Práctica 3. IA en la empresa

FERNANDO JAVIER BORAO ZACCHEO - 100506847

ADRIÁN CERROS SÁNCHEZ - 100405342

CLARA MIRANDA GARCÍA - 100518506

LUIS VICENTE ARÉVALO - 100505651

Índice

Introducción

La Inteligencia Artificial (IA) se ha establecido como una herramienta transformadora en el ámbito empresarial, impulsando la innovación y eficiencia en una amplia gama de sectores. Lejos de reemplazar la inteligencia y la creatividad humanas, la IA actúa como un instrumento de apoyo que amplía nuestras capacidades, especialmente en el procesamiento y análisis de grandes volúmenes de datos. Esto permite a las empresas anticipar necesidades, optimizar procesos y tomar decisiones estratégicas con una velocidad y precisión sin precedentes.

Las aplicaciones de IA en las empresas son diversas y profundamente impactantes, extendiéndose desde la automatización de procesos rutinarios mediante el uso de softwares de Automatización de Procesos Robóticos (RPA), hasta sistemas complejos de gestión como los ERP (Enterprise Resource Planning), CRM (Customer Relationship Management) y SCM (Supply Chain Management). Estas herramientas no solo mejoran la eficiencia operativa, sino que también transforman las interacciones con los clientes y la gestión de la cadena de suministro, ofreciendo un enfoque más integrado y basado en datos para la toma de decisiones.

Estado del arte

Este ejercicio se sitúa dentro del campo del Procesamiento del Lenguaje Natural (PLN), un área que ha experimentado notables desarrollos recientemente. Destacan, por ejemplo, diversas soluciones enfocadas en la clasificación temática de textos, tal como se requiere en este proyecto. Asimismo, existen modelos innovadores como el GPT-3 de OpenAI, capaz de generar texto continuando desde un punto inicial, o GitHub Copilot, diseñado para escribir código automáticamente interpretando comentarios en lenguaje natural sobre las funcionalidades deseadas.

Con los progresos en PLN, se ha establecido un proceso estándar que comienza con la extracción y preparación de textos, lo cual incluye la eliminación de términos innecesarios o redundantes, como preposiciones y palabras de significado similar dentro de un mismo campo semántico. Posteriormente, se convierten estos textos en vectores que representan las palabras contenidas. La etapa final implica clasificar cada texto según las palabras que lo componen.

Para llevar a cabo este ejercicio, utilizaremos RapidMiner, un software diseñado para simplificar el análisis y la minería de datos. Gracias a su interfaz gráfica, RapidMiner facilita la tarea de arrastrar y unir operadores para configurar y ejecutar el análisis deseado.

Pre-procesado

El pre-procesado, véase el Anexo 1 del documento, se compone de las siguientes etapas:

- Tokenización: Dividir el texto en unidades más pequeñas llamadas tokens, en este caso, en palabras.
- Eliminar stopwords: Las stop-words son palabras que son poco significativas para el análisis y pueden aumentar el ruido en los datos. Incluyen, por ejemplo, preposiciones, conjunciones y artículos. También se puede aplicar un filtrado de las stopwords en base a su aparición en el fichero dado. De esta manera se eliminan palabras en concreto.
- Estandarización : Normalizar el texto para asegurar consistencia. En este caso se convierte todo el texto a minúsculas. Ayuda a reducir la complejidad y la variabilidad dentro de los datos de texto.
- Filtrado de tokens: Se filtran aquellas palabras de dos o menos letras, así se elimina ruido adicional en los datos de texto, pues las palabras muy cortas a menudo son menos significativas y pueden no aportar valor relevante al análisis. En concreto se considera eliminar aquellas con longitud de palabra menor a dos caracteres y mayor que veinticinco, considerando estas últimas como excepciones en el texto.
- Stemming: Reduce las palabras a su raíz o forma base, por lo general eliminando sufijos. Para este trabajo, se hace uso del Snowball Stemming, útil para estandarizar variantes de una palabra y simplificar el análisis de texto en tareas como la búsqueda y clasificación de información.

Propuesta desarrollada

Modelo supervisado

Para el modelo supervisado, véase el Anexo 2, se carga un conjunto de datos que servirá para enseñar al modelo las características y patrones que debe aprender. También se cargan datos de prueba para evaluar la capacidad del modelo de generalizar a nuevos ejemplos que no ha visto durante el entrenamiento.

Con los datos de entrenamiento se aplica un algoritmo de DeepLearning. Según la documentación de RapidMiner respecto a este algoritmo, éste se fundamenta en una red neuronal artificial que avanza a través de múltiples capas y que se entrena

utilizando el método del gradiente descendente estocástico con la técnica de retropropagación. La red incorpora numerosas capas ocultas con neuronas que funcionan con funciones de activación tales como tanh (tangente hiperbólica), rectificador y maxout.

Cada nodo de procesamiento entrena una versión local de los parámetros globales del modelo en sus propios datos de manera asincrónica (multihilos) y aporta periódicamente al modelo global mediante el promedio de modelos en la red.

El operador inicia un clúster H2O local de un solo nodo para ejecutar el algoritmo. Aunque se utilice un único nodo, la ejecución se realiza de manera paralela. Por defecto, se utiliza el número de hilos recomendado para el sistema.

Dentro del algoritmo hay una serie de hiper parámetros que pueden ser ajustados. El primero es la función de activación, que utilizarán las neuronas en las capas ocultas. La elegida es la Rectifier, que elige un máximo de (0, x) donde x es el valor de entrada. Seguidamente, se establece un tamaño de las capas ocultas de la red. Se tienen dos capas con 50 neuronas cada una. Esto implica que cada capa intentará aprender diferentes representaciones de los datos de entrada, y cada una de las 50 neuronas en una capa puede aprender a activarse para diferentes patrones o características de los datos.

