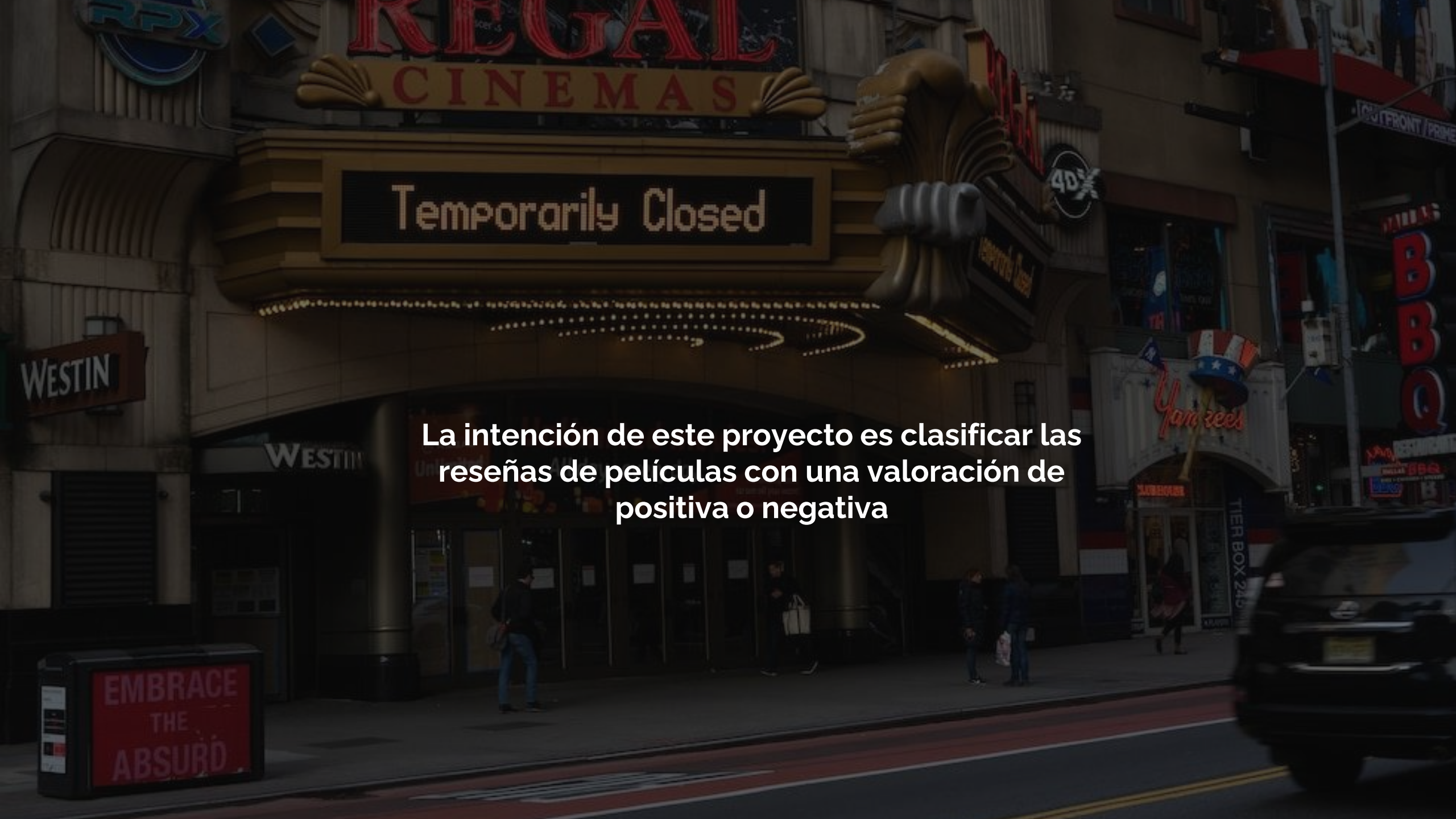


Equipo 5 ~ Análisis de datos con Python

Clasificación de reseñas de películas



La intención de este proyecto es clasificar las reseñas de películas con una valoración de positiva o negativa

- 1 Presentación
- 2 Exploración de datos
- 3 Análisis de datos y visualización
- 4 Limpieza de datos
- 5 Preprocesamiento
- 6 Entrenamiento del modelo
- 7 Evaluación del modelo
- 8 Visualización



1 Presentación

PLANTEAMIENTO

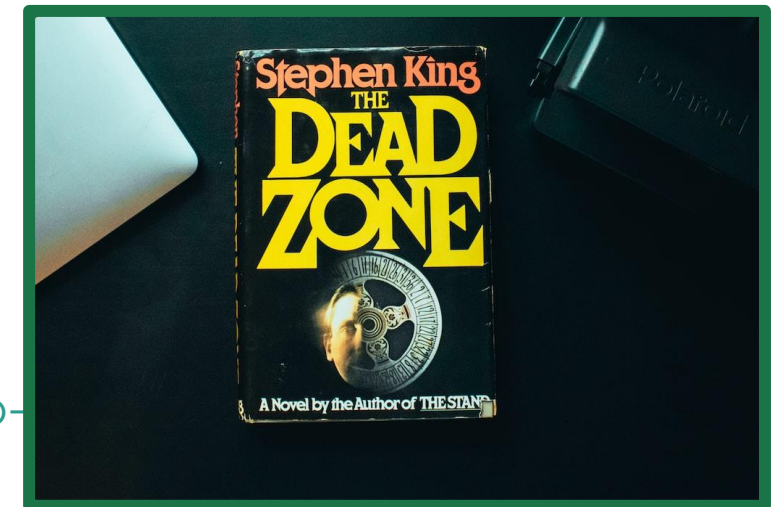
- Para llevar a cabo el proyecto se realizaron diversas técnicas de NLP para la visualización y comprensión de los datos
- Se utilizó Regresión Logística para realizar la clasificación
- Además se aplicaron de diversas métricas para medir el desempeño del modelo

INTEGRANTES

1. Rodrigo Garmendia
2. Jorge Arista
3. Isaac Moreno

ENLACE

<https://github.com/IsaacJumito/IsaacJumito-BEDU-ProyectoEquipo5-AnalisisdatosPython>





