Reporte de Ingeniería de Datos

La ingeniería de datos es un componente esencial en el ciclo de vida de los datos, permitiendo la recopilación, transformación y almacenamiento de datos de manera eficiente y efectiva. Este informe describe un proceso de ingeniería de datos que abarca la extracción de datos desde una fuente, la transformación y análisis exploratorio de datos preliminar en Python, así como el almacenamiento en Google Cloud Platform (GCP) utilizando Google Cloud Storage, Cloud Functions y BigQuery.

-Tratamiento preliminar (Python):

El tratamiento preliminar de datos en la ingeniería de datos es el primer paso esencial para garantizar que los datos estén limpios, bien estructurados y listos para su procesamiento y análisis. Esta etapa proporciona una base sólida para la construcción de pipelines de datos efectivos y para la obtención de insights valiosos a partir de la información.

Esto nos permite garantizar calidad y fiabilidad de los datos, asegurar consistencia, mejorar eficiencia, prevenir sesgos y errores posteriores, una de las principales causas para nuestro caso es la selección de características relevantes: La etapa preliminar también permite identificar las características más relevantes para el objetivo del proyecto. Esto simplifica el pipeline al reducir la cantidad de datos que deben ser procesados y analizados en profundidad, lo que mejora la eficiencia y evita la inclusión de información redundante.

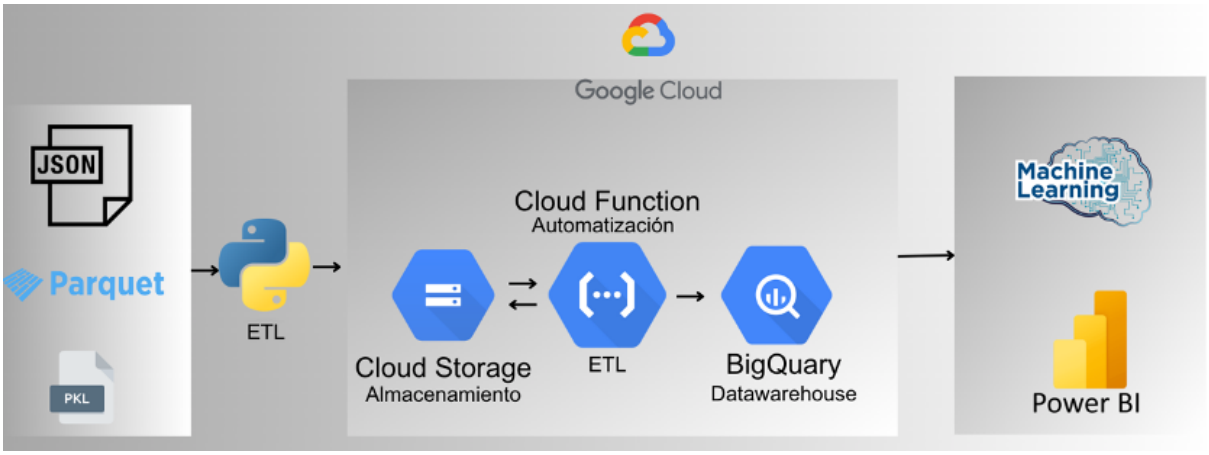
También mejora la planificación y diseño del pipeline; el tratamiento preliminar proporciona información crucial para planificar y diseñar el pipeline de ingeniería de datos de manera más efectiva. Permite definir claramente los pasos que serán necesarios para abordar las particularidades de los datos y prepararlos para su posterior procesamiento y análisis.

-Elección de Google Cloud Platform (GCP)

La elección de Google Cloud Platform se vio fortalecida por la disponibilidad de un crédito inicial gratuito de $400, y la elección de Google Cloud Functions se basó en su capacidad para integrarse con otros servicios de GCP, su escalabilidad sin servidor, eficiencia y su capacidad para responder a eventos clave en el proceso de ingeniería de datos sumado a que las funciones se pueden desarrollar en Python lo que simplifica la creación y mantenimiento de tareas automatizadas.. Estos factores hacen que GCP y GCF sean opciones sólidas para este proyecto específico.

-Arquitectura para el proceso de datos.

La elección de la arquitectura Big Data, está respaldada por la tecnología de GOOGLE CLOUD PLATFORM, junto con la organización de datos en CLOUD STORAGE, la automatización de tareas mediante CLOUD FUNCTIONS, y la explotación de datos a través de BIG QUERY, POWER BI y aplicaciones de MACHINE LEARNING, conforma una estrategia sólida y escalable para gestionar y analizar los extensos conjuntos de datos de YELP y GOOGLE BUSINESS.



En Cloud Storage se utilizan dos buckets distintos. Uno de ellos está destinado exclusivamente al resguardo de los conjuntos de datos en su estado original, sin procesar. El segundo bucket se emplea para albergar los datos una vez que han sido sometidos a procesos de limpieza y transformación.

