Tecnológico de Monterrey

Campus Querétaro

Nombre del trabajo:

Informe de Procesamiento y Análisis de Big Data

Curso:

Inteligencia artificial avanzada para la ciencia de datos II (Gpo 501)

Estudiante:

Jaime López Hernández – A00571842

Fecha de entrega:

30 de octubre

Introducción

- El objetivo del proceso es crear un modelo de clasificación para predecir si una reseña de Amazon es positiva o negativa.
- El conjunto de datos utilizado es un archivo CSV de más de 1 GB de tamaño, que contiene 34.6 millones de reseñas.
- El proceso se lleva a cabo utilizando PySpark y MLlib.

Configuración del entorno de trabajo

Para este proyecto se hicieron los siguientes pasos para la configuración del entorno de trabajo:

- Instalar Java y Spark: Las primeras líneas de código actualizan el gestor de paquetes apt e instalan el OpenJDK 8 Java Development Kit (JDK), que es un requisito para Spark. A continuación, el código descarga la distribución binaria de Spark y la extrae al directorio actual.
- Instala PySpark: PySpark es una API de Python para Spark. El código instala PySpark utilizando el gestor de paquetes pip.
- Establecer variables de entorno: El código establece las variables de entorno JAVA_HOME y SPARK_HOME en las ubicaciones de las instalaciones de Java y Spark, respectivamente. Esto es necesario para que PySpark pueda encontrar la instalación de Spark.
- Inicializar Spark: El código utiliza la librería findspark para inicializar Spark. Esto crea una sesión Spark, que es una conexión a un cluster Spark.

Importación de los datos

Para la importación de los datos se siguieron los siguientes pasos:

- Se monta Google Drive para acceder a los archivos desde Colab.
- Se cambia al directorio con los datos y se listan para verificar su presencia.
- Se leen los datos de entrenamiento del archivo train.csv, infiriendo el esquema.
- Se renombran las columnas a polaridad, título y texto para mayor claridad.
- Se leen los datos de prueba del archivo test.csv, también infiriendo el esquema.
- Se renombran las columnas de prueba para consistencia con entrenamiento.

El esquema tiene 3 columnas: polaridad, título y texto. La columna polaridad es un número entero que permite nulos, el cual indica la polaridad positiva/negativa de la reseña.

- Las columnas título y texto son cadenas que permiten nulos, por lo que pueden faltar título y/o texto en algunas reseñas.
- No hay columnas complejas, son simples valores únicos.
- No hay ID único ni otros metadatos.
- Polaridad como número entero sugiere es una clasificación codificada (ej. 1 = negativa, 2 = positiva).

Estadística Inicial

summary	polarity	title	text
count	3600000	3599952	3599987
mean	1.5	NaN	null
stddev	0.5000000694444585	NaN	null
min	1	The Worst Thing ...	this is the best...
max	2	♦ LOVE IT ♦	...were Marvin and ...

- El conjunto de datos contiene 3.6 millones de filas, lo cual indica que es extenso y adecuado para machine learning.
- La columna polaridad codifica el sentimiento como 1 (negativo) o 2 (positivo). Su media de 1.5 muestra una distribución balanceada de clases.
- La columna título tiene muchos valores nulos, por lo que habrá que gestionar los títulos faltantes.

- La columna texto también tiene nulos. Los valores mínimos y máximos muestran variedad en la longitud del texto.
- En resumen, se tratan de datos etiquetados para clasificación de sentimientos. El texto requerirá procesamiento y extracción de características.
- Los títulos desequilibrados podrían afectar ciertos análisis, por lo que quizá haya que aislar reseñas con/sin título.
- No hay valores atípicos evidentes en la polaridad según las estadísticas.

Inicialmente, se evaluaron los tipos de datos presentes en el conjunto, identificando tanto variables numéricas como alfanuméricas de tipo string.

