Grupo 1
Google Maps

Integrantes:

Cristian Alexander García Abril

Alejandro del Gerbo Actis

Fernando Blas De Olano

Mario Nahuel Vargas

Yamila Abigail Galiano

Entregable - Semana 1

1 - Entendimiento de la situación actual

Este proyecto implica realizar un análisis de mercado en los Estados Unidos para un cliente que es parte de un conglomerado de restaurantes y negocios relacionados.

Tenemos disponibles para analizar, las opiniones de los usuarios en Google Maps con respecto a hoteles, restaurantes y otros negocios relacionados con el turismo. El proyecto implica la recopilación, limpieza y disponibilidad de los datos, la realización de un análisis utilizando técnicas de aprendizaje automático y la formulación de recomendaciones en base a los hallazgos.

Los datos se extraerán de la plataforma de revisión de Google Maps en los EE. UU. y pueden incluir información sobre la ubicación, la categoría, las puntuaciones promedio y las revisiones realizadas por los usuarios. Se pueden utilizar fuentes de datos adicionales para complementar el análisis, como las cotizaciones de acciones y las tendencias en las redes sociales y los medios.

El proyecto también puede involucrar la mejora de estrategias de marketing y la creación de sistemas de recomendación para ubicaciones específicas, como restaurantes y hoteles.

2 - Objetivos

MVP

- [Business Plan]
  Dar un plan de negocio a la empresa que nos contrata.
- [Fortalecimiento Branding]
  Brindar un sistema de recomendación de actividades y servicios <usando servicio GMaps> para los usuarios de Wyndham y así darle la posibilidad de mejorar la experiencia completa de la estadía.

- [App]
  Brindar un sistema de recomendación de restaurantes para los usuarios de Google y así darle la posibilidad de conocer nuevos sabores a partir de sus experiencias previas.

EXTRA

- [Modelo de Monetización]
  Dar mayor conocimiento de turismo y ocio a las empresas del sector en relación a un área para facilitar la toma de decisiones e invertir en expansión de nuevas oportunidades.

  nota: Antes de esta etapa usaremos el producto (MVP) para nuestro cliente “Wyndham” como test del mercado antes de lanzar los banners publicitarios.

3 - Alcance

Límites del proyecto:

[Brindar recomendaciones a la empresa Wyndham]
Brindar recomendaciones a la empresa de nuestro servicio de data, detectando puntos de mejoras y servicios más requeridos.

[Análisis mercado y competencia]
Los rubros de los negocios (relacionados a la empresa) que más crecerán o decaerán del listado de categorías del dataset de Henry.

[Branding: Experiencia de usuario]
Tener un sistema de recomendación de servicios extras para mejorar la experiencia de los usuarios generando lealtad a la marca.

[Análisis de sentimiento Reviews]
Ofrecer información para mejorar el negocio en base al análisis de sentimientos de las reviews de los usuarios.

[Ubicaciones estratégicas Wyndham]
Dar recomendación de ubicaciones estratégicas para abrir el nuevo negocio de la empresa (hotel). Esto se limita a EEUU, sobre las categorías obtenidas del dataset de Henry.

[Features]
A futuro poder proporcionar esta información a diferentes clientes-empresas. Ej: Marriott y Hilton ocupan las mayores cuotas del mercado.

[Monetización de Producto]
A futuro también ofrecer datos sobre la mejora del negocio, como información geográfica, social, etc para agregar valor al negocio. Vale destacar que la monetización del producto debe hacerse usando publicidad y no mediante ranquear a las empresas “recomendadas”.

4 - Objetivos y KPIs asociados (planteó)

- Mejora de rating en base al análisis de las reviews y comparación con rating de negocios cercanos (área).
- Mejorar el tiempo de respuesta de las reviews de los usuarios, el objetivo es ser más rápido que los otros negocios relacionados.
- Tasa de crecimiento del mercado. Proyectar el crecimiento del rubro.
- Satisfacción de clientes. Sentimiento general. Generado a partir de los comentarios de cada hotel.