**La base de datos se obtuvo de la plataforma
Kaggle, dicha base cuenta con un total de 50,000
reseñas de películas extraídas de la página de
IMDB**

1 Presentación

2 Exploración de datos

BASE DE DATOS

- El *dataset* se subió en cuatro archivos distintos en Github por motivos de limitación de espacio. Posteriormente estos archivos se leyeron y unieron nuevamente en el *Notebook*

Se importan y leen los 4 archivos que contienen el *data set* a explorar

```
data1 = pd.read_csv('https://raw.githubusercontent.com/ruderikissa/BEDU/main/neg_rev1', index_col=0)
data2 = pd.read_csv('https://raw.githubusercontent.com/ruderikissa/BEDU/main/neg_rev2', index_col=0)
data3 = pd.read_csv('https://raw.githubusercontent.com/ruderikissa/BEDU/main/pos_rev1', index_col=0)
data4 = pd.read_csv('https://raw.githubusercontent.com/ruderikissa/BEDU/main/pos_rev2', index_col=0)
```

Se unen los archivos y se verifica que se haya realizada correctamente

```
[ ] data = pd.concat([data1,data2,data3,data4],ignore_index=True)
data
```

	review	sentiment
0	Basically there's a family where a little boy ...	negative
1	This show was an amazing, fresh & innovative i...	negative
2	Encouraged by the positive comments about this...	negative

1 Presentación

2 Exploración de datos

BIBLIOTECAS

- Se importaron las bibliotecas requeridas para la exploración y análisis de los datos. Dado que estamos trabajando con archivos de texto, entre estas herramientas se encuentra NLTK para el procesamiento de lenguaje natural

```
[ ] !pip install nltk
import nltk # herramientas para el análisis de NLP
import seaborn as sns
from pylab import * # numpy y matplotlib.pyplot
import pandas as pd
import re # análisis de patrones en str
nltk.download('punkt') # lista de palabras vacías
nltk.download('stopwords') # y puntuación en inglés
```

1 Presentación

2 Exploración de datos

INFORMACIÓN

- Notamos que el data set únicamente cuenta con dos columnas: la reseña hecha y la valoración positiva o negativa de la misma, ambas columnas de tipo 'object'
- Revisamos cuántas reseñas hay de acuerdo a cada sentimiento
- Para hacer una análisis con mayor detalle dividiremos el data frame en dos: cada uno conteniendo las reseñas de un tipo de sentimiento.

```
# Data frame con las reseñas positivas
pos_rev = data.query('sentiment== "positive"').reset_index(drop=True)
pos_rev
```

	review	sentiment
0	One of the other reviewers has mentioned that ...	positive
1	A wonderful little production. The...	positive
2	I thought this was a wonderful way to spend ti...	positive
3	Petter Mattei's "Love in the Time of Money" is...	positive
4	Probably my all-time favorite movie, a story o...	positive
...
24995	I loved it, having been a fan of the original ...	positive

```
data.groupby('sentiment').count()
```

	review
sentiment	
negative	25000
positive	25000

A vintage slide projector is positioned on a wooden surface. The projector is a light-colored, boxy device with a lens on the right side and a slide tray on top. It has several control knobs and buttons on its front panel, including one labeled 'FOCUS'. The background is dark and filled with a thick, white, smoky or misty atmosphere. The lighting is dramatic, highlighting the projector against the dark background.

**Una vez explorado el data set, procedemos a
analizar la información con la que contamos**

1 Presentación

2 Exploración de datos

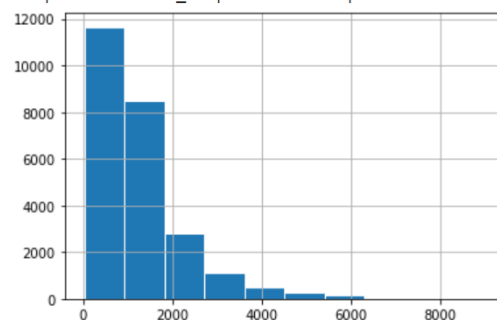
3 Análisis de datos y visualización

CANTIDAD DE CARACTERES

- Comparamos las reseñas de acuerdo al sentimiento conforme a la división que hicimos previamente
- Empezamos explorando la distribución de la cantidad de caracteres por reseña

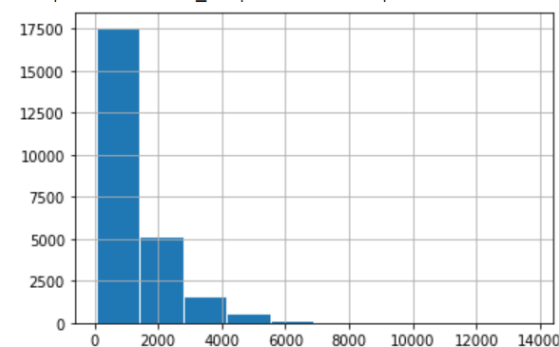
```
[ ] # Histograma de cantidad de caracteres para reseñas negativas
neg_rev_len = neg_rev.review.apply(lambda x: len(x))
neg_rev_len
neg_rev_len.hist( ec='w' )
```

<matplotlib.axes._subplots.AxesSubplot at 0x7f994ecc0c50>



```
[ ] # Histograma de cantidad de caracteres para reseñas positivas
pos_rev_len = pos_rev.review.apply(lambda x: len(x))
pos_rev_len.hist(ec='w')
```

