Universidad de los Andes

Inteligencia de Negocios

Proyecto

Etapa 1

Grupo 4

Juan Martin Santos

Ángela Vargas

Daniel Osorio

Análisis de sentimientos de películas

Tabla de contenidos

1. Entendimiento del negocio y enfoque analítico

¿Cuáles son los objetivos del negocio?	En este caso asumimos que nuestro cliente es una página de streaming que desea tomar las reviews de las películas de su plataforma para determinar si esta debe reemplazarse por una nueva película que pueda generar un mejor sentimiento a la audiencia o no. El sentimiento viene dado por la cantidad de buenas y malas reviews que tenga, entre mas reviews negativas tenga, peor es el sentimiento de la película.
¿Cuáles son los criterios de éxito?	Nuestro principal criterio de éxito es la precisión, ya que determinar erróneamente el sentimiento de una película puede afectar al negocio en la toma de decisiones sobre el estado de una película dentro de su plataforma.
Oportunidad/problema negocio	Al usar aprendizaje automático, el negocio se puede beneficiar al ser más eficiente en la forma en que determinar si una película debe quedarse o no, haciendo que el público en general esté más satisfecho con el contenido que consume aumentando la fidelidad y posibles ganancias.
Enfoque analítico (Descripción del requerimiento desde el punto de vista de aprendizaje automático)	Desde el punto de vista del aprendizaje automático, el problema se enfoca en algoritmos de clasificación en el análisis de textos. De esta forma, se está realizando el análisis de sentimientos de películas. Este proyecto tiene comentarios de películas en español, que deben ser clasificadas en las categorías de positivo, negativo. Esto por medio de técnicas para el procesamiento de lenguaje natural.
¿Quiénes están siendo beneficiados?	El principal beneficiado a parte del negocio son los clientes que van a poder determinar tener más decisión sobre el catálogo que ofrece la empresa de streaming.
Técnicas y algoritmos a utilizar	Haremos uso de Multinomial Naives Bayes (Daniel Osorio), Random forest (Juan Martín Santos) y regresión logística (Ángela Vargas).

2. Entendimiento y preparación de los datos

Entendimiento

Primero se cargaron los datos en un DataFrame, que es una estructura de datos que nos proporciona la librería Pandas. Luego, se contaron el número de registros que tenemos, que son 5000 filas y 3 columnas. La primera columna corresponde al identificador del comentario, la segunda columna contiene la reseña en español sobre la

película, y la tercera columna es el sentimiento asociado al comentario (positivo o negativo).

Luego, se revisaron los tipos de datos de las variables, los cuales fueron int64 para el identificador, y object (string) para las columnas del review y el sentimiento. Para la columna 'sentimiento' se contaron sus valores y se encontraron que existen 2500 comentarios de películas positivos, y 2500 comentarios de películas negativos. Se contaron los valores nulos para las filas y se encontró que existe una completitud de los datos del 100%. Además se contaron el número de filas duplicadas y se encontraron únicamente 2 filas.

La última exploración de los datos fue confirmar que los comentarios de las películas estuvieran en realidad en español, para lo cual se utilizó una librería llamada *langdetect*, con la cual se obtuvo que, en realidad, habían 196 comentario en inglés, 1 comentario en indonesio, y el resto sí efectivamente en español.

Preparación de los datos

En primera instancia se reemplazaron los valores de la variable de sentimiento, que son positivo y negativo, por 0 (negativo) y 1 (positivo). Luego se quitaron las filas duplicadas.
Posteriormente se hizo un recorrido por cada fila del conjunto de datos, y a cada fila se le detectó el lenguaje utilizando la librería *langdetect*. Todas las filas que su lenguaje fuese distinto al español fueron eliminadas. Al final quedaron XXXX filas luego de hacer esta limpieza de los datos.

Como preparación de los datos para los algoritmos que se van a implementar, se hizo además una limpieza de la columna 'review_es'. Más específicamente, se tomó cada comentario/review de la película y se estableció una lista de palabras vacías (stopwords en inglés). Estas palabras vacías son aquellas que no aportan significado alguno a una oración, ya sea porque son conectores, conjunciones o cualquier otro tipo de palabra más de la estructura del idioma que de la semántica.

Por otro lado, se utilizó el Snowball stemming algorithm para transformar las palabras a su forma raíz. En el notebook se hace un ejemplo con las palabras 'corriendo' y 'correr', que en realidad hacen referencia a la misma acción pero en diferente temporalidad. Entonces lo que hace esto es transformar ambas palabras a la palabra 'corr', que es la forma base de todas las combinaciones de este verbo. Esto se hace para mejorar la tokenización que es lo siguiente que se hace.