Estos KPIs pueden ayudar al cliente a comprender mejor su posición en el mercado. Algunas métricas que se proponen además, son las siguientes:

- Turismos vs reviews. Búsqueda de reviews negativas, en lugares donde el turismo está creciendo para ocupar esos mercados.
- Puntuación de google. Promedio y TOP 10 por regiones.
- Número de opiniones.
- Tasa de respuestas.
- Sentimiento general

- Palabras claves que diferencian al negocio.

Estas métricas permiten comprender a los clientes y determinar áreas de mejora.

5 - Repositorio Github

Link de repositorio: https://github.com/YamiGaliano/google-map

6 - Solución propuesta

Deben detallar qué tareas harán para cumplir los objetivos de trabajo propuestos previamente, con qué herramientas (stack tecnológico) y cómo lo harán (metodologías de trabajo, forma de organización, distribución de tareas, roles de cada uno dentro del equipo, etc). También, deben detallar qué productos surgirán de su trabajo y en qué etapa los presentarán, teniendo en cuenta los requerimientos generales (entregables esperados) para cada etapa del proyecto.

A su vez, deben realizar una estimación de tiempo para cada tarea, contemplando los tiempos de ejecución globales y los hitos previstos para cada semana; y plasmar esa estimación en un diagrama de Gantt.

❖ Stack tecnológico

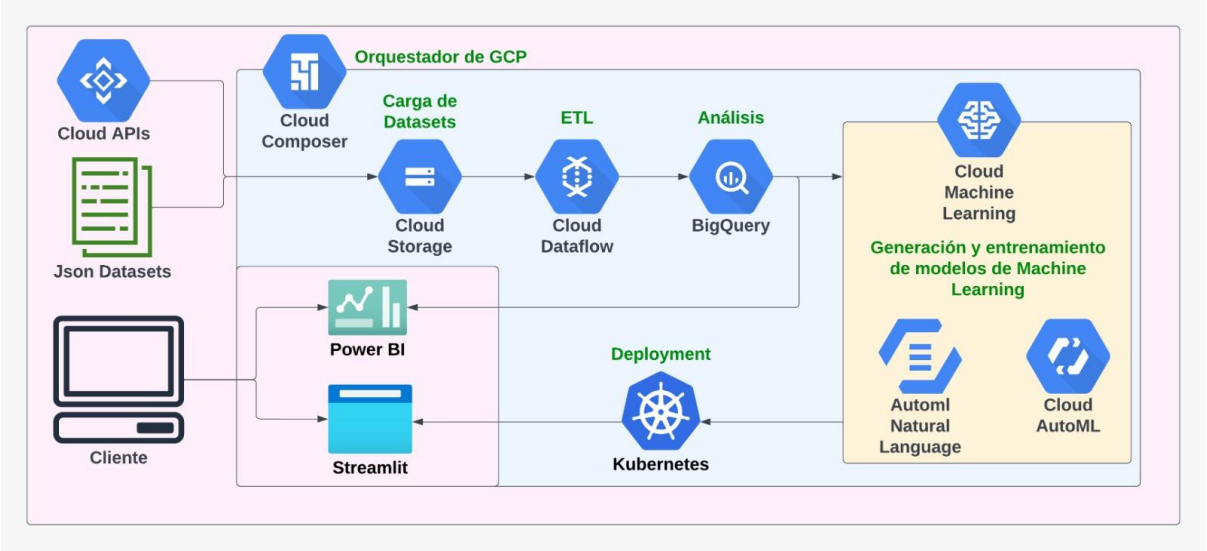
Tras analizar las opciones disponibles online para nuestra infraestructura, decidimos utilizar los servicios de Google Cloud Platform, debido a que este posee toda la arquitectura necesaria para nuestro proyecto. Entre estos vamos a utilizar los siguientes.

- ➢ Google Storage, para almacenar nuestros datasets previamente filtrados por categorías.
- ➢ Cloud Dataflow, para el análisis de datos.
- ➢ Big Query, para realizar el ETL de nuestros datos.
- ➢ Cloud Auto ML, sistema de entrenamiento de machine learning de múltiples propósitos.