<matplotlib.axes._subplots.AxesSubplot at 0x7f994ec0f9d0>



1 Presentación

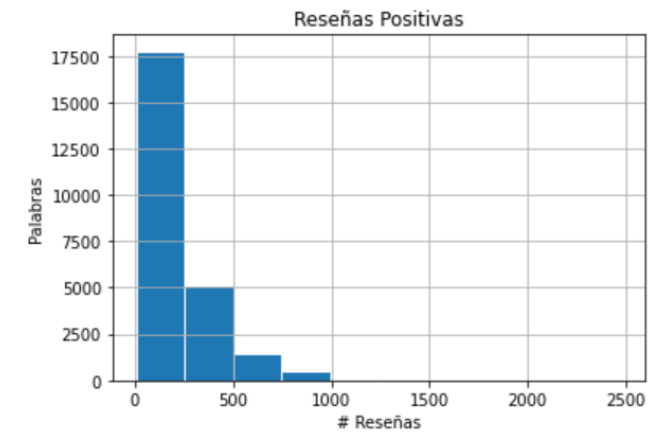
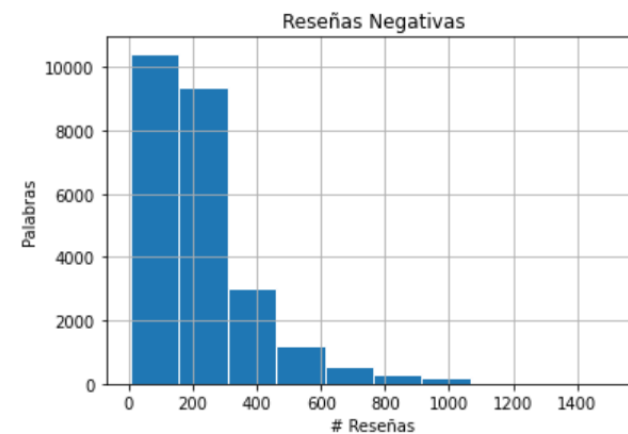
2 Exploración de datos

3 Análisis de datos y visualización

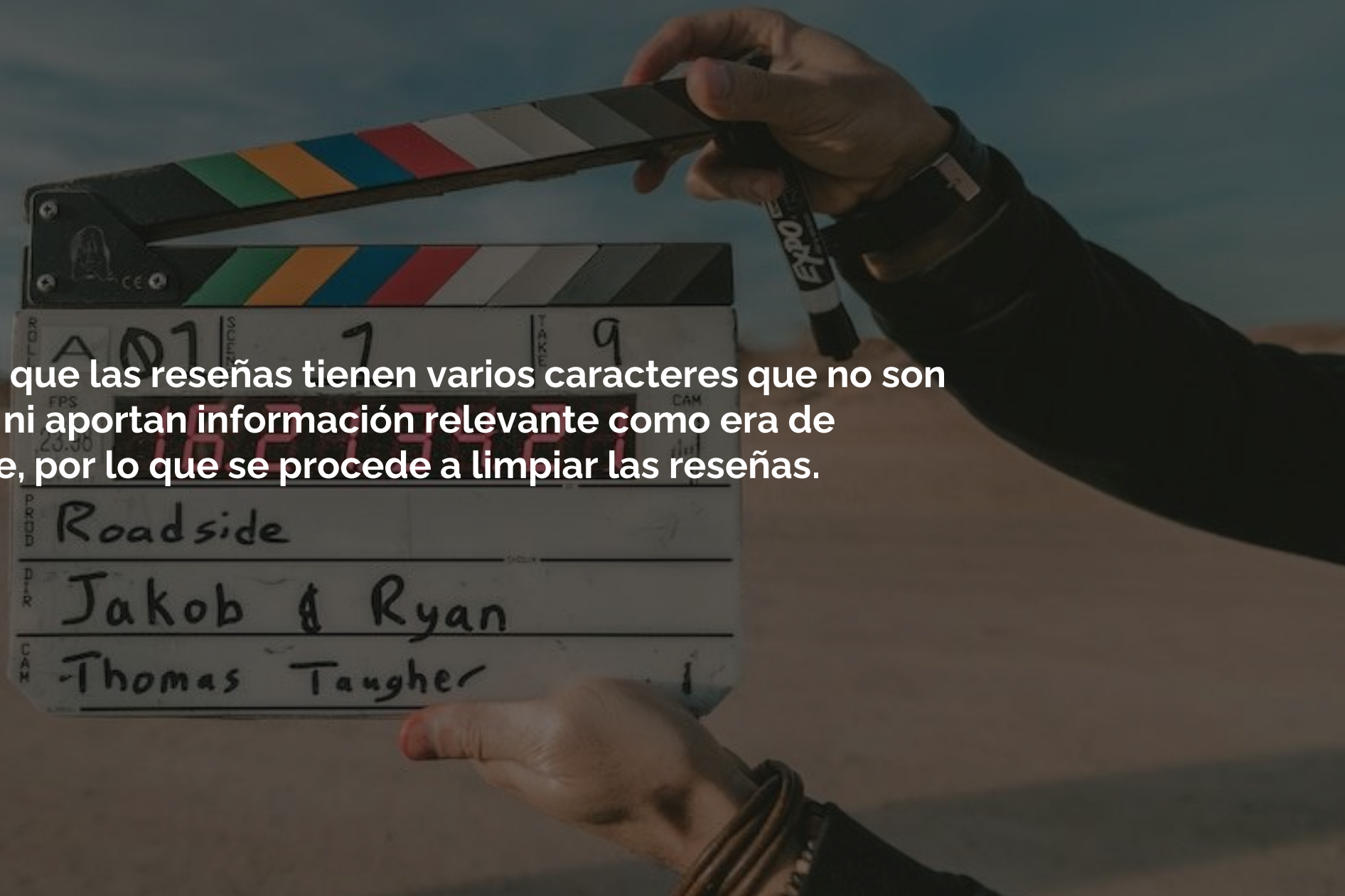
CANTIDAD DE PALABRAS

- Después vimos la cantidad de palabras de reseñas por cada clasificación
- Definimos la función contadora de palabras y graficamos

```
[ ] #Definimos la función contadora d palabras
def word_count(review):
    words = review.split()
    return len(words)
```



Notamos que las reseñas tienen varios caracteres que no son palabras ni aportan información relevante como era de esperarse, por lo que se procede a limpiar las reseñas.



