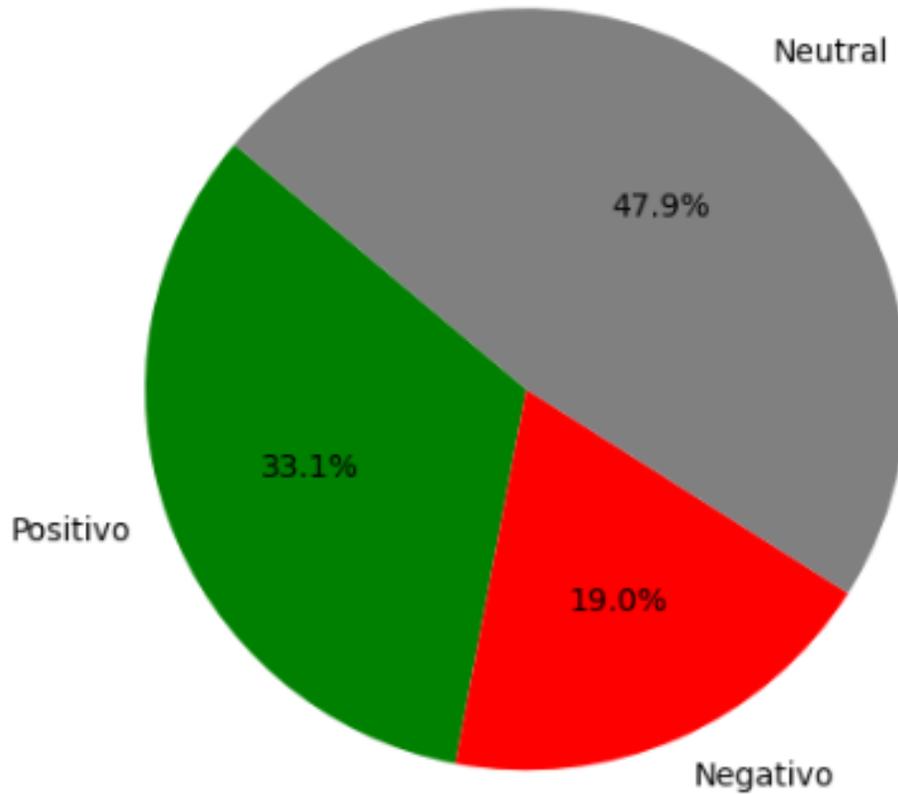
```
# Mostrar el DataFrame con los resultados del  
análisis de sentimientos  
print(df)
```



Otra forma de analizar dichos sentimientos parte de la distribución de polaridad de sentimientos. Con el análisis gráfico, se observará la frecuencia con respecto a la polaridad, permitiendo observar que tan bueno o malo es el sentimiento de acuerdo con la media.



Porcentaje de Sentimientos



Además, se puede detallar los porcentajes de los Tweets con un gráfico de barras. Esto es del siguiente scripts.

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



```
from textblob import TextBlob
```

```
# Leer el archivo CSV
```

```
df = pd.read_csv('twitter_sentiment_data.csv')
```

```
# Función para realizar el análisis de sentimientos
```

```
def analizar_sentimientos(texto):
```

```
    # Utilizar TextBlob para obtener la polaridad del sentimiento
```

```
    blob = TextBlob(texto)
```

```
    polaridad = blob.sentiment.polarity
```

```
    return polaridad
```

```
# Aplicar la función de análisis de sentimientos a cada tweet
```

```
df['polaridad']
```

```
df['message'].apply(analizar_sentimientos)
```

```
=
```

```
# Obtener métricas de análisis de sentimientos
```

```
media_polaridad = df['polaridad'].mean()
```

```
sentimientos_positivos = df[df['polaridad'] > 0]
```

```
sentimientos_negativos = df[df['polaridad'] < 0]
```

```
sentimientos_neutrales = df[df['polaridad'] == 0]
```



```
# Mostrar gráficos
```

```
plt.figure(figsize=(10, 6))
```

```
# Histograma de polaridad
```

```
sns.histplot(data=df, x='polaridad', bins=30, kde=True,  
color='skyblue')
```

```
plt.title('Distribución de Polaridad de Sentimientos')
```

```
plt.xlabel('Polaridad')
```

```
plt.ylabel('Frecuencia')
```

```
plt.axvline(x=media_polaridad, color='red', linestyle='--',  
label='Media')
```

```
plt.legend()
```

```
plt.show()
```

```
# Gráfico de torta de sentimientos
```

```
sizes = [len(sentimientos_positivos),  
len(sentimientos_negativos),  
len(sentimientos_neutrales)]
```

```
labels = ['Positivo', 'Negativo', 'Neutral']
```

```
plt.pie(sizes, labels=labels, autopct='%1.1f%%', colors=  
['green', 'red', 'gray'], startangle=140)
```

```
plt.axis('equal')
```

```
plt.title('Porcentaje de Sentimientos')
```

```
plt.show()
```



```
# Información sobre el análisis de sentimientos
print(f"Media de polaridad: {media_polaridad}")
print(f"Cantidad de sentimientos positivos:
{len(sentimientos_positivos)}")
print(f"Cantidad de sentimientos negativos:
{len(sentimientos_negativos)}")
print(f"Cantidad de sentimientos neutrales:
{len(sentimientos_neutrales)}")
```