En este artículo, te guiaré a través del proceso para realizar análisis de sentimiento en una gran cantidad de datos. Utilizaremos el archivo Reviews.csv del conjunto de datos de Amazon Fine Food Reviews de Kaggle para realizar el análisis.

Utilizaremos Jupyter Notebook para todos los análisis y visualización, pero cualquier IDE de Python hará el trabajo.

Paso 1: Leer el dataframe

Revisamos los primeros datos del dataframe:

	Id	ProductId	UserId	ProfileName	HelpfulnessNumerator	HelpfulnessDenominator	Score	Time	Summary	Text
0	1	B001E4KFG0	A3SGXH7AUHU8GW	delmartian	1	1	5	1303862400	Good Quality Dog Food	I have bought several of the Vitality canned d...
1	2	B00813GRG4	A1D87F6ZCVE5NK	dll pa	0	0	1	1346976000	Not as Advertised	Product arrived labeled as Jumbo Salted Peanut...
2	3	B000LQOCH0	ABXLMWJIXXAIN	Natalia Corres "Natalia Corres"	1	1	4	1219017600	"Delight" says it all	This is a confection that has been around a fe...
3	4	B000UA0QIQ	A395BORC6FGVXV	Karl	3	3	2	1307923200	Cough Medicine	If you are looking for the secret ingredient i...
4	5	B006K2ZZ7K	A1UQRSCLF8GW1T	Michael D. Bigham "M. Wassir"	0	0	5	1350777600	Great taffy	Great taffy at a great price. There was a wid...

Podemos observar que el dataframe contiene productos, usuarios e información de la reseña.

Los datos que más vamos a utilizar para el análisis será “Summary”, “Text” y “Score”.

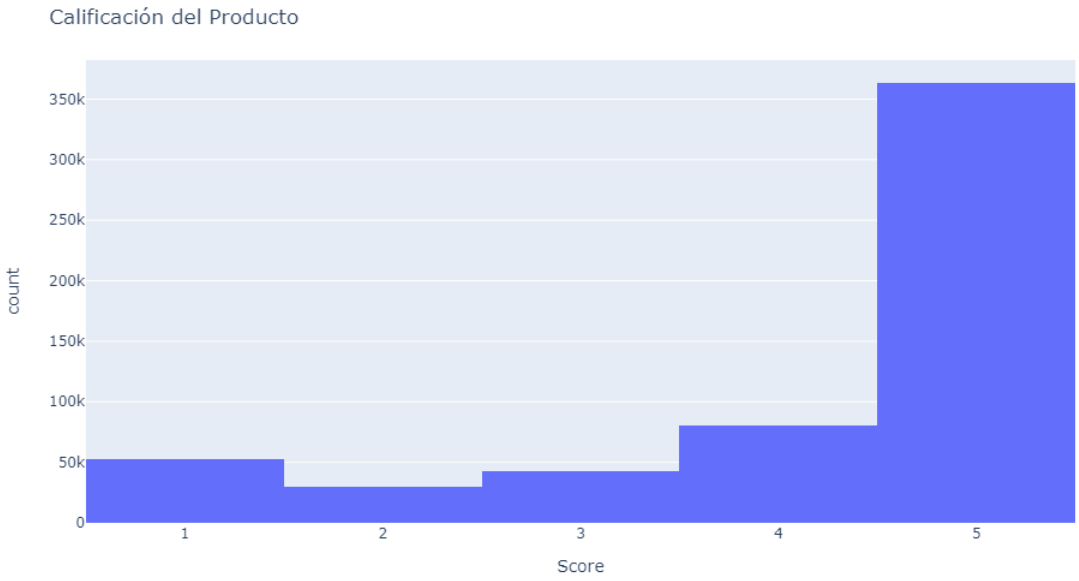
- **Text** – Es la variable que contiene la reseña completa.
- **Summary** – Es el resumen de la reseña.
- **Score** – Es la calificación del producto provista por el cliente.

Paso 2: Análisis de Datos

Ahora, lo que haremos será mirar la variable “Score” para revisar si la mayoría de las calificaciones son positivas o negativas.

Para realizar esto utilizaremos la librería **Plotly,** que tendrás que tenerla instalada previamente.

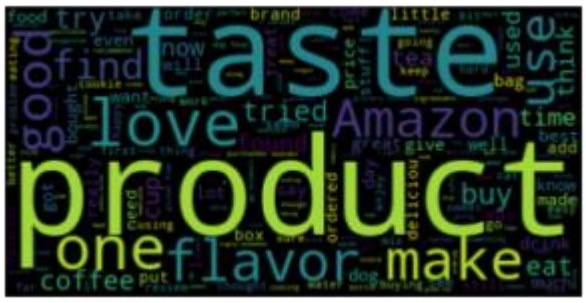
El resultado del gráfico se ve como esto:



Desde de acá podemos ver que la mayoría de las calificaciones de los clientes son positivas. Esto nos conlleva a pensar que la mayoría de las reseñas serán también positivas, las cuales analizaremos en breve.

