Grado en Ingeniería Informática

Inteligencia Artificial en las Organizaciones 2024-2025
Grupo 81

Práctica 2

"Data Mining"

Álvaro Guerrero Espinosa – 100472294
César López Mantecón – 100472092
Paula Subías Serrano – 100472119
Irene Subías Serrano – 100472108

Equipo 4

Profesor

Agapito Ledezma Espino

Índice

Índice de figuras

Índice de tablas

1. Introducción

En este documento se recoge el desarrollo de la segunda práctica de la asignatura *Inteligencia Artificial en las Organizaciones*. El objetivo será la predicción de la valoración que un sujeto dará a su experiencia volando con una aerolínea a partir de distintos campos incluidos en una reseña. Dos de estos campos se tratan de datos textuales. Todo el estudio se llevará a cabo en la herramienta Altair AI Studio.

Se obtendrán 3 modelos siguiendo distintas aproximaciones para predecir la clase a la que pertenece una crítica de un cliente en una aerolínea. Las estrategias a seguir son las siguientes:

- Modelo básico: se construye un modelo eliminando todos los atributos textuales.
- Modelo de *text mining I*: se construye un modelo con procesamiento de texto, pero sin emplear análisis de sentimientos.
- Modelo de *text mining II*: se construye un modelo con procesamiento de texto y análisis de sentimientos. Además, en este modelo se hará uso de bigramas y trigramas.

El uso de análisis textual puede resultar útil dado el tipo de problema que se pretende resolver, ya que entre los datos aparecen campos textuales no estructurados. Este tipo de datos no podrán ser utilizados por otro tipo de modelos de aprendizaje automático debido a que, por si mismos, no son capaces de reconocer patrones o información incluidos en datos textuales. De manera complementaria, el uso de *análisis de sentimientos* ayudará a la predicción al depender la variable objetivo fuertemente de la opinión del autor de la valoración. Además, al construir varios modelos se podrá comparar el aporte del *text mining* y el *análisis de sentimientos* frente al modelo básico sin campos textuales.

El número de pasajeros de avión a lo largo de los últimos 10 años ha crecido enormemente, haciendo imposible usar técnicas tradicionales para el análisis de la gran cantidad de datos que se disponen. Si se toma como referencia el año 2019, año previo a la pandemia del *COVID-19*, hubo 4490 millones de pasajeros [1]. Aunque únicamente un pequeño porcentaje de ellos dejara una valoración de su experiencia, la cantidad de datos para procesar sobrepasaría la capacidad de cualquier mecanismo tradicional de análisis de datos. Es por esto que el uso de algoritmos de *minería de datos* es crucial para obtener rendimiento de la información en este ámbito.

Además, esta clase de modelos cuentan con multitud de aplicaciones en diversos campos: la detección temprana de problemas en los servicios, la creación de marketing dirigido, el desarrollo de algoritmos de respuestas automáticas y sistemas de recomendación o el análisis competitivo de compañías para destacar sus fortalezas y puntos flacos en el mercado. Especialmente en el contexto actual de recuperación postpandemia, el uso de

algoritmos de minería de datos permite a las aerolíneas tomar decisiones más informadas y responder de forma más eficiente a las necesidades emergentes del mercado [2].

2. Análisis exploratorio de los datos (EDA)

Se ha realizado un estudio de los datos con el objetivo de su entendimiento para determinar el uso y significado de los mismos. Este análisis se ha realizado a través de las herramientas de Rapid Miner Statistics [3] y Correlation-Matrix [4]. Además, se han explorado únicamente los datos de entrenamiento para evitar problemas de *data_leak*.

El *dataset* cuenta con 13643 instancias de reseñas de vuelos. Para cada reseña se incluyen 19 atributos de distinta naturaleza: campos textuales, fechas, datos categóricos y datos numéricos. A continuación haremos un breve repaso por cada uno de los atributos, explorando su tipo y su significado.

- **Airline Name:** representa el nombre de la aerolínea asociado al vuelo reseñado. Se trata de un atributo *nominal*.
- **Aircraft:** representa el modelo de avión empleado en el vuelo reseñado. Se trata de un atributo *nominal*.
- **Cabin Staff Service:** puntuación numérica del 0 al 50 sobre el servicio del personal a bordo. Se trata de un atributo *numérico*. Cabe destacar que los valores que toma son siempre múltiplos de 10.
- **Date Flown:** fecha en la que se realizó el vuelo.
- **Food & Beverages:** puntuación numérica del 0 al 50 sobre el servicio de comida en el vuelo. Se trata de un atributo *numérico*. Cabe destacar que los valores que toma son siempre múltiplos de 10.
- **Ground Service:** puntuación numérica del 0 al 50 sobre el servicio en tierra. Se trata de un atributo *numérico*. Cabe destacar que los valores que toma son siempre múltiplos de 10.
- **Inflight Entertainment:** puntuación numérica del 0 al 50 sobre el servicio de entretenimiento durante el vuelo. Se trata de un atributo *numérico*. Cabe destacar que los valores que toma son siempre múltiplos de 10.
- **Overall Rating:** puntuación numérica del 1 al 9. Representa el grado de satisfacción del cliente. Se trata de un atributo *numérico*. Además, se trata de la **variable objetivo**.
- **Recommended:** valor *booleano*. Representa si el cliente recomendaría el servicio. Destaca un desbalanceo del 69.95 % hacia el valor negativo.

- **Review:** atributo *textual*. Contiene el comentario del usuario sobre el vuelo.
- **Review Date:** fecha en la que se realizó la reseña.
- **Review Title:** campo *textual*. Da título a la reseña.
- **Route:** campo *textual*. Representa la ruta del vuelo.
- **Seat Comfort:** puntuación numérica del 0 al 50 sobre la comodidad de los asientos. Se trata de un atributo *numérico*. Cabe destacar que los valores que toma son siempre múltiplos de 10.
- **Seat Type:** atributo *nominal*. Describe el tipo de asiento. Puede tomar los valores "Ecoclass", "Business Class", "Premium Economy" y "First Class".
- **Type Of Traveller:** atributo *nominal*. Representa el tipo de viajero. Puede tomar los valores "Solo Leisure", "Couple Leisure", "Family Leisure" y "Business".
- **Value For Money:** puntuación numérica del 0 al 50 sobre el valor percibido con respecto al dinero gastado. Se trata de un atributo *numérico*. Cabe destacar que los valores que toma son siempre múltiplos de 10.
- **Verified:** atributo *booleano*. Representa si la persona que realiza la reseña está verificada. Ambas clases están balanceadas, con un 55.77 % de inclinación hacia la clase mayoritaria.
- **Wifi & Connectivity:** puntuación numérica del 0 al 50 sobre la conexión a internet durante el vuelo. Se trata de un atributo *numérico*. Cabe destacar que los valores que toma son siempre múltiplos de 10.

