Tarea 3

IRVING DANIEL ESTRADA LÓPEZ
Facultad de Ciencias Físico Matemáticas
Universidad Autónoma de Nuevo León
Nuevo León, México
irving.estradalo@uanl.edu.mx

I. Introducción

Los modelos predictivos son un conjunto de técnicas que mediante los algoritmos de Machine Learning, la recolección de datos históricos, el Big Data y el reconocimiento de patrones, pretenden dar resultados futuros. Su objetivo es precisar la toma de decisiones, por esto mismo son de gran relevancia en la actualidad. Las empresas deben tomar decisiones, incluso en tiempo real.

El Machine Learning tiene ciertas ventajas sobre los modelos estadísticos tradicionales y los modelos econométricos. Es más eficiente al momento de trabajar con grandes volúmenes de datos, permite modelos más complejos de manera más efectiva que una regresión lineal. Los modelos econométricos son más utilizados para encontrar causalidad y relación entre las variables mientras que el Machine Learning se enfoca en la predicción y clasificación de los datos.

En este artículo se llevarán a cabo modelos predictivos acerca de las calificaciones del producto "Iphone SE". Primero se hará un análisis descriptivo de nuestro conjunto de datos para identificar posibles adversidades que afecten a los modelos. Después se llevará a cabo el preprocesado de nuestros datos para darles la estructura necesaria donde se incluye: preprocesado de texto, preprocesado del conjunto de datos no balanceados, entre otros. Se realizarán tres modelos predictivos: KNN (K-nearest neighbors), Decision Tree y Random Forest. Por último, se evaluarán los modelos y se seleccionará el que obtuvo mejores resultados.

II. Datos

El conjunto de datos con el que se trabajó en este artículo es de dominio público y fue obtenido de Kaggel. Nuestro conjunto de datos contiene 9,713 registros los cuales fueron extraídos de Flipkart utilizando Selenium y BeautifulSoup. Como podemos ver en la figura 1 el conjunto de datos contiene la valoración que le dan al producto Iphone SE, un comentario y la reseña de los clientes en el e-commerce de la India Flipkart.

Fig. 1.

	Ratings	Comment	Reviews
0	5	Super!	Great camera for pics and videos Battery life ...
1	5	Must buy!	Great device. Let me tell the Pros..1. Superb ...
2	5	Great product	Who all loves older size i.e., 4.7 inch type s...
3	5	Simply awesome	This iPhone SE is the best phone ever you get....
4	5	Classy product	This is my second iphone after iphone 4s. I've...

En la tabla 1 podemos revisar los atributos de nuestro conjunto de datos junto a una descripción, es importante destacar que la variable Ratings va de 1 a 5. Siendo la calificación más baja 1 y siendo la mejor 5.

Variable	Representación
Ratings	La calificación que los clientes le dieron al producto.
Comment	Comentario, es el título de la reseña.
Reviews	La reseña del cliente hacia el producto.

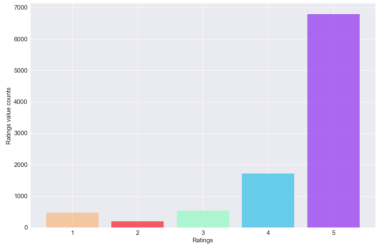
TABLE I
Variables del Conjunto de datos

III. Metodología

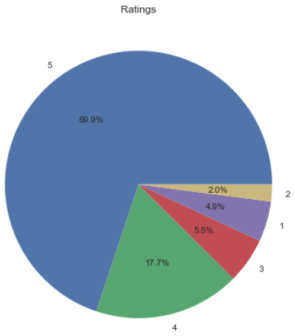
A. Análisis Exploratorio

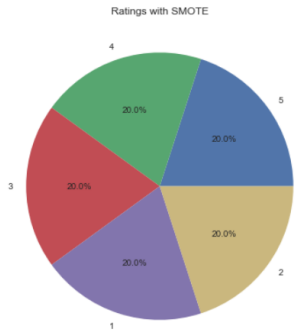
En la figura 2 tenemos el conteo de los registros de las respectivas calificaciones, podemos identificar que las calificaciones de 5 son las que predominan en el conjunto de datos. La clase con menos registros son las que contienen 2 como calificación. Ninguna de las otras calificaciones está cerca a la cantidad de observaciones que tienen 5 como calificación, esto nos indica que es un conjunto de datos no balanceado.

