Integrantes:

Santiago Sinisterra – 202022177

Lina Ojeda - 202112324

Ana Sofía Padilla Daza - 202021748

**Proyecto 1, Etapa 2 – Analítica de textos**

Inteligencia de Negocios

Tabla de contenido

[Proceso de automatización del proceso de preparación de datos 1](#_Toc164246242)

[Construcción y persistencia del modelo 1](#_Toc164246243)

[Acceso por medio de API 1](#_Toc164246244)

[Desarrollo de la aplicación y justificación 2](#_Toc164246245)

[Rol de la organización que va a utilizar la aplicación y proceso de negocio que va a apoyar 2](#_Toc164246246)

[Mejoras en base a los comentarios del grupo de estadística. 2](#_Toc164246247)

[Trabajo en equipo 2](#_Toc164246248)

[Referencias 4](#_Toc164246249)

# Proceso de automatización del proceso de preparación de datos

Para realizar la automatización del proceso de preparación de datos este se dividió en tres partes. La primera fue identificar los datos externos al programa que se necesitaban, siendo estos el *bag of words* creado en la etapa anterior y las *stopwords* usadas para filtrar los textos. Para obtener el *bag of words* este se descargó como un archivo csv y se agregó a los archivos del programa. Y para las stopwords se utilizó la librería con este nombre de nltk, más adiciones que consideramos relevantes. Para la segunda parte se creó una función que limpiara, filtrara y tokenizara los textos ingresados, siendo esta removerTodos(). En esta función se tokeniza el texto, se eliminan las stopwords y se lematizan las palabras del texto; esta función retorna una lista con las palabras relevantes. La función se encuentra dentro de la clase classification.py, al igual que la función encargada de la siguiente parte: la función vectorización(). Esta se encarga de preparar el texto ingresado para que pueda cumplir los requisitos que pide el modelo. Para esto usa la información del bag of words importado y retorna el dataframe listo para ser usado por el algoritmo. Con estos tres pasos se crea ya la función “classify\_single\_text”, la cual recibe un texto, si no es un dataframe lo cambia a este formato, luego le aplica los filtros, la limpieza y la vectorización para luego cargar el modelo y predecir la clase del texto ingresado, la cual retorna.

## 

## Construcción del modelo

Para la construcción y persistencia del modelo utilizamos el modelo y usamos la librería *joblib* para descargar el pipeline creado a partir del modelo en la etapa previa. Escogimos el modelo de Random Forest. Después de entrenarlo y probarlo con varios datos se guardó el modelo en un pipeline que se descargó con el nombre de modelo.joblib. Ya con eso se utilizó la función de load de esta librería para clasificar los textos después de pasarlos por el proceso de preparación de datos ya automatizado.

# Acceso por medio de API

Para poder acceder al modelo creamos un back-end con las pautas dadas en la práctica de “Despliegue de modelos de aprendizaje automático”. Con base en esto

Con esto se logró conectar la construcción del modelo y la aplicación creada.

# Desarrollo de la aplicación y justificación

El administrador dentro de una organización desempeña un papel fundamental en la supervisión y la toma de decisiones estratégicas. Es responsable de coordinar y gestionar diferentes aspectos del negocio, incluyendo la atención al cliente, la calidad del servicio y la reputación de la marca. Para llevar a cabo estas responsabilidades de manera efectiva, el administrador necesita acceder a información relevante y oportuna que le permita evaluar el desempeño del negocio y tomar decisiones informadas.

La aplicación que hemos desarrollado se alinea perfectamente con las necesidades del administrador al proporcionar una herramienta integral para la gestión de comentarios y reseñas.

# Rol de la organización que va a utilizar la aplicación y proceso de negocio que va a apoyar

Para el rol hemos considerado que la persona para utilizar la aplicación es un administrador en la organización, la aplicación que hemos desarrollado desempeña un papel crucial en la optimización del proceso de gestión de comentarios y reseñas. La aplicación actúa como un puente directo entre el usuario y el proceso de negocio, al conectarse con una API que utiliza un pipeline de análisis de datos y modelos de predicción. Esta conexión permite al administrador recibir de manera rápida y precisa la clasificación de las reseñas, generada por algoritmos entrenados, facilitando así la toma de decisiones informadas. La existencia de esta aplicación es de suma importancia para el rol de administrador, ya que le brinda una herramienta efectiva para evaluar y gestionar la satisfacción del cliente, identificar tendencias y áreas de mejora, y tomar medidas proactivas para mantener la reputación y el éxito del negocio. Con esta aplicación, el administrador puede agilizar el proceso de revisión de comentarios, aumentar la eficiencia operativa y mejorar la experiencia del cliente de manera significativa.

