Proyecto de analitica de textos - etapa #2

Presentado para la empresa Turismo de los Alpes

Domingo 7 de abril del año 2024

El Ministerio de Comercio, Industria y Turismo de Colombia, la Asociación Hotelera y Turística de Colombia – COTELCO, cadenas hoteleras de la talla de Hilton, Hoteles Estelar, Holiday Inn y hoteles pequeños ubicados en diferentes municipios de Colombia están interesados en analizar las características de sitios turísticos que los hacen atractivos para turistas locales o de otros países, ya sea para ir a conocerlos o recomendarlos. De igual manera, quieren comparar las características de dichos sitios, con aquellos que han obtenido bajas recomendaciones y que están afectando el número de turistas que llegan a ellos. Adicionalmente, quieren tener un mecanismo para determinar la calificación que tendrá un sitio por parte de los turistas y así, por ejemplo, aplicar estrategias para identificar oportunidades de mejora que permitan aumentar la popularidad de los sitios y fomentar el turismo.

Durante la primera etapa de desarrollo de este proyecto se hizo la entrega de 3 modelos construidos por nuestros desarrolladores que acompañarán a COTELCO en la predicción de clasificación de reviews, y que nos permitirá establecer, con buenas métricas, que podría hacer que un destino turístico pueda obtener una buena calificación, todo ello basado en las reviews dadas por los usuarios.

Es por ello que desde el equipo de ingeniería de datos, de software y ciencia de datos hemos desarrollado una solución integral que permite a COTELCO obtener feedback real de una review por medio de una calificación de 1 a 5 estrellas para predecir qué tan bueno podría ser el sitio turístico basado en esa review ingresada por el usuario.

a. **(20%) Proceso de automatización del proceso de preparación de datos, construcción de modelo, persistencia del modelo y acceso por medio de API**

En lo que respecta a la fase de entendimiento, preparación de datos, creación de modelo y evaluación de este mismo, se recomienda acceder al siguiente link que lo redireccionará al repositorio de GitHub donde se encuentra de manera detallada el proceso realizado para cada una de estas etapas.

En general, para el entendimiento de los datos se compartieron datos de 7875 reseñas de turistas de los cuales tenemos una única columna object "Review" en el que se encuentra toda la información de la reseña escrita. Por otro lado, hay una única columna numérica "Class" que guarda el nivel de satisfacción del turista

con base en la reseña. Luego nos dimos cuenta de que los datos suministrados inicialmente para el entreno de la inteligencia artificial no poseen problemas con el tipo de dato al que debería corresponder, siendo este integer, ya que no vimos presencia de algún dato que no esté en el conjunto {1,2,3,4,5}. Adicionalmente en lo que respecta a la calidad de datos se llegaron a las siguientes conclusiones después de realizar un análisis:

- Todos los datos están completos en términos de “datos vacíos”, es decir, no existen datos tipo null en el conjunto dado para el entrenamiento.

- No existen valores duplicados en los datos, es decir, no hay reseñas duplicadas.

- No se observan inconsistencias frente a los valores y lo que representan en la vida real, todas las reviews van de acuerdo a lo que se puede considerar una review en la vida real y no se observan anomalías que hagan que estas sean extremadamente insignificantes para el modelo.

Para la preparación de los datos de forma general lo que se hizo fue una limpieza de datos, tokenización y vectorización, en específico se hicieron los siguientes cambios:

- Eliminar signos de puntuación, caracteres especiales, tildes y convertir todo a minúsculas.

- Elimina las palabras vacías. A estas se les conocen como stop-words, y son palabras que se caracterizan por no poseer una significación semántica para una frase, por lo que generan ruido y no se considera necesario incluirlas en el modelo. Para llevar a cabo esta eliminación, se hizo uso de una base de datos de la liberia de lenguaje natural nltk.

- Lematizar las palabras. Esto hace referencia a tratar a las palabras de cada review en su forma basal, es decir, de una palabra pueden existir diferentes combinaciones que se pueden realizar, las cuales surgen principalmente por el uso de géneros en el lenguaje u otros elementos, y palabras como “correr” y “corremos” en general deberían de tener una significación equivalente para una máquina, es por ello que en ese caso la palabra basal podría ser “corr” y así evitamos que la combinación “corremos”, la cual puede no ser tan común en los datos, pierda relevancia y también haga que la propia palabra “correr” la pierda, de manera que conservamos la relevancia de ambas en una sola.