En las siguientes tablas se recogen los datos relevantes para cada atributo:

Atributo	Airline Name	Aircraft	Cabin Staff Service	Date Flown
Tipo	Nominal	Nominal	Numérico	Fecha
Min	-	-	0	04/01/2012
Max	-	-	50	07/01/2023
Media	-	-	26.259	-
Moda	Tiger Air Australia	A320	10	05/01/2023
Missing (%)	0	70.57 %	16.81 %	11.82 %

Tabla 1: *Datos de los atributos - 1.*

Atributo	Food & Beverages	Ground service	Inflight Entertainment
Tipo	Numérico	Numérico	Numérico
Min	0	10	0
Max	50	50	50
Media	23.298	20.646	19.792
Moda	10	10	10
Missing (%)	38.15%	17.32%	53.10%

Tabla 2: *Datos de los atributos - 2.*

Atributo	Overall Rating	Recommended	Review	Review Date	Review Title
Tipo	Numérico	Booleano	Textual	Fecha	Textual
Min	1	-	-	22/11/2003	-
Max	9	-	-	27/07/2023	-
Media	3.368	-	-	-	-
Moda	1	NO	-	16/07/2023	-
Missing (%)	0%	0%	0%	0%	0%

Tabla 3: *Datos de los atributos - 3.*

Atributo	Route	Seat Confort	Seat Type	Type of Traveller
Tipo	Nominal	Numérico	Nominal	Nominal
Min	-	0	-	-
Max	-	50	-	-
Media	-	23.831	-	-
Moda	Melbourne to Sydney	10	Economy Class	Solo Leisure
Missing (%)	12.22%	16.28%	1.71%	11.79%

Tabla 4: *Datos de los atributos - 4.*

Atributo	Value For Money	Verified	Wifi & Connectivity
Tipo	Numérico	Booleano	Numérico
Min	0	-	10
Max	50	-	50
Media	22.050	-	15.228
Moda	10	Verdadero	10
Missing (%)	1.62%	0%	72.69%

Tabla 5: *Datos de los atributos - 5.*

2.1. Análisis de la variable objetivo

Si se observa en más profundidad la variable objetivo, se puede ver que se trata de una puntuación del 1 al 9 que muestra el grado de satisfacción general del autor de la reseña. A continuación se muestra un gráfico de su distribución.

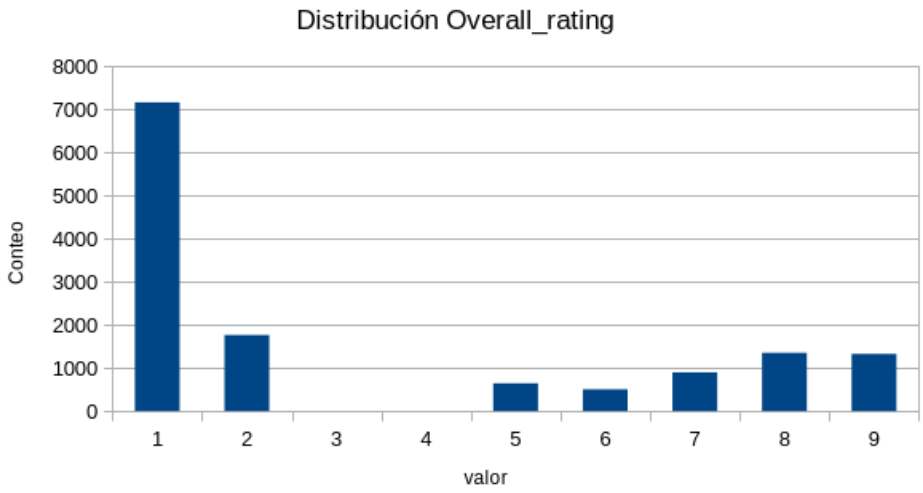


Fig. 1: *Distribución de los valores de OVERALL_RATING.*

Se aprecia como se trata de un atributo con una distribución desigual, con concentraciones en los valores extremos. Particularmente, la calificación 1 es la más común con 7159 registros. Esto parece indicar un alto grado de insatisfacción y un sesgo hacia las calificaciones negativas. También es destacable la ausencia de reseñas con calificaciones intermedias (3 y 4).

Estos datos podrían ser problemáticos para el entrenamiento de modelos debido al fuerte desbalanceo que existe entre clases.

3. Preprocesado

En este capítulo se describen las diferentes transformaciones aplicadas a los datos para su posterior uso en el entrenamiento de modelos.

3.1. Transformación de *OVERALL_RATING* a ordinal

Para poder construir un clasificador se ha transformado el atributo *OVERALL_RATING* de numérico a ordinal; codificándolo en 3 clases:

- 1: agrupa los valores comprendidos entre 1 y 3, ambos incluidos.

- 2: agrupa los valores comprendidos entre 4 y 6, ambos incluidos.
- 3: agrupa los valores comprendidos entre 7 y 9, ambos incluidos.

De esta forma también se trata de compensar la poca representación de algunos valores en el conjunto de datos. Cabe destacar que aun agrupando sigue habiendo una representación muy pobre de la clase *2*.

Clase	1	2	3
Número de instancias	8921	1148	3574
Porcentaje	65.39 %	8.41 %	26.20 %

Tabla 6: *Porcentaje de representación de cada clase.*

La sobrerrepresentación de la clase correspondiente a las evaluaciones más bajas se puede explicar a través de la tendencia de las personas a publicar una crítica en caso de descontento. A continuación se muestra un gráfico de tarta con la distribución de cada una de las clases:

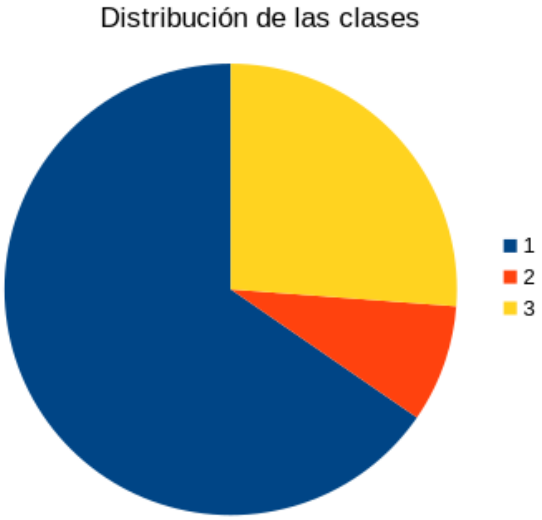


Fig. 2: *Distribución de las clases.*

3.2. Eliminación de atributos

Se han seguido 3 criterios para la eliminación de atributos:

- Relevancia: algunos datos no aportan información relevante de cara al entrenamiento del modelo. Por este criterio se han eliminado *Date Flown* y *Date Review*.

- Afectar negativamente al modelo: algunos atributos nominales cuentan con un número de clases muy numeroso. En el caso de *Route*, tratar con todas ellas añade una gran complejidad que no se traduce a un mejor rendimiento. En consecuencia, se ha eliminado este atributo.
- Numerosos valores faltantes: se han eliminado todos los atributos con un número de *missing values* superior al 35 % de los valores totales. Esto es porque se considera que no contamos con información suficiente para imputar estos datos y que hacerlo envenenaría los datos reales.

A continuación se muestra un gráfico con el número de valores faltantes por atributo:

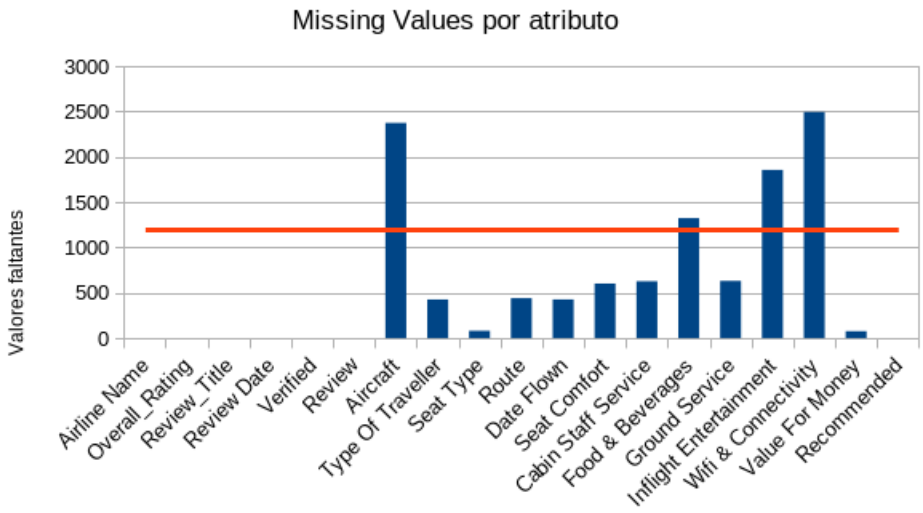


Fig. 3: *Número de valores faltantes por atributo.*

La línea sobre el gráfico 3 señala el 35 % de las instancias. Por lo tanto, toda barra que rebase la línea se corresponde con un atributo a eliminar. Los atributos afectados han sido: *Aircraft, Food & Beverages, Inglight Entertainment* y *Wifi & Connectivity*.