# Mejoras en base a los comentarios del grupo de estadística.

Después de mostrarles la aplicación obtuvimos una sugerencia por parte del grupo de estadística la cual consiste en que la clasificación podría ser explicada nos comentario que para el usuario obtener solo un número no sería conveniente. Por esta razón se agregó una tabla con los posibles valores a a obtener y explico cómo se entiende ese número de acuerdo con nuestro modelo implementado.

# Trabajo en equipo

Integrate 1:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Integrate | Rol | Algoritmos trabajados | Retos enfrentados | Solución a los retos |
| Santiago Sinisterra | Líder de datos | Arboles de decisión y Random Forest | Manejo del tiempo de las ejecuciones, saber que filtros eran los adecuado para usar. | Para el manejo del tiempo de las educaciones no se pudieron realizar muchas pruebas y nos conformamos con las realizadas y para la decisión sobre los filtros adecuados se realizó una evaluación de los datos con cada filtro y su eficiencia. |

Integrante 2:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Integrate | Rol | Algoritmos trabajados | Retos enfrentados | Solución a los retos |
| Lina Ojeda | Lider de negocio | Logistic Regression | Entendimiento de cómo funcionaba el algoritmo, saber que variables aceptaba y cuales no, y manejo de tiempo. | Leer documentación acerca de logistic regression, revisar junto con el profesor el error y encontrar una solución pertinente. Aunque no se pudieron realizar las pruebas esperadas por el tiempo de ejecución que tomaba se decidió trabajar con lo que teníamos. |

Integrante 3:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Integrate | Rol | Algoritmo trabajado | Retos enfrentados | Solución a los retos |
| Ana Sofia Padilla Daza | Lider de proyecto y Líder de analítica | Naive Bayes | El manejo del tiempo, el entendimiento completo del problema. | Para poder entender mejor el problema y los pasos a seguir para desarrollar el proyecto pedí ayuda tanto a monitores y profesores como a mis compañeros de equipo. Para optimizar el manejo del tiempo y la carga de trabajo realicé una planeación de lo que iba a hacer cada día y utilicé la técnica Pomodoro para poder lograr los objetivos diarios. |

Repartición de puntos entre los integrantes:

|  |  |  |
| --- | --- | --- |
| Integrante | Puntos sobre 100 | Horas dedicadas al proyecto en total |
| Santiago Sinisterra | 34 | 24 horas |
| Lina Ojeda | 33 | 20 Horas |
| Ana Sofía Padilla Daza | 33 | 20 horas |

Reuniones:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Numero de la reunión | Propósito | Fecha | Duración | Asistentes |
| 1 | Entender el problema. | 25/03/2024 | 2 horas | Lina Ojeda  Santiago Sinisterra  Ana Sofia |
| 2 | Realizar preprocesamiento de los datos | 26/03/2024 | 8 horas | Lina Ojeda  Santiago Sinisterra  Ana Sofia |
| 3 | Realizar modelación de los datos | 31/03/2024 | 4 horas | Lina Ojeda  Santiago Sinisterra  Ana Sofia |
| 4 | Finalización del documento y conclusión de detalles | 06/04/2024 | 2 horas | Lina Ojeda  Santiago Sinisterra  Ana Sofia |

# Referencias

1. Espinosa-Zúñiga, Javier Jesús. (2020). Aplicación de algoritmos Random Forest y XGBoost en una base de solicitudes de tarjetas de crédito. Ingeniería, investigación y tecnología, 21(3), 00002. Epub 02 de diciembre de 2020.https://doi.org/10.22201/fi.25940732e.2020.21.3.022
2. Economía, R. (2024, 19 enero). Hoteles tuvieron un 2023 de caídas en ocupación e ingresos: ¿qué dicen las cifras? ELESPECTADOR.COM. <https://www.elespectador.com/economia/hoteles-tuvieron-un-2023-de-caidas-en-ocupacion-e-ingresos-que-dicen-las-cifras-noticias-colombia/>
3. ¿Qué es un árbol de decisión? | IBM. (s. f.). <https://www.ibm.com/es-es/topics/decision-trees>
4. ¿Qué es la regresión logística? | IBM. (s. f.). <https://www.ibm.com/mx-es/topics/logistic-regression>
5. Raschka, Sebastian, and Vahid Mirjalili. Python Machine Learning: Machine Learning and Deep Learning with Python, Scikit-Learn, and TensorFlow 2. 3rd ed. Birmingham: Packt Publishing, Limited, 2019. Print.