1

Presentación

2

Exploración de
datos

3

Análisis de datos
y visualización

4

Limpieza de
datos

LIMPIEZA

- Definimos la función de limpieza y la aplicamos a las reseñas

```
[ ] # Se define la función de limpieza para las reseñas
    from nltk.corpus import stopwords
    def clean_review(review):
        # Se transforman en minúsculas todas las letras
        review = review.lower()
        # Se eliminan los caracteres de html
        review = re.sub(r'<.*?>', ' ', review)
        # Se eliminan las ligas de internet
        review = re.sub(r'http[s]?.*', ' ', review)
        # Se eliminan los dígitos
        review = re.sub(r'\d', ' ', review)
        # Se eliminan el resto de caracteres que no sean letras.
        review = re.sub(r'[^a-zA-Z]', ' ', review)
        # Se eliminan posibles espacios extras
        review = review.strip()
        return review
```


1

Presentación

2

Exploración de
datos

3

Análisis de datos
y visualización

4

Limpieza de
datos

LIMPIEZA

- Ejemplo previo y posterior a la limpieza por cada sentimiento

1. Reseña negativa sin limpieza:

```
[ ] neg_example = neg_rev.loc[24994][0]
neg_example
```

'Robert Colomb has two full-time jobs. He's known throughout the world as a globetrotting TV reporter. Less well-known but equally effortful are his exploits as a full-time philanderer.

I saw 'Vivre pour Vivre' dubbed in English with the title 'Live for Life.' Some life! Robert seems to always have at least three women in his life: one mistress on her way out, one on her way in, and the cheated wife at home. It helps that Robert is a glib liar. Among his most useful lies are 'I'll call you tomorrow' and 'My work took longer than planned.' He spends a lot of time and money on planes, trains and hotel rooms for his succession of liaisons. You wonder when this guy will get caught with his pants down.

Some may find his life exciting, but I thought it to be tedious. His companions, including his wife, Catherine, are all attractive and desirable women. But his lifestyle is so hectic and he is so deceitful, you wonder if he's enjoying all this.

Adding to the te...'

2. Reseña negativa limpia:

```
[ ] neg_clean_example = neg_rev_clean[24994]
neg_clean_example
```

'robert colomb has two full time jobs he s known throughout the world as a globetrotting tv reporter less well known but equally effortful are his exploits as a full time philanderer i saw vivre pour vivre dubbed in english with the title live for life some life robert seems to always have at least three women in his life one mistress on her way out one on her way in and the cheated wife at home it helps that robert is a glib liar among his most useful lies are i ll call you tomorrow and my work took longer than planned he spends a lot of time and money on planes trains and hotel rooms for his succession of liaisons you wonder when this guy will get caught with his pants down some may find his life exciting but i thought it to be tedious his companions including his wife catherine are all attractive and desirable women but his lifestyle is so hectic and he is so deceitful you wonder if he s enjoying all this adding to the tedium is considerable footage that doesn t further the plot ther...'

1 Presentación

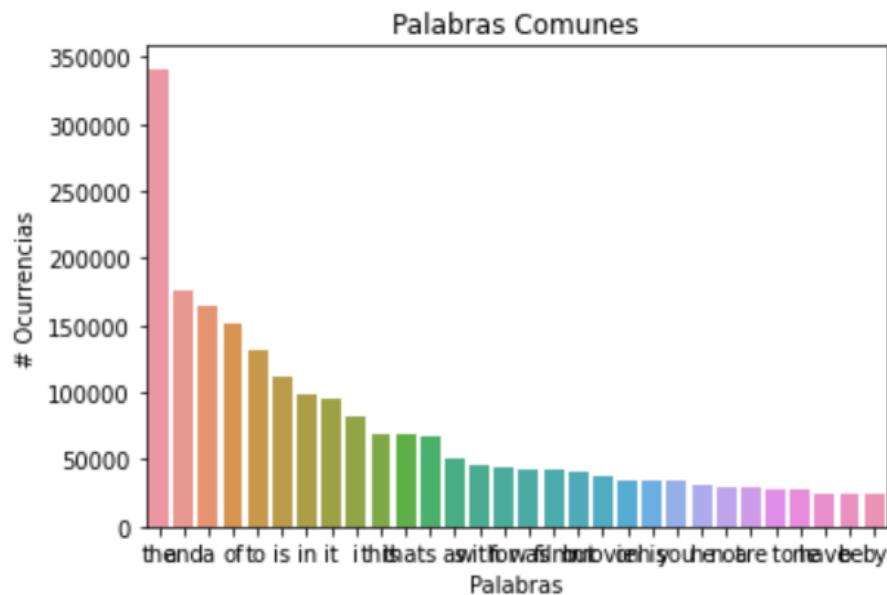
2 Exploración de datos

3 Análisis de datos y visualización

4 Limpieza de datos

FRECUENCIA DE PALABRAS

- Ya hecha la limpieza, procedemos a 'tokenizar' las reseñas para seguir con nuestro análisis
- Creamos un 'corpus' de palabras que contienen cada tipo de reseña
- Se crea un diccionario con los frecuencias de cada palabra en los data frames y visualizamos cuáles son las más frecuentes para cada sentimiento

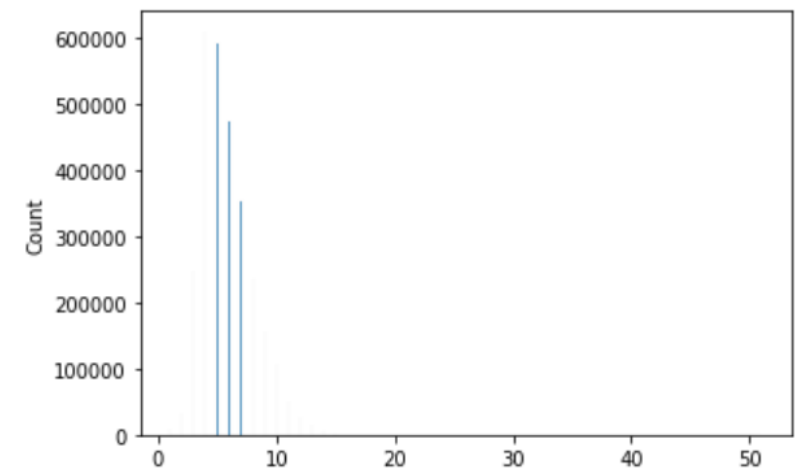
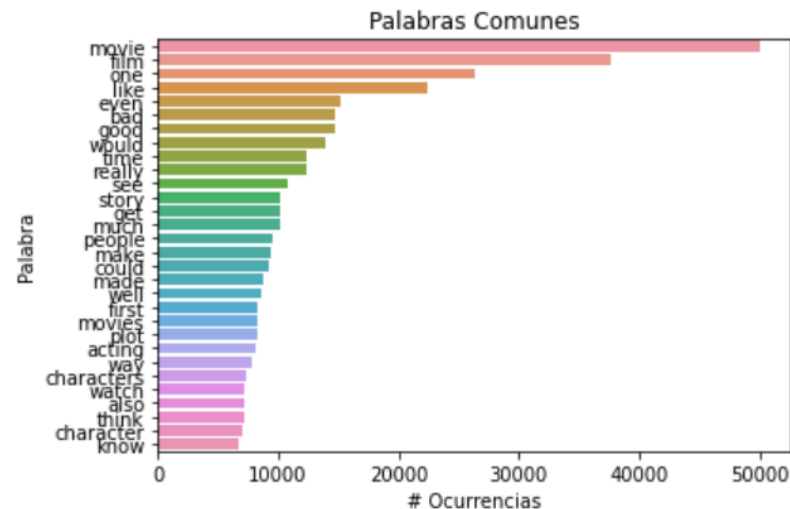