- Convertir números en su representación textual en español en una reseña.

- Vectorizar las palabras de las review. Esto implica la transformación de palabras en su forma textual a una representación numérica que una máquina pueda entender, como convertirlas a 1s y 0s.

Para la construcción y persistencia del modelo, se escoge el mejor modelo encontrado en la etapa anterior, es decir, se escoge el modelo entrenado por Regresión Logística Multinomial que fue construido utilizando las librerías de python de sklearn. Una vez obtenido el modelo, se procede a persistir mediante el uso de un Pipeline, igualmente de la librería de sklearn, en el que se busca serializar el proceso de vectorización de la reseña y clasificación de la reseña. Una vez conseguido el Pipeline se persiste mediante un archivo ".joblib", el cual puede ser usado para la aplicación del modelo en la solución web desarrollada.

Para el desarrollo de la API de la aplicación, se decide usar el framework de FastAPI. En este framework se configura para que reciba dos endpoints "/predict" y "/predicts", estos endpoints cargan el modelo ".joblib" mencionado anteriormente. Como el Pipeline no serializa la preparación de la nueva reseña si no hasta la vectorización, entonces los pasos de preprocesamiento anteriores se deben hacer mediante el uso de funciones configuras en el proyecto sin el uso del Pipeline, después de preparadas las reseñas, están si se pasan al Pipeline para que pueda clasificarlas. Estas clasificaciones dadas son devueltas como respuestas a esas peticiones a la API. Por términos de facilidad se decide simplemente probar el API de forma local, accediendo a la dirección "http://127.0.0.1:8000/<endpoint>".

b. (40%) Desarrollo de la aplicación

Tabla de actores:

Rol dentro de la empresa	Tipo de actor	Beneficio	Riesgo
Turistas	Cliente	Se benefician al recibir recomendaciones de viaje más personalizadas basadas en las predicciones del modelo.	Existe el riesgo de que las predicciones del modelo no se alineen con sus preferencias personales, lo que podría llevar a una experiencia de viaje insatisfactoria.
Inversores	Financiador	Pueden ver un retorno de la inversión si el modelo mejora la satisfacción del cliente y aumenta las ventas.	El riesgo es que si el modelo no funciona como se esperaba, puede no ver un retorno de su inversión.

Hoteles, restaurantes, operadores turísticos	Proveedor	Se benefician al recibir más clientes de la empresa de turismo debido a las recomendaciones personalizadas.	El riesgo es que si las predicciones del modelo son inexactas, pueden recibir críticas negativas de los clientes insatisfechos.
Comunidad local	Beneficiados	Se benefician del aumento del turismo y del gasto turístico.	Sin embargo, el aumento del turismo también puede llevar a problemas como el sobre turismo y el desgaste de los recursos locales.
Gerente de experiencia del cliente de COTELCO y cadenas hoteleras	Usuario - Cliente	Se beneficia al poseer una herramienta de inteligencia artificial que le puede dar un análisis rápido respecto a la calificación de un sitio turístico y así este puede hacer un mejor análisis de qué salió mal y darle importancia a las reviews más negativas o más positivas.	Es probable que en una clasificación se presente sesgo por parte de de la inteligencia artificial, dando lugar a un falso positivo (una review que es clasificadamente como altamente importante negativamente o positivamente) lo cual provoque que se pierdan reviews que podrían tener insight importante para el Gerente.

El usuario objetivo de la aplicación sería para un Gerente de Experiencia del Cliente de COTELCO y de las principales cadenas hoteleras, y el proceso de negocio que soporta es la automatización de calificación de reseñas, la importancia que tiene el usuario para la aplicación es que este puede ingresar diferentes reseñas de clientes y que la aplicación le de una calificación basado en el modelo y verse beneficiado para estudiar la satisfacción de los clientes en los diferentes hoteles.

Para el diseño de la aplicación se tomaron diferentes decisiones, para la aplicación web se decidió manejar un backend el cual se encarga de cargar el modelo ya entrenado además de recibir los textos de las reseñas, predecir la clasificación a partir del modelo, para esta parte se eligió usar FastAPI ya que consideramos que es una herramientas rápida y fácil de usar. Por otro parte se decidió manejar un frontend el cual se encarga de recibir las reseñas del usuario, enviarselas al back para que prediga la clasificación y mostrarsela al usuario de una forma simple e ilustrativa, el framework usado para el frontend fue React ya que tenemos experiencia con ella. En términos de funcionalidad de la aplicación se decidió tener dos procesos principales para la clasificación de reseñas, la primera funcionalidad se refiere a consultar el nivel de satisfacción de una única reseña que se debe escribir manualmente, para la segunda funcionalidad se refiere a pasar un archivo .csv que contengan grandes cantidades de reseñas para su clasificación, este archivo se pasa a la aplicación y posteriormente se procesa y muestra la clasificación en formato de número de estrellas.