- ➢ ML APIS, sistema de entrenamiento específico, en nuestro caso utilizaremos Cloud Natural Language.
- ➢ Kubernets Engine, para el despliegue y ajuste de aplicaciones.
- ➢ Cloud Composer, como orquestador de google cloud.

Además de GCP utilizaremos otras plataformas externas.

- ➢ Streamlite o a definir, como interfaz entre el usuario y nuestro aplicativo.
- ➢ Power BI o Looker Studio, como dashboard para la representación de nuestro análisis.



❖ Metodología de trabajo

Scrum híbrido. Si bien no vamos a trabajar con un scrum completamente, pretendemos trabajar con dailys, weeklys y dateline para cada tarea. Generamos además un calendario para organizarnos con el equipo, un Trello y un diagrama de Gantt y para dar seguimiento a las tareas.

❖ Diseño detallado - Entregables

Vamos a entregar un dashboard (Power BI o Looker Studio) con información del análisis aplicado en los datasets con una implementación de machine learning adaptado a los requerimientos del usuario.

Una interfaz para poder brindar recomendaciones al usuario. Ya sea una empresa que busca analizar su negocio o generar uno nuevo y para usuarios que pretenden vivir una experiencia nueva con recomendaciones de los mejores lugares.

API con la información disponible, lista para ser consultada y consumida por otras aplicaciones.

Informe escrito con detalle.

❖ Equipo de trabajo - Roles y responsabilidades

Destacamos en cada uno las áreas en donde mejor nos podemos desempeñar y las que no.

Nahuel - Tech Lead

- ➢ - Estadística
- ➢ - Machine Learning
- ➢ - Visual
- ➢ - Documentación
- ➢ + Arquitectura
- ➢ + Infra / Data engineer
- ➢ + Analitics
- ➢ + BBDD
- ➢ + Código
- ➢ + Automatización

Yami - Team Manager

- ➢ - Machine learning
- ➢ - Arquitectura
- ➢ + Visual
- ➢ + Analitics
- ➢ + BBDD
- ➢ + Código
- ➢ + Automatización
- ➢ API

Blas - Business Analyst

- ➢ - Infra / data engineer
- ➢ + visual
- ➢ + analitics/Machine learning
- ➢ +Estadística
- ➢ +Código
- ➢ +Automatizar
- ➢ + API stuff

Cristian - Data Engineer

- ➢ - liderazgo/management
- ➢ - infraestructura
- ➢ -Arquitectura
- ➢ + Automatización
- ➢ + Machine Learning
- ➢ + Visual
- ➢ + data engineer
- ➢ + código

Ale - Data Scientist

- ➢ + Análisis
- ➢ + Estadística
- ➢ + Visual
- ➢ + Machine learning
- ➢ - Arquitectura
- ➢ - BBDD
- ➢ - Infra / data engineer
- ➢ - Código

❖ Cronograma general - Gantt

Link: Google Maps - Diagrama de Gantt

❖ Análisis preliminar de calidad de datos

Informe sobre exploración EDA del set "metadata"

Descripción:

El conjunto de metadatos consta de información sobre diversos comercios, oficinas públicas, establecimientos educativos, puntos geográficos destacados, iglesias, parques y demás lugares de interés catalogados por Google, incluyendo nombres, categorías e identificadores únicos de Google Maps. Después de efectuar la limpieza de datos, el conjunto de metadatos incluye información correspondiente a2.154.098 sitios únicos. También se eliminaron duplicados en la columna *"gmap_id"*.

“gmap_id” es una columna que está representada tanto en el dataset de “metadata” de los negocios, como en el dataset de “reviews” de los mismo.
Por lo tanto, es nuestra manera de vincularlos.

Es importante tener en cuenta que algunos valores en varias columnas son "null" y probablemente necesiten ser sustituidos.