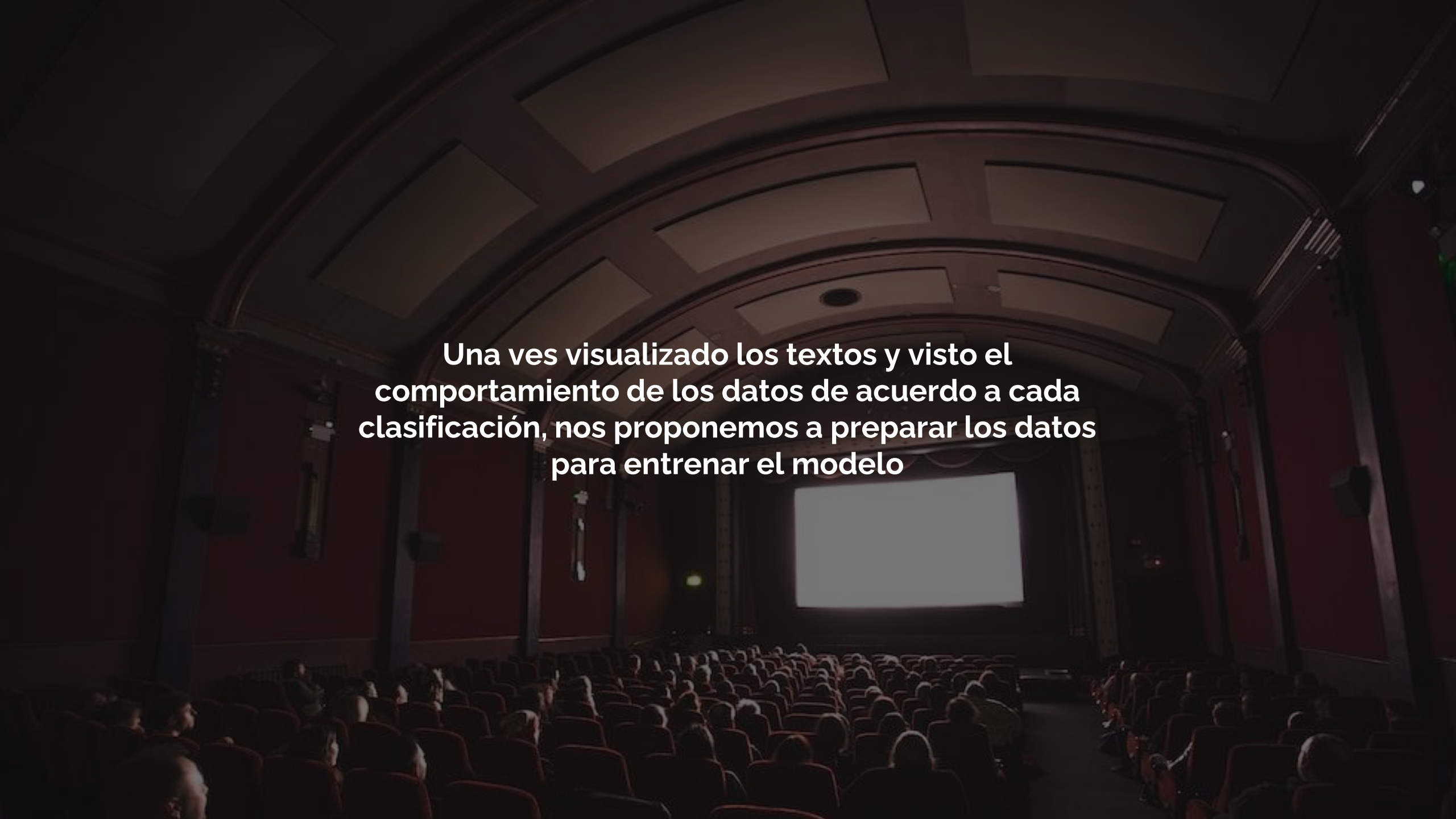
- 1 Presentación
- 2 Exploración de datos
- 3 Análisis de datos y visualización
- 4 Limpieza de datos

FRECUENCIA DE PALABRAS

- Como era de esperarse las palabras más frecuentes son palabras vacías, es decir sirven como conectores sintácticos pero no aportan información, por lo que procedemos a eliminarlas
- Creamos nuevamente el diccionario de frecuencias sin las palabras vacías y visualizamos palabras comunes y longitud de las mismas

```
[ ] stopwords = stopwords.words('english') #Palabras vacías en inglés  
neg_corpus_clean = [word for word in neg_corpus if word not in stopwords]  
pos_corpus_clean = [word for word in pos_corpus if word not in stopwords]
```





Una vez visualizado los textos y visto el comportamiento de los datos de acuerdo a cada clasificación, nos proponemos a preparar los datos para entrenar el modelo

- 1 Presentación
- 2 Exploración de datos
- 3 Análisis de datos y visualización
- 4 Limpieza de datos
- 5 Preprocesamiento

