Clasificación de reseñas de Amazon utilizando NLP y Random Forest

Classification of Amazon reviews using NLP and Random Forest

Marcos Ruvalcaba García[1], Misael López Ramírez[2], Eduardo Cabal Yépez[3], Rafel Guzmán Cabrera[4]
[1 2 3 4]Universidad de Guanajuato
m.ruvalcabagarcia@ugto.mx[1], lopez.misael@ugto.mx[2], educabal@ugto.mx[3], guzmanc@ugto.mx[4]

Resumen

En este trabajo se realizó el análisis y clasificación de reseñas de productos de Amazon contenidas en un conjunto de datos. Primeramente, fue necesario llevar a cabo un preprocesamiento a dichas opiniones, con la finalidad de aplicarles una depuración previo a la clasificación. Las variables que se tomaron en cuenta para la depuración fueron: eliminación de palabras de parada o "stop words", lematización, eliminación de palabras de poca frecuencia y ganancia de información. La clasificación de las reseñas se efectuó mediante el método de aprendizaje Random Forest, para esto se utilizó la herramienta de software Weka. Se eligió esta herramienta debido a que es ampliamente utilizada para minería de datos, aprendizaje automático y diversas tareas relacionadas con el análisis de datos. Los resultados obtenidos en la clasificación fueron muy alentadores sobrepasando el 80% de precisión para el método de aprendizaje seleccionado.

Palabras clave: Análisis de sentimientos, Inteligencia artificial, Aprendizaje automático, Weka.

Introducción

El problema de la detección de opiniones en textos no estructurados consiste en detectar las opiniones de las personas o usuarios en textos que no siguen una estructura como tal o un formato ya establecido (Chauhan et al., 2020). Esto se puede observar claramente de las opiniones que se dan en redes sociales como Facebook, Instagram, Twitter, etc. Otro claro ejemplo se puede observar en las reseñas que dan las personas al comprar un producto en tiendas electrónicas, mostrando su satisfacción o inconformidad con el producto o artículo que han adquirido.

El desafío que conlleva la detección de opiniones se debe que es complicado realizar dicha tarea puesto que, la mayoría de las veces, los usuarios expresan sus opiniones dejándose llevar por la subjetividad (Nandal et al., 2020). Esto se debe a que el criterio de cada persona puede variar de manera significativa, otorgando opiniones más directas y explicitas o, de forma contraria, dando opiniones que suelen caer en la ambigüedad. Por tal motivo, es importante aplicar métodos que ayuden a la detección de este tipo situaciones, con el objetivo de lograr una mayor comprensión de las opiniones de los usuarios.

Existen diferentes técnicas que pueden ayudar a dar solución a este problema, como los son el procesamiento del lenguaje natural y el aprendizaje automático. En estas técnicas se incluyen la tokenización de las palabras, la detección de emociones, la clasificación de textos, entre otras. De igual manera es posible aplicar enfoques en modelos de aprendizaje profundo, como lo son las redes neuronales recurrentes, esto con la finalidad de extraer la información contextual y obtener una mejora en la precisión de la identificación de este tipo de textos. Tal es el caso en (Alroobaea, 2022), donde los autores presentaron un método utilizando redes neuronales recurrentes para predecir el sentimiento de reseñas de usuarios de Amazon. En este estudio los autores compararon su modelo propuesto con otros modelos de redes neuronales, obteniendo excelentes resultados.

Un enfoque diferente se aplicó en (Daniel & Meena, 2021), en donde se presentó un marco híbrido que combina enfoques de aprendizaje automático y basados en léxico para un análisis preciso de sentimientos en reseñas de productos de Amazon. En este trabajo los autores integraron una reducción de características basada en el algoritmo de enjambre tunicados para mejorar la escalabilidad y el rendimiento general. Por otro lado, analizando de igual forma reseñas de Amazon, pero enfocado a teléfonos móviles, en (Sivakumar & Reddy, 2021) los autores presentaron un sistema inteligente para el análisis de sentimientos utilizando redes neuronales LSTM y lógica difusa. El sistema fue capaz de clasificar las oraciones de las reseñas de los consumidores en diferentes aspectos con alta precisión.