Fig. 2.



En la figura 3 podemos confirmar lo anteriormente mencionado ya que casi el 70% de nuestros datos tienen 5 como





calificación, las demás categorías no están cerca de la cantidad de registros de dicha categoría.

Fig. 3.

B. *Preprocesado*

A continuación, se presenta el preprocesado del texto. Nos enfocaremos en la variable "Reviews", en base a ella haremos las predicciones de la variable "Ratings". Este preprocesado ya ha sido discutido en los artículos anteriores. La intención de este proceso es limpiar nuestro texto para que de esta manera se pueda llevar a cabo un análisis correcto, liberando carga en los procesos posteriores que se tienen que realizar, beneficiando su rendimiento. El texto con el preprocesado es guardado en una nueva columna llamada "Cleaned_Reviews", la cual será con la que estaremos trabajando. A continuación, se presentan los pasos a seguir en el preprocesado de texto:

1) **Remover patrones de ruido**
2) **Remover Emojis**
3) **Remover URL**
4) **Remover signos de puntuación y números**
5) **Convertir a minúsculas**
6) **Revisión de Ortografía**
7) **Obtener el lemma de cada una de las palabras**
8) **Tokenization**
9) **Remover Stop Words**

Una vez llevado a cabo nuestro preprocesado de texto, lo siguiente es realizar el TF-IDF (Term Frequency — Inverse Data Frequency). Lo que busca esta técnica es medir la frecuencia de las palabras y después ponderarlas por su relevancia, es decir, que las palabras menos comunes serán las que más peso tengan, de esta manera éstas sean las que mejor expliquen de qué habla cada oración. Por esta parte era importante eliminar las Stop Words, ya que se hace una matriz respecto a cada palabra con su respectivo TF-IDF que con las Stop Words se convertiría en una matriz mucho más grande, obteniendo cada una de las Stop Words un TF-IDF de 0 debido a que no tienen significancia en las oraciones. En la figura 4 tenemos la matriz anteriormente mencionada y podemos identificar que cuenta con 1,050 columnas, éstas representan cada una de las palabras. Cabe destacar que este paso nos lleva de trabajar con datos no estructurados que en este caso fue el texto, a trabajar con una matriz con valores numéricos.

Fig. 4.

0	1	2	3	4	5	6	7	8	9	...	1040	1041	1042	1043	1044	1045	1046	1047	1048	1049
0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	...	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.138207	0.0
0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	...	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	0.0
0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	...	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	0.0
0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	...	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	0.0
0.0	0.0	0.0	0.152588	0.0	0.0	0.0	0.0	0.0	0.0	...	0.124142	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.144405	0.0

Como habíamos mencionado anteriormente en la parte de análisis descriptivo, las clases que tenemos en nuestro conjunto de datos no están balanceadas. Para que nuestros modelos tengan un buen desempeño al momento de aprender o identificar los patrones de cada una de las clases, éstas deben de tener un número cercano de observaciones entre sí, de lo contrario nuestros modelos no tendrán oportunidad de aprender correctamente las clases que cuenten con pocas observaciones. Para realizar el balance en nuestro conjunto de datos utilizaremos el algoritmo SMOTE (Synthetic Minority Over-sampling Technique) el cual genera datos sintéticos para equilibrar las respectivas clases. Este sobremuestreo se aplica a nuestra matriz de TF-IDF, de esta forma generando nuevas observaciones en base a las del conjunto de datos. Una ventaja que nos brinda esta solución es que no es necesario reducir el número de observaciones de la clase mayoritaria, perjudicando al número de observaciones en el conjunto de entrenamiento.

En la figura 5 podemos identificar que después de llevar a cabo lo anteriormente mencionado, nuestras clases ahora están balanceadas y representadas cada una de las 5 tipos de calificaciones como el 20% de nuestro conjunto total.