DISEÑO DE COLUMNAS

- Empezamos creando una nueva columna donde se almacenaran los datos ya preparados. Lo primero que hicimos es aplicar la función `clean_review` definida previamente
- Luego, 'estematizamos' las palabras, es decir, reducimos a su raíz las palabras que tienen un origen común
- Codificamos numéricamente los sentimientos, asignamos el valor de 0 para los negativos y de 1 a los positivos

```
[ ] stemmer = nltk.PorterStemmer()
def more_cleaning(review):
    stemmer = nltk.PorterStemmer()
    words = [stemmer.stem(word) for word in review if word not in stopwords]
    return words
```

	review	sentiment	review_clean	sentiment_encoded
0	One of the other reviewers has mentioned that ...	positive	[one, review, mention, watch, oz, episod, hook...	1
1	A wonderful little production. The...	positive	[wonder, littl, product, film, techniqu, unass...	1
2	I thought this was a wonderful way to spend ti...	positive	[thought, wonder, way, spend, time, hot, summe...	1
3	Petter Mattei's "Love in the Time of Money" is...	positive	[petter, mattei, love, time, money, visual, st...	1
4	Probably my all-time favorite movie, a story o...	positive	[probabl, time, favorit, movi, stori, selfless...	1

- 1 Presentación
- 2 Exploración de datos
- 3 Análisis de datos y visualización
- 4 Limpieza de datos
- 5 Preprocesamiento

CONJUNTOS PROPORCIONALES

- Después dividimos nuestro data set en conjuntos de entrenamiento y prueba. Para garantizar que haya una proporción idéntica de datos con las etiquetas 0 y 1 en ambos conjuntos, lo que hacemos es dividir por separado los data frames 'pos_rev' y 'neg_rev' en prueba y entrenamiento para posteriormente unir los similares
- Creamos una máscara con los índices de 'train_pos' y aplicamos a los demás conjuntos, esto para simplificar la separación y garantizar la misma proporción de reseñas con las dos etiquetas en ambos conjuntos

```
[ ] # Ordenamos los índices para optimizar el cómputo
idx = train_pos.sort_index().index.to_list()
# Creamos la máscara
mask = np.array([True if x not in idx else False for x in range(25000)])
# Aplicamos la máscara a 'pos_rev'
test_pos = pos_rev[mask]
test_pos
```

	review	sentiment	review_clean	sentiment_encoded
6	If you like original gut wrenching laughter yo...	positive	[like, origin, gut, wrench, laughter, like, mo...	1
7	This a fantastic movie of three prisoners who ...	positive	[fantast, movi, three, prison, becom, famou, o...	1
10	After the success of Die Hard and it's sequels...	positive	[success, die, hard, sequel, surpris, realli, ...	1
14	'War movie' is a Hollywood genre that has been...	positive	[war, movi, hollywood, genr, done, redon, mani...	1
22	Preston Sturgis' THE POWER AND THE GLORY was u...	positive	[preston, sturgi, power, glori, unseen, public...	1
...
24967	I first saw this movie in the night program of...	positive	[first, saw, movi, night, program, one, favour...	1

- 1 Presentación
- 2 Exploración de datos
- 3 Análisis de datos y visualización
- 4 Limpieza de datos
- 5 Preprocesamiento

APROXIMACIÓN FRECUENTISTA

- Para entrenar el modelo vamos a realizar una aproximación frecuentista, es decir, los valores con los que vamos a alimentar el algoritmo representan las veces que cada palabra se encontró en las reseñas positivas y negativas
- Unimos los conjuntos positivos y negativos para entrenamiento y prueba, mezclando el orden para no sesgar el algoritmo

```
[ ] # Creamos los corpus para ambas etiquetas con los datos de entrenamiento
train_neg_corpus = [words[i] for words in train_neg.review_clean for i in range(len(words))]
train_pos_corpus = [words[i] for words in train_pos.review_clean for i in range(len(words))]
```

```
[ ] print('Longitud del corpus positivo:\t', len(train_pos_corpus))
print('\nLongitud del corpus negativo', len(train_neg_corpus))
```

```
Longitud del corpus positivo:    2397697
```

```
Longitud del corpus negativo 2341733
```

```
[ ] data_test.head()
```


	review	sentiment	review_clean	sentiment_encoded
5522	This Columbo episode is probably noted more fo...	positive	[columbo, episod, probabl, note, director, ste...	1
5361	Michael (played by Steven Robertson) has cereb...	positive	[michael, play, steven, robertson, cereb, pals...	1
5372	A remarkable piece of documentary, giving a vi...	positive	[remark, piec, documentari, give, vivid, depic...	1
4403	First of all, I was expecting "Caged Heat" to ...	negative	[first, expect, cage, heat, along, line, llsa,...	0
4231	Think "stage play". This is worth seeing once ...	negative	[think, stage, play, worth, see, perform, lion...	0

- 1 Presentación
- 2 Exploración de datos
- 3 Análisis de datos y visualización
- 4 Limpieza de datos
- 5 Preprocesamiento

TRADUCIR TEXTO A NÚMEROS

- Traducimos el texto a números de forma que el algoritmo pueda leerlo, así que creamos la función 'vectorize_review', la cual a cada palabra la convierte en un vector con entradas iguales a la frecuencia con que se encuentran en cada corpus y la suma de esos vectores representan el vector asociado a cada reseña
- Después realizamos una serie de pasos para preparar la información y que esta sea fácil de procesar al momento de entrenar. Ej: Definimos el 'target' como el sentimiento codificado y los 'features' como la reseña vectorizada

	review	sentiment	review_clean	sentiment_encoded	review_vectorized
28763	I liked this a lot. The camera ang...	positive	[like, lot, camera, angl, cool, jumpi, like, b...	1	[[1.0, 267735.0, 284534.0]]
31760	Watching Fire and Ice for the first time remin...	positive	[watch, fire, ice, first, time, remind, experi...	1	[[1.0, 738725.0, 756284.0]]
16002	The final pairing of Nelson Eddy and Jeanette ...	negative	[final, pair, nelson, eddi, jeanett, macdonald...	0	[[1.0, 132506.0, 134226.0]]
32249	If you like the excitement of a good submarine...	positive	[like, excit, good, submarin, drama, fun, good...	1	[[1.0, 231787.0, 225605.0]]
34400	This indie film looks at the lives of a group ...	positive	[indi, film, look, live, group, peopl, take, a...	1	[[1.0, 300639.0, 272523.0]]

A nighttime photograph of the Chicago Theatre, a historic landmark in Chicago. The building is illuminated, with its iconic marquee and vertical sign clearly visible. The marquee displays the text "#IMOMSOHARD-MOM'S NIGHT OUT ROUND 2" and "JULY 12". The vertical sign reads "CHICAGO". The surrounding area includes modern glass-fronted buildings and a street with some vehicles and pedestrians.