Como se vio anteriormente, existen muchas técnicas y métodos que se han aplicado para el análisis de sentimientos, obteniendo resultados prometedores en cada uno de ellos. Ahora bien, para este trabajo se propone realizar la misma tarea pero aplicando un enfoque diferente. Se plantea analizar reseñas de Amazon contenidas en un corpus, buscando realizar variaciones en el preprocesamiento con el fin de obtener la mejor precisión posible. Dentro de estas variaciones se tomarán en cuenta, eliminar palabras de poca frecuencia, lematización, eliminación de palabras de parada y ganancia de información. Por último, se llevará a cabo la clasificación en el software Weka, utilizando Random Forest como método de aprendizaje.

Fundamento Teórico

Software Weka

Según (Witten et al., 2016) el banco de trabajo de Weka es un conjunto de algoritmos de aprendizaje automático y herramientas para el procesamiento de datos. Este software está diseñado para poder realizar múltiples pruebas con los diferentes métodos existentes en varios conjuntos de datos. Proporciona un gran soporte para el proceso de extracción de datos, incluyendo la preparación de los datos que se toman como entrada, la evaluación estadística de cada uno de los modelos de aprendizaje disponibles y la visualización de los datos de entrada, son olvidar, el resultado que cada método de aprendizaje implementado.

Además de la increíble variedad de algoritmos de aprendizaje, este también incluye una amplia gama de herramientas de procesamiento que se pueden aplicar a los datos. Es posible acceder a este conjunto de instrumentos diverso y completo a través de una interfaz graficas para que los usuarios logren comparar los diferentes métodos disponibles e identificar aquellos que podrían ser los que les otorguen mejores resultados en el problema en que se estén enfocando.

Archivo ARFF

De acuerdo con (Dhiman, 2020) ARFF (Attribute-Relation File Format) es el formato de archivo predeterminado para Weka, aunque de igual manera acepta otro tipo de archivos, como los archivos separados por comas, formato C4.5, entre otros. El archivo ARFF es un archivo de texto ASCII; el cual brinda una lista de instancias que comparten un conjunto de atributos.

Este tipo de archivos está compuesto por dos secciones distintas. La primera sección pertenece al encabezado, en esta se encuentran los detalles de la relación, los atributos y sus tipos. La segunda sección pertenece a la declaración de los datos, la cual contiene los datos más relevantes.

Cada una de las instancias es representada en una solo línea, con retornos que indican el final de la instancia. Los valores de cada atributo para cada una de las instancias están delimitados por comas. Estos deben aparecer en el orden en que fueron declarados previamente, en la sección del encabezado.

En la Figura 1 se muestra un ejemplo de la estructura de un archivo ARFF.

```
@RELATION TextClassification

@ATTRIBUTE book NUMERIC
@ATTRIBUTE like NUMERIC
@ATTRIBUTE good NUMERIC
@ATTRIBUTE read NUMERIC
@ATTRIBUTE movie NUMERIC
@ATTRIBUTE buy NUMERIC
@ATTRIBUTE time NUMERIC
@ATTRIBUTE great NUMERIC
@ATTRIBUTE use NUMERIC
@ATTRIBUTE love NUMERIC
@ATTRIBUTE think NUMERIC
@ATTRIBUTE work NUMERIC
@ATTRIBUTE find NUMERIC
@ATTRIBUTE story NUMERIC
@ATTRIBUTE look NUMERIC
@ATTRIBUTE year NUMERIC
```

Figura 1. Ejemplo de la sección del encabezado de un archivo ARFF.
Fuente: Elaboración propia.

Random Forest

De acuerdo con (Yohei Mishina, 2015) Random Forest es un algoritmo de clasificación multiclase que introduce aleatoriedad a través del uso de bagging y selección de características, y puede ser fácilmente paralelizado. Sin embargo, para lograr buenos resultados, se necesita una gran cantidad de memoria para su construcción.

Por otro lado, Random Forest es un método que reduce el consumo de memoria utilizado por los árboles de decisión. Esto se logra al utilizar la misma función de decisión para los nodos de ramificación en el mismo nivel. No obstante, simplemente reducir el número de nodos conlleva una disminución en el rendimiento de la clasificación (Yohei Mishina, 2015).