Fig. 5.

C. *KNN (K-nearest neighbors)*

La clasificación del algoritmo KNN está basada en aprendizaje por analogía, donde cada uno de los registros de nuestro conjunto de datos representa un punto en una dimensión n. Al momento de agregar un nuevo registro, el clasificador del KNN busca el patrón del espacio para los k puntos de nuestro conjunto de datos que estén más cerca del punto desconocido. Los puntos más cercanos son definidos por la distancia Euclidiana, así el punto desconocido es asignado a la clase más común entre sus k vecinos. El clasificador de vecinos más cercanos es basado en instancias o también conocido como "lazy learner" ya que almacena todo el conjunto de entrenamiento y no hace una clasificación hasta que una nueva observación tenga que ser clasificada. Algunas de las ventajas del algoritmo KNN son que es no paramétrico, es un algoritmo simple, es insensible a

valores atípicos, entre otras cosas. Sus desventajas son que es basado en instancias y puede ser computacionalmente costoso debido a que almacena los datos de entrenamiento.

D. Decision Tree

El árbol de decisión es un método no paramétricos de aprendizaje supervisado usado para la clasificación y regresión. Aprende generando reglas inferidas por las características de los datos. Algunas de las ventajas de los árboles de decisión son que son simples de interpretar, requieren poco preprocesado de datos, el costo de usar el árbol es logarítmico con respecto al número de puntos de datos usado en el entrenamiento del árbol. Algunas de las desventajas que tienen los árboles de decisión son que hay riesgo de un sobreajuste por la generación de reglas, también hay riesgo de que estén sesgados con respecto a los datos de entrenamiento. Cualquier pequeño cambio en los datos de entrada pueden suponer un árbol de decisión completamente diferente. También se cuenta con el problema de generar el numero óptimo de árboles de decisión.

E. Random Forest

El modelo de Random Forest como su nombre lo dice es un conjunto de árboles de decisión individuales que operan en conjunto. Cada uno de estos árboles de decisión en el Random Forest crea una clase de predicción y la clase con más votos es aquella que se convierte en nuestro modelo. La razón por la que el Random Forest es considerado un excelente modelo es debido a que un gran número de modelos relativamente no correlacionados superarán a cualquier modelo individual. Cada uno de los árboles de decisión se "protegerán" entre ellos con respecto a su error individual, mientras que algunos árboles no serán correctos, otros si los serán y se complementarán no tomando una decisión absoluta con un solo modelo. Una de las desventajas de este modelo es que se considera una caja negra, debido a su complejidad.

IV. Resultados

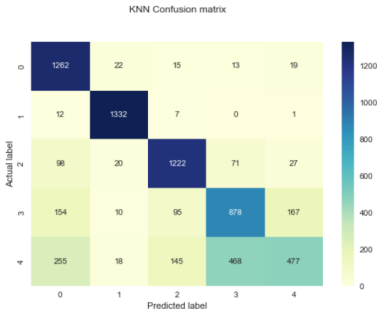
A continuación, se presentan los resultados de cada uno de los modelos. Se presenta la matriz de confusión, junto al reporte de clasificación. Nos centraremos en la métrica de evaluación F1-Score ya que dicha métrica engloba las métricas de precision y recall. También nos apoyaremos con el accuracy de cada uno de los modelos. Con ambas métricas evaluaremos cada modelo y definiremos cuál es el que tuvo mejor desempeño con nuestro conjunto de datos. Se realizó un Análisis de Componentes Principales (PCA) para reducir la dimensión de nuestros datos, de esta manera poder graficarlos y graficar la clasificación de cada uno de los modelos, sin embargo, la proporción de varianza acumulada de los primeros dos componentes alcanza solamente el 10%. La gráfica que pudiéramos obtener con esta proporción de varianza acumulada solamente nos explicaría el 10% de los datos. Por lo tanto, no se alcanza una buena proporción de varianza acumulada, para poder graficar nuestros datos con los dos primeros componentes.

A. Predicciones con KNN

En la figura 6 tenemos la matriz de confusión del modelo KNN, aparentemente obtuvo resultados aceptables. Tuvo problemas diferenciando entre las calificaciones 4 y 5, hay sospecha de que en dichas clases se utilicen palabras iguales o bastante parecidas ya que el número más bajo de la diagonal principal es de la última clase.

Viéndolo desde el punto de vista del modelo, existe la posibilidad de que en el espacio de dimensión n con respecto a éstos registros, con los cuales fue diseñado dicho espacio y representados como puntos de datos, estén en un área bastante cercana debido a su simitulud de palabras. Al momento de agregar nuevas observaciones y clasificarlas, el modelo debe de tener problemas, de esta manera identificando los k vecinos más cercanos de la clase equivocada. Podemos darnos cuenta en la matriz de confusión en el último renglón, la mayoría de los errores cometidos en la clase 5, están acumulados en en la clase 4. En las primeras 3 categorías aparentemente obtuvo buenos resultados, generando la sospecha de que se utilizan palabras lo suficiente diferentes para poder identificar correctamente dichas observaciones en el plano de dimensión n.

Fig. 6.



En la figura 7 tenemos el reporte de clasificación el cual complementa lo anteriormente mencionado, el modelo obtuvo malos resultados con la categoría 5, obteniendo un 46% de f1-score. De lo contrario la clase 2 fue la que obtuvo excelentes resultados con un 97% de f1-score. Este reporte apoya lo que identificamos en la matriz de confusión, las clases 4 y 5 son las más bajas en nuestras métricas de evaluación. El modelo tuvo un desempeño aceptable, alcanzando el 76% de accuracy.

Fig. 7.

	precision	recall	f1-score	support
1	0.71	0.95	0.81	1331
2	0.95	0.99	0.97	1352
3	0.82	0.85	0.84	1438
4	0.61	0.67	0.64	1304
5	0.69	0.35	0.46	1363
accuracy			0.76	6788
macro avg	0.76	0.76	0.74	6788
weighted avg	0.76	0.76	0.75	6788

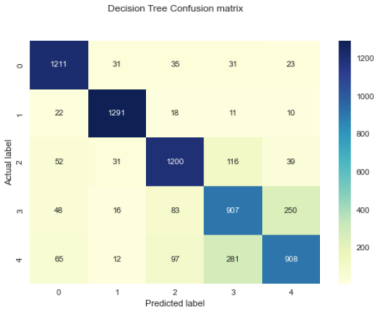
0.7617855038302888

B. Predicciones con Decision Tree

La matriz de confusión del modelo, ilustrada en la figura 8 nos indica que aparentemente tiene buenos resultados. A diferencia del KNN, éste modelo pareciera diferenciar de una

manera más efectiva las clases 4 y 5. Sin embargo, al igual que el modelo anterior podemos notar que sigue existiendo ese inconveniente de clasificar la clase 4 como 5 y viceversa. Estos resultados complementan la sospecha del modelo anterior que nuestros datos de dichas categorías utilizan palabras similares. A pesar de ser modelos con principios diferentes el KNN y el Decision tree, ya que el árbol de decisión es un algoritmo basado en reglas, ambos tuvieron inconvenientes clasificando dichas clases. Las demás clases aparentan tener una excelente clasificación. Tomando en cuenta solamente la matriz de confusión pudiéramos decir que tiene un buen desempeño el modelo clasificando los Ratings de acuerdo con las palabras, sin embargo, hay que apoyarnos en el reporte de clasificación para complementar nuestras observaciones.

Fig. 8.



Con la figura 9 podemos confirmar el buen desempeño del modelo, siendo la clase 4 la que tuvo mayor dificultad de ser clasificada. El accuracy del modelo es bueno, alcanzando el 81%. Las tres primeras clases son con las que aparentemente clasifica con mayor facilidad, a diferencia de las ultimas 2 clases.

Fig. 9.