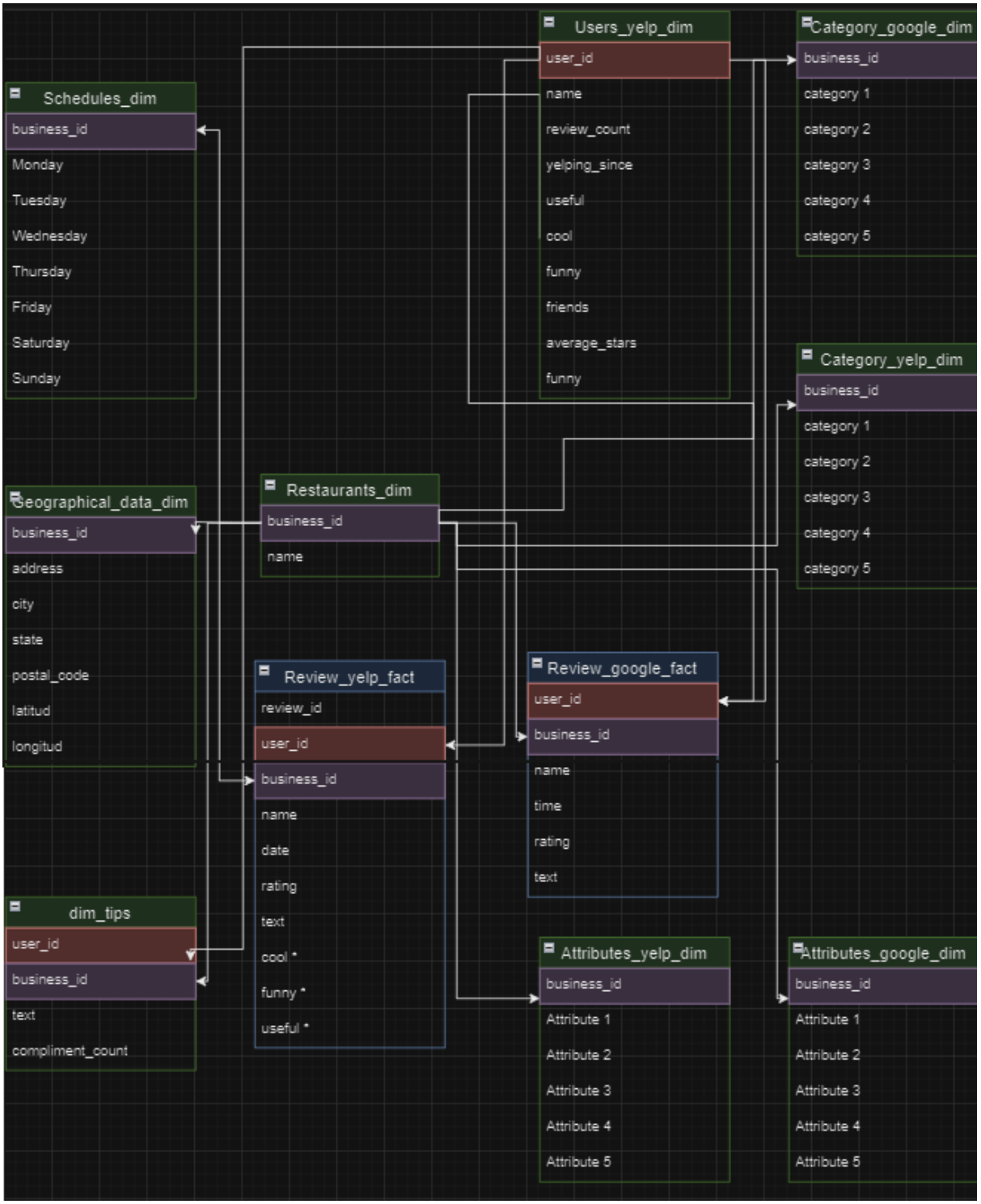
La automatización de tareas se ha logrado mediante el uso de Cloud Functions. Se han desarrollado dos funciones programadas en Python:

Function-2: Esta función se encarga de ejecutar el proceso de Extracción, Transformación y Carga (ETL) de los datos, asegurando su adecuada preparación para su posterior análisis.

Function-3: Esta función se ocupa de importar los datos procesados desde el bucket que alberga los datos procesados hacia su almacenamiento en el Datawarehouse, el cual se realiza con Big Query.

Finalmente, se utiliza Power Bi para la creación y diseño de dashboards y paneles de control que permitirán una visualización efectiva de los datos. Además, se implementan procesos de Machine Learning para la creación de un sistema de recomendación basado en técnicas de aprendizaje automático.

Diagrama de relación de tablas



Diccionario de datos

Datos eliminados

Google

Tabla metadatos

La columna relative_results hace referencia de recomendación a otros negocios pero dado que nos interesa hacer un sistema propio de recomendaciones, se eliminó.

Esta columna nos dice el estado en el que se encuentra el restaurante al momento que se extrajeron los datos por lo que no es relevante para nuestro análisis.

Tabla restaurantes

Había diferencia en el número de "reviews" en la columna num_of_reviews por consecuencia también en "avg_rating" y no se sabía la fecha en que fueron calculados, por lo que no sería correcto simplemente sumarlos y obtener el promedio en avg_rating por lo que se eliminaron.

YELP

Tabla business_id

La columna is_open se eliminó ya que no parecía útil para nuestro análisis. Con respecto a review_count y stars tenían datos faltantes y de por sí podemos obtener esos datos, pero más certeros durante el análisis más profundo que realicen los analistas de datos.

Tabla user

La columna elite se eliminó porque contenía datos en distintos formatos que dificultan su normalización y de por sí no estábamos seguros de que tan útil nos pudiera llegar a ser cualquier información que se obtuviera de los datos.

Tabla categories

la columna categories fue desanidada y de allí se obtuvieron alrededor de 400 columnas de categorías. Solo nos quedamos con las relevantes en relación con nuestro análisis sobre restaurantes de comida rápida.

Tabla attributes

Como ocurrió con categories, en attributes se desanidó la columna para obtener una nueva tabla de atributos y nos quedamos con aquellas que suponemos de utilidad para los analistas de datos, ya que muchas de ellas no solo no guardaban relación con nuestro enfoque sino que también tenían hasta el 70% de los datos faltantes. Algunas de las columnas eliminadas son: "BusinessParking", "WiFi", "ByAppointmentOnly", "AcceptsInsurance","BikeParking","BusinessAcceptsBitcoin", "DietaryRestrictions", "AgesAllowed", entre otras.

Resultado

Restaurants_dim

- **business_id:** identificador para cada restaurante único
- **name:** nombre del restaurante

Schedules_dim

- **business_id:** identificador para cada restaurante único
- Esta tabla contiene información sobre los **horarios** de cada **día de la semana** de los restaurantes

Attributes_google_dim

- **business_id:** identificador para cada restaurante único
- esta tabla cuenta con una serie de **atributos** o **propiedades** con datos relacionados a restaurantes de **Google** como: **tipos de pago**, **características** del **establecimiento**, o de los **servicios** que brinda

Attributes_yelp_dim

- **business_id:** identificador para cada restaurante único
- esta tabla cuenta con una serie de **atributos** o **propiedades** con datos relacionados a restaurantes de **Yelp** como: **tipos de pago**, **características** del **establecimiento**, o de los **servicios** que brinda

Review_google_fact

- **business_id:** identificador para cada restaurante único

- **user_id:** identificador para cada usuario
- **name:** nombre del usuario
- **time:** fecha de la reseña
- **rating:** puntuacion indicada por el usuario según su experiencia en el establecimiento
- **text:** contenido escrito en relación con la opinión del usuario con respecto a su experiencia en el establecimiento

Review_yelp_fact

- **business_id:** identificador para cada restaurante único
- **review_id:** identificador para cada reseña única
- **user_id:** identificador para cada usuario único
- **name:** nombre del usuario
- **date:** fecha de la reseña
- **rating:** puntuación indicada por el usuario según su experiencia en el establecimiento
- **text:** contenido escrito en relación con la opinión del usuario con respecto a su experiencia en el establecimiento
- **stars:** puntuación del usuario según su experiencia en el establecimiento
- Esta tabla contiene indicadores de puntuación (**useful**, **cool**, **funny**) en relación con la reseña
-

Tips_yelp_dim

- **user_id:** identificador para cada usuario
- **business_id:** identificador para cada restaurante único
- **text:** contenido escrito en relación a la opinion del usuario con respecto a su experiencia en el establecimiento
- **compliment_count:** cantidad de veces que se puntuo la reseña

Category_google_dim

- **business_id:** identificador para cada restaurante único

- esta tabla cuenta con una serie de **características** relacionadas a restaurantes de **Google** sobre el **tipo de restaurante o comida** que se sirve como**: comida rápida**, **mexicana**, **italiana**, **pastas**, etc

Category_yelp_dim

- **business_id:** identificador para cada restaurante único
- esta tabla cuenta con una serie de **características** con relacionadas a restaurantes de **Yelp** sobre al **tipo de restaurante o comida** que se sirve como**: comida rápida**, **mexicana**, **italiana**, **pastas**, etc

Users_yelp_dim

- **user_id:** identificador para cada usuario experiencia en el establecimiento
- **name:** nombre del usuario
- **review_count:** cantidad de reseñas realizadas por el usuario
- **yelping_since**: fecha desde la cual el usuario hace uso de la aplicación
- Esta tabla contiene indicadores de puntuación (**useful**, **cool**, **funny**) en relación a la reseña
- **friends:** datos sobre amistades en la aplicación
- **fans:** seguidores del usuario
- **average_stars:** promedio de estrellas del usuario
- Esta tabla contiene indicadores de puntuación para usuarios como (**compliment_hot**, **compliment_funny**, **compliment_photo**) en relación a la reseña

Geographical_data_dim

- **business_id:** identificador para cada restaurante único
- **address**: datos de la dirección del establecimiento
- **city**: datos sobre la ciudad en la cual se ubica el establecimiento
- **state:** datos sobre el estado
- **postal_code**: código postal del establecimiento
- **latitud**: latitud para la ubicación geográfica
- **longitud**: longitud para la ubicación geográfica