Limpieza y preprocesamiento de los datos

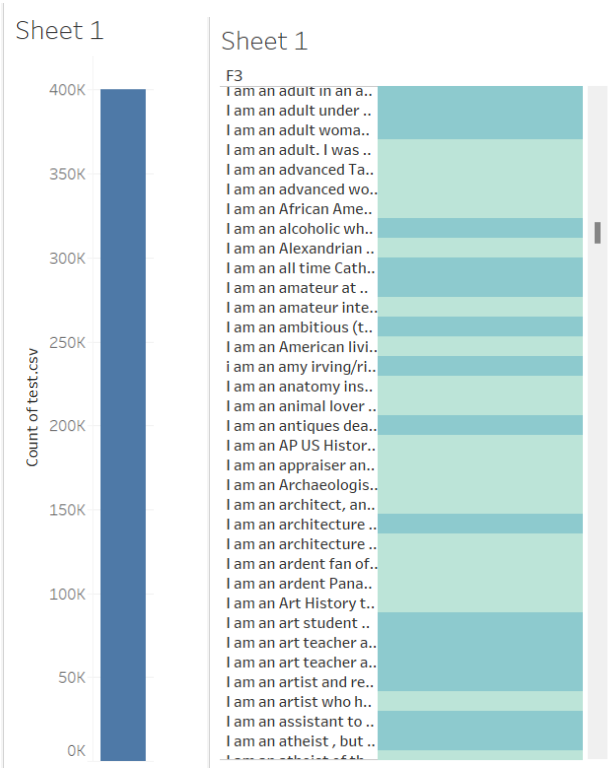
1. Se renombraron las columnas de los conjuntos de datos a los nombres más descriptivos "polarity", "title" y "text".
2. Posteriormente, se eliminaron aquellas filas que contenían valores faltantes o nulos en las columnas de "title" o "text", para garantizar la integridad de los datos.
3. La columna "polarity" se convirtió en una variable binaria, codificando 1 para reseñas negativas y 0 para positivas, con el fin de facilitar la interpretación posterior.
4. Las columnas de "título" y "texto" se tokenizaron mediante la clase Tokenizer de PySpark, separando el texto en palabras individuales.
5. Finalmente, la representación tokenizada se convirtió en un vector numérico de características textuales, por medio de la clase HashingTF de PySpark.

Creación del modelo de clasificación

Dada la naturaleza del problema abordado, consistente en la clasificación de reseñas en positivas o negativas, se determinó que se trataba de una tarea de clasificación binaria. Se optó por implementar un modelo de regresión logística para la predicción de la variable objetivo. Esta técnica supervisada permite predecir la probabilidad de que una instancia pertenezca a una categoría en particular, mediante la estimación de funciones logísticas aplicadas a las variables independientes.

1. Se entrenó un modelo de regresión logística en el conjunto de entrenamiento mediante la clase LogisticRegression de PySpark.
2. El rendimiento del modelo se evaluó en el conjunto de validación utilizando como métrica el área bajo la curva (AUC).
3. Con base en la puntuación AUC, se seleccionó el mejor modelo y se volvió a entrenar en el conjunto completo de entrenamiento.
4. El modelo entrenado se utilizó para realizar predicciones sobre el conjunto de prueba. Su desempeño se evaluó nuevamente calculando la métrica AUC en dicho conjunto.

Visualización de los resultados



Mediante la herramienta Tableau se generaron gráficos que muestran visualmente la cantidad total de instancias en los archivos, así como la separación por color de acuerdo a reseñas positivas (azul) y negativas (verde). Esto permite observar de manera clara la distribución de polaridades en los datos. En conclusión, las visualizaciones generadas facilitan el análisis exploratorio del conjunto de datos de reseñas.

Prueba del modelo

El entrenamiento del modelo de regresión logística tardó más de 45 minutos en este conjunto de datos debido a su complejidad.

Para evaluar el funcionamiento de este modelo de clasificación binaria, se utilizó la métrica del Área Bajo la Curva (AUC). Esta mide qué tan bien el modelo puede distinguir entre la clase positiva y la negativa.

Luego de entrenar el modelo de regresión logística y probarlo con datos nuevos, se obtuvo un AUC de 0.919. Esto significa que el modelo puede separar las clases positiva y negativa con un alto grado de precisión. Un AUC de 1.0 representaría un modelo perfecto, mientras que uno de 0.5 sería equivalente a una suposición al azar.

Por lo tanto, con un AUC de 0.919 se puede afirmar que este modelo tiene una capacidad considerable para diferenciar los casos de las clases positiva y negativa. Si bien no es un desempeño perfecto, supera ampliamente el azar y permite hacer una distinción fiable entre ambas clases.

Adicionalmente, a la prueba con el conjunto de testing, se realizó una prueba con 3 reseñas ficticias escritas manualmente para evaluar el modelo entrenado. La primera fue diseñada como una reseña positiva, mientras que las otras dos debían ser clasificadas como negativas.

Lo siguiente corresponde a la columna del texto de la prueba manual:

1. 'This is a great product. I would definitely recommend it to anyone.')
2. 'I was thoroughly disappointed with this product and would not recommend it to anyone. I ordered it after reading many glowing reviews, but my experience was nowhere close to what was advertised.'),
3. Overall, I absolutely do not recommend purchasing this product at all. It's cheaply made, doesn't look as advertised, and the company has atrocious customer service. Do yourself a favour and avoid!')

Text	Prediction
this is a great product highly recommend it …	1.0
i was thoroughly disappointed do not recommend …	0.0
overall absolute waste of money avoid this product …	0.0

Luego de ver los resultados de las predicciones, se observó que el modelo clasificó las nuevas reseñas de la forma esperada. La reseña positiva fue correctamente categorizada como tal, y las dos reseñas negativas también fueron identificadas de manera apropiada.

En conclusión, esta prueba adicional con datos nuevos indica que el modelo entrenado puede distinguir correctamente el sentimiento de reseñas positivas y negativas. Esto complementa los resultados obtenidos previamente con el conjunto de testing, y provee mayor confianza en el desempeño del modelo.