Ahora crearemos la nube de palabras "wordclouds" más utilizadas en las reseñas empleando la librería "NLTK" y la librería "wordcloud".

Ejecutando el correo obtenemos la siguiente nube de palabras:



Se puede identificar que palabras populares son "Taste", "Love", "Product", "Amazon", "One". Estas palabras son en su mayoría positiva, lo que también nos indica que la mayoría de las reseñas del conjunto de datos expresan un sentimiento positivo.

Paso 3: Clasificando los tweets.

En este paso vamos clasificar las reseñas como “positivas” y “negativas”, de esta forma podemos usarla nuestros datos como entrenamiento para nuestro modelo de clasificación de sentimiento.

Reseñas positivas serán clasificadas como +1, y reseñas negativas serán clasificadas como -1.

Clasificaremos todas las reseñas con “Score”>3 como +1, “Score”<3 serán negativas. Se eliminarán las reseñas con “Score”=3 por que son valores neutrales. Nuestro modelo solo clasificará reseñas como positivas o negativas.

Revisando nuevamente nuestro data frame, ahora podemos observar una nueva columna llamada “sentiment”.

	Id	ProductId	UserId	ProfileName	HelpfulnessNumerator	HelpfulnessDenominator	Score	Time	Summary	Text	sentiment
0	1	B001E4KFG0	A3SGXH7AUHU8GW	delmartian	1	1	5	1303862400	Good Quality Dog Food	I have bought several of the Vitality canned d...	1
1	2	B00813GRG4	A1D87F6ZCVE5NK	dll pa	0	0	1	1346976000	Not as Advertised	Product arrived labeled as Jumbo Salted Peanut...	-1
2	3	B000LQOCH0	ABXLMWJIXXAIN	Natalia Corres "Natalia Corres"	1	1	4	1219017600	"Delight" says it all	This is a confection that has been around a fe...	1
3	4	B000UA0QIQ	A395BORC6FGVXV	Karl	3	3	2	1307923200	Cough Medicine	If you are looking for the secret ingredient i...	-1
4	5	B006K2ZZ7K	A1UQRSCLF8GW1T	Michael D. Bigham "M. Wassir"	0	0	5	1350777600	Great taffy	Great taffy at a great price. There was a wid...	1

Paso 4: Más análisis de datos.

Ahora que ya hemos clasificado nuestros tweets en positivos y negativos, vamos a construir una nube de palabras para ambos casos.

Primero, crearemos dos dataframes, uno para las reseñas positivas y otro para las reseñas negativas.

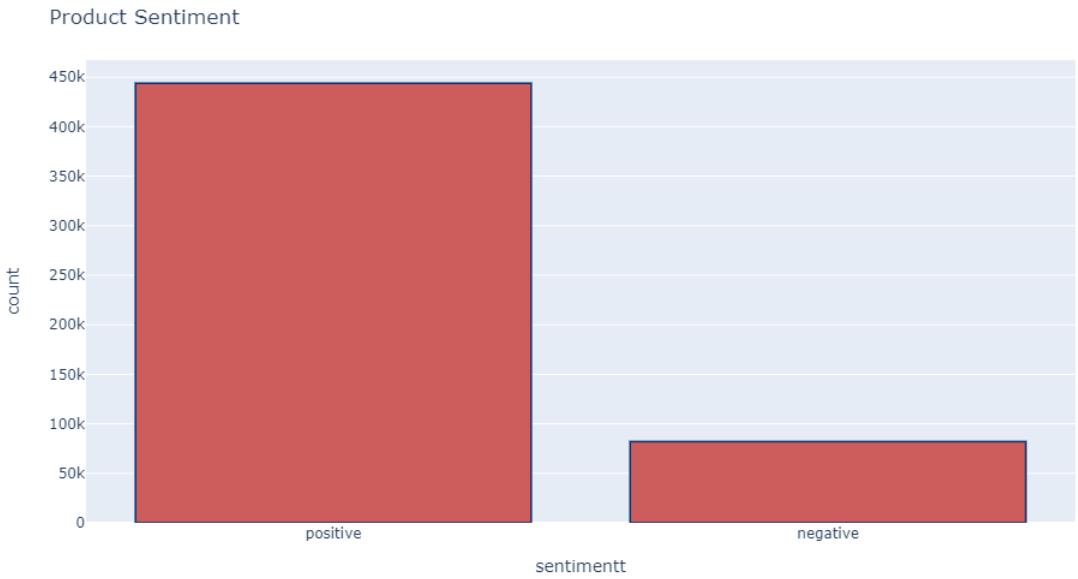
Nube de palabras para Positivos

Nube de palabras para Negativos

Tal como se muestra en las dos nubes generadas, la nube positiva refleja palabras como “love”, “best” y “delicious”, mientras que en la nube de sentimientos negativos las palabras que más aparecen son “dissapointed” y “horrible”.