3.3. Formateo de los datos

Se han sustituido los valores de las columnas booleanas (i.e. *Verified* y *Recommended*) por valores binarios. Los valores nominales se han codificado mediante *One-hot encoding* (*OHE*), a excepción de *Airline* (esto es por limitaciones de la implementación). Los atributos codificados a través de *OHE* han sido *Seat Type* y *Type of Traveller*.

3.4. Imputación de valores faltantes

En cuanto a la imputación de valores faltantes, se han reemplazado mediante el operador *Replace Missing Values* de *Altair AI Studio* [5] por la moda de cada atributo. Los atributos

afectados han sido *Type Of Traveller, Seat Type, Food & Bervarges, Groud Service, Seat Confort* y *Value For Money*.

4. Modelo básico sin usar datos textuales

Se ha entrenado un modelo básico de *deep learning* para utilizarlo como referencia y cuantificar los beneficios proveídos por el *text mining*. Además del preprocesado de los datos especificado anteriormente, se han eliminando los atributos textuales *Review* y *Review Title*.

Tanto en el proceso de entrenamiento como en la evaluación se observa una *accuracy* aproximada del 92 %. Se puede observar en la matriz de confusión que los resultados tanto de *precission* como de *recall* son muy buenos para las clases 1 y 3, siendo los valores en ambos casos cercanos al 90 %. Sin embargo, en el caso de la clase 2, estos los valores no alcanzan el 55 % de *precission*. Se concluye que esto se debe a la poca representación de los datos de esta clase (8 %).

	True 3	True 2	True 1	precision
pred. 3	3355	387	25	0.8906
pred. 2	192	482	206	0.5477
predl 1	27	279	8690	0.9660
recall	0.9387	0.4199	0.9741	

Tabla 7: *Matriz de confusión en el conjunto de train - Modelo base.*

5. Proceso de entrenamiento

En este capítulo se describen los procesos de entrenamiento para la construcción de los modelos de la parte 1 y parte 2. Para que el experimento sea reproducible se ha fijado una semilla a 1992 y se han marcado las opciones pertinentes en el software de *Altair AI Studio* para la ejecución en un único hilo.

Cabe destacar que para la construcción de estos modelos se han empleado exclusivamente los atributos textuales *Review* y *Review Title*.

5.1. Parte 1: Modelado sin análisis de sentimientos

Para el entrenamiento de distintos modelos se ha trabajado con variaciones sobre los parámetros del preprocesado de los datos y los modelos usados. Además, se han empleado distintos tipos de modelo con sus parámetros por defecto con el fin de tener una mayor variedad y encontrar el más efectivo.

De esta forma, se ha realizado un *grid-search* con los siguientes valores:

- Preprocesado: {*TF-IDF*, *Binary Term Occurrence*, *prune below* 3 %, *prune below* 10 %}.
- Modelos: {*Deep Learning*, *Random Forest*, *Naive Bayes*, *Decision Trees*}

La función de evaluación de término determina qué se necesita para considerar que un término es relevante en un documento. Se han seleccionado "TF-IDF" y "Binary Term Occurrence" por representar 2 enfoques muy distintos. En el primer caso se necesita que el término sea muy frecuente en el documento pero muy poco frecuente en otros documentos, lo que puede significar que es especialmente relevante en el documento. El segundo caso, sin embargo, es mucho más simple: únicamente comprueba si el término aparece en el documento o no.

Con respecto al *pruning*, se ha seleccionado una cota inferior del 3 y 10 % para eliminar los términos muy poco frecuentes según la función de evaluación de término. Se considera que estos términos no tendrán relevancia en los resultados, por lo que eliminarlos permitirá construir modelos más simples y rápidos de entrenar sin cambiar los resultados. Inicialmente también se seleccionó una cota inferior del 0 %, pero esto causó que hubiese demasiados términos en el resultado, incrementando demasiado el tiempo de ejecución del entrenamiento. Debido a esto, se decidió descartar este valor.

Al concluir el entrenamiento se han obtenido los siguientes resultados:

Modelo	Parámetros[1]	Accuracy
1	(*TF-IDF*, 3 % *Deep Learning*)	$0{,}842 \pm 0{,}009$
2	(*TF-IDF*, 3 % *Random Forest*)	$0{,}660 \pm 0{,}006$
3	(*TF-IDF*, 3 % *Naive Bayes*)	$0{,}778 \pm 0{,}007$
4	(*TF-IDF*, 3 % *Decision Trees*)	$0{,}665 \pm 0{,}024$
5	(*Binary Term Occurrence*, 3 % *Deep Learning*)	$0{,}843 \pm 0{,}005$
6	(*Binary Term Occurrence*, 3 % *Random Forest*)	$0{,}805 \pm 0{,}010$
7	(*Binary Term Occurrence*, 3 % *Naive Bayes*)	$0{,}748 \pm 0{,}009$
8	(*Binary Term Occurrence*, 3 % *Decision Trees*)	$0{,}798 \pm 0{,}007$
9	(*TF-IDF*, 10 %, *Deep Learning*)	$0{,}819 \pm 0{,}008$
10	(*TF-IDF*, 10 %, *Random Forest*)	$0{,}743 \pm 0{,}009$
11	(*TF-IDF*, 10 %, *Naive Bayes*)	$0{,}748 \pm 0{,}008$
12	(*TF-IDF*, 10 %, *Decision Trees*)	$0{,}749 \pm 0{,}023$

Tabla 8: *Parámetros y Accuracy de los diferentes modelos - Parte 1.*

[1] El formato de los parámetros es el siguiente: (*Vector creation*, porcentaje de *prune below*, modelo usado)

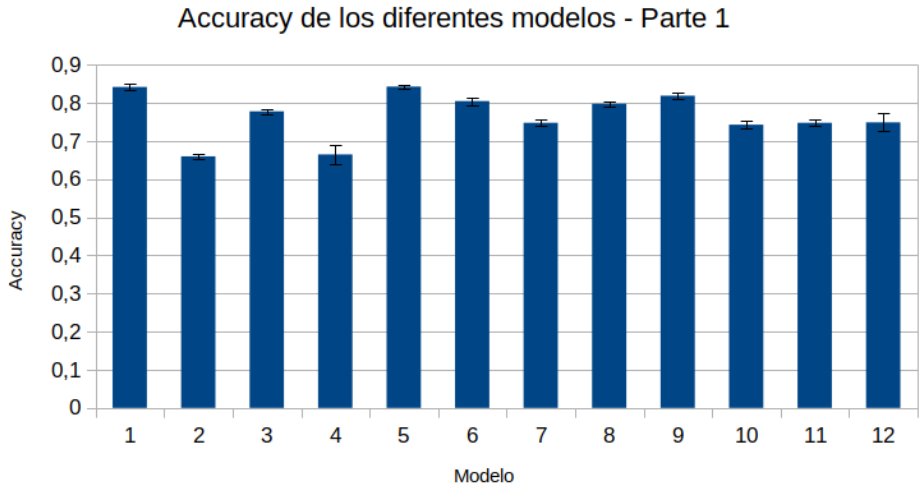


Fig. 4: *Accuracy de los diferentes modelos - Parte 1.*

Al observar la comparación por pares proporcionada por el *t-test* (figura 5), destacan los modelos 1 y 5 por obtener los mejores resultados y ser significativamente diferentes de todos los demás. Ambos de estos modelos usan un *prune below* del 3 % y "Deep Learning". El modelo 9 cuenta con el siguiente mejor error, aunque este es significativamente distinto del error de los modelos 1 y 5. Este modelo cuenta con un *prune below* del 10 %, lo que puede eliminar demasiados términos, llegando a quitar términos potencialmente relevantes y explicando el peor rendimiento. En los tres modelos se usa la familia de modelos "Deep Learning", lo que indica que, en general, esta familia de modelos tiene una mejor capacidad para adaptarse a los datos y generalizar los patrones encontrados. Los modelos 1 y 5 se diferencian en la función de evaluación de término, pero esto no crea diferencias significativas en el error.

