



Análisis de sentimiento: Clasificación de reseñas de películas



Integrantes:

- Rodrigo Garmendia
 - Jorge Arista
-

Equipo 5 - Proyecto
Final

A photograph of a man with a beard and long hair, wearing white 3D glasses and holding a large bucket of popcorn. He is looking upwards and to the right with a smile on his face. The background is dark.

Contenido

- Planteamiento
- Objetivo
- Exploración de datos
- Análisis de datos y visualización
- Limpieza de datos
- Preprocesamiento
- Modelo 1: enfoque frecuentista
- Modelo 2: Word embeddings
- Conclusiones



Planteamiento

Obtener información por medio del análisis del lenguaje natural ha sido un tema difícil a través del tiempo debido a la complejidad del lenguaje humano.

Las plataformas digitales se han vuelto el principal medio por el cuál las personas expresan sus opiniones respecto a un servicio o producto.





Nuestro objetivo es clasificar las reseñas de películas en positivas y negativas



Se llevaron a cabo diversas técnicas de NLP para visualizar y comprender los datos.

A photograph showing a person's hands typing on a silver laptop keyboard. The laptop is resting on a light-colored wooden desk. In the background, there is a red brick wall. A white coffee cup with a lid sits on the desk next to the laptop. On the far left edge of the frame, a portion of a notebook is visible.

Exploración de datos

Base de datos obtenida en la plataforma Kaggle.com, cuenta con un total de 50,000 reseñas de películas en IMDB.

Exploración de datos



- El *data set* se subió en 4 archivos distintos por limitación de espacio, se leyó y se unió posteriormente en el *Notebook*.
- Se importan las bibliotecas correspondientes.
- Observamos que el data set solo contiene 2 columnas tipo object: reseña y tipo de valoración (positiva o negativa), por lo que el data frame se divide en dos.

Análisis de datos y visualización

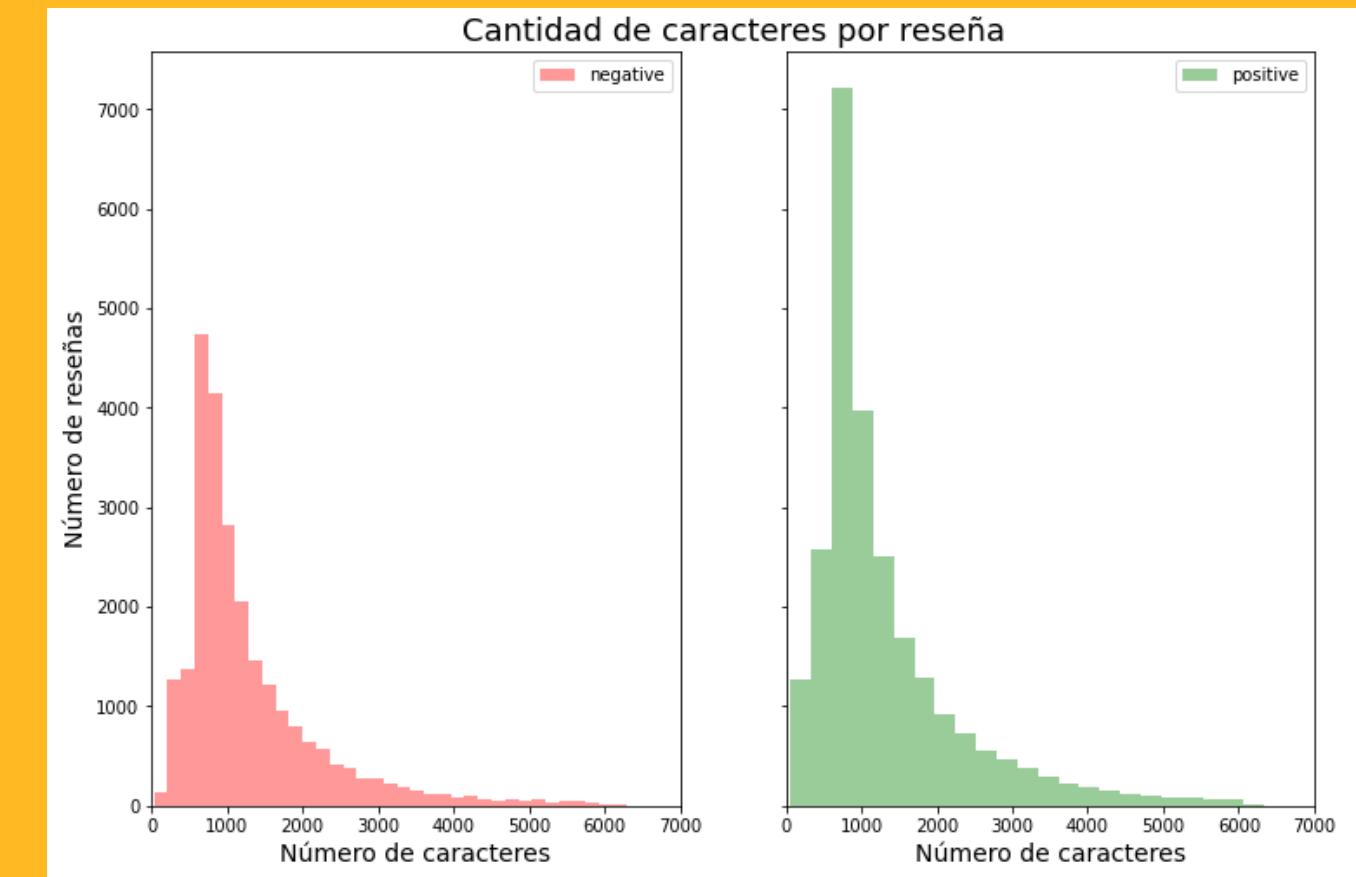
Después de haber realizado la exportación del *data set*, la analizamos comparando el tipo de reseñas.



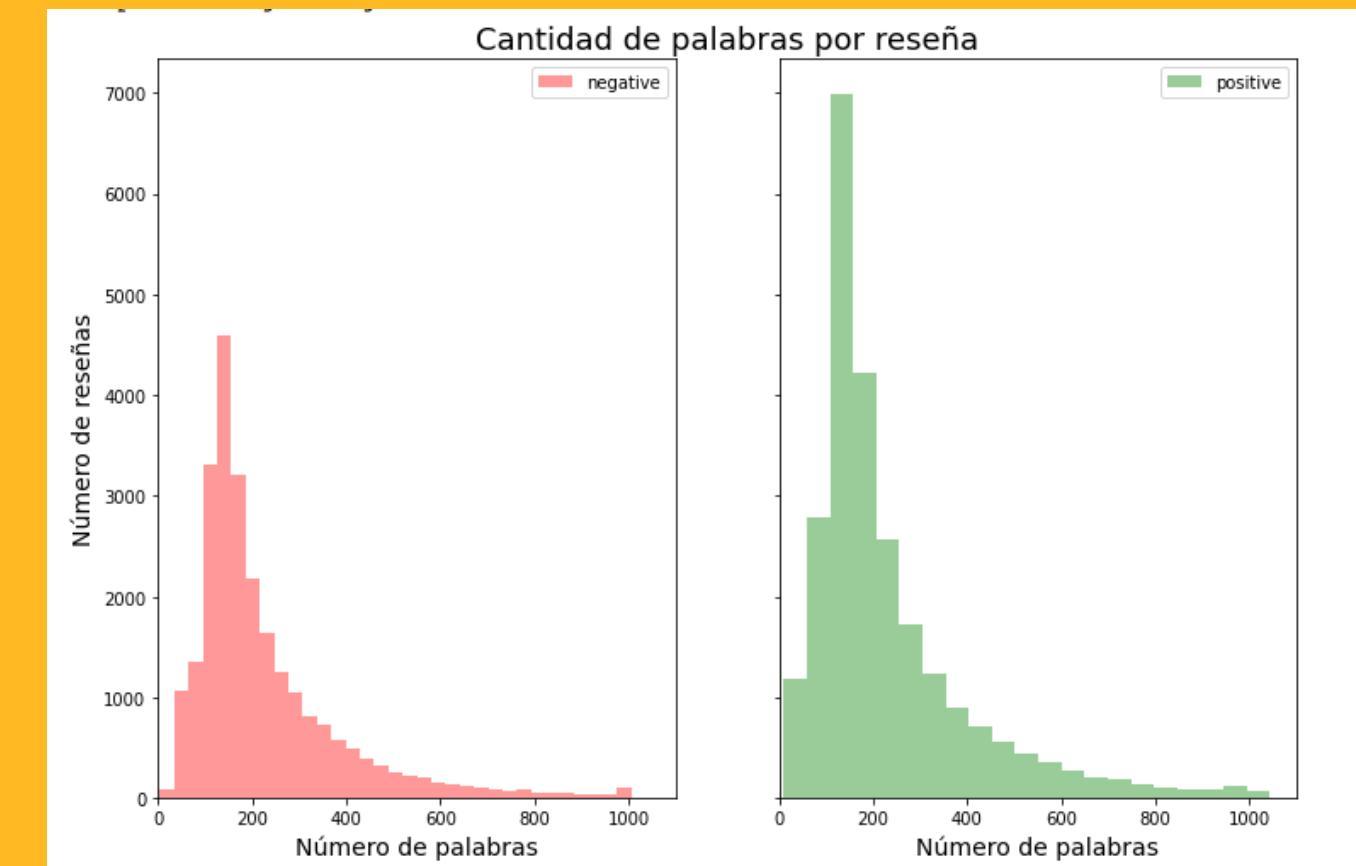
Análisis de datos y visualización



- Primero revisamos la distribución de cantidad de caracteres por reseña.



- Observamos la cantidad de palabras por reseña.