Una vez finalizado el preprocesamiento, nos
proponemos a crear el modelo

- 1 Presentación
- 2 Exploración de datos
- 3 Análisis de datos y visualización
- 4 Limpieza de datos
- 5 Preprocesamiento
- 6 Entrenamiento del modelo

ENTRENAMIENTO DEL MODELO

- Dado que nuestro objetivo es predecir una clasificación binaria, ocuparemos el algoritmo de Regresión Logística

```
[ ] # Importamos la regresión logística
    from sklearn.linear_model import LogisticRegression
```

```
[ ] # Definimos el modelo y sus hiperparámetros
    LR = LogisticRegression(C=0.1, solver='saga')
```

```
[ ] # Entrenamos el modelo
    LR.fit(x_train,y_train)

    LogisticRegression(C=0.1, solver='saga')
```


1989/COLOR/103 MIN./R/STEREO/

-177AK5

CAT'S EYE HORROR

STEPHEN KING'S

MAXIMUM OVERDRIVE

VHS 395

GRAVEYARD SHIFT



A Paramount Communications Company

32512

1990/COLOR/89 MIN./R/STEREO/

NEW WORLD VIDEO PRESENTS

CREEPSHOW 2

CREEPSHOW

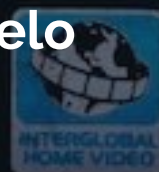
VHS
VA5081

GHOULIES



VHS
VA5081

GHOULIES



STEPHEN KING'S NIGHTSHIFT COLLECTION VOLUME TWO

STEPHEN KING'S SLEEPWALKERS

COLUMBIA
TRISTAR



HOME VIDEO

STEPHEN KING'S

SILVER BULLET



Con el modelo entrenado, nos proponemos a
realizar las predicciones y evaluar la eficacia del
modelo

- 1 Presentación
- 2 Exploración de datos
- 3 Análisis de datos y visualización
- 4 Limpieza de datos
- 5 Preprocesamiento
- 6 Entrenamiento del modelo
- 7 Evaluación del modelo

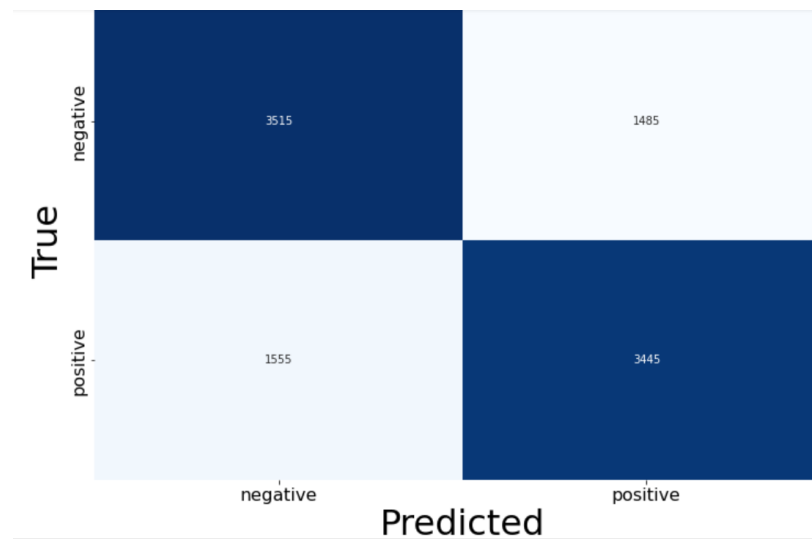
EVALUACIÓN DEL MODELO

- Empezamos midiendo la precisión del modelo para darnos una idea de su desempeño
- ***Vemos que aproximadamente 70% de las reseñas las clasificó correctamente, lo cual consideramos como un desempeño regular***

```
[ ] print('Precisión del modelo:\t ',(accuracy_score(y_test, y_hat)))
```

```
Precisión del modelo:      0.696
```

- Creamos una matriz de confusión para visualizar las clasificaciones hechas y la cantidad de falsos positivos y falsos negativos