Cabe destacar también que los modelos basados en "Decision Trees" obtienen una tolerancia en el error significativamente mayor que los demás modelos. Esto puede indicar que este tipo de modelos dependen mucho de los datos de entrenamiento usados, lo que puede dificultar la estimación de su rendimiento para nuevos datos.

A	B	C	D	E	F	G	H	I	J	K	L	M
	0.842 +/- 0.009	0.660 +/- 0.006	0.778 +/- 0.007	0.665 +/- 0.024	0.843 +/- 0.005	0.805 +/- 0.010	0.748 +/- 0.009	0.798 +/- 0.007	0.819 +/- 0.008	0.743 +/- 0.009	0.748 +/- 0.008	0.749 +/- 0.023
0.842 +/- 0.009		0.000	0.000	0.000	0.783	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.660 +/- 0.006			0.000	0.575	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.778 +/- 0.007				0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.001
0.665 +/- 0.024					0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.843 +/- 0.005						0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.805 +/- 0.010							0.000	0.069	0.003	0.000	0.000	0.000
0.748 +/- 0.009								0.000	0.000	0.263	0.818	0.681
0.798 +/- 0.007									0.000	0.000	0.000	0.000
0.819 +/- 0.008										0.000	0.000	0.000
0.743 +/- 0.009											0.151	0.461
0.748 +/- 0.008												0.969
0.749 +/- 0.023												

Fig. 5: *Resultado del t-test de los modelos en el conjunto de train - Parte 1.*

Con toda esta información se concluye que el mejor modelo y con el que se hará la

predicción es el modelo 5, debido a que es el modelo con mejor *accuracy*, manteniendo una gran simplicidad al usar *Binary Term Occurrence* como función de evaluación de término. En la siguiente figura se muestra el *recall* y *precision* de las diferentes clases para este modelo.

	True 3	True 2	True 1	precision
pred. 3	2952	549	323	0.7720
pred. 2	284	156	208	0.2407
predl 1	338	443	8390	0.9148
recall	0.8260	0.1359	0.9405	

Tabla 9: *Matriz de confusión del modelo 5 en el conjunto de train - parte 1.*

5.2. Parte 2: Modelado con análisis de sentimientos

Al igual que en la parte anterior, se ha trabajado con variaciones sobre los parámetros del preprocesado de los datos y los modelos usados.

De esta forma se realiza un *grid-search* con los siguientes valores:

- Preprocesado: {*bigramas*, *trigramas*, *prune below* 3 %, *prune below* 10 %}.
- Modelos: {*Deep Learning*, *Random Forest*, *Naive Bayes*, *Decision Trees*}

La longitud de los n-gramas permite reconocer términos compuestos por múltiples palabras, pero si es demasiado grande puede generar muchos términos sin significado semántico. Se han seleccionado longitudes de 2 y 3 ya que permiten reconocer la mayoría de términos compuestos sin permitir generar términos compuestos por demasiadas palabras.

Con respecto al *pruning*, se ha seleccionado una cota inferior del 3 y 10 % para eliminar los términos sin significado semántico que pueden generar los n-gramas. Estos términos serán muy poco frecuentes en los datos, por lo que estarán por debajo de esta cota inferior. Inicialmente también se seleccionó una cota inferior del 0 %, pero esto causó que hubiese demasiados términos en el resultado, incrementando en gran medida el tiempo de ejecución del entrenamiento, sin que esto se tradujera a un mejor rendimiento. Debido a esto, decidimos descartar este valor.

Al concluir el entrenamiento se han obtenido los siguientes resultados:

Modelo	Parámetros[2]	Accuracy
1	(*Bigramas*, 3 %, *Deep Learning*)	$0{,}845 \pm 0{,}007$
2	(*Bigramas*, 3 %, *Random Forest*)	$0{,}655 \pm 0{,}001$
3	(*Bigramas*, 3 %, *Naive Bayes*)	$0{,}784 \pm 0{,}007$
4	(*Bigramas*, 3 %, *Decision Trees*)	$0{,}669 \pm 0{,}024$
5	(*Trigramas*, 3 %, *Deep Learning*)	$0{,}847 \pm 0{,}006$
6	(*Trigramas*, 3 %, *Random Forest*)	$0{,}656 \pm 0{,}003$
7	(*Trigramas*, 3 %, *Naive Bayes*)	$0{,}784 \pm 0{,}007$
8	(*Trigramas*, 3 %, *Decision Trees*)	$0{,}670 \pm 0{,}025$
9	(*Bigramas*, 10 %, *Deep Learning*)	$0{,}815 \pm 0{,}007$
10	(*Bigramas*, 10 %, *Random Forest*)	$0{,}730 \pm 0{,}015$
11	(*Bigramas*, 10 %, *Naive Bayes*)	$0{,}761 \pm 0{,}010$
12	(*Bigramas*, 10 %, *Decision Trees*)	$0{,}752 \pm 0{,}038$

Tabla 10: *Parámetros y Accuracy de los diferentes modelos - Parte 2.*

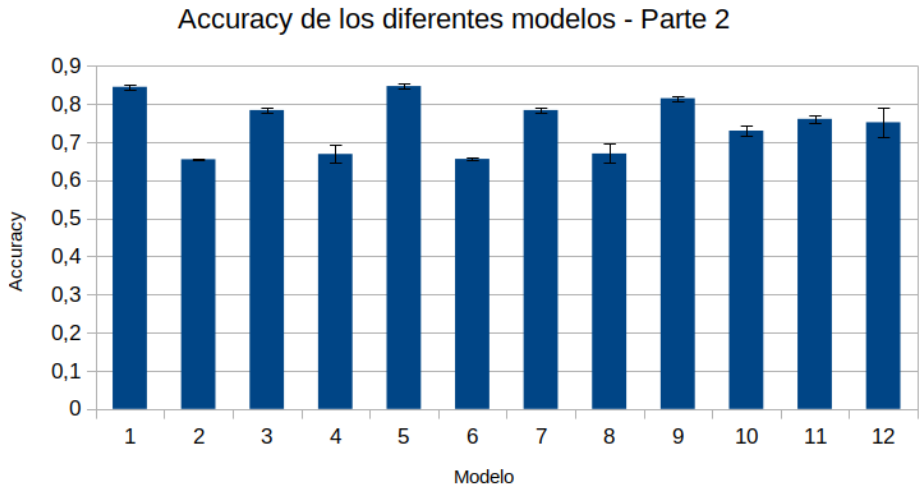


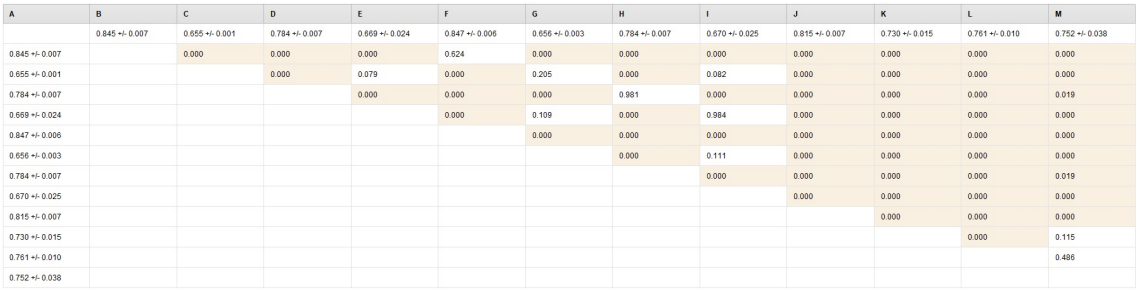
Fig. 6: *Accuracy de los diferentes modelos - Parte 2.*