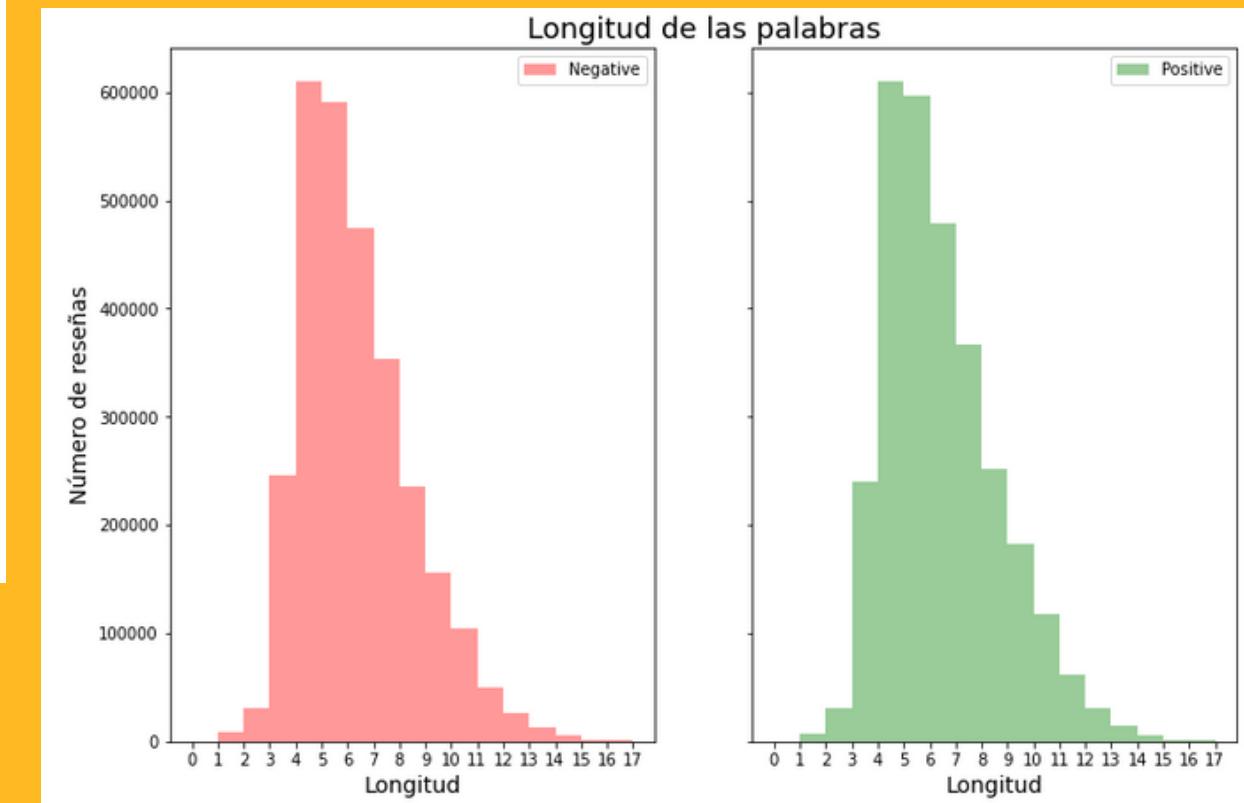
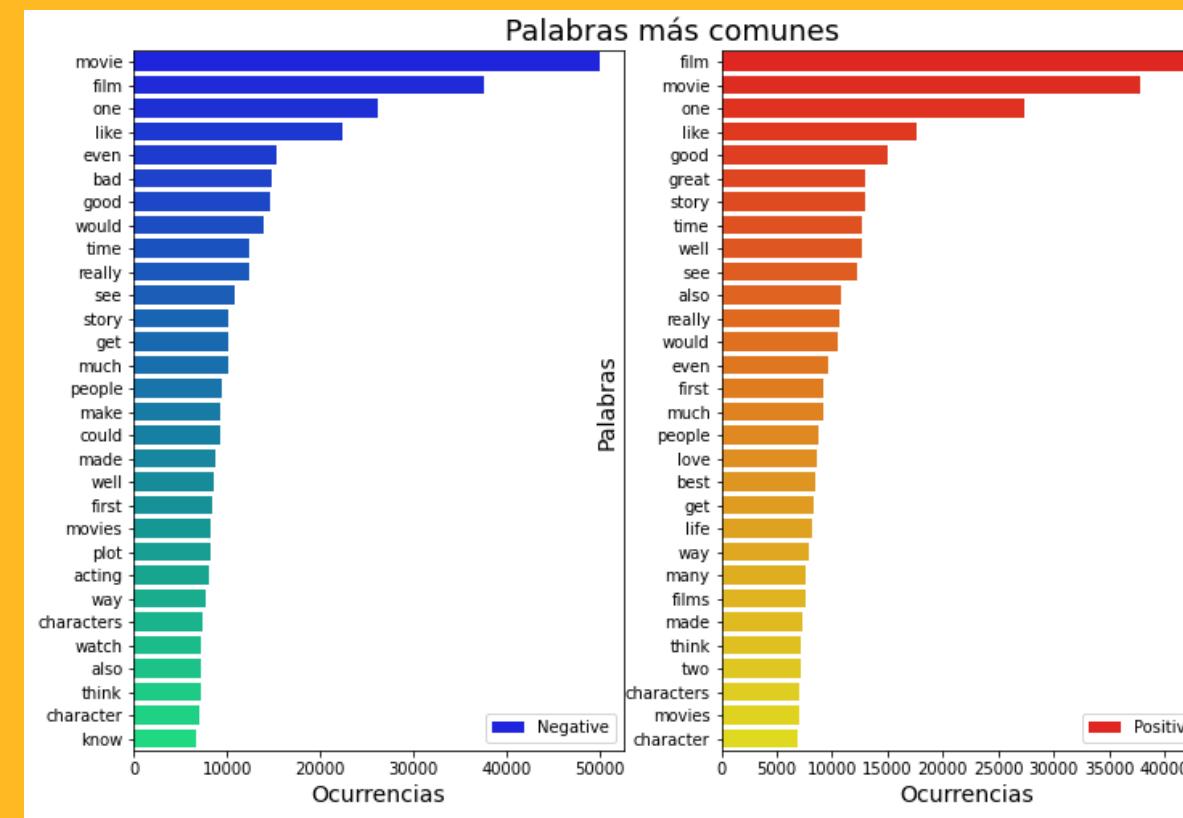