Además, se activa la opción de reproducible lo que implica que el sistema está configurado para asegurar que los resultados puedan ser reproducidos de manera exacta, pero a costa de la eficiencia en la ejecución y se hace uso de 50 epochs por lo que el modelo pasará por el conjunto completo de datos de entrenamiento 50 veces.

Finalmente, se desactiva el cálculo de la importancia de cada característica, activar esta característica implicaría que se evaluarán qué tan importantes son las diferentes variables de entrada para la precisión predictiva del modelo.

Una vez que el modelo ha sido entrenado, se aplica a los datos de prueba para hacer predicciones y se evalúan las predicciones del modelo con las respuestas reales de los datos de prueba usando métricas de rendimiento.

Modelo no supervisado LDA

Para identificar los temas latentes dentro de un conjunto de documentos, se optó por utilizar Latent Dirichlet Allocation (LDA) de manera no supervisada, prescindiendo así del etiquetado previo de los datos. El proceso de implementación del modelo se dividió en varias etapas, tal como se detalla en el Anexo 3 del documento adjunto. Se siguieron los pasos fundamentales que incluyen la carga de los datos provenientes de la colección de documentos proporcionada, seguida de la conversión del texto en una

estructura de documentos utilizando el operador “Data to Documents” de RapidMiner, lo que permitió preparar los datos para las etapas posteriores.

En lo que respecta al preprocesamiento del texto, se aplicaron diversas técnicas de preprocesamiento textual, como se menciona en la sección correspondiente del documento.

A continuación, se procedió a extraer los temas utilizando LDA. Los parámetros utilizados en el proceso de extracción de temas fueron los siguientes:

- Número de topics: Se exploraron diferentes valores para el número de temas, incluyendo 5 y 8. La elección del número de topics dependió del objetivo específico del análisis. Se esperaba que 5 topics proporcionaran un tema para cada etiqueta predefinida, mientras que 8 topics podrían capturar subtemas adicionales o matices dentro de los datos.

- Número de iteraciones: Se realizó un total de 1000 iteraciones durante el proceso de inferencia de temas. Este parámetro controla la cantidad de veces que el algoritmo recorre los datos para ajustar los parámetros del modelo.

- Top palabras por topic: Se seleccionaron las 5 palabras con mayor peso para cada topic, lo que proporcionó una representación sintética de los temas identificados.

- Uso de heurísticas alpha: Se utilizó la configuración predeterminada para el parámetro "use_alpha_heuristics", que ajusta automáticamente el parámetro alpha del modelo LDA para optimizar el proceso de inferencia de temas.

- Uso de heurísticas beta: Similarmente, se utilizó la configuración por defecto para el parámetro "use_beta_heuristics", que ajusta automáticamente el parámetro beta del modelo LDA para optimizar el proceso de inferencia de temas.

- Distribución inicial de probabilidades sobre temas (alpha): La distribución inicial de probabilidades sobre temas se estableció mediante la configuración por defecto del parámetro "alpha_sum".

- Distribución inicial de probabilidades sobre palabras en un tema (beta): La distribución inicial de probabilidades sobre las palabras en un tema se determinó mediante la configuración por defecto del parámetro "beta".

- Optimización de hiperparámetros: Se habilitó la optimización de hiperparámetros, permitiendo que el algoritmo ajuste automáticamente los parámetros del modelo para maximizar la coherencia de los temas extraídos.

- Uso de semilla aleatoria local: Se utilizó la semilla aleatoria local para garantizar la reproducibilidad de los resultados. Se configuró con la opción predeterminada "use_local_random_seed" y "local_random_seed".

Modelo no supervisado word2vec

Además del modelo no supervisado con LDA, se ha utilizado otro modelo de Word2vec, redes neuronales poco profundas de dos capas. Funciona de tal manera que asigna a cada palabra única un vector específico compuesto por una lista de números. Estos vectores se seleccionan de manera que la similitud entre ellos refleja la cercanía semántica entre las palabras que representan.

El proceso, representado en el Anexo 4, sigue una estructura similar al modelo no supervisado LDA mencionado anteriormente en cuanto a los primeros pasos. Respecto a los hiper parámetros del algoritmo se detallan a continuación:

- Minimal Vocab Frequency: Frecuencia mínima de palabra en el conjunto de datos para ser considerada en el entrenamiento del modelo. Cualquier palabra que aparezca menos de veces este valor será ignorada.
- Layer Size: Número de neuronas en la capa oculta del modelo Word2Vec, que a su vez determina la dimensión del vector de palabras (word embeddings).
- Window Size: Este parámetro especifica el tamaño de la "ventana" de contexto alrededor de la palabra objetivo durante el entrenamiento. Un tamaño de ventana de 5 significa que el modelo considera las 5 palabras antes y después de la palabra objetivo para predecir su representación vectorial.
- Use Negative Samples: Muestreo negativo, una técnica de optimización para reducir el número de actualizaciones de peso durante el entrenamiento. Por ejemplo, un número 5 indica que para cada palabra objetivo, se seleccionarán cinco "ejemplos negativos" (palabras no contexto) para actualizar los pesos durante el entrenamiento.
- Iterations: Cuántas veces el algoritmo pasará por el conjunto de datos durante el entrenamiento.
- Down Sampling Rate: Umbral para el muestreo descendente de palabras frecuentes. En este caso, el valor 1E-4 (o 0.0001) es un umbral comúnmente utilizado donde palabras con una frecuencia mayor que el umbral se muestrean menos.

Seguidamente, se hace un almacenamiento del modelo en el repositorio y de ahí se hace una extracción del vocabulario, que consiste en extraer los vectores de palabras

del modelo previamente entrenado y exportarlos a un conjunto de ejemplos. En este trabajo se hace exportando un conjunto aleatorio de palabras, pues si se hace de todo el vocabulario puede consumir una cantidad significativa de memoria.

Modelo no supervisado k-medias

En este trabajo se ha realizado un agrupamiento de k-medias, véase Anexo 5. Para ello, se cargan y pre procesan los datos al igual que en los modelos anteriores. Seguidamente, se realiza el clustering de k-medias que consiste en dividir un conjunto de n elementos en k grupos. La asignación de cada elemento a un grupo se hace en función de la proximidad al promedio (o centroide) más cercano de ese grupo. A continuación, se explican los parámetros del algoritmo:

- add cluster attribute: Esta opción agrega un atributo al conjunto de datos de salida que indica el clúster asignado a cada punto de datos.
- add as label: El atributo del clúster se agregará como una etiqueta.
- remove unlabeled: Se elimina del conjunto de datos todos los puntos que no tengan una etiqueta asignada.
- k: Especifica el número de clústeres que se deben encontrar. Se ha probado de manera similar a la versión LDA con dos parámetros distintos, uno con cinco clústeres y otro de ocho.
- max runs: Número máximo de veces que el algoritmo se ejecutará con diferentes inicializaciones centroides. Cada "run" comienza con una selección diferente de los puntos de partida para los centroides. Se ha elegido un valor de 10 ejecuciones.
- determine good start values: El algoritmo intentará determinar valores iniciales que probablemente resulten en una mejor convergencia del modelo.
- measure types: Determina la medida de distancia utilizada para calcular la similitud entre los puntos de datos y los centroides. Se ha seleccionado 'BregmanDivergences'. Esto especifica que se utilizarán las divergencias de Bregman como la medida de distancia durante el agrupamiento.
- divergence: Tipo de divergencia que se utilizará para calcular la distancia entre puntos y centroides. 'SquaredEuclideanDistance' se ha elegido como la divergencia específica, que usa la distancia euclidiana al cuadrado entre puntos para el cálculo de la pertenencia al clúster.

- max optimization steps: Define el número máximo de iteraciones que el algoritmo realizará para optimizar los centroides de los clústeres. Una vez alcanzado este límite, el algoritmo se detendrá aunque no haya convergido. Se limita a 100 pasos.

Tras la ejecución del k-medias, se realiza una copia independiente del clustering y se visualiza.

Herramientas

Para el desarrollo de las soluciones se decidió utilizar la herramienta de minería de texto RapidMiner, además de diversas extensiones de este software que apoyan el pre-procesamiento de texto, Text Processing, Operator Toolbox para la solución LDA y word2vec.

Resultados

Modelo supervisado

Para este modelo se ha obtenido una matriz de confusión, véase el Anexo 6, y dos métricas diferentes: una precisión del 90% y un coeficiente de Kappa de 0.874.

Desglosando la matriz de confusión:

- Negocios: De todas las predicciones de la clase Negocios, el 88.46% eran correctas, y el modelo identificó correctamente el 92% de todas las instancias reales de la clase Negocios.
- Entretenimiento: De todas las predicciones de la clase Entretenimiento, el 89.66% eran correctas, y el modelo identificó correctamente el 86.67% de todas las instancias reales de la clase Entretenimiento.
- Política: De todas las predicciones de la clase Política, el 83.72% eran correctas, y el modelo identificó correctamente el 90% de todas las instancias reales de la clase Política.
- Deportes: De todas las predicciones de la clase Deportes, el 97.78% eran correctas, y el modelo identificó correctamente el 88% de todas las instancias reales de la clase Deportes.
- Tecnología: De todas las predicciones de la clase Tecnología, el 90.24% eran correctas, y el modelo identificó correctamente el 92.50% de todas las instancias reales de la clase Tecnología.

La precisión global del modelo refleja la capacidad general del modelo para etiquetar correctamente una instancia independientemente de la clase. Sin embargo, esta medida no distingue entre las diferentes clases; por ejemplo, no indica si algunas clases son más difíciles de predecir que otras o si el modelo tiende a favor alguna clase sobre otra.

Se ha decidido utilizar además el coeficiente de Kappa, ya que tiene en cuenta el equilibrio de las clases y la posibilidad de que las predicciones correctas se deban al azar.

Ajusta la precisión del modelo teniendo en cuenta la precisión que se obtendría si las clasificaciones se hicieran al azar. También es útil al encontrarse con muchas más instancias de una clase en concreto que otras, es decir sigue mostrando un resultado representativo incluso con clases desbalanceadas.

Un kappa de 1 indicaría un acuerdo perfecto, mientras que un valor de 0 demuestra que el acuerdo es no mejor que el azar. Un kappa de 0.874, sugiere un acuerdo muy bueno entre las predicciones del modelo y las etiquetas reales, mostrando la eficacia del modelo.

El alto valor de kappa junto con la alta precisión general obtenida nos indica que el modelo es efectivo en la clasificación y no está sesgado simplemente hacia las clases más frecuentes.

Como se ve en la matriz de confusión, la clase con mejor tasa de “true positives” y “true negatives” sería “Deportes” con más de un 97% en precisión, pero con un 88% en la sensibilidad. Esto indica que hay un claro espacio de mejora a la hora de determinar algunos casos positivos, seguramente el modelo no esté logrando capturar todas las características relevantes de la clase o no esté está bien representada en los datos de entrenamiento.

Modelo no supervisado LDA

Este modelo muestra un resultado en forma de tabla compuesta por temas y con pesos asociados que representan la importancia o contribución de cada palabra dentro de su respectivo tema.

Como se observa en el Anexo 7, el resultado muestra cinco temas diferentes, y para cada tema, se presentan varias palabras clave junto con su peso. Estos pesos muestran la relevancia de la palabra dentro del tema, siendo un número más alto indicativo de mayor importancia.

Los diferentes tópicos obtenidos se detallan a continuación, basado en las palabras clave más pesadas:

- Tópico 0: Las palabras como "film", "award", "music", "star" y "show" sugieren que este tópico está relacionado con el entretenimiento.
- Tópico 1: Palabras como "company", "market", "firm", "bank" y "share" apuntan a un tema centrado en negocios.
- Tópico 2: "government", "people", "party", "election" y "labour" indican un tema político.
- Tópico 3: Palabras como "game", "play", "win", "England" y "player" muestran un tema centrado en los deportes.
- Tópico 4: Las palabras "game", "use", "technology", "mobile" sugieren un tema relacionado con la tecnología.

Se observa una clara diferenciación de temas entre los tópicos seleccionados, pudiendo ser identificado de manera sencilla el tema relacionado con el tópico.

Además, se ha realizado esta misma operación utilizando esta vez ocho tópicos diferentes y se ha obtenido otra tabla, véase Anexo 8. Similar a cómo se ha realizado previamente, se procede a analizar los distintos tópicos.

- Tópico 0: Temas económicos con palabras como "rate", "market", "price", “growth” y “economy”.
- Tópico 1: Entretenimiento y medios con palabras como "film", "award", "music", “star”, “show”.
- Tópico 2: Política y gobierno con palabras como "parti", "elect", “labour”, "govern" y “people”.
- Tópico 3: Negocios y corporaciones con palabras como "compani", "firm", "share", “deal” y “profit”.
- Tópico 4: Leyes con palabras como "law", "court", "case", “polic” y “rule”.
- Tópico 5: Tecnología o juegos, con palabras como "people", "game", "use", "technolog" y “mobile”
- Tópico 6: Deportes o actividades de ocio, con palabras como "game", "club", "play", “time” y “player”.
- Tópico 7: Actividad deportiva en un área geográfica específica, relacionada con Inglaterra y Gales, como se indica por "England", "Wales", y palabras relacionadas con juegos y deportes como "game", "play" y “win”.

En este segundo análisis se puede ver claramente como hay palabras repetidas que pueden aparecer en distintos topics, esto es debido a que al estar identificando subtemas dentro de lo que podría ser un tema general es común encontrar estas repeticiones.

En esta segunda versión se diferencian temas de manera más sutil, e incluso se llegan a diferenciar los deportes/actividades de ocio con el deporte local, haciendo una diferenciación clave al seleccionar la región específica.

Modelo no supervisado word2vec

Los resultados obtenidos de este modelo son a partir del operador de la extracción del vocabulario, que transforma los vectores obtenidos en un set de ejemplo como se observa en el Anexo 9.

Se presenta cada palabra como un vector de 10 dimensiones, donde cada dimensión representa una característica numérica que ha sido aprendida durante el entrenamiento. Las palabras con contextos similares tendrán vectores similares.

Por ejemplo, algunas palabras poseen valores similares en varias de las dimensiones, como "counterfeit" y "fcc" (posiblemente referido a la Federal Communications Commission). Esto indica que ambas palabras comparten un contexto, posiblemente uno relacionado con la legalidad.

Modelo no supervisado k-medias

En la versión utilizando cinco clústers distintos, véase Anexo 10, y por orden de peso de la palabra se obtuvo:

- Cluster 0: Este clúster parece estar relacionado con temas políticos, ya que contiene palabras como "labour", "elect" y "parti", que sugieren discusiones sobre elecciones y partidos políticos.
- Cluster 1: Aquí encontramos palabras relacionadas con la industria del entretenimiento, como "film", "star" y "role", lo que indica discusiones sobre películas y actores.
- Cluster 2: Este clúster parece estar relacionado con temas económicos, dado que contiene palabras como "bn", "growth" y "price", que sugieren discusiones sobre negocios y crecimiento económico.
- Cluster 3: Palabras como "music", "technolog" y "websit" sugieren discusiones sobre tecnología y música, lo que indica posibles debates sobre el impacto de la tecnología en la industria musical.
- Cluster 4: Este clúster parece estar relacionado con eventos deportivos, ya que contiene palabras como "match", "game" y "side", que sugieren discusiones sobre partidos y equipos deportivos.

En la versión con ocho clústers, véase Anexo 11, se obtuvo:

- Cluster 0: Similar al Cluster 0 de la versión anterior, está relacionado con temas políticos, con palabras como "elect", "labour" y "parti".
- Cluster 1: Este clúster parece estar relacionado con temas deportivos, ya que contiene palabras como "game", "player" y "releas", sugiriendo discusiones sobre jugadores y lanzamientos de juegos.
- Cluster 2: Similar al Cluster 2 de la versión anterior, está relacionado con temas económicos, con palabras como "growth", "bn" y "analyst".
- Cluster 3: Este clúster parece estar relacionado con la industria del entretenimiento, con palabras como "music", "show" y "perform", sugiriendo discusiones sobre espectáculos y actuaciones.
- Cluster 4: Similar al Cluster 4 de la versión anterior, parece estar relacionado con temas políticos y gubernamentales, con palabras como "govern", "case" y "minist".
- Cluster 5: Similar al Cluster 1 de la versión anterior, parece estar relacionado con la industria del entretenimiento, con palabras como "film", "star" y "director".
- Cluster 6: Este clúster parece estar relacionado con tecnología y uso de internet, con palabras como "technolog", "use" y "websit".
- Cluster 7: Este clúster parece estar relacionado con eventos deportivos específicos de una región geográfica, con palabras como "match", "england" y "side", sugiriendo discusiones sobre partidos en Inglaterra.

Éstos resultados también se pueden observar en forma de mapa de calor en los Anexos 12 y 13 para las versiones de cinco y ocho clusters respectivamente.

El modelo LDA sobresale por su capacidad para generar tópicos amplios y abstractos, ofreciendo una visión panorámica de los temas que existen en los datos. Basado en la distribución de palabras dentro de cada tópico, LDA proporciona una perspectiva global, lo que resulta especialmente útil para comprender tendencias generales en grandes conjuntos de datos.

En contraste, el enfoque de k-medias se inclina hacia la identificación de clústers más específicos y detallados. Este modelo agrupa documentos según la similitud de palabras, lo que conduce a una identificación más precisa de subtemas con menos solapamientos entre ellos.

En términos de interpretación y aplicabilidad, cada modelo presenta distintas visiones sobre la clasificación de temas. Mientras que LDA ofrece una visión más abstracta y general, útil para comprender las grandes tendencias, k-medias destaca en la identificación de subtemas específicos y detallados.

Conclusiones

Este estudio realizado sobre el procesamiento de lenguaje natural y la clasificación de documentos mediante diferentes modelos de aprendizaje automático revela importantes hallazgos sobre la eficacia y aplicabilidad de dichas técnicas en la minería de textos.

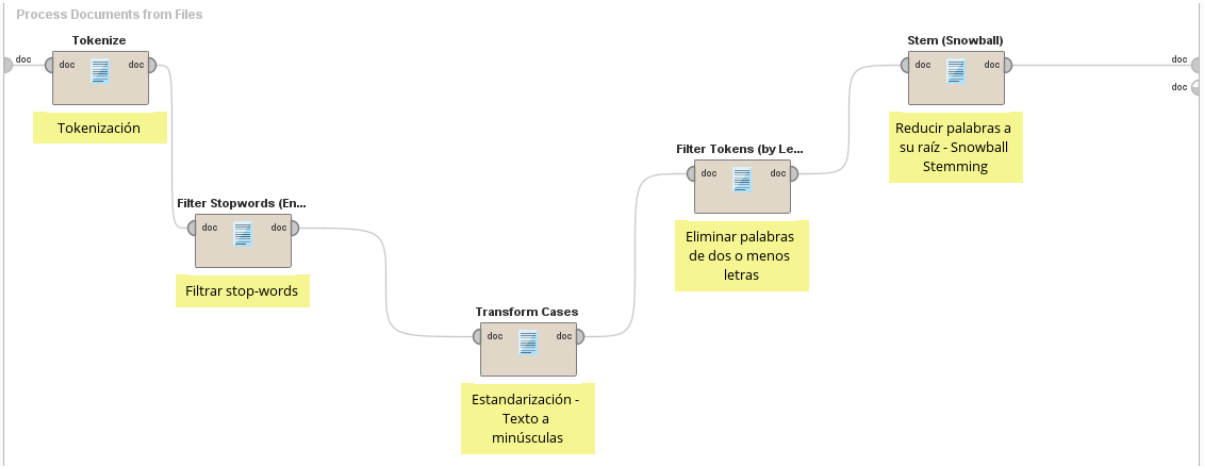
En primer lugar, se destaca la capacidad de los modelos supervisados para clasificar documentos con una precisión significativa, lo que demuestra su utilidad en escenarios donde se dispone de conjuntos de datos etiquetados. Esta precisión se ve respaldada por métricas como el coeficiente de Kappa, evidenciando un acuerdo sustancial entre las predicciones del modelo y las etiquetas reales. Técnicas como esta pueden aplicarse en la monitorización de la satisfacción del cliente y el análisis de sentimientos, lo cual permite a las empresas responder de manera más efectiva a las necesidades y preocupaciones de los clientes.

Por otro lado, los modelos no supervisados, como LDA, word2vec y k-medias, proporcionan una perspectiva más amplia al identificar temas y clústers en los documentos sin la necesidad de etiquetas predefinidas. El análisis de temas podría revelar tendencias emergentes en las preferencias de los consumidores o cambios en el sentimiento del mercado, permitiendo a las empresas adaptarse rápidamente a las nuevas demandas del mercado. Además, pueden facilitar la segmentación del mercado, permitiendo a las empresas personalizar sus estrategias de marketing y productos para satisfacer las necesidades de segmentos específicos del mercado.

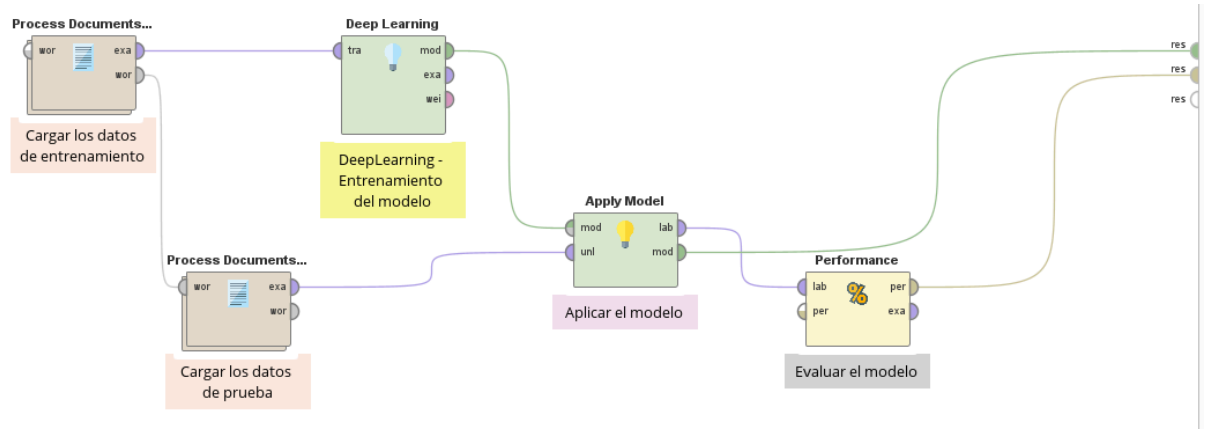
Este trabajo destaca el papel fundamental que desempeñan los modelos de aprendizaje automático en el análisis de texto y la minería de datos en las estrategias empresariales. Tanto los modelos supervisados como los no supervisados ofrecen herramientas para la clasificación y segmentación de documentos, permitiendo extraer conocimientos significativos de grandes volúmenes de datos textuales. Esto no solo mejora la capacidad de las empresas para tomar decisiones basadas en datos sino que también promueve la innovación al proporcionar nuevas perspectivas y comprensiones de los datos a su disposición.