Al observar la comparación por pares proporcionada por el *t-test* (figura 7), destacan de nuevo los modelos 1 y 5 (*Deep Learning* con *prune below* 3 %). De la misma forma que en la parte anterior, el siguiente modelo con el mejor error es el 9 (*prune below* 10 %), verificando la hipótesis realizada anteriormente de que este valor de *prune below* es demasiado alto y elimina términos relevantes. Al igual que en la parte anterior, los modelos con mejor rendimiento usan la familia de modelos de "Deep Learning", confirmando la hipótesis de que esta familia tiene una mejor capacidad de aprendizaje y generalización.

[2]El formato de los parámetros es el siguiente: (Longitud *n*-gramas, porcentaje de *prune below*, modelo usado)

Por último, se puede ver que los modelos basados en “Decision Trees” siguen teniendo una tolerancia en el error significativamente más alta que los demás modelos, confirmando la hipótesis de que estos modelos dependen mucho de los datos utilizados para el entrenamiento.

En este caso, la diferencia entre los modelos 1 y 5 es la longitud de los n-gramas usados. Como no hay diferencias significativas en los errores de estos modelos, se puede concluir que no hay n-gramas de longitud 3 relevantes en los datos.



A	B	C	D	E	F	G	H	I	J	K	L	M
	0.845 +/- 0.007	0.655 +/- 0.001	0.784 +/- 0.007	0.669 +/- 0.024	0.847 +/- 0.006	0.656 +/- 0.003	0.784 +/- 0.007	0.670 +/- 0.025	0.815 +/- 0.007	0.730 +/- 0.015	0.761 +/- 0.010	0.752 +/- 0.038
0.845 +/- 0.007		0.000	0.000	0.000	0.624	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.655 +/- 0.001			0.000	0.079	0.000	0.205	0.000	0.082	0.000	0.000	0.000	0.000
0.784 +/- 0.007				0.000	0.000	0.000	0.981	0.000	0.000	0.000	0.000	0.019
0.669 +/- 0.024					0.000	0.109	0.000	0.984	0.000	0.000	0.000	0.000
0.847 +/- 0.006						0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.656 +/- 0.003							0.000	0.111	0.000	0.000	0.000	0.000
0.784 +/- 0.007								0.000	0.000	0.000	0.000	0.019
0.670 +/- 0.025									0.000	0.000	0.000	0.000
0.815 +/- 0.007										0.000	0.000	0.000
0.730 +/- 0.015											0.000	0.115
0.761 +/- 0.010												0.486
0.752 +/- 0.038												

Fig. 7: *Resultado del t-test de los modelos en el conjunto de train - Parte 2.*

Con toda esta información se concluye que el mejor modelo y con el que se hará la predicción es el modelo 1, debido a que es el modelo con mejor *accuracy* y es de los modelos más simples al usar *bigramas*. En la siguiente figura se muestra el *recall* y *precision* de las diferentes clases para este modelo.

	True 3	True 2	True 1	precision
pred. 3	2946	534	278	0.7839
pred. 2	304	186	240	0.2548
predl 1	324	428	8403	0.9179
recall	0.8243	0.1620	0.9419	

Tabla 11: *Matriz de confusión del modelo 1 en el conjunto de train - parte 2.*

6. Comparación de resultados

En este capítulo se comparan los resultados obtenidos en los diferentes procesos para cada uno de los modelos.

6.1. Procesos de entrenamiento

Durante el proceso de entrenamiento destaca positivamente el modelo base, contando con un *Accuracy* superior al 91 %. Los modelos basados en *minería de texto* cuentan con un

rendimiento similar, entorno al 84 %. Es relevante mencionar que la clase mejor predicha en todos los modelos es la clase 1, hecho que se explica por la sobrerrepresentación de esta clase en el conjunto de los datos. En contraste, la clase con la puntuación más baja es la clase 2. A continuación se muestra una tabla y un gráfico que recogen el *Accuracy* por cada clase y modelo.

Modelo	Acc. clase1	Acc. clase2	Acc. clase 3
Modelo base	0.9741	0.4199	0.9387
Modelo p1	0.9405	0.1359	0.826
Modelo p2	0.9419	0.162	0.824

Tabla 12: *Accuracy por modelo y clase en el conjunto de entrenamiento.*

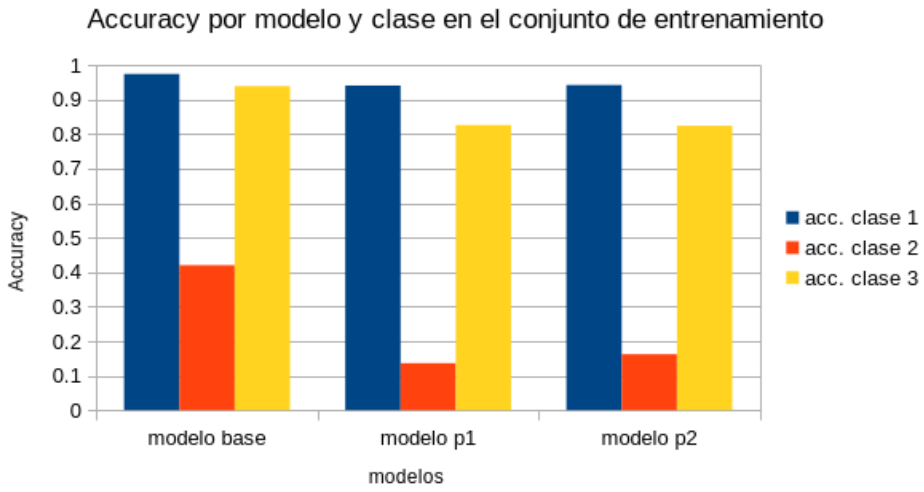


Fig. 8: *Accuracy por modelo y clase en el conjunto de train.*

Tal y como se esperaba en un inicio, las clases mejor predichas son aquellas con un mayor número de instancias. Cabe destacar el rendimiento superior del modelo base, especialmente hablando de la clase 2. Estos resultados parecen indicar que los campos textuales no contienen suficiente información para distinguir las reseñas intermedias (clase 2) de las de otro tipo.

Al observar a la matriz de confusión de cada modelo se aprecia cómo los valores de *precision* y *recall* son similares para las clases 1 y 3 en todos los modelos. Esto parece indicar que se tratan de modelos robutos que son capaces de capturar y clasificar correctamente todas las instancias de estas clases. Para la clase 2, los valores de *precision* y *recall* son similares únicamente en el caso del modelo base. Para los demás modelos, estos valores son muy bajos y desbalanceados hacia la *precision*. En cualquier caso, las medidas realizadas sobre la clase 2 para todos los modelos indican que ninguno ha sido capaz de

capturar y clasificar correctamente las instancias de esta clase; muy probablmente debido a la poca representación de la misma en el conjunto de los datos.