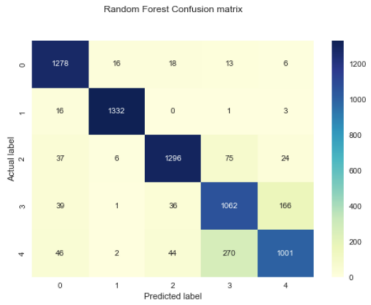
	precision	recall	f1-score	support
1	0.87	0.91	0.89	1331
2	0.93	0.95	0.94	1352
3	0.84	0.83	0.84	1438
4	0.67	0.70	0.68	1304
5	0.74	0.67	0.70	1363
accuracy			0.81	6788
macro avg	0.81	0.81	0.81	6788
weighted avg	0.81	0.81	0.81	6788

0.8127578078962876

C. Predicciones con Random Forest

La matriz de confusión del Random Forest en la figura 10 parece obtener excelentes resultados. Como habíamos mencionado anteriormente el Random Forest entra en la categoría de aprendizaje en conjunto, siendo sus bloques de construcción los árboles de decisión. Esta es una ventaja sobre los dos modelos pasados, ya que el aprendizaje en conjunto elabora un modelo más completo y menos sesgado. Podemos destacar que entre los otros dos modelos éste tiene los mejores resultados en la diagonal principal, este modelo al igual que los anteriores tiene problemas identificando las últimas dos categorías, sin embargo, en menor proporción de esta manera es el que mejores resultados ha obtenido.

Fig. 10.



Su reporte de clasificación en la figura 11 nos confirma lo anteriormente mencionado, alcanza una excelente acccuracy, con un 88%. Las primeras 3 categorías son las que clasifica con mayor facilidad, al igual que los anteriores dos modelos, sin embargo, obtiene un f1-score mayor que las anteriores. Pasa lo mismo con las últimas dos categorías, sin embargo, a diferencia de las primeras 3, éstas se quedan atrás.

Fig. 11.

	precision	recall	f1-score	support
1	0.90	0.96	0.93	1331
2	0.98	0.99	0.98	1352
3	0.93	0.90	0.92	1438
4	0.75	0.81	0.78	1304
5	0.83	0.73	0.78	1363
accuracy			0.88	6788
macro avg	0.88	0.88	0.88	6788
weighted avg	0.88	0.88	0.88	6788

0.879345904537419

D. Comparación entre modelos

Los tres modelos predictivos obtuvieron un buen desempeño, sin embargo, hay puntos a destacar de cada uno de ellos. Los tres modelos tienen características diferentes, el KNN está basado en aprendizaje por analogía, el decision tree genera reglas basadas en las características de los datos, y el random forest entra en la clasificación de aprendizaje en conjunto.

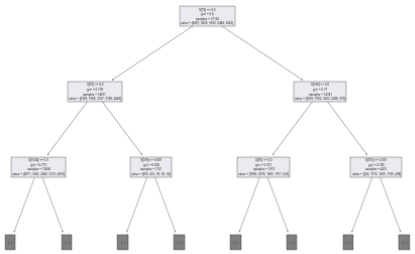
En la figura 12 tenemos una tabla comparativa de los F1-Score de cada una de las clases de los respectivos modelos, así como su accuracy. Las primeras 3 clases son las que mejor fueron clasificadas, esto pasó en todos los modelos. Se podría decir que las 3 primeras clases utilizan palabras lo suficientemente diferentes como para diferenciar con facilidad dichas clases, sin importar los métodos de clasificación utilizados en nuestros modelos. Por otro lado, las ultimas 2 clases son lo contrario. Dichas clases son las que obtuvieron menor F1-Score, sin importar el modelo. Cabe destacar que la complicación que más perjudico al KNN fue la clase 5, siendo el que obtuvo peor accuracy ya que esto es un reflejo de sus resultados en el F1-Score.

Fig. 12.