Por otro lado, se revisó la tabla de actores con los estudiantes de estadística y se concluyó que la tabla de actores cumple con lo esperado por lo que no se hicieron cambios.

c. (18%) Interpretación de resultados obtenidos

Se logró un desarrollo correcto y bueno de una solución web que permite a COTELCO obtener clasificaciones de reviews por medio de un modelo entrenado por el equipo de científicos e ingenieros de datos, el cual se basa en el uso del algoritmo de regresión logística multinomial por sus buenas métricas de f1 score de 0.48, precision y recall que nos permiten obtener predicciones de buena calidad, aunque no perfectas. Adicionalmente, se propone que por medio de la implementación de un large language model (LLM) basado en lenguaje natural se puede lograr un sistema, que dada una review y datos acerca la estadía de los autores de este y de eventos ocurridos importantes, cree recomendaciones y análisis de ultimo nivel respecto a la justificación de hechos mencionados en la review, lo cual puede facilitar mucho más el trabajo de análisis de niveles satisfactorios e insatisfactorios en visitas a diversos sitios turísticos.

El Gerente de Experiencia del Cliente de COTELCO puede tomar dos acciones importantes al recibir información de la clasificación de las reseñas de hoteles y sitios turísticos:

Identificar áreas de mejora: Se requiere analizar las reseñas para identificar patrones de insatisfacción o problemas comunes en los servicios o instalaciones de los hoteles y sitios turísticos. Una de estas formas es entender cuáles palabras tienen mayor peso en la clasificación de las reseñas. Esto ayuda a determinar en qué áreas se deben hacer mejoras para incrementar la satisfacción de los clientes.

Implementar estrategias de reconocimiento: Las reseñas también pueden destacar aspectos positivos de los hoteles y sitios turísticos. El gerente puede aprovechar esta información para establecer estrategias de reconocimiento para mantener la calidad del servicio en los puntos fuertes identificados.

Adicionalmente, nosotros desde el equipo de análisis e ingeniería de datos y software hemos identificado un patrón de conjunto de palabras que suelen seguir la tendencia de dar review con malas calificaciones y buenas calificaciones, las cuales son las siguientes:

Las 30 palabras más positivas: ['bien' 'gente' 'subir' 'gran' 'bueno' 'buena' 'historia' 'agradable' 'viejo' 'desayuno' 'precio' 'siquiera' 'mojitos' 'vendedores' 'dentro' 'demasiado' 'control' 'salon' 'mojito' 'ser' 'habitación' 'piso' 'caro' 'dos' 'guia' 'bastante' 'visita' 'interesante' 'igual' 'pareció']

Las 30 palabras más negativas: ['peor' 'horrible' 'asco' 'pésima' 'cover' 'pésimo' 'ayer' 'recorrido' 'chico' 'jamás' 'robo' 'rota' 'nunca' 'cubano' 'nardo' 'persona' 'sucias' 'vuelta' 'tardó' 'noma' 'llegada' 'sabado' 'respuesta' 'cero' 'sucia' 'asqueroso' 'quemas' 'lamentable' 'terminar' 'seguridad']

Finalmente, con respecto a la usabilidad y accesibilidad de la aplicación el grupo de estadística valido su correcto funcionamiento e intuitiva y fácil interacción con las funcionalidades y diseño de la interfaz gráfica.