6.2. Evaluación

Tras realizar predicciones sobre el conjunto de *test* los resultados obtenidos con cada modelo han sido los siguientes:

Modelo	Acc. clase1	Acc. clase2	Acc. clase 3
Modelo base	0.9837	0.4184	0.8896
Modelo p1	0.9851	0.0	0.7038
Modelo p2	0.9435	0.2163	0.7934

Tabla 13: *Accuracy por modelo y clase en el conjunto de test.*

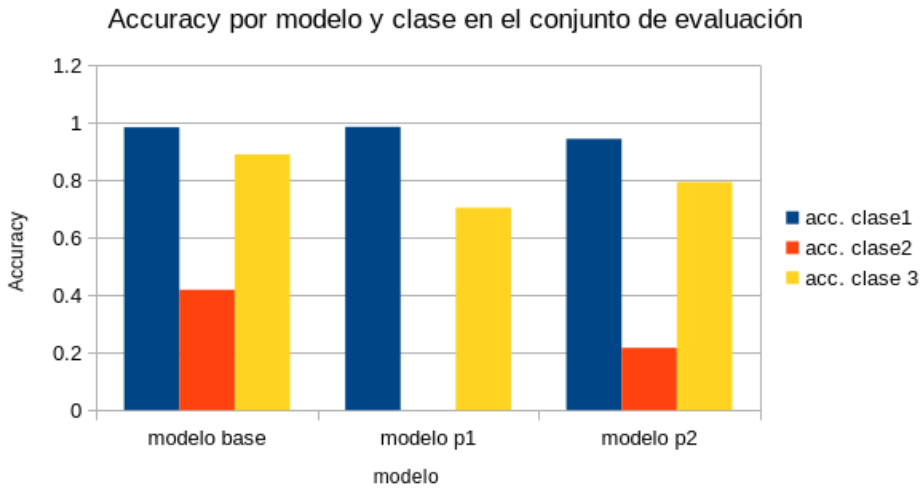


Fig. 9: *Accuracy por modelo y clase en el conjunto de test.*

Se aprecia un comportamiento similar de los modelos al compararlos con los resultados obtenidos en el entrenamiento. Esto significa que ninguno de los modelos ha sufrido de *overfitting*. Destaca que el modelo de la parte 1 no ha sido capaz de predecir correctamente ninguna de las instancias de la clase 2. También es notable cómo el modelo base cuenta con un rendimiento superior al resto de modelos. Los modelos de la parte 1 y parte 2 se comportan de forma similar, destacando el segundo por mostrar un mejor rendimiento en las clases minoritarias.

A continuación, con el fin de hacer un análisis más profundo, se desglosarán las predicciones en una matriz de confusión para cada modelo.

6.2.1. Modelo base

La matriz correspondiente al modelo base, construído exclusivamente con atributos no textuales, es la siguiente:

	True 3	True 2	True 1	precision
pred. 3	814	75	4	0.9115
pred. 2	88	118	32	0.4958
predl 1	13	89	2178	0.9553
recall	0.8896	0.4184	0.9837	

Tabla 14: *Matriz de confusión del modelo base con el conjunto test.*

Se aprecia que las medidas para *precision* y *recall* son altas y similares para las clases mayoritarias. Esto significa que se trata de un modelo robusto capaz de capturar y predecir correctamente las instancias pertenencientes a las clases 1 y 3. Para la clase 2 contamos con las medidas más altas entre los 3 modelos. No obstante, siguen siendo medidas muy bajas que muestran un rendimiento pobre para esta clase de instancias. Cabe destacar que el modelo clasifica una cantidad mayor de instancias de la clase 3 como clase 2. Esto indica que existe una separación más clara entre las instancias de la clase 1 y la clase 2 que entre la clase 3 y la clase 2. Esto se puede corroborar al observar la distribución de los valores de la variable objetivo *OVERALL_RATING* previo a la transformación del atributo de numérico a ordinal (figura 1).

6.2.2. Parte 1: modelo con *text mining*

La matriz correspondiente al modelo de la parte 1, construído con atributos exclusivamente textuales y sin uso de análisis de sentimientos, es la siguiente:

	True 3	True 2	True 1	precision
pred. 3	644	105	33	0.8235
pred. 2	3	0	0	0.0
pred. 1	268	177	2181	0.8305
recall	0.9851	0.0	0.7038	

Tabla 15: *Matriz de confusión del modelo de la parte 1 con el conjunto test.*

Este modelo emplea técnicas de *text mining* para analizar los campos *Review* y *Review title* y una red neuronal (operador *DeepLearning* [6] del software).

De forma similar al modelo anterior, las medidas de *precision* y *recall* son altas y similares para las clases mayoritarias. De nuevo, se trata de un modelo robusto capaz de

reconocer y clasificar correctamente las instancias pertenencientes a las clases mayoritarias. El modelo muestra un rendimiento notablemente superior en la clasificación de instancias de la clase 1, hecho esperable ya que se trata de la clase mayoritaria. Sin embargo, muestra un rendimiento paupérrimo para la clase 2, seguramente debido a la poca representación de esta clase. El modelo no ha sido capaz de capturar ninguna de las 282 instancias de la esta clase.

6.2.3. Parte 2: modelo con *text mining*, *sentiment analysis* y *n-gramas*

La matriz correspondiente al modelo de la parte 2, construído con atributos exclusivamente textuales y empleando análisis de sentimientos y construcción de n-gramas, es la siguiente:

	True 3	True 2	True 1	precision
pred. 3	726	122	67	0.7934
pred. 2	112	61	58	0.2641
pred. 1	77	99	2089	0.9223
recall	0.7934	0.2163	0.9435	

Tabla 16: *Matriz de confusión del modelo de la parte 2 con el conjunto test.*

Este modelo emplea técnicas de *text mining* para analizar los campos *Review* y *Review title*. Además, realiza un análisis de sentimientos y construye n-gramas de tamaño 2. Por último, emplea una red neuronal (operador *DeepLearning* [6] del software) para realizar la predicción.

Al igual que en los anteriores modelos, las medidas de *precision* y *recall* son altas y similares para las clases mayoritarias; demostrando ser un modelo robusto para las clases 1 y 3. El modelo muestra un rendimiento significativamente superior en la clasificación de instancias de la clase 2 al compararlo con el modelo 1, lo que parece demostrar que el analisis de sentimientos y la construccion de bigramas permiten reconocer nuevos patrones capaces de separar las instancias de la clase 2. Sin embargo, el rendimiento para esta clase sigue siendo muy bajo, mostrando como el conjunto de datos no cuenta con representación suficiente de esta clase como para construir un modelo capaz de predecir correctamente este tipo de instancias.

6.3. Conclusiones de los resultados

Los tres modelos predicen correctamente las instancias pertenecientes a las clases mayoritarias, destacando el modelo base al mostrar mejor rendimiento para ambas clases. Sin embargo, cabe destacar que los modelos basados en *text mining* han empleado únicamente 2 atributos del conjunto de datos.

También se aprecia que, pese a que el modelo de la parte 1 tenga un mejor rendimiento para la clase 1, el análisis de sentimientos y la construcción de n-gramas hacen que el modelo de la parte 2 tenga un mejor rendimiento general.

En cuanto a la clase 2, ninguno de los modelos ha logrado tener un rendimiento suficiente en el reconocimiento de este tipo de instancias, siendo el que mejores resultados presenta el modelo base. Destaca que el modelo de la parte 2 es capaz de capturar más instancias que el modelo de la parte 1, por lo que se concluye que el análisis de sentimientos permite segregar mejor el conjunto de los datos.