Modelo	F1-Score					Accuracy
	1	2	3	4	5	
KNN	0.81	0.97	0.84	0.64	0.46	0.76
Decision Tree	0.89	0.94	0.84	0.68	0.7	0.81
Random Forest	0.93	0.98	0.92	0.78	0.78	0.88

El modelo con el mejor desempeño en base al conjunto de datos que estamos trabajando y de acuerdo con la figura anterior, fue el Random Forest. Como hemos visto a través de este artículo, el Random Forest tiene una ventaja al ser aprendizaje en conjunto. El aprendizaje en conjunto usualmente tiene un mayor porcentaje de accuracy en las predicciones, comparado con los modelos individuales. Los sesgos que puede haber en las predicciones desaparecen, también el sobreajuste. Usualmente los modelos de aprendizaje en conjunto son más estables. Sin embargo, a diferencia de los modelos individuales, el aprendizaje en conjunto es difícil de interpretar, como mencionamos anteriormente es una caja negra. El costo computacional usualmente es mayor que los modelos individuales. El Random Forest comprobó ser un excelente modelo para el conjunto de datos que estuvimos trabajando. Comparando el Random Forest con el Decision Tree, podemos decir que el Random Forest es una versión más robusta y está formado con Decision Trees. El Decision Tree supera al Random Forest en velocidad, también lo superan al momento de ser interpretados debido a que es posible ilustrarlo, como podemos ver en la figura 13 tenemos nuestro árbol de decisión recortado debido a que el real consiste de mayor profundiad . El Decision tree y el KNN fueron superados por éste, sin embargo, cada algoritmo de predicción está diseñado para situaciones diferentes.

Fig. 13.



V. Conclusión

La intención con los análisis de datos es poder tomar decisiones correctas y efectivas. Tanto el sector público como el privado se han visto beneficiados por estos tipos de análisis. Los datos no estructurados circulan en la actualidad en grandes cantidades.

El análisis de texto ha cobrado relevancia en los últimos años. La extracción de información en base a una reseña o incluso a comentarios en redes sociales pueden ayudar a las empresas a tomar decisiones e incluso detectar posibles amenazas de sus productos o servicios. Poder tener una clasificación de la calificación o incluso de los sentimientos por medio del texto del usuario trae beneficios a los que proveen el producto o dicho servicio. Además de poder mejorar la experiencia del usuario, podemos identificar áreas de oportunidad en nuestro producto o incluso la necesidad de entrenamiento de los empleados que brindan el servicio evaluado.

Los modelos de Machine Learning nos permiten clasificar en base a los registros, estos aprenden de acuerdo con los datos de entrenamiento. En base a ellos nos permite detectar patrones, así como automatizar la clasificación. Estos modelos son herramientas poderosas en la toma de decisiones. Cabe destacar que no existe el modelo que lo resuelva todo, cada uno de los modelos de Machien Learning tienen sus ventajas y desventajas, así como lo vimos en éste artículo. Algunos modelos se desempeñan mejor en ciertas situaciones, es importante hacer comparativas de los modelos. Existen modelos que tienen un excelente desempeño con grandes volúmenes de información, pero los mismos modelos al momento de trabajar con cantidades pequeñas de información no tienen un buen desempeño. Pasa lo mismo con los datos estructurados y no estructurados. Es importante conocer los modelos y poder elegir de acuerdo con la situación del problema.

Los modelos al final son herramientas, como científicos de datos hay que saber cuándo utilizar cada una. Tenemos que saber interpretar los datos y los resultados, para de esta manera comunicar información valiosa y poder tomar decisiones.

References

[1] KAMAL DAS. (2021). Apple iPhone SE reviews ratings. Mayo del 2022, de Kaggle Sitio web: https://www.kaggle.com/datasets/kmldas/apple-iphone-se-reviews-ratings

[2] Irving Estrada. Github. 2022,Sitio web: https://github.com/Irving-Estrada/Procesamiento

[3] Thair Nu Phyu. (March 2009). Survey of Classification Techniques in Data Mining. Proceedings of the International MultiConference of Engineers and Computer Scientists , I.

[4] Gongde Guo, Hui Wang , David Bell , Yaxin Bi , and Kieran Greer . (2003). KNN Model-Based Approach in Classification. Springer-Verlag Berlin Heidelberg , 986-996.

[5] Nitin Bhatia, Vandana. (2010). Survey of Nearest Neighbor Techniques. (IJCSIS) International Journal of Computer Science and Information Security, 8, 2.