Además, se identificaron algunas de las principales compañías hoteleras y de alimentos en los Estados Unidos y se categorizaron en "hotel_top_companies" y "food_top_companies".

Este conjunto de metadatos puede ser útil para explorar patrones y tendencias en la presencia de diferentes tipos de comercios en una zona específica.

7 - EDA

Columnas

El conjunto de datos de metadata incluye 15 columnas que describen diferentes aspectos de los comercios. A continuación, se describen cada una de las columnas:

1. "name": representa el nombre del negocio. Ej: Porter Pharmacy

> **Nota**: esta columna sirve para identificar los Hoteles de las corporaciones Top 10 en USA y las cadenas de comida Top 10 en USA, realizando una búsqueda de palabras claves.

2. "address": representa la dirección completa del negocio. También representa el código postal de la dirección del negocio. Ej: Porter Pharmacy, 129 N Second St, Cochran, GA 31014

> **Nota**: este dato puede ser completado utilizando una conexión con la API.

3. "gmap_id": representa el identificador único de un sitio en Google Maps. Ej: 0x88f16e41928ff687:0x883dad4fd048e8f8

> **Nota**: campo clave para concatenar datasets y conectar con la API.

4. "description": representa la descripción del negocio. Ej: "Chain selling pharmaseutical."

> **Nota**: por ahora dato innecesario, posee muchos nulos, probablemente esta columna sea eliminada.

5. "latitude": representa la latitud geográfica de la dirección del sitio. Ej: "32.3883"

6. "longitude": representa la longitud geográfica de la dirección del sitio. Ej: "-83.3571"

7. "category": representa la categoría asignada por Google Maps para el comercio. Ej: "['Pharmacy']"

> **Nota**: Al exportar del dataset los valores únicos de esta columna (con filas que en algunos casos contenían listas de categorías, ya que un sitio puede ubicarse en más de una de ellas), se detectó la existencia de 4.472 categorías distintas.

Explorando sus descripciones, y comparando con listados de categorías de Google Maps encontrados en la web, concluimos en que los datasets provistos contienen la información de la totalidad de las categorías de sitios relevados por Google. A efectos entonces de reducir sustancialmente el tamaño de los datasets y trabajar a partir de este momento con los datos que sean relevantes para nuestras tareas, decidimos efectuar una exploración manual de las categorías, eliminando aquellas que no nos serían de utilidad (Ejemplos: “Proveedor de alternadores”, “Cirujano de Mano” o “Casa Embrujada”).
Luego de eliminadas las categorías innecesarias, dejando solo aquellas relativas a Hoteles, sitios para concurrir a comer y otros relacionados con el ocio y el esparcimiento, nos quedaron 471 categorías. Eliminamos por lo tanto el 89% de las categorías de sitios incluidas en el dataset inicial, con la finalidad posterior de reducir los registros en proporción similar tanto del dataset de sitios como en el de reviews . Creemos que sería una buena idea generar una columna feature que conforme una supra categoría para nuestros hoteles y restaurantes y los de la competencia. Esta columna también posee muchos valores nulos que deben ser reemplazados.

8. "avg_rating": representa la calificación promedio de los usuarios para el negocio. Ej: 4.9

9. "num_of_reviews": representa el número total de calificaciones de los usuarios para el negocio. Ej: 16

10. "price": Representa un rango de precios en forma de rating, usando valores de símbolo monetario como si fueran las clásicas estrellas.
De esta manera el rango va a de 1 a 5 signos monetarios, donde 5 representa un servicio costoso y 1 un servicio económico. Ej: $$

Nota: Esta columna posee una cantidad muy grande de valores nulos. Esto se debe a que el dataset incluye en su mayoría sitios o servicios que no responden a un servicio oneroso (“parque”) o que no pueden ser catalogados fácilmente dentro de este esquema (“ingeniero civil”) Esta situación debería regularizarse y la cantidad de nulos descender dramáticamente luego de eliminar del dataset las categorías mencionadas, ya que las categorías de alojamientos y sitios para comer

son aquellas en las que más se utiliza este esquema de clasificación de costo o precio.