Anexo

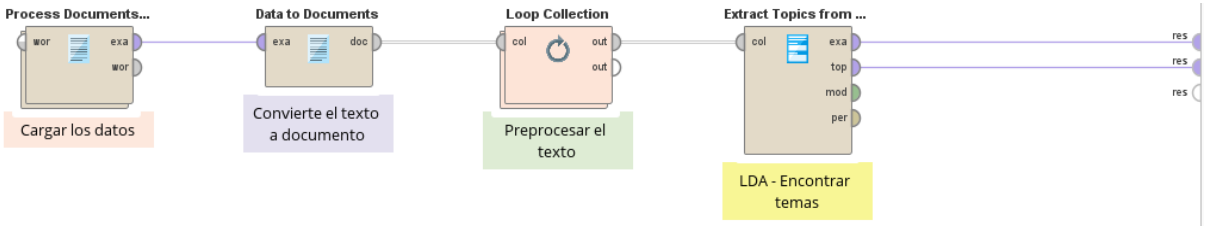
Anexo 1: Pre-procesado de datos



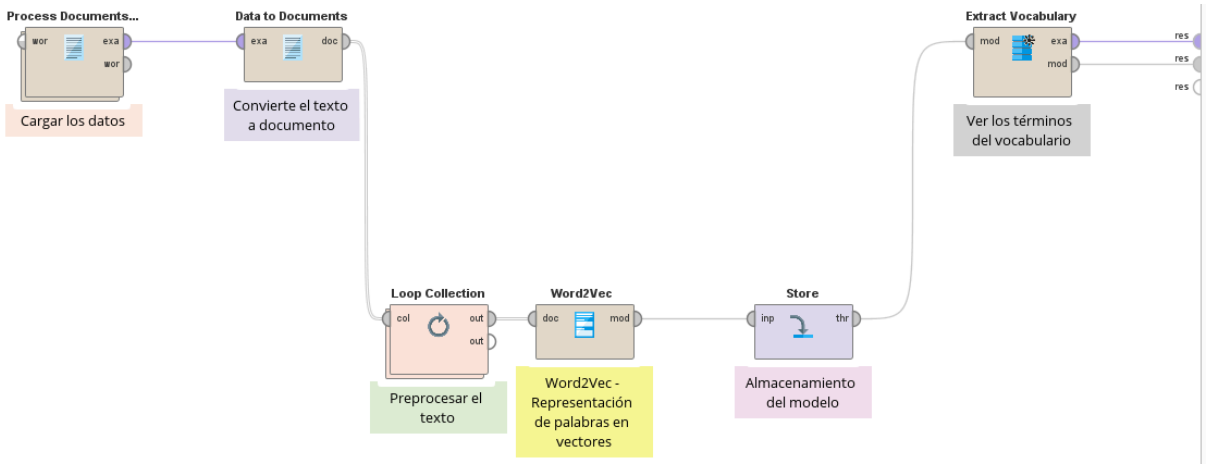
Anexo 2: Modelo supervisado



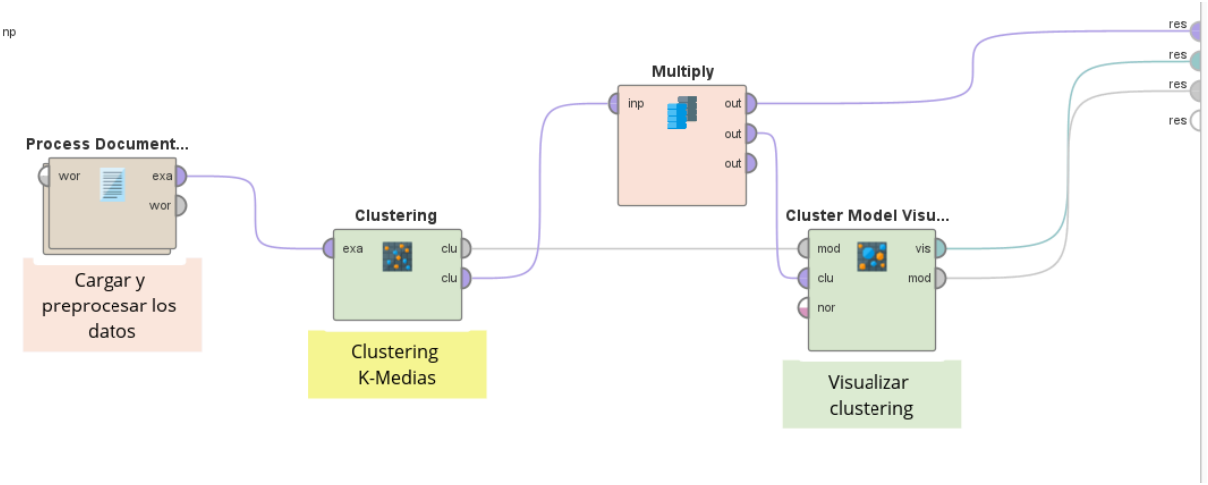
Anexo 3: Modelo no supervisado LDA



Anexo 4: Modelo no supervisado word2vec



Anexo 5: Modelo no supervisado k-medias



Anexo 6: Matriz de confusión del modelo supervisado

	true Negocios	true Entretenimiento	true Política	true Deportes	true Tecnología	class precision
pred. Negocios	46	0	3	2	1	88.46%
pred. Entretenimiento	0	26	0	2	1	89.66%
pred. Política	3	2	36	1	1	83.72%
pred. Deportes	0	1	0	44	0	97.78%
pred. Tecnología	1	1	1	1	37	90.24%
class recall	92.00%	86.67%	90.00%	88.00%	92.50%	

Anexo 7: Resultados modelo no supervisado LDA con 5 topics

Row No.	topicId	word	weight
1	0	govern	876
2	0	peopl	704
3	0	parti	688
4	0	elect	680
5	0	labour	675
6	1	peopl	897
7	1	game	816
8	1	use	720
9	1	technolog	643
10	1	mobil	582
11	2	game	692
12	2	play	632
13	2	win	544
14	2	player	491
15	2	england	486
16	3	compani	641
17	3	market	571
18	3	firm	525
19	3	bank	491
20	3	share	429
21	4	film	990
22	4	award	582
23	4	music	523
24	4	star	450
25	4	show	442