La idea principal detrás de Random Forest es la combinación de múltiples árboles de decisión entrenados de forma independiente utilizando diferentes subconjuntos de datos y características.

Metodología Propuesta

El corpus que se utilizó para el desarrollo de este trabajo fue el corpus “Amazon Reviews for Sentiment Analysis”. Este conjunto datos está compuesto por un total de 4 millones de reseñas de Amazon de diferentes productos, se construyó tomando en cuenta el puntaje asignado a cada reseña, 1 y 2 como negativo y, 4 y 5 como positivo. Las muestras con puntuación 3 se toman como neutras por lo tanto se descartan ya que no son criticas para su estudio.

El conjunto de datos está compuesto por dos clases, la clase 1 que corresponde a las reseñas negativas y la clase 2 que corresponde a las reseñas positivas. Para cada clase se tiene un conjunto de 1,800,000 muestras para el entrenamiento y 200,000 muestras para la implementación de pruebas.

Para el desarrollo de este trabajo se tomó una muestra de 2000 reseñas positivas y 2000 reseñas negativas, dando un total de 4000 instancias, esto siendo el 2% del total de reseñas que se encuentran disponible en el conjunto de datos.

Como se muestra en la Figura 2, la metodología que se implementó consta de varios pasos, los cuales se siguieron estrictamente para un correcto desarrollo del trabajo. Inicialmente, fue necesario realizar la conversión del corpus en archivos con formato txt. Esto debido a que, los archivos del conjunto de datos originalmente se encuentran en formato separado por comas. La finalidad de esto es crear un archivo tipo txt por cada una de las reseñas de los usuarios de Amazon, y así tener separadas cada una de las clases del conjunto de datos (negativa y positiva).

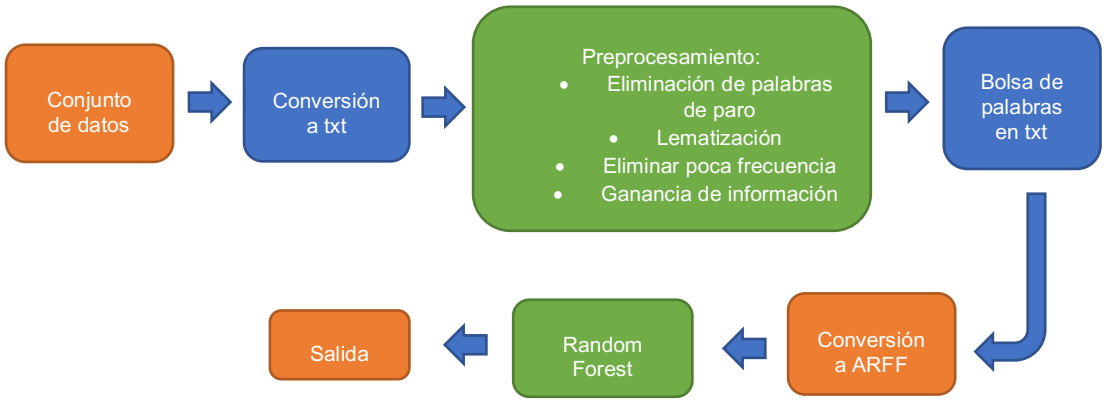

Figura 2. Metodología propuesta.
Fuente: Elaboración propia.

Es importante aclarar que solo se tomó la tercera columna de la estructura de las reseñas, debido a que este campo contiene la reseña como tal. Por lo tanto, las demás columnas se descartaron ya que no eran relevantes para este caso de estudio.

Una vez hecha la conversión a archivos txt, se procedió a realizar el procesamiento de los datos. Dicho procesamiento se realizó mediante el lenguaje de programación Python, de hecho, la mayoría de los procesos de este estudio se realizaron mediante este lenguaje. En estos procesos se incluye la conversión de los archivos, el procesamiento de los datos, la conversión a ARFF, y la obtención de archivo con ganancia de información.

Siguiendo con el procesamiento de los datos, se utilizaron diversas librerías que desempeñaron un papel fundamental en este proceso. Una de estas fue "NLTK", la cual simplificó varias tareas, como la tokenización y la eliminación de caracteres especiales y palabras de parada. Además, se implementó la librería "Spacy", que resultó especialmente beneficiosa para la lematización, otorgando una ventaja significativa en comparación con otras bibliotecas disponibles, gracias a su capacidad para identificar de manera más eficaz los verbos irregulares.

Una vez completados los procesos anteriores, se genera y guarda una bolsa de palabras en un archivo de texto. Posteriormente, se lleva a cabo la conversión de este archivo a un formato ARFF para permitir su lectura por el software Weka. Este archivo debe incluir todos los atributos o palabras clave necesarios, así como la declaración de los datos, para garantizar una clasificación precisa.

Por último, se abrió el archivo ARFF en el software Weka. Dentro de esta plataforma, se aplicó el método de aprendizaje automático Random Forest a los datos procesados, con el propósito de llevar a cabo la clasificación de estos. En la Figura 3 se muestra un ejemplo de cómo se observan estos datos en el software Weka.

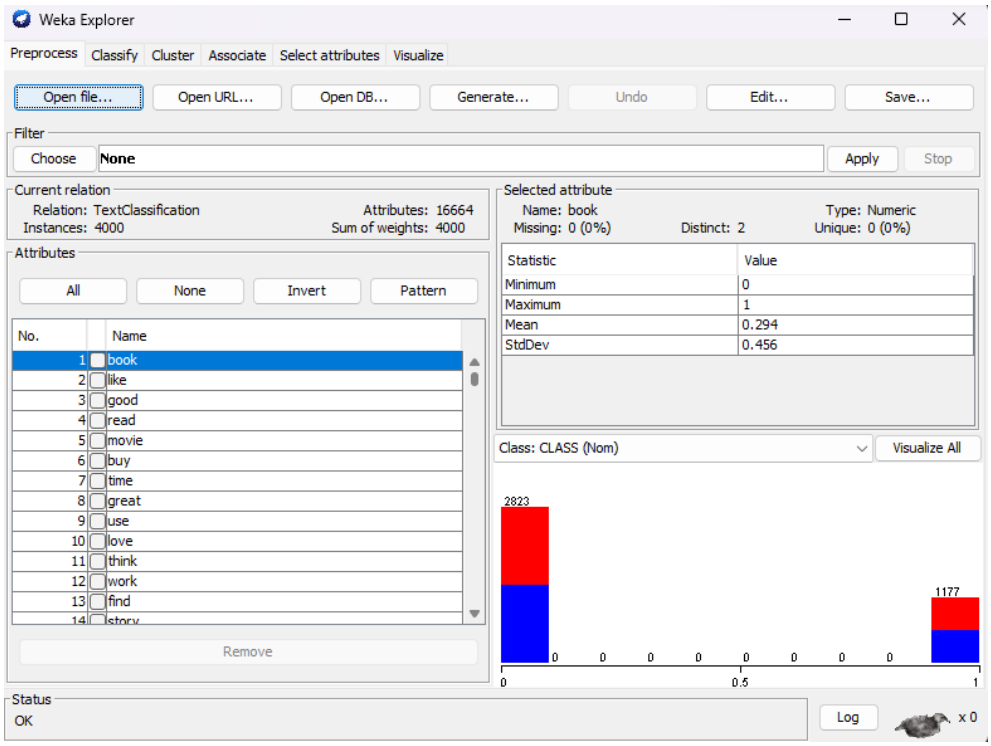

Figura 3. Datos en el software Weka.
Fuente: Elaboración propia.

Resultados

Se crearon cuatro archivos para su análisis en Weka, con el objetivo de examinar el comportamiento de cada uno de ellos. El primer archivo, denominado "amazonBL" (baseline), se generó sin aplicar ningún procesamiento significativo, limitándose a la tokenización de palabras y la eliminación de caracteres para que pudiera ser procesado por el software.

El segundo archivo se creó a partir del primero como punto de partida. En este caso, se le aplicó la eliminación de "stop words" y se eliminaron las palabras con frecuencia de baja aparición. Este archivo recibió el nombre de "amazonST".