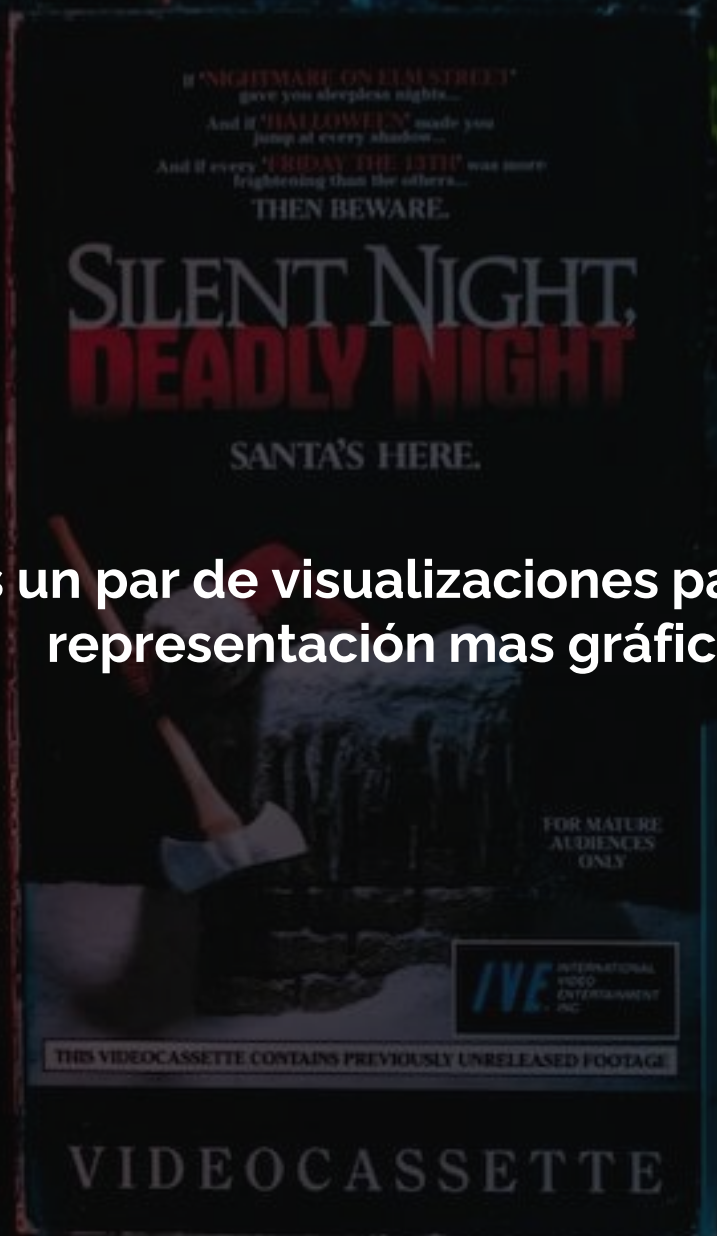


- 1 Presentación
- 2 Exploración de datos
- 3 Análisis de datos y visualización
- 4 Limpieza de datos
- 5 Preprocesamiento
- 6 Entrenamiento del modelo
- 7 Evaluación del modelo

EVALUACIÓN DEL MODELO

- Por último, creamos el reporte con las diferentes métricas que nos enriquecen la información previamente visualizada
- Realizamos un entrenamiento supervisado el cual nos dió una confianza al momento de realizar el análisis de los resultados cercano al 70 %

	negative	positive	accuracy	macro avg	weighted avg
precision	0.703000	0.689000	0.696	0.696000	0.696098
recall	0.693294	0.698783	0.696	0.696038	0.696000
f1-score	0.698113	0.693857	0.696	0.695985	0.696015
support	5070.000000	4930.000000	0.696	10000.000000	10000.000000



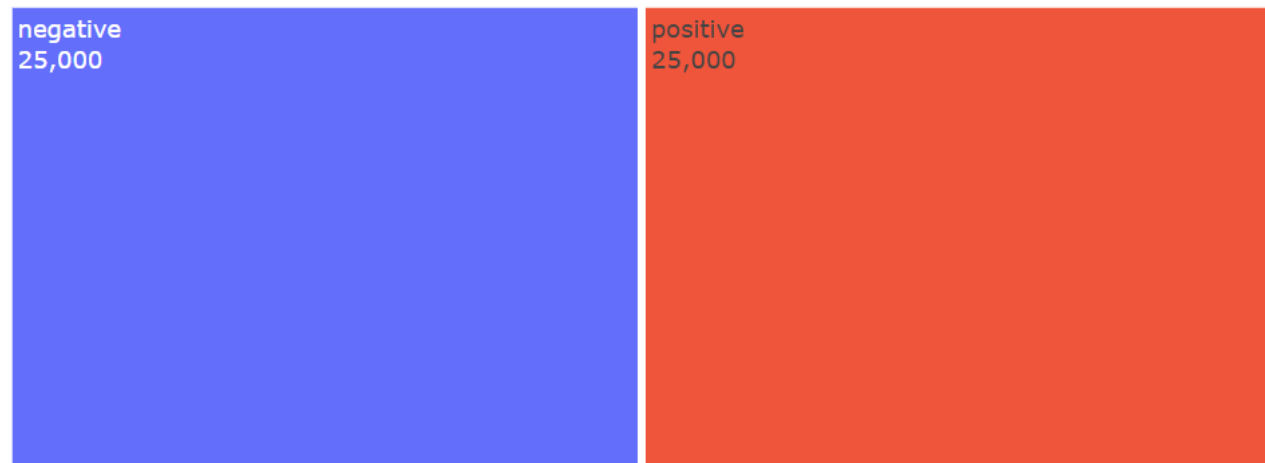
Añadimos un par de visualizaciones para tener una representación mas gráfica

- 1 Presentación
- 2 Exploración de datos
- 3 Análisis de datos y visualización
- 4 Limpieza de datos
- 5 Preprocesamiento
- 6 Entrenamiento del modelo
- 7 Evaluación del modelo
- 8 Visualización

VISUALIZACIÓN

- Para tener una visualización mas gráfica de nuestra data decidimos realizar dos formas de visualizarlas
- Primero haremos un "TreeMap" con nuestros valores de reseñas negativas y positivas

```
[ ] import plotly.express as px
dataTree=pd.DataFrame(data.groupby('sentiment').count())#Realizamos la agrupación para obtener el total de Reviews
dataTree['sentiment'] = dataTree.index #Convertimos nuestros indices en las etiquetas a usar
fig = px.treemap(dataTree, path=['sentiment'], values='review', width=800, height=400)
fig.data[0].textinfo='label+value'
fig.layout.hovermode = False
fig.show()
```



1

Presentación

2

Exploración de datos

3

Análisis de datos y visualización

4

Limpieza de datos

5

Preprocesamiento

6

Entrenamiento del modelo

7

Evaluación del modelo

8

Visualización

VISUALIZACIÓN

- La siguiente será un "WordCloud", con la cual podremos identificar de manera gráfica las palabras mas usadas según cada clasificación
- Podemos apreciar que las palabras mas usadas en comentarios **negativos son Bad y Film**, mientras que de las valoraciones **positivas Son Wonder, Right y Love**

Valoraciones Negativas

```
[ ] wordcloud.generate(str(neg_words))
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis('off')
```

(-0.5, 399.5, 199.5, -0.5)



Valoraciones Positivas

```
[ ] wordcloud = WordCloud(colormap='autumn')
wordcloud.generate(str(pos_words))
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis('off')
```

(-0.5, 399.5, 199.5, -0.5)



Conclusión