Finalmente, se utiliza la función *wordpunct_tokenize()* de la librería NLTK (Natural Language Tool-Kit) para tokenizar las distintas reviews de películas, esto lo que hace es darnos las palabras únicas de cada review, representadas como tokens.
Posteriormente, lo que se hace es aplicar la función *CountVectorizer()*, que nos da la

cuenta de cada token (cada palabra) para cada review, que es lo que finalmente utilizan los algoritmos para crear los modelos que se presentan a continuación.

3. Modelado y evaluación

Se utilizaron los siguientes algoritmos para crear los distintos modelos: Multinomial Naive Bayes, Random Forest y Regresión logística. A continuación se explicará brevemente en qué consiste cada uno y sus respectivas evaluaciones de desempeño.

Multinomial Naive Bayes

El algoritmo de Naive Bayes Multinomial es un método de aprendizaje supervisado que se utiliza comúnmente en clasificación de texto. Primero, se realiza una transformación de los comentarios de texto en vectores de características (como se describió en la preparación de los datos). Para esto, se utilizó el *CountVectorizer* para contar la frecuencia de cada palabra en los comentarios. Luego, se aplicó el algoritmo de Naive Bayes Multinomial para entrenar el modelo. El modelo utiliza la frecuencia de cada palabra en los comentarios para predecir la probabilidad de que un comentario sea positivo o negativo. El algoritmo de Naive Bayes asume que las palabras en el comentario son independientes entre sí y utiliza la probabilidad de cada palabra en el comentario para predecir la probabilidad de que el comentario sea positivo o negativo.
Finalmente, se evaluó el rendimiento del modelo utilizando métricas como precisión, recall y F1-score. Estas métricas nos permiten medir cuán bien el modelo está clasificando los comentarios en positivo o negativo. Nuestro modelo obtuvo las siguientes métricas:

- Exactitud sobre test: 82%
- Recall: 80.1%
- Precisión: 82.9%
- Puntuación F1: 81.5%

A continuación se muestra la matriz de confusión sobre los datos de test.

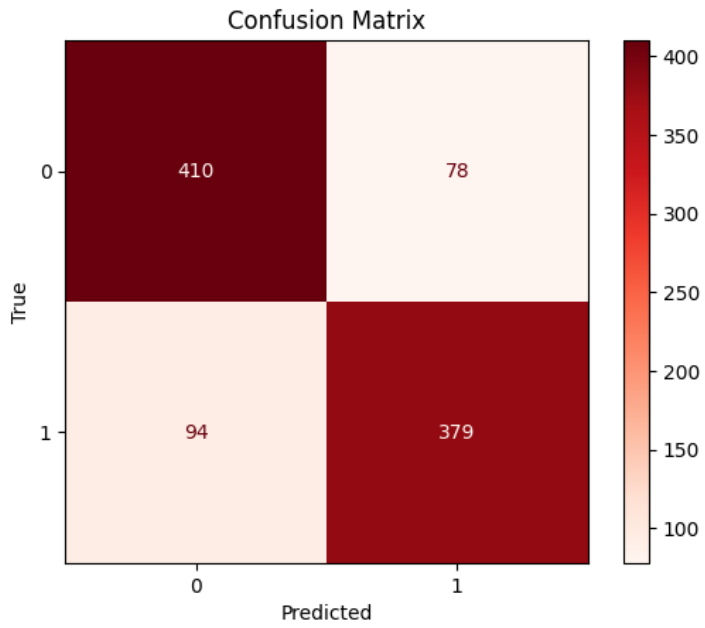


Figura 1. Matriz de confusión del algoritmo Naive Bayes Multinomial.

Como se puede observar, el modelo da buenos resultados, sin embargo no es perfecto puesto que tuvo 94 falsos negativos y 78 falsos positivos.

Random Forest

Escogimos este algoritmo porque este es capaz de manejar características de alta dimensión lo que favorece en el análisis de texto por la extensión de los mismos. Además, maneja muy bien el overfitting. El algoritmo se basa en árboles de decisión y funciona de la siguiente forma: Selecciona aleatoriamente un subconjunto de muestras; Selecciona aleatoriamente un subconjunto de características; con las muestras y características seleccionadas se construyen varios árboles de decisión; por último se selecciona el árbol de decisión que obtiene mayor votación dados sus resultados. Se obtuvieron los siguientes resultados:

- Exactitud sobre test: 84%
- Recall: 85.2%
- Precisión: 83.4%
- Puntuación F1: 84.3%

A continuación se muestra la matriz de confusión sobre los datos de test.