El tercer archivo se desarrolló a partir del segundo. En este caso, se aplicó el proceso de lematización, y se le asignó el nombre de "amazonLE".

El último archivo se generó a partir del tercero. El propósito de este archivo era obtener una ganancia de información mediante la selección de atributos con un umbral mayor a cero. Este archivo se denominó "amazonIG".

Después de realizar diversas pruebas en Weka, el archivo ARFF con el que se obtuvieron los mejores resultados fue “amazonIG” para el método de aprendizaje seleccionado. La construcción de este modelo se demoró 25.92 segundos. Para el cual, se configuro una validación cruzada de 4 carpetas para la clasificación. Como se muestra en la Figura 4 se obtuvo una precisión general de 80.9% con 3236 instancias correctas. De igual forma, en esta misma Figura se logra observar la matriz de confusión generada.

```
Time taken to build model: 25.92 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances        3236               80.9    %
Incorrectly Classified Instances       764               19.1    %
Kappa statistic                          0.618
Mean absolute error                      0.3242
Root mean squared error                  0.3832
Relative absolute error                 64.8488 %
Root relative squared error             76.638  %
Total Number of Instances             4000

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.795    0.177    0.818      0.795    0.806      0.618    0.877     0.870     Negative
                 0.824    0.206    0.800      0.824    0.812      0.618    0.877     0.862     Positive
Weighted Avg.    0.809    0.191    0.809      0.809    0.809      0.618    0.877     0.866

=== Confusion Matrix ===

    a    b   <-- classified as
 1589  411 |    a = Negative
  353 1647 |    b = Positive
```

Figura 4. Resultados con Random Forest para el archivo "amazonIG".
Fuente: Elaboración propia.

Referencias

Alroobaea, R. (2022). Sentiment analysis on Amazon product reviews using the Recurrent Neural Network (RNN). International Journal of Advanced Computer Science and Applications, 13(4), 314-318. https://doi.org/10.14569/ijacsa.2022.0130437.

Chauhan, U. A., Afzal, M. T., Shahid, A., Abdar, M., Basiri, M. E., & Zhou, X. (2020). A comprehensive analysis of adverb types for mining user sentiments on Amazon product reviews. World Wide Web, 23(3), 1811-1829. https://doi.org/10.1007/s11280-020-00785-z.

Daniel, D. A. J., & Meena, M. J. (2021). A novel sentiment analysis for Amazon data with TSA based feature selection. Scalable Computing: Practice and Experience, 22(1), 53-66. https://doi.org/10.12694/scpe.v22i1.1839.

Attwal, K. P. S., & Dhiman, A. S. (2020). Exploring data minig tool-Weka and using Weka to build and evaluate predictive models. Advances and Applications in Mathematical Sciences, 19(6), 451-469. https://www.mililink.com/upload/article/131953194aams_vol_196_april_2020_a3_p451-469_kanwal_preet_singh_attwal.pdf.

Witten, I. H., Frank, E., Hall, M. A., & Pal, C. J. (2016). Data Mining: Practical Machine Learning Tools and Techniques. Morgan Kaufmann. https://www.google.com.mx/books/edition/Data_Mining/1SylCgAAQBAJ?hl=en&gbpv=1&dq=Data+Mining:+Practical+Machine+Learning+Tools+and+Techniques.+Morgan+Kaufmann&printsec=frontcover.

Nandal, N., Tanwar, R., & Pruthi, J. (2020). Machine Learning based aspect level sentiment analysis for Amazon products. Spatial Information Research, 28(5), 601-607. https://doi.org/10.1007/s41324-020-00320-2.

Sivakumar, M., & Reddy, U. S. (2021). Aspect-based sentiment analysis of mobile phone reviews using LSTM and fuzzy logic. International journal of data science and analytics, 12(4), 355-367. https://doi.org/10.1007/s41060-021-00277-x.

Mishina, Y., Murata, R., Yamauchi, Y., Yamashita, T., & Fujiyoshi, H. (2015). Boosted Random Forest. IEICE Transactions on Information and Systems, E98.D(9), 1630-1636. https://doi.org/10.1587/transinf.2014opp0004.