Limpieza de datos

- Se observa que hay palabras y caracteres que no aportan información por lo que se realiza limpieza de las reseñas para eliminarlos.
- Procedemos a "tokenizar" las reseñas.
- Creamos un "corpus" de palabras para cada sentimiento.

Limpieza de datos

- Con las frecuencias de cada palabra de los *data frames* creamos un diccionario para visualizar las más comunes por cada sentimiento.



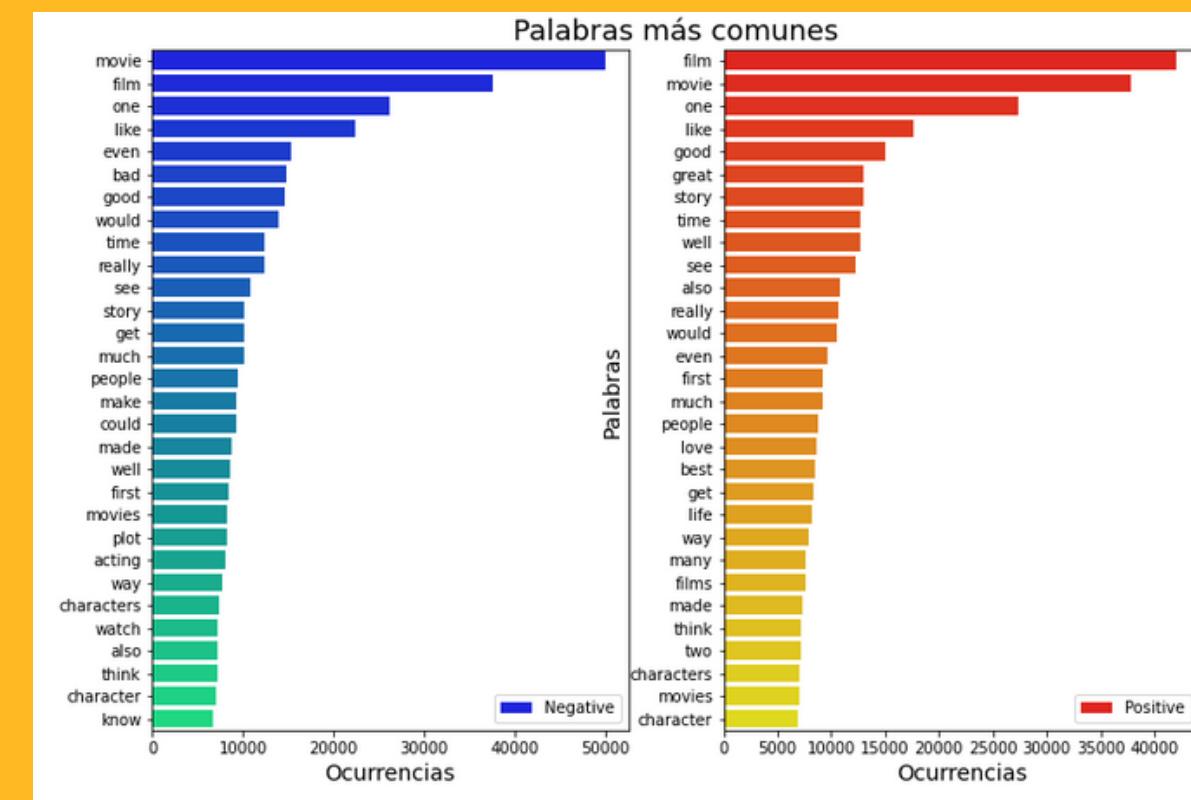
- Observamos que las palabras más repetidas tampoco aportan información relevante por lo que se eliminan.



Limpieza de datos



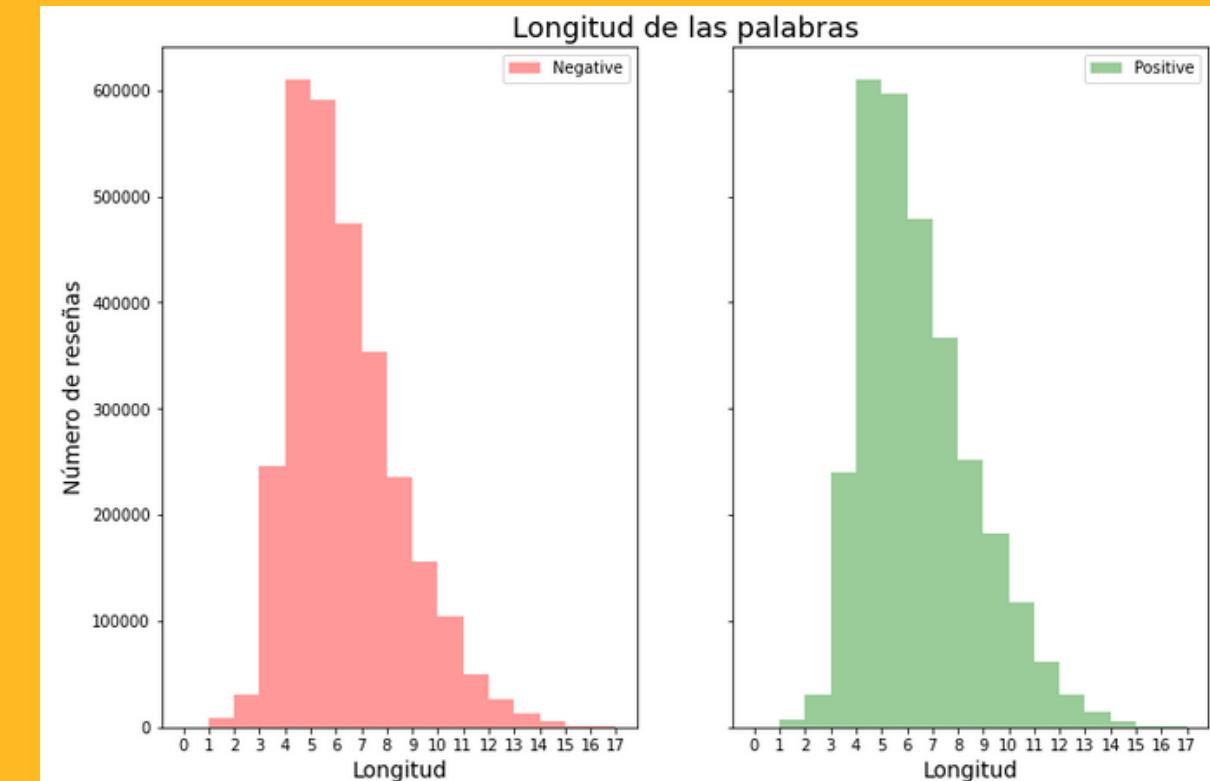
- Creamos de nuevo el diccionario de frecuencias para visualizar las palabras comunes y su longitud.



Palabras positivas más
frecuentes: wonderful y love.



Palabras negativas más
frecuentes: bad y family.





Preprocesamiento

- Primero, se creó una columna para almacenar los datos preparados. Aplicamos la función *clean_review* y "tokenizamos" los textos.
- "Estematzamos" las palabras para reducir su cantidad.
- Codificamos numéricamente los sentimientos. Negativos (0) y positivos (1).



Preprocesamiento

- Se divide el *data set* en conjuntos de prueba y entrenamiento. Se dividen por separado los *data frames* "pos_rev" y "neg_rev".
- Para la prueba, unimos los conjuntos negativos y positivos y mezclamos el orden.





Modelo 1: Enfoque frecuentista

Primero se realiza una aproximación frecuentista: los valores que se usarán para el algoritmos deben representar las veces que una palabra se encontró en ambos tipos de reseñas. Se crean "*corpus*" y diccionarios de frecuencias correspondientes.

Creación de los vectores

- Traducimos el texto a números para que el algoritmo pueda leerlo creando la función "*vectorize_review*" que convierte cada palabra en vector.
- Definimos como "target" al sentimiento codificado y como "features" a la reseña vectorizada. Esta última se separa por entradas en los conjuntos de prueba y entrenamiento. Validamos.



Entrenamiento del modelo

Después de realizar la preparación de datos, creamos el modelo.