Las palabras “good” y “great” inicialmente aparecían en la nube de sentimientos negativos, a pesar de ser palabras positivas. Esto se debe probablemente que estas palabras se usaron en un contexto negativo como, “not good”, “not great”. Por este motivo es que fueron removidas de la nube de palabras.

Finalmente podemos dar una mirada a la distribución de las reseñas a traves del dataset:



Paso 5: Construir el modelo

Finalmente podemos construir el modelo de análisis de sentimiento.

Este modelo tomara las reseñas como entrada (input). Luego proporcionará una predicción si la reseña es positiva o negativa.

Esta tarea es una modelo de clasificación, por lo que entrenaremos nuestro modelo en una simple regresión logística.

Como referencia, volvamos a revisar el dataframe.

	Id	ProductId	UserId	ProfileName	HelpfulnessNumerator	HelpfulnessDenominator	Score	Time	Summary	Text	sentiment
0	1	B001E4KFG0	A3SGXH7AUHU8GW	delmartian	1	1	5	1303862400	Good Quality Dog Food	I have bought several of the Vitality canned d...	1
1	2	B00813GRG4	A1D87F6ZCVE5NK	dll pa	0	0	1	1346976000	Not as Advertised	Product arrived labeled as Jumbo Salted Peanut...	-1
2	3	B000LQOCH0	ABXLMWJIXXAIN	Natalia Corres "Natalia Corres"	1	1	4	1219017600	"Delight" says it all	This is a confection that has been around a fe...	1
3	4	B000UA0QIQ	A395BORC6FGVXV	Karl	3	3	2	1307923200	Cough Medicine	If you are looking for the secret ingredient i...	-1
4	5	B006K2ZZ7K	A1UQRSCLF8GW1T	Michael D. Bigham "M. Wassir"	0	0	5	1350777600	Great taffy	Great taffy at a great price. There was a wid...	1

Solo se requieren algunos pasos adicionales que hacer:

- **Limpieza de datos**

Estaremos usando los datos de la columna “summary” para determinar las predicciones. Primero necesitamos remover todas las **puntaciones** de los datos.

- **Dividimos el dataframe**

Los nuevos datos solo deberían contar con dos columnas “Summary (reseña)” y “sentiment (target)”

Miremos ahora como queda el nuevo dataframe:

	Summary	sentiment
0	Good Quality Dog Food	1
1	Not as Advertised	-1
2	Delight says it all	1
3	Cough Medicine	-1
4	Great taffy	1

Ahora solo resta dividir el dataframe para entrenamiento y para test. 80% de los datos serán empleados para entrenamiento y 20% para test.

- **Creamos una bolsa de palabras**

Ahora, usaremos el “count vectorize” de la librería Scikit-learn.

Esto transformará el texto en nuestro dataframe en una bolsa modelo de palabras, la cual contendrá una matriz dispersa de numero enteros. Se contará e imprimirá el número de ocurrencias de cada palabra.

Necesitaremos convertir el texto en un modelo de bolsa de palabras ya que el algoritmo de regresión logística no puede entender texto.

- **Importamos el modelo de regresión logística.**
- **Dividimos las variables independientes del objetivo (target)**
- **Ajustamos el modelo**
- **Hacemos predicciones**

Finalmente hemos construido un simple modelo de regresión logística y entrenado los datos. También hemos realizado predicciones utilizando el modelo.

Paso 6: Test

Ahora resta realizar las pruebas para determinar la precisión del modelo.

Obtendremos la matrix de confusión tal como se muestra en la siguiente figura:

```
array([[11610,  2303],
       [ 5874, 91530]], dtype=int64)
```

Ahora imprimos el reporte:

	precision	recall	f1-score	support
-1	0.66	0.83	0.74	13913
1	0.98	0.94	0.96	97404
accuracy			0.93	111317
macro avg	0.82	0.89	0.85	111317
weighted avg	0.94	0.93	0.93	111317

La precisión general del modelo en los datos de prueba es del 93%, lo cual es bastente bueno considerando que no se hicieron mayores trabajos de procesado.