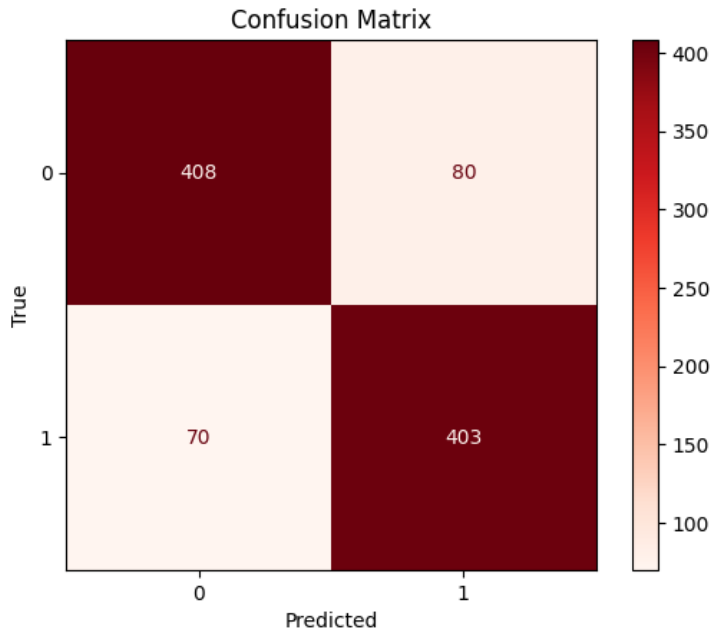


Figura 2. Matriz de confusión del algoritmo Random Forest.

Como se puede observar, el modelo da buenos resultados, incluso mejores resultados que con el Naive Bayes Multinomial. Este tuvo 14 falsos negativos menos que el algoritmo anterior, sin embargo tuvo 2 falsos positivos de más comparado con Naive Bayes Multinomial.

Regresión Logística

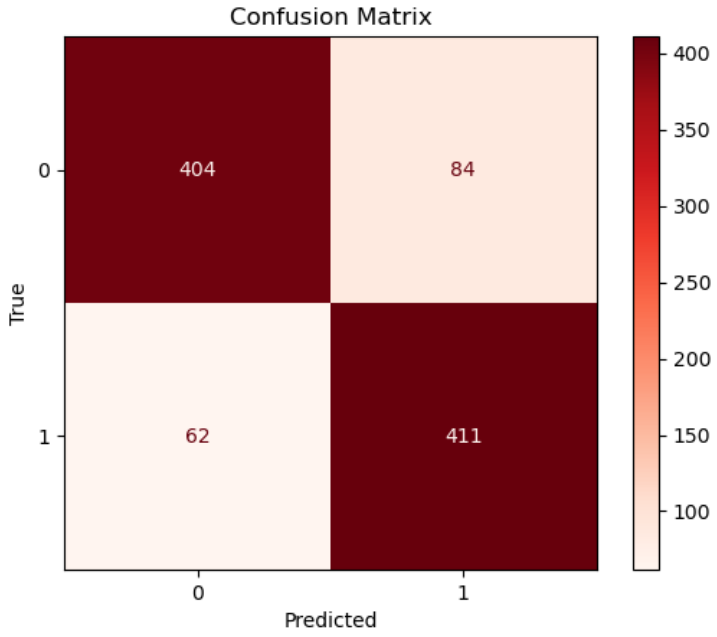
Escogimos este algoritmo porque la regresión logística es particularmente adecuada para problemas de clasificación binaria, es decir, cuando solo hay dos categorías posibles (en este caso, positivo o negativo). En estos casos, la regresión logística puede ser más eficiente y precisa que otros algoritmos. Los coeficientes del modelo pueden utilizarse para determinar qué palabras o características tienen más influencia en la clasificación de los comentarios como positivos o negativos. Además puede escalar bien con grandes conjuntos de datos y se puede mejorar mediante técnicas avanzadas de optimización.

Se implementaron 3 versiones de este algoritmo, al correr la primera versión con todas los parámetros por defecto se encontró una sugerencia de aumentar el número de iteraciones. Para la segunda versión se cambió este del predeterminado que es 100 a 1000 pero no se observó ninguna mejora considerable por lo cual se consideró en cambiar el algoritmo de optimización, el usado por defecto es lbfgs que utiliza el método de optimización de cuasi-Newton Broyden-Fletcher-Goldfarb-Shanno (BFGS), se probó con saga el cual utiliza una versión mejorada del método de promedio de gradientes estocásticos que admite regularización elástica y regularización L1, no obstante los resultados obtenidos fueron muy similares a los de la primera versión de regresión

logística utilizados. Finalmente se probó con el algoritmo de optimización newton-cg el cual utiliza el método de Newton conjugado para optimizar los parámetros del modelo. La mejor versión fue la de regresión lineal con newton-cg.
Los resultados obtenidos se muestran a continuación

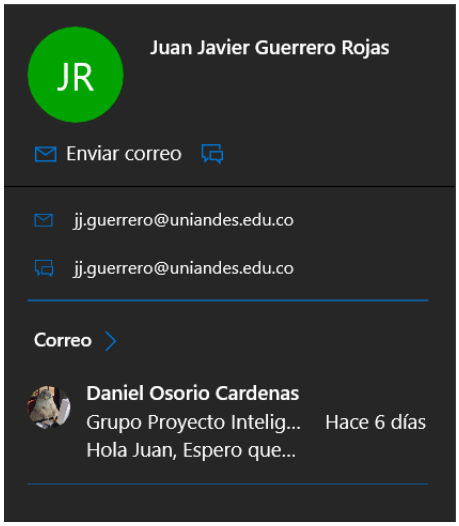
Metrica	**RL Pred**	**RL con 1000 iter**	**RL con saga**	**RL con newton-cg**
Exactitud sobre el test	85%	85%	85%	85%
Recall	86.89%	86.89%	87.1%	86.89%
Precisión	83.03%	83.03%	82.9%	83.03%
Puntuación	84.92%	84.92%	84.95%	84.92%

Se muestra la matriz de confusión sobre los datos del set para la última versión del algoritmo de regresión logística



Validación con el experto en temas de estadística

Lastimosamente, no pudimos contactar a Juan Guerrero. Se le envió un correo desde el martes de la semana pasada, pero no respondió. Preguntamos a otros grupos que tenían asignado a la misma persona y nos dijeron que tampoco habían recibido respuesta. Adjuntamos evidencia del intento de contactarlo.



4. Resultados

La siguiente tabla resume los resultados de los algoritmos probados (del algoritmo de regresión lineal solo se muestra el mejor)

Metrica	Multinomial NB	Random Forest	RL con newton-cg
Exactitud sobre el test	82%	84%	85%
Recall	80.13%	85.2%	86.89%
Precisión	82.93%	83.44%	83.03%
Puntuación	81.5%	84.31%	84.92%

Inicialmente se había estipulado que el criterio de éxito principal es la precisión, teniendo esto en cuenta el modelo que se sugiere utilizar es el de random forest. No obstante, se sugiere comparar el resultado con el de regresión logística,puesto que presenta una precisión similar y en general las demás métricas son mejores..

Adicionalmente, dado que se quiere utilizar la información para determinar qué películas deben seguir en la plataforma se define qué se debe tener especial cuidado con eliminar peliculas que tengan una buena acogida del público, por lo tanto se prefieren los falsos positivos a los falsos negativos: si una película que en realidad es positiva para los usuarios es clasificada como negativa puede llevar a su eliminación del catálogo perdiendo así la confianza de la audiencia, mantener algunas películas que no son positivas para la audiencia no generará este impacto negativo en los usuarios. Teniendo esto en cuenta y revisando la matriz de confusión de los diferentes algoritmos aplicados

el algoritmo que mejor funciona en este caso sería el de la regresión logística ya que es el que menos falsos negativos presenta. Se debe validar con el cliente si esta condición de preferencias de falsos positivos es correcta para el negocio pues de no ser así indudablemente el mejor método pasaría a ser el de random forest.

5. Referencias

- *Introducción al procesamiento del lenguaje natural-1 - Mi Blog*. (2017, 2 enero).https://josearcosaneas.github.io/python/r/procesamiento/lenguaje/2017/01/02/procesamiento-lenguaje-natural-0.html
- Saranya, G., & Geetha, T. V. (2020). A review on machine learning algorithms for sentiment analysis. International Journal of Advanced Science and Technology, 29(4), 732-738.
- Wang, S., & Li, X. (2019). A comparative study of machine learning algorithms for sentiment analysis of short texts. Journal of Information Science, 45(6), 801-818.