Con todo lo anterior, se concluye que en el conjunto de datos existe suficiente información en los atributos no textuales para la construcción de modelos robustos de clasificación. Sin embargo, las técnicas de *text mining* y análisis de sentimientos han demostrado ser muy potentes y capaces de reconocer patrones y extraer información a partir de menos atributos; logrando acercarse al rendimiento del modelo base. Además, las redes de neuronas artificiales han demostrado ser capaces de generalizar correctamente y extraer la información relevante de los conjuntos de datos. Es importante mencionar que para la construcción de un modelo capaz de trabajar con las instancias de la clase 2 se proponen 3 alternativas:

- Mayor recopilación de datos de estas instnacias: recoger más datos de reseñas con puntuaciones entre 3 y 6.
- Utilización de técnicas de *Undersampling* de las clases mayoritarias: especialmente de la clase 1.
- Otras divisiones de los datos: se podría trabajar con 2 clases en lugar de 3, agrupando la clase 2 y la clase 3 en una única clase. Sin embargo, seguiría existiendo un desbalanceo hacia la clase 1 automático.

7. Parte opcional: Topic modelling

Se ha realizado un último análisis de los datos como parte de la sección opcional de la investigación basado en el *topic modelling*. El objetivo de este es identificar patrones en los datos que puedan agrupar las reseñas en distintos grupos. Para esto se va a utilizar el operador de RapidMiner "Extract Topics from Data" [7], que usa el algoritmo de Latent Dirichlet Allocation para identificar y agrupar temas de los documentos.

El preprocesado de los datos para la aplicación de este algoritmo será similar al utilizado en las dos partes anteriores de este trabajo, con la excepción de la eliminación del atributo objetivo. Este preprocesado es realizado para poder comparar los resultados con los datos de manera efectiva, aunque en el operador solo se utilice la columna "Review". Se buscará agrupar las reseñas de dos maneras modificando el parámetro *number of topics*:

- En dos grupos, con la hipótesis de que se dividirán en positivas y negativas. En este caso se compararán los grupos obtenidos con la columna de *Recommended*, considerándose esta una buena medida de la positividad de la reseña.
- En tres grupos que se compararán con el valor de *OVERALL_RATING* ya reducido a tres opciones en el preprocesado. El objetivo es comprobar si estos textos cuentan con agrupaciones naturales que se relacionen con la valoración final del usuario.

Además, en ambos casos se comprobarán las *top words* proporcionadas por cada una de las agrupaciones con el objetivo de identificar los temas obtenidos. Cabe resaltar que se utilizará la opción de 10 *top words* por ser este un número lo suficientemente grande como para recabar información aún siendo manualmente analizable. Este parámetro no afecta al agrupamiento final o al rendimiento del proceso.

Se van a utilizar 1000 iteraciones para ambos procesos, siendo este un número que consigue una buena precisión en los resultados sin sacrificar el rendimiento. Se han explorado rangos más bajos y más altos de iteraciones, sin éxito. En el primer caso la reducción de precisión era demasiado significativa y en el segundo el aumento del tiempo no llevaba a una mejora sustancial en la efectividad del modelo.

Con estos parámetros se han obtenido los resultados expuestos a continuación.

7.1. Generación de 2 grupos y comparación con *Recommended*

Una vez ejecutado el modelo se consiguen dos salidas: las instancias clasificadas en dos grupos distintos y las 10 palabras más comunes en ambos grupos. Las instancias clasificadas han sido comparadas con su valor en el atributo *Recommended*, obteniendo el siguiente resultado:

TopicID	Nº Instancias	Recommended = 0	Recommended = 1
0	8216	7949	367
1	5327	1603	3724

Tabla 17: *Resultados de topic modelling basado en Recommended.*

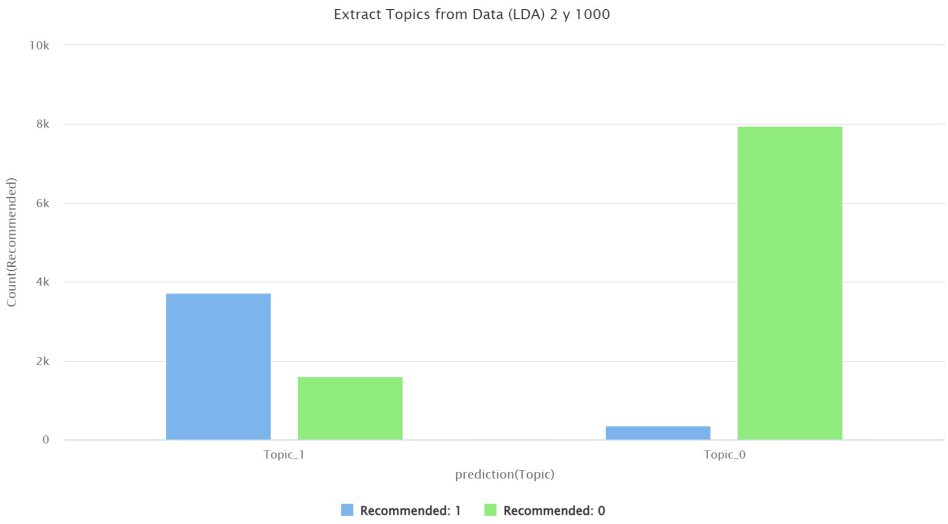


Fig. 10: *Resultados de topic modelling basado en Recommended.*

Se puede observar que ambos grupos cuentan con una mayoría de instancias de una de las categorías de *Recommended*, lo cual puede indicar que las agrupaciones corresponden en cierta medida con la opinión general de los usuarios. En el caso del grupo 0 la diferencia parece mucho menor, pero esto probablemente esté causado por el desbalance que existe en las clases de *Recommended*, siendo la clase 0 casi tres veces más común que la 1. Así, se puede concluir que la clasificación en grupos del LDA es correcta en relación con este atributo.

Al analizar las palabras más comunes de ambas agrupaciones se pueden encontrar nombres y adjetivos que pueden apoyar la idea de que efectivamente se han creado dos grupos de experiencias positivas y negativas.

topicid	word	weight	topicid	word	weight
0	flight	17165	1	flight	7211
0	airline	6157	1	good	3358
0	hours	4662	1	time	2970
0	get	4303	1	crew	2955
0	time	4248	1	seats	2907
0	airport	4090	1	service	2842
0	service	4083	1	seat	2773
0	would	4075	1	food	2629
0	told	3723	1	cabin	2237
0	one	3280	1	staff	1862

Tabla 18: *Top words en las agrupaciones por Recommended.*

En el grupo 0, que corresponde a las experiencias negativas, se encuentran palabras

como “hours” que probablemente tengan relación con malas experiencias de tiempos de espera o cancelaciones. En el segundo grupo, sin embargo, se encuentran palabras como “good” o “staff” que representan opiniones positivas sobre el vuelo y los empleados. Cabe destacar que aunque existan palabras útiles para el análisis en ambos casos, la mayoría de palabras son comunes entre ambos grupos y no cuentan con una connotación negativa o positiva, siendo inútiles para conseguir conclusiones. Especialmente, “flight” o “time” dependen completamente del calificativo del que van acompañadas, por lo que sería necesario realizar un análisis que permita expresiones de varias palabras.