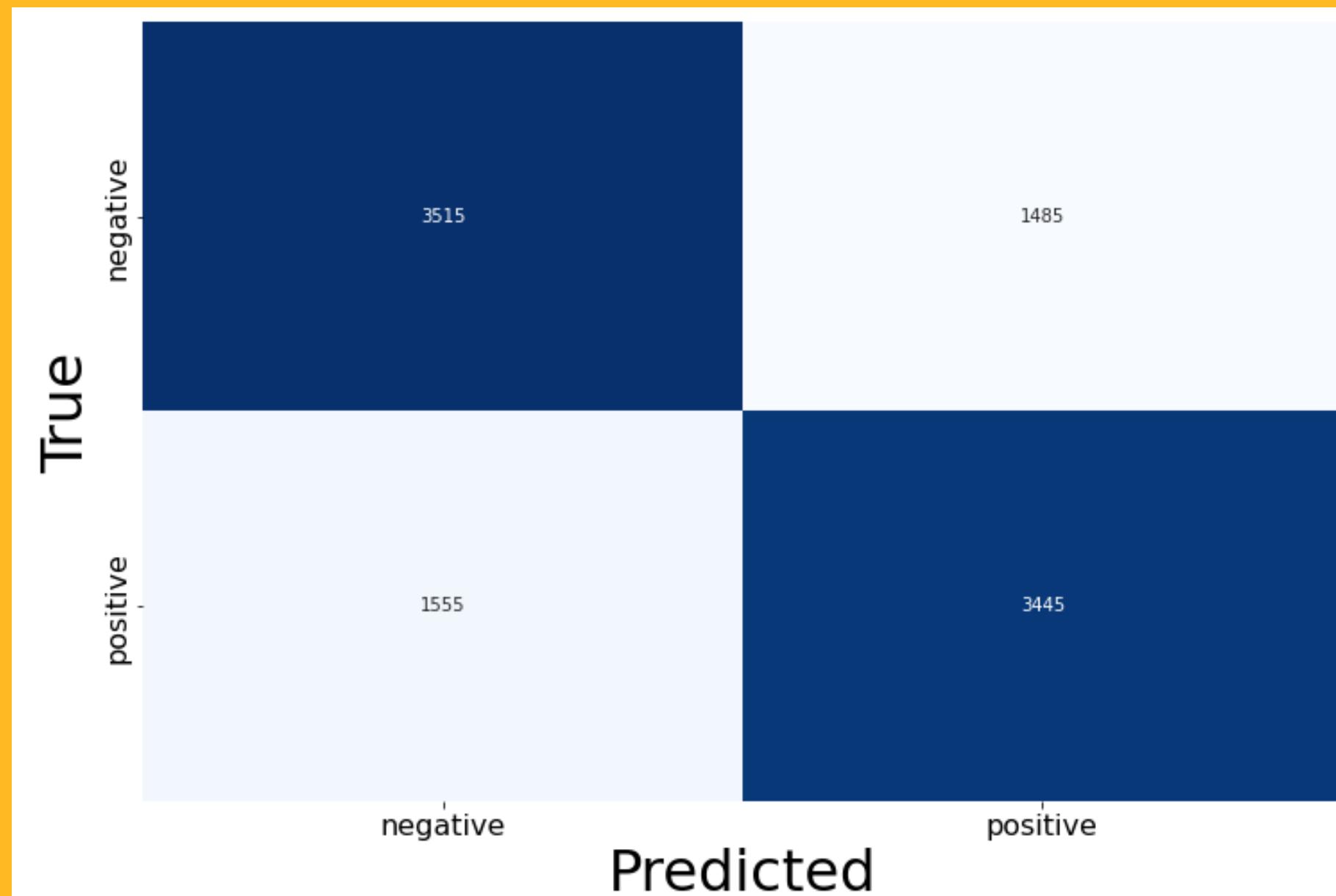
Para cumplir nuestro objetivo, ocupamos diversos algoritmos para comprobar su precisión y medir el desempeño general del enfoque frecuentista.

Evaluación del modelo

Realizamos las predicciones y evaluamos la eficacia con el modelo entrenado.

Primero medimos la precisión para darnos idea sobre su desempeño. En este caso, obtuvimos un desempeño regular (70% de las reseñas correctas).

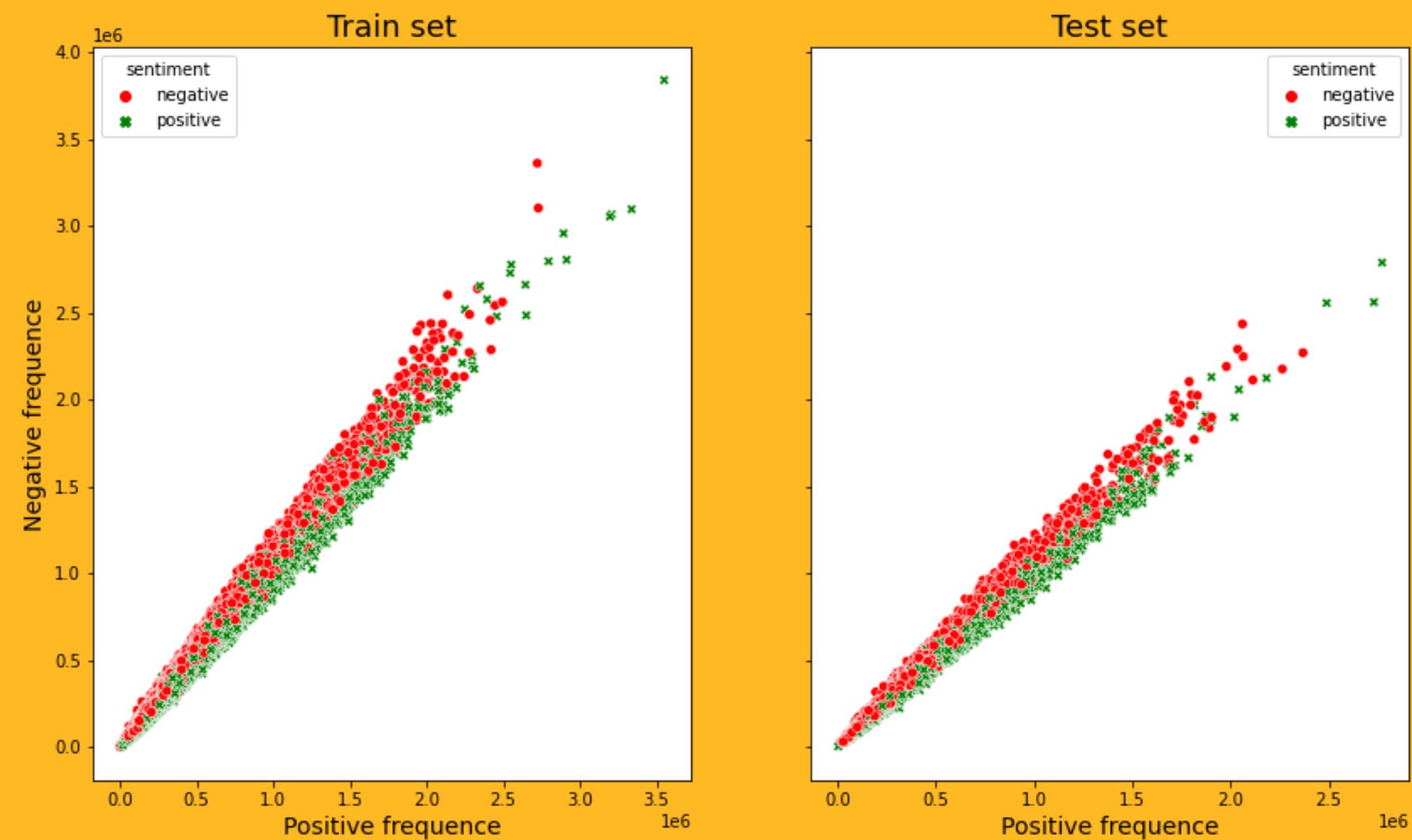
Vamos a visualizar las clasificaciones realizadas y el número de falsos positivos y falsos negativos por medio de una matriz de confusión.