d. (10%) Trabajo en equipo

Integrante	Rol	Tareas realizadas	Horas dedicadas	Retos	Puntos de mejora	Puntaje
Jefferson Hernandez 202120242	Líder del proyecto Ingeniero de software responsable de desarrollar la aplicación final	**Algoritmo trabajado: Regresión logística multinomial** Entendimiento de los datos. Evaluación y análisis de los resultados. Creación del modelo de regresión logística multinomial. Desarrollo de API por medio de Fast Api. Desarrollo de archivos de python que permiten la automatización de limpieza e implementación de ellos en el API. Implementación de múltiples componentes de React en la app web.	12 hrs	Creación de API: para crear un API existen diferentes maneras, sin embargo, debemos de escoger una que fuera sencilla y que nos permitiera fácilmente aplicar metodología AGILE, es por ello que finalmente se decidió usar Fast Api después de analizar la competencia.	Como mejora hubiera sido mejor estudiar más algoritmos de clasificación para tener un ambiente más amplio y buscar mejores modelos. También me gustaría conocer acerca de casos de uso específicos para el modelamiento de API y toda la metodología que se debe llevar a cabo idealmente.	**33%**
Samuel	Ingeniero de	**Algoritmo trabajado: Árboles de**	12 hrs	Desarrollo de interfaz de la	Para mejorar a futuro	**33%**

Goncalves 202122595	datos Ingeniero de software responsable del diseño de la aplicación y resultados	**Decisión** Preparación de los datos. Modelado de los datos. Creación del modelo de Árboles de Decisión. Creación de pipeline que permite la automatización del procesamiento de datos. Creación de componente de Reviews de React y de otros archivos tipo js en la carpeta de app web de GitHub.		aplicación web: durante el desarrollo de la aplicación debemos de pensar en una forma de implementar la interfaz web y que esta sea fácil de usar y manejar por el cliente objetivo y que así mismo facilita la visualización de resultado de clasificación de una review, es por ello se optó por el diseño actual que también posee un marco de una estrella que indica visualmente cuantas estrellas obtendría la zona turística.	sería mejor estudiar otros modelos y probar diferentes transformaciones para elegir la que mejor se adapte a lo requerido por el negocio. Adicionalmente, me gustaría para futuras entregas ser capaz de diseñar diferentes mock-ups para la página y posiblemente buscar un público objetivo que me permita evaluar cuál podría ser el mejor diseño y qué cosas cambiarían.	
Ronald Pardo 202111309	Ingeniero de datos Ingeniero de software responsable del diseño de la aplicación y resultados	**Algoritmo trabajado: Máquinas de vector de soporte (SVM)** Desarrollo del documento final. Desarrollo de la presentación hecha en Canva para la realización del video. Creación de repositorio de GitHub. Creación de documento para la segunda etapa de entrega. Diseño de presentación en Canva para	12 hrs	Desarrollo de front-end de la solución planteada: se tuvo que decidir en qué framework se desarrolla, si este sería Angular JS o React JS, finalmente llegamos a la conclusión de que React JS nos facilita mucho el trabajo y que este nos permitirá trabajar mejor en equipo gracias a las abstracciones de este framework y a nuestros conocimientos	Sería ideal tener una mejor organización de tiempo, pero la dificultad de ello implicó que ciertos días se realizará más trabajo que en otros días. Adicionalmente, me gustaría estar más pendiente a la realización de tareas por parte de mis compañeros, ya que ellos finalizaron la gran mayoría de sus tareas	**33%**

		la segunda entrega del proyecto 1. Desarrollo de componente de página principal. Análisis y preparación de los datos.		más recientes de este framework.	días antes, mientras yo todavía estaba en la fase de planeación y de desarrollo de estas.	

Reuniones realizadas:

Título de la reunión	Fecha	Descripción de reunión
Reunión de lanzamiento y planeación	12 de abril de 2024	Se definieron los roles de cada persona y adicionalmente se leyó el enunciado de la segunda etapa del proyecto para definir qué tecnologías y metodologías usamos, así mismo, se definió qué cambios se deben de hacer al análisis y preparación de los datos para mejorar la calidad de ellos. Esta reunión la hicimos con acompañamiento de la estudiante de estadística Carol Clavijo.
Reunión de seguimiento #1	15 de abril de 2024	En esta reunión se discutió el avance que tenía cada uno hasta entonces y se discute cómo se juntarían las tecnologías que se iban desarrollando poco a poco para la solución final de aplicación web con api de modelo integrado. Aquí probamos el funcionamiento de cada componente, como la del API del modelo y la aplicación web.
Reunión de finalización	20 de abril de 2024	Esta reunión se hizo en conjunto con la estudiante de estadística Carol Clavijo, con ella se discutió lo desarrollado en el proyecto, el algoritmo usado y se hizo una demostración del demo para la cual Carol

		nos dio sus sugerencias, preguntas e indicaciones de cosas bien hechas. Así mismo, se discutieron detalles del documento a presentar y del análisis a los resultados.