



PIPELINE

de

Análisis del Dataset de Letras de Canciones

ALUMNO: ALVAREZ ASTETE, JHEEREMY MANUEL
PROFESORA: CUADROS VALDIVIA, ANA MARIA



1. ¿Qué problemas identifican en el dataset?

- Valores nulos o vacíos en variables importantes
- Etiquetas `Sentiment` con categorías inconsistentes o muy desbalanceadas
- Valores anómalos o sucios en `Decade` (ej. `nan0`)
- Formato complejo en `Vader_Score` (es un string que parece diccionario)

1. ¿Qué problemas identifican en el dataset?

```
INFO GENERAL DEL DATASET
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4583 entries, 0 to 4582
Data columns (total 18 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   Unnamed: 0                            4583 non-null   int64
1   Singer                                4583 non-null   object
2   Song                                  4583 non-null   object
3   Genre                                4580 non-null   object
4   Vader_Score                           4583 non-null   object
5   Valence                               4583 non-null   float64
6   Arousal                               4583 non-null   float64
7   Decade                                4583 non-null   object
8   Sentiment                             4583 non-null   object
9   Total_Word_Count                      4583 non-null   int64
10  Unique_Word_Count                     4583 non-null   int64
11  Fraction_Unique_Words                  4583 non-null   float64
12  Numer_Unique_Word_Lengths              4583 non-null   int64
13  Min_Word_Length                        4583 non-null   int64
14  Max_Word_Length                        4583 non-null   int64
15  Average_Word_Length                    4583 non-null   float64
16  Sum_All_Word_Lengths                   4583 non-null   int64
17  Cluster_Label                           4583 non-null   object
dtypes: float64(4), int64(7), object(7)
memory usage: 644.6+ KB
None
```

1. ¿Qué problemas identifican en el dataset?

```
Nulos por columna:
Unnamed: 0      0
Singer          0
Song            0
Genre           3
Vader_Score     0
Valence         0
Arousal         0
Decade          0
Sentiment       0
Total_Word_Count 0
Unique_Word_Count 0
Fraction_Unique_Words 0
Numer_Unique_Word_Lengths 0
Min_Word_Length 0
Max_Word_Length 0
Average_Word_Length 0
Sum_All_Word_Lengths 0
Cluster_Label   0
dtype: int64

Valores únicos en 'Sentiment': ['Relaxed' 'Angry' 'Sad' 'Happy']

Valores únicos en 'Decade': ['nan0' '1970' '2010' '2000' '1980' '1990' '1960' '1950' '1930' '1920']

Ejemplo de 'Vader_Score':
0  {'neg': 0.057, 'neu': 0.735, 'pos': 0.207, 'co...
1  {'neg': 0.016, 'neu': 0.898, 'pos': 0.086, 'co...
2  {'neg': 0.142, 'neu': 0.808, 'pos': 0.049, 'co...
3  {'neg': 0.131, 'neu': 0.817, 'pos': 0.052, 'co...
4  {'neg': 0.308, 'neu': 0.632, 'pos': 0.061, 'co...
Name: Vader_Score, dtype: object
```

2. ¿Qué descubrieron al analizar los datos?

- Sentiment tiene muchas más canciones en la categoría "Relaxed" (desbalance)
- Algunas filas tienen Decade mal formateado (limpieza necesaria)
- Valence, Arousal, Total_Word_Count tienen rangos dentro de lo esperado
- Vader_Score necesita transformación para obtener el valor compound

2. ¿Qué descubrieron al analizar los datos?

```
# Contar por Sentiment
sentiment_counts = df['Sentiment'].value_counts()
print("Distribución Sentiment:\n", sentiment_counts)

# Estadísticas básicas de variables numéricas
print("\nEstadísticas Valence, Arousal y Total_Word_Count:")
print(df[['Valence', 'Arousal', 'Total_Word_Count']].describe())

# Extraer el score 'compound' de Vader_Score
import ast
df['Vader_compound'] = df['Vader_Score'].apply(lambda x: ast.literal_eval(x)['compound'])

print("\nEstadísticas de Vader_compound:")
print(df['Vader_compound'].describe())
```

2. ¿Qué descubrieron al analizar los datos?

Distribución Sentiment:

Sentiment

Relaxed 4128

Sad 258

Happy 127

Angry 70

Name: count, dtype: int64

Estadísticas Valence, Arousal y Total_Word_Count:

	Valence	Arousal	Total_Word_Count
count	4583.000000	4583.000000	4583.000000
mean	5.900379	4.286748	315.121318
std	0.580384	0.387666	216.950667
min	3.107586	3.195156	52.000000
25%	5.560333	4.010104	168.000000
50%	5.961364	4.266053	250.000000
75%	6.300595	4.521461	394.500000
max	7.412500	6.422523	2828.000000

Estadísticas de Vader_compound:

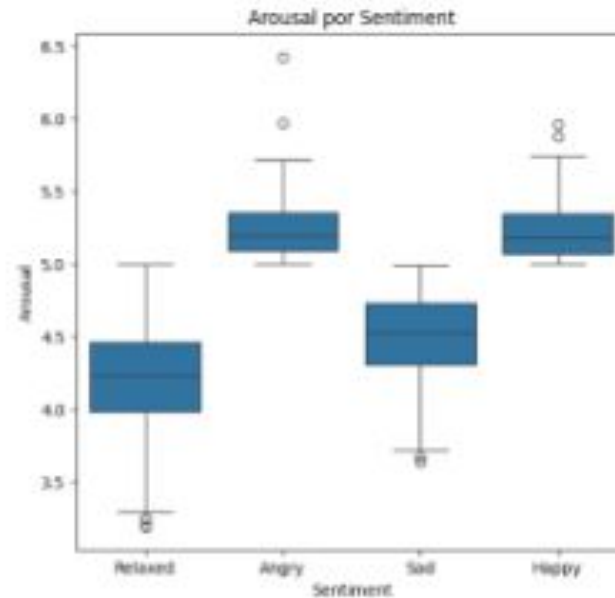
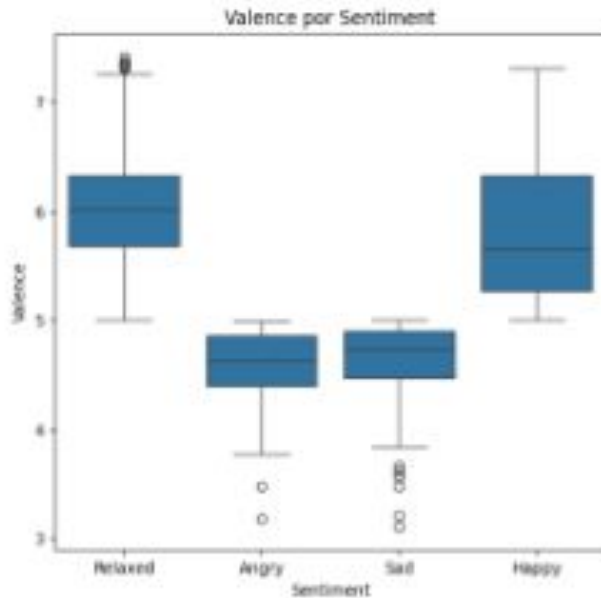
count	4583.000000
mean	0.264129
std	0.856248
min	-0.999900
25%	-0.878000
50%	0.860800
75%	0.984100
max	0.999900

Name: Vader_compound, dtype: float64

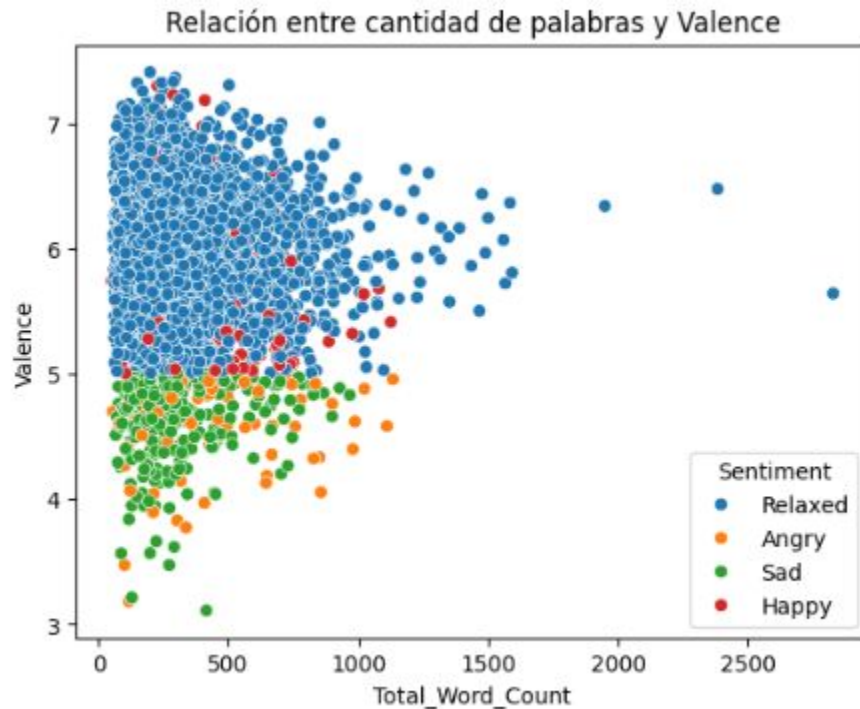
3. ¿Qué reflejan los patrones de tendencia?

- **Valence es más estable mientras que Arousal varía según el tipo de Sentiment**
- **Emociones negativas (Sad o Angry) tienen baja valencia y alta excitación (arousal)**
- **Letras más largas (Total_Word_Count) están asociadas a emociones complejas (por ejemplo, tristeza o reflexión)**

3. ¿Qué reflejan los patrones de tendencia?



3. ¿Qué reflejan los patrones de tendencia?



4. ¿Cómo es afectado el comportamiento humano (lenguaje, emoción) en el dataset?

- El lenguaje refleja el estado emocional: letras con más palabras tienden a expresar emociones más complejas y profundas (mayor variabilidad en Valence y Arousal)
- El dataset muestra que las emociones no son homogéneas, hay diversidad en la expresión humana a través de las letras y su estructura
- La combinación de análisis manual (Sentiment) y automático (Vader) ayuda a entender mejor estas emociones y su impacto en el comportamiento

4. ¿Cómo es afectado el comportamiento humano (lenguaje, emoción) en el dataset?

```
corr = df[['Valence', 'Arousal', 'Total_Word_Count', 'Vader_compound']].corr()
print("Correlaciones:\n", corr)

sns.heatmap(corr, annot=True, cmap='coolwarm')
plt.title('Correlación entre variables emocionales y texto')
plt.show()
```

Correlaciones:

	Valence	Arousal	Total_Word_Count	Vader_compound
Valence	1.000000	-0.219290	-0.056895	0.521155
Arousal	-0.219290	1.000000	0.142734	-0.196101
Total_Word_Count	-0.056895	0.142734	1.000000	-0.209630
Vader_compound	0.521155	-0.196101	-0.209630	1.000000