7.2. Generación de 3 grupos y comparación con *Overall_Rating*

En la segunda aproximación los resultados de la clasificación son:

TopicID	Nº Instancias	Overall Rating = 1	Overall Rating = 2	Overall Rating = 3
0	5281	4762	230	289
1	3354	3196	73	85
2	5008	963	845	3200

Tabla 19: *Resultados de topic modelling basado en Overall_Rating.*

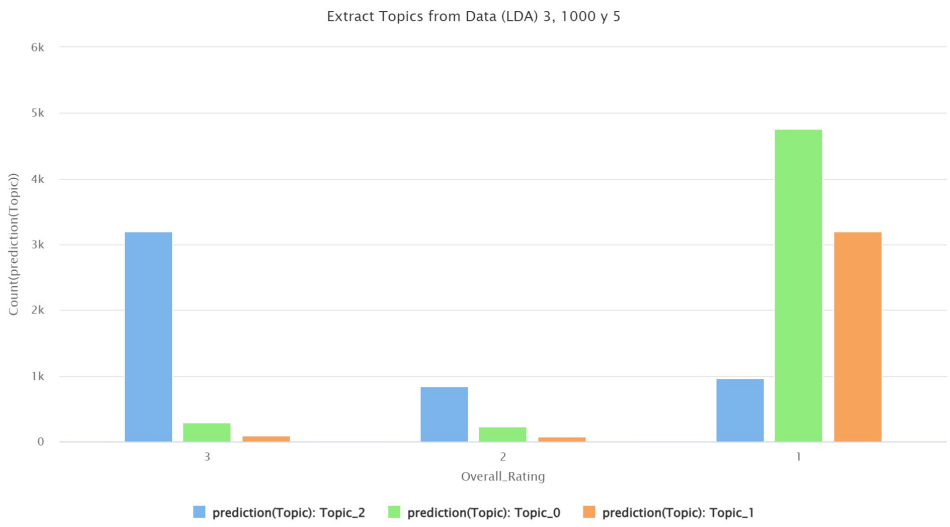


Fig. 11: *Resultados de topic modelling basado en Overall_Rating.*

Una vez más se observa que el desbalance de las clases del atributo provoca una diferencia en la precisión de los resultados entre los grupos. En este caso, aunque el *topic* 2 representa de manera significativa las valoraciones positivas de los usuarios, los *topics* 0 y 1 parecen ambos corresponder con las instancias negativas. La poca existencia de instancias que representen la clase 2 de *Overall Rating* probablemente se corresponde con la

falta de existencia de una agrupación natural de los datos para esta. Así, se puede teorizar que solo existen dos agrupaciones de datos, siendo el grupo intermedio insignificante. Esta teoría viene además acompañada de los datos de las *top words* de cada agrupación:

topicid	word	weight	topicid	word	weight	topicid	word	weight
0	flight	10718	1	flight	7478	2	flight	6180
0	hours	3755	1	airline	2841	2	good	3313
0	airline	3513	1	service	2521	2	service	2701
0	airport	3449	1	customer	2429	2	time	2680
0	time	3225	1	refund	2287	2	crew	2644
0	luggage	3133	1	ticket	2121	2	seats	2489
0	plane	3023	1	cancelled	1871	2	food	2408
0	staff	2856	1	would	1835	2	seat	2362
0	get	2824	1	flights	1789	2	cabin	2074
0	one	2702	1	get	1667	2	flight	1591

Tabla 20: *Top Words en las agrupaciones por Overall_Rating.*

La agrupación 2 cuenta con palabras claramente positivas como “good”, pero ambas agrupaciones 0 y 1 cuentan con palabras de connotación negativa como “hours”, “worst” o “cancelled”.

Tras estos resultados se ha decidido realizar una comparación de las dos agrupaciones de la primera sección con los datos de *Overall_Rating* con el objetivo de probar la teoría. Esto resulta en una distribución de dos grupos claros en los que la mayoría de instancias de valoraciones positivas se encuentran en uno de los grupos y la mayoría de las negativas en otro.

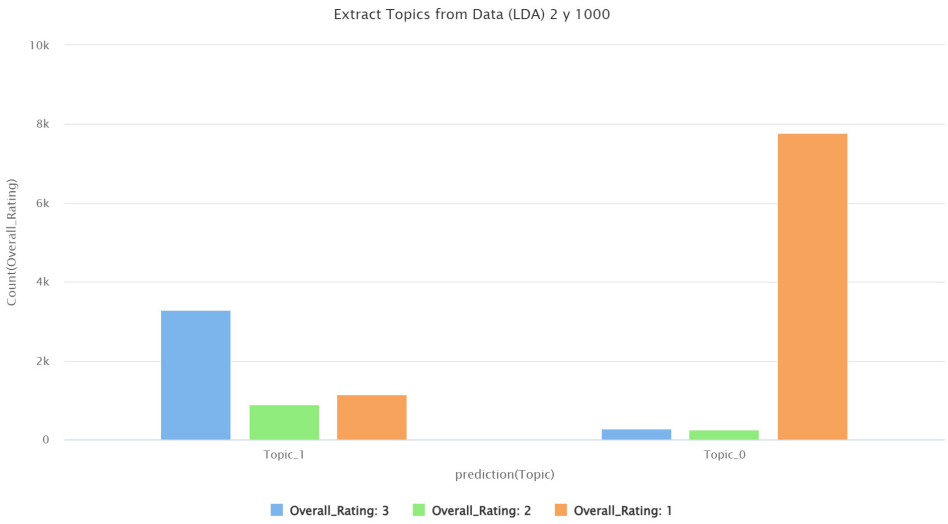


Fig. 12: *Resultados de topic modelling basado en Overall_Rating.*

Se concluye que el método de *topic modelling* en este caso puede resultar una herramienta útil para la clasificación de instancias en dos grupos, existiendo patrones en los datos que contribuyen a dos agrupaciones naturales de estos en reseñas positivas y negativas.

8. Conclusiones de la práctica

Esta práctica ha permitido la exploración de nuevas técnicas para la construcción de modelos. En concreto, se ha podido comprobar como el *text mining* y el *análisis de sentimientos* son herramientas muy últiles capaces de inferir información a partir de pocos campos textuales. Resulta verdaderamente interesante comprobar como el campo de la inteligencia artificial ha evolucionado hasta extremos donde es capaz de reconocer la actitud de un agente hacia un dominio a través de una información tan compleja como es el lenguaje natural.

También se considera importante destacar cómo esta clase de herramientas deben usarse con conocimiento ya que muchas presentan una gran complejidad que afecta fuertemente al tiempo de entrenamiento y el rendimiento de los modelos. Esto deja ver la importancia de un correcto entendimiento de las técnicas y modelos empleados; destacando nuestro valor como ingenieros en el campo de la Inteligencia Artificial.

Bibliografía

[1] I. C. A. O. via Airlines for America, *Global number of airline passengers [dataset]*, Processed by Our World in Data, Consultado: 30 octubre, 2024, 2023. [En línea]. Disponible en: https://ourworldindata.org/grapher/number-airline-passengers.

[2] D. Perez-Campuzano, P. M. Ortega, L. R. Andrada y A. López-Lázaro, «Artificial Intelligence potential within airlines: a review on how AI can enhance strategic decision-making in times of COVID-19,» *Journal of Airline and Airport Management*, vol. 11, n.º 2, pp. 53-72, 2021. DOI: https://doi.org/10.3926/jairm.189.

[3] Altair. «Statistics - Altair RapidMiner Documentation.» (2024), [En línea]. Disponible en: https://docs.rapidminer.com/2024.1/studio/operators/cleansing/data_statistics.html (Acceso: 23-10-2024).

[4] Altair. «Correlation Matrix - Altair RapidMiner Documentation.» (2024), [En línea]. Disponible en: https://docs.rapidminer.com/2024.1/studio/operators/modeling/correlations/correlation_matrix.html (Acceso: 23-10-2024).

[5] Altair. «Replace Missing Values - Altair RapidMiner Documentation.» (2024), [En línea]. Disponible en: https://docs.rapidminer.com/2024.1/studio/operators/cleansing/missing/replace_missing_values.html (Acceso: 23-10-2024).

[6] Altair. «DeepLearning - Altair RapidMiner Documentation.» (2024), [En línea]. Disponible en: https://docs.rapidminer.com/latest/studio/operators/modeling/predictive/neural_nets/deep_learning.html (Acceso: 26-10-2024).

[7] Altair. «LDA - Altair RapidMiner Documentation.» (2024), [En línea]. Disponible en: https://docs.rapidminer.com/2024.1/studio/operators/extensions/Operator%20Toolbox/text_processing/lda_exampleset.html (Acceso: 30-10-2024).