ExampleSet (25 examples,0 special attributes,3 regular attributes)

Anexo 8: Resultados modelo no supervisado LDA con 8 topics

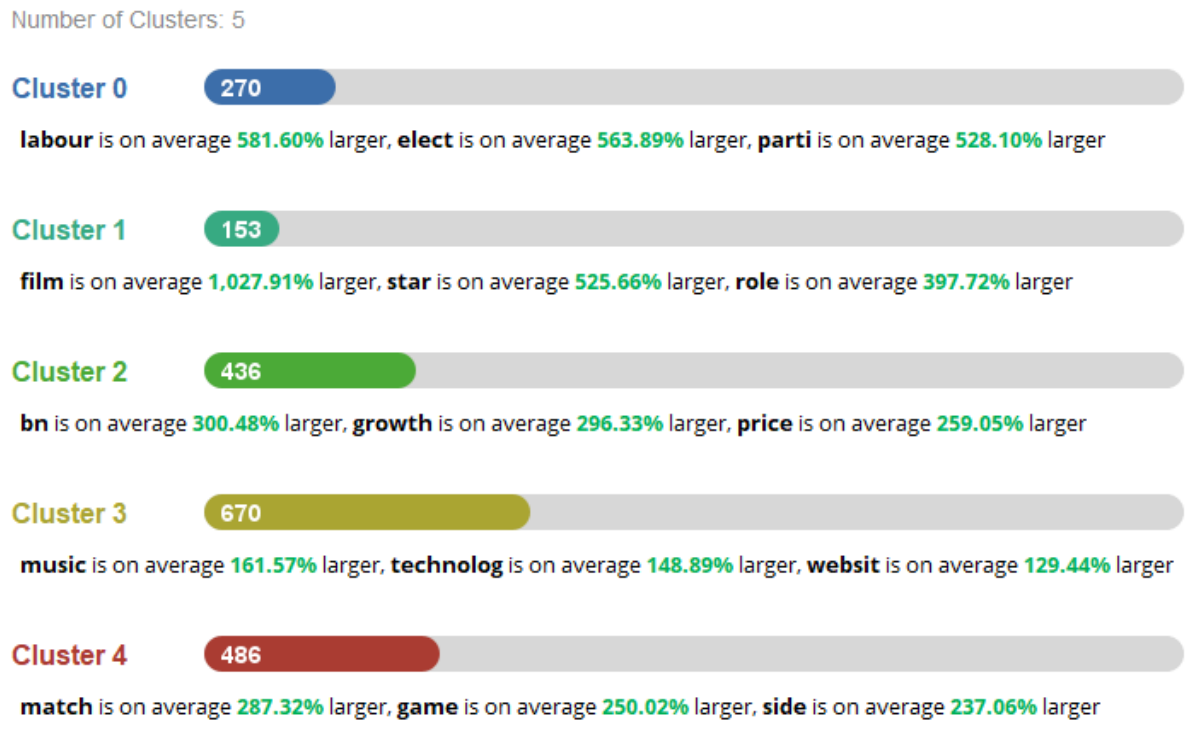
Row No.	topicId	word	weight
1	0	rate	379
2	0	market	375
3	0	price	374
4	0	growth	373
5	0	economi	366
6	1	film	989
7	1	award	581
8	1	music	539
9	1	star	454
10	1	show	423
11	2	parti	685
12	2	elect	676
13	2	labour	671
14	2	govern	662
15	2	peopl	585
16	3	compani	572
17	3	firm	475
18	3	share	350
19	3	deal	268
20	3	profit	206
21	4	law	385
22	4	court	278
23	4	case	260
24	4	polic	247
25	4	rule	241
26	5	peopl	889
27	5	game	792
28	5	use	710
29	5	technolog	638
30	5	mobil	578
31	6	game	406
32	6	club	396
33	6	play	332
34	6	time	309
35	6	player	252
36	7	england	407
37	7	game	331
38	7	play	328
39	7	wale	300
40	7	win	289

ExampleSet (40 examples,0 special attributes,3 regular attributes)

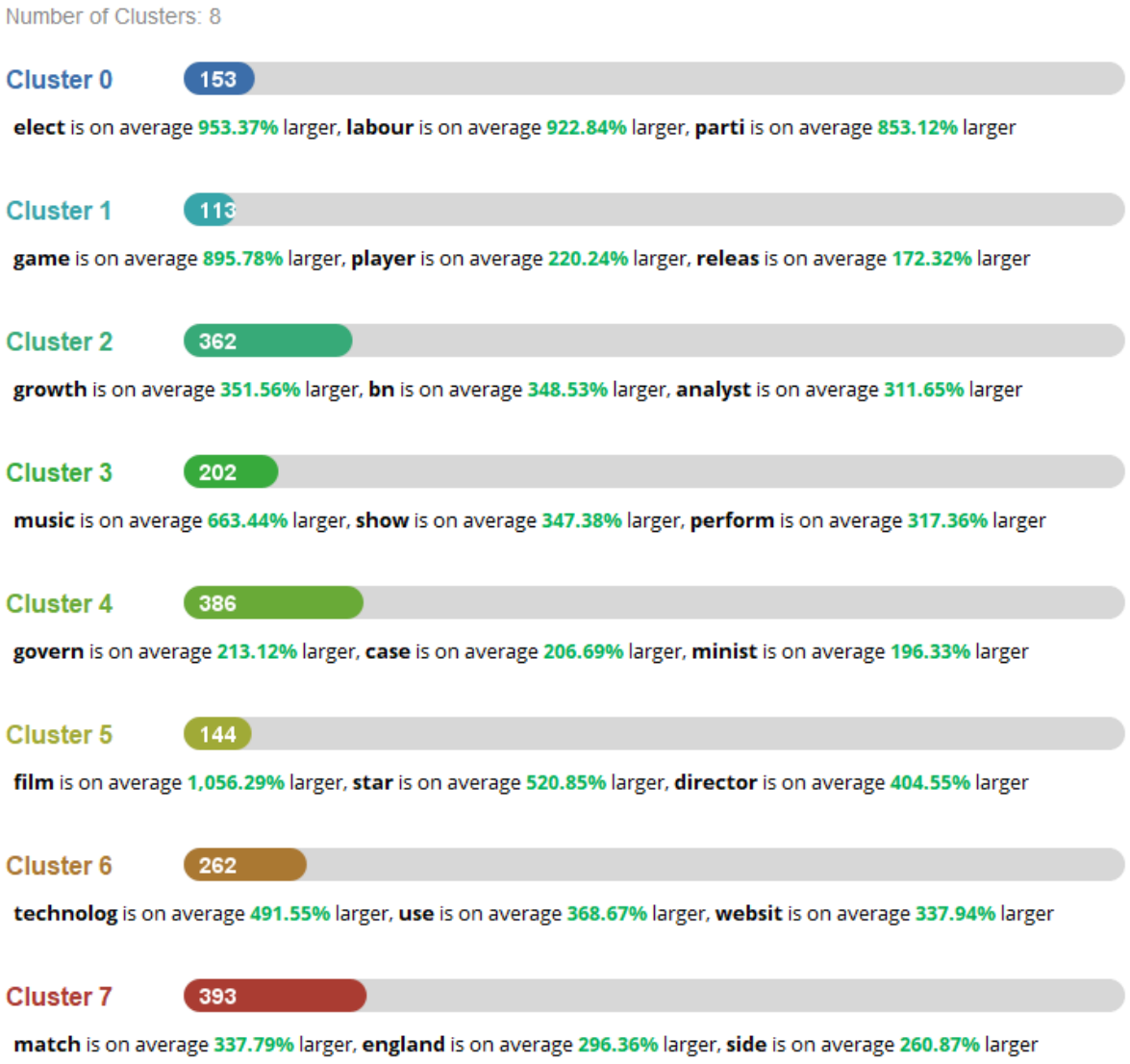
Anexo 9: Fragmento del vocabulario extraído en word2vec

Row No.	word	dimension_0	dimension_1	dimension_2	dimension_3	dimension_4	dimension_5	dimension_6	dimension_7	dimension_8	dimension_9
1	counterfeit	-0.610	0.083	-0.069	0.053	-0.021	-0.209	0.092	0.151	-0.042	0.732
2	fcc	-0.664	-0.000	-0.124	0.254	0.027	-0.220	-0.070	0.091	0.010	0.646
3	headlin	-0.687	0.041	-0.130	0.290	0.033	-0.265	-0.068	0.059	0.143	0.570
4	hard	-0.674	0.008	-0.108	0.225	0.104	-0.272	-0.089	0.115	0.051	0.613
5	reserv	-0.272	-0.012	-0.374	-0.253	-0.072	-0.457	-0.274	0.019	-0.322	0.573
6	trip	-0.554	0.002	-0.220	0.568	0.174	0.082	-0.133	-0.251	0.376	0.250
7	wembley	-0.552	0.047	-0.355	0.212	-0.007	-0.158	-0.100	-0.022	0.039	0.697
8	prof	-0.652	0.038	-0.127	0.170	-0.003	-0.220	-0.084	0.146	-0.014	0.672
9	fourth	-0.304	0.111	-0.493	0.050	-0.015	-0.022	-0.279	-0.469	0.283	0.521
10	minist	-0.365	-0.336	-0.041	0.347	0.378	-0.463	-0.200	0.394	-0.280	0.022
11	lo	-0.712	-0.009	-0.146	0.369	0.074	-0.114	-0.122	-0.088	0.162	0.518
12	prioriti	-0.663	-0.037	-0.146	0.308	0.115	-0.252	-0.198	0.151	-0.072	0.547
13	corp	-0.431	0.057	-0.202	-0.110	-0.136	-0.280	-0.000	0.068	-0.153	0.796
14	pirat	-0.465	0.106	0.157	-0.305	-0.179	-0.116	0.161	0.217	-0.071	0.729
15	ac	-0.514	0.015	-0.233	0.544	0.134	0.157	-0.134	-0.234	0.380	0.355
16	wing	-0.559	-0.036	-0.254	0.586	0.176	0.113	0.029	-0.296	0.329	0.196
17	delet	-0.676	-0.067	-0.103	0.224	-0.004	-0.192	-0.035	0.146	0.032	0.646
18	welfar	-0.610	-0.137	-0.090	0.146	0.115	-0.364	-0.159	0.328	-0.275	0.475
19	bizarr	-0.635	0.037	-0.202	0.362	0.056	-0.078	-0.118	-0.078	0.184	0.600
20	interest	-0.354	-0.025	-0.345	-0.315	-0.047	-0.493	-0.207	0.068	-0.370	0.476
21	reynold	-0.625	-0.028	-0.136	0.267	-0.027	-0.302	-0.091	0.093	0.039	0.639
22	shown	-0.703	0.192	0.062	0.414	0.004	-0.137	-0.165	0.134	0.320	0.357
23	cathol	-0.722	-0.050	-0.115	0.425	0.068	-0.156	-0.126	0.093	0.098	0.468
24	anymor	-0.648	0.033	-0.202	0.276	0.053	-0.173	-0.115	0.041	0.081	0.639
25	ireland	-0.425	-0.047	-0.300	0.482	0.111	0.257	-0.127	-0.461	0.421	0.104

Anexo 10: 5 clústers de k-medias



Anexo 11: 8 clústers de k-medias



Anexo 12: Importancia palabras por cluster K-Medias, 5 clusters

Anexo 13: Importancia palabras por cluster K-Medias, 8 clusters