Evaluación del modelo



Creamos el reporte con las diferentes métricas y graficamos los vectores oración.





Modelo 2: Word embeddings

Crearemos un vector n-dimensional por cada palabra, estos se realizan principalmente por medio de redes neuronales. Usaremos el método **CBOW** (Bag of words) para crear nuestros propios "Word embeddings".

Creación de los vectores- oración y entrena- miento



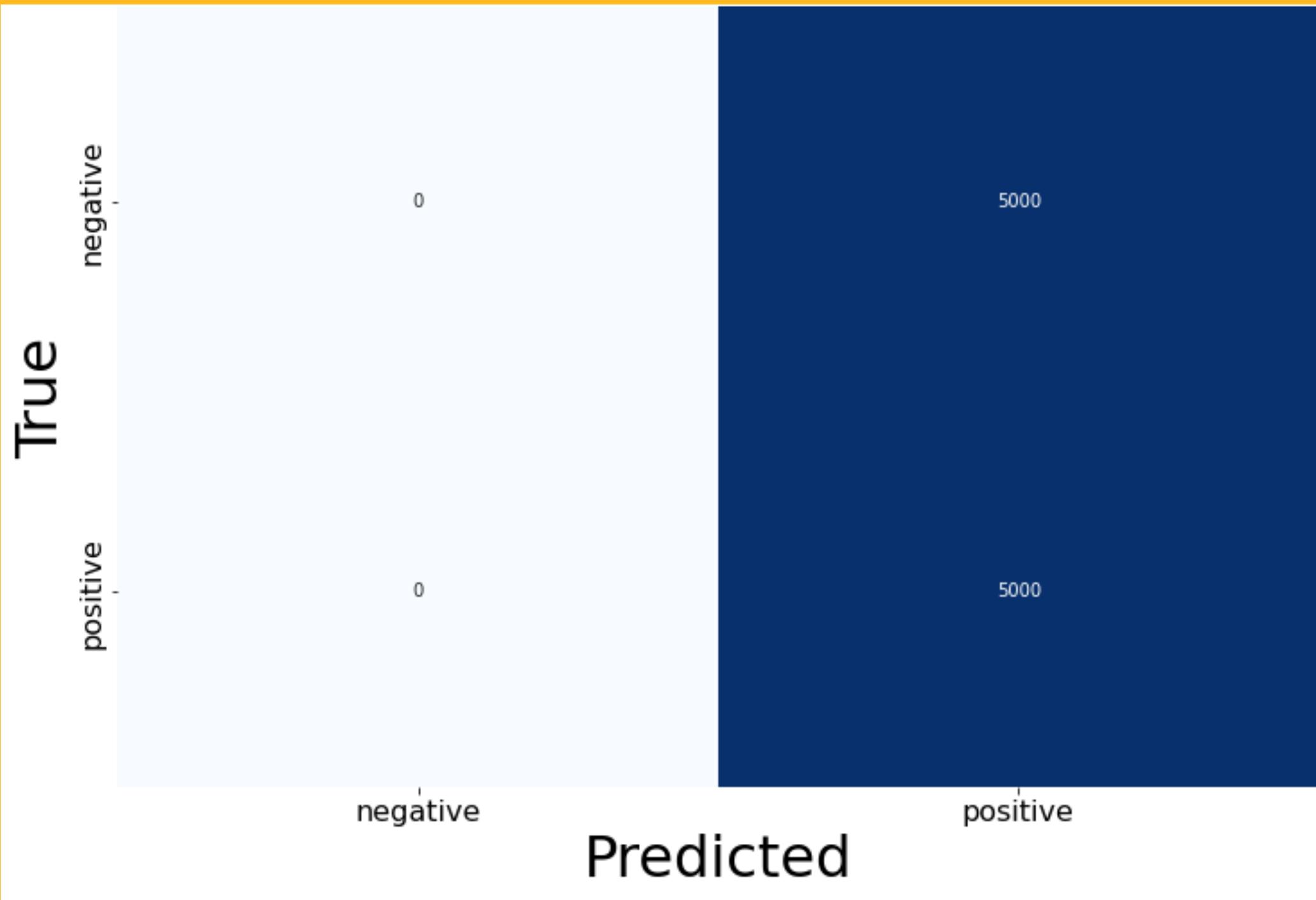
Después de crear lo "Word embeddings", vectorizamos toda la reseña.

- Filtramos las palabras que aparecen menos de 5 veces.
- Sumamos los *vectores-palabra* entrada por entrada.

Para el entrenamiento del modelo, utilizamos los mismos algoritmos con los mismos hiperparámetros del modelo anterior para que puedan ser comprables.

Evaluación del modelo

El desempeño promedio decayó al 50% por lo que se harán ajustes reduciendo su dimensionalidad



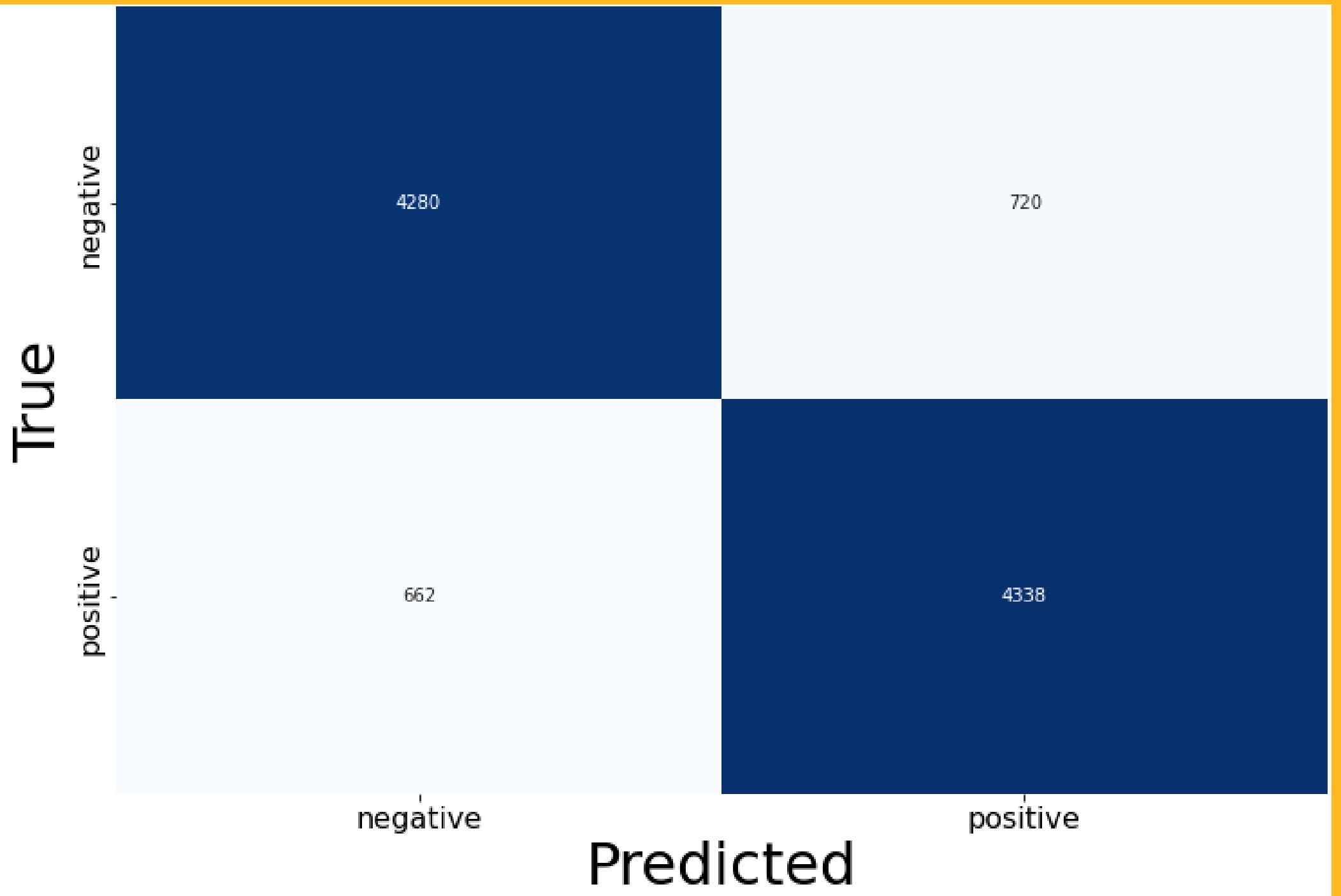
Reducción de dimensionalidad del modelo: PCA

Con esta herramienta reducimos vectores de alta dimensionalidad a espacios más pequeños conservando la mayor cantidad de información.

Se reasignan los conjuntos de entrenamiento y prueba a estos vectores.

Entrenamiento y evaluación del modelo

Con este ajuste la
precisión incrementó
un 35% respecto a los
vectores completos y
15% en comparación al
enfoque frecuentista.



> Conclusiones

- El método de 'word embeddings'-PCA provó ser más eficaz
- Es un método que combina efectividad con optimización de recursos
- El análisis de sentimiento es muy útil para obtener información de textos
- • Se puede ampliar su uso a otras categorías
-