11. "hours": representa el rango de horarios de servicio del negocio en un formato de lista de listas con los días de la semana y el rango de horario de atención. Ej: [['Friday', '8AMâ€“6PM'], ['Saturday', '8AMâ€“12PM'], ['Sunday', 'Closed'], ['Monday', '8AMâ€“6PM'], ['Tuesday', '8AMâ€“6PM'], ['Wednesday', '8AMâ€“12PM'], ['Thursday', '8AMâ€“6PM']]

12. "MISC": representa valores categóricos que pueden servir para el análisis posterior y desarrollo de features pero también posee gran cantidad de nulos. Ej: {'Service options': ['In-store shopping', 'Same-day delivery'], 'Health & safety': ['Mask required', 'Staff required to disinfect surfaces between visits'], 'Accessibility': ['Wheelchair accessible entrance'], 'Planning': ['Quick visit']}

13. "state": representa el estado en tiempo real de apertura o cierre del negocio. Ej: Open â‹… Closes 6PM

N**ota**: Probablemente necesite conexión con una API (Place API?)

14. "relative_results":representa los resultados de una búsqueda ordenada del negocio por cercanía, mostrando una lista con las "gmap_id"s de los negocios resultantes de la búsqueda ordenados por más cercanos. Ej:['0x88f16e41929435cf:0x5b2532a2885e9ef6',
'0x88f16c32716531c1:0x5f19bdaa5044e4fa',
'0x88f16e6e3f4a21df:0xcf495da9bb4d89ea']

15. "url": indica la dirección web del negocio representada en la interfaz de GMaps. Ej: *https://www.google.com/maps/place//data=!4m2!3m1!1s0x88f16e41928ff687:0x883dad4fd048e8f8?authuser=-1&hl=en&gl=us*

Nota: muy útil para comprobaciones manuales de todos los datos del set (visual).

Nulls

Una vez descartados los campos duplicados por "gmap_id" que representa el valor único de nuestra unidad de negocios, podemos comprobar la cantidad de nulos por columna:

name	37
address	80.511
gmap_id	0
description	2.770.722
latitude	0
longitude	0
category	17.419
avg_rating	0
num_of_reviews	0
price	2.749.808
hours	787.405
MISC	690.834
state	746.455
relative_results	295.058
url	0

Cantidad total de registros sin duplicados y data Type

2.998.428 es el número de registros sin duplicados

Data columns (total 15 columns):

#	Column	Dtype
0	name	object
1	address	object
2	gmap_id	object
3	description	object
4	latitude	float64
5	longitude	float64
6	category	object
7	avg_rating	float64
8	num_of_reviews	int64
9	price	object
10	hours	object
11	MISC	object
12	state	object
13	relative_results	object
14	url	object

Corporaciones Top por Hoteleria y Gastronomia

Hotel Companies:
- Marriott International
- Hilton Worldwide Holdings
- InterContinental Hotels Group
- Accor
- Wyndham Hotels & Resorts
- Hyatt Hotels Corporation
- Choice Hotels
- Best Western Hotels & Resorts
- Radisson Hotel Group
- AccorLive Limitless (ALL)

Food Companies:
- McDonald's Corporation
- Starbucks Corporation
- Yum! Brands, Inc. (KFC, Pizza Hut, Taco Bell)
- Subway
- Dunkin' Brands Group, Inc. (Dunkin' Donuts)
- The Wendy's Company
- Burger King Worldwide, Inc.
- Domino's Pizza, Inc.
- Jack in the Box Inc.
- Sonic Corp.

Nota: Una vez concluida la definición de los posibles valores para la columna "category" se va a buscar cual de estas empresas sino todas están presentes en el dataset.

Informe sobre exploración EDA del set "reviews"

Se compone de 611 archivos json ubicados en 51 carpetas, correspondientes a los diferentes estados de Estados Unidos.

Contiene las reviews individuales realizadas por usuarios de Google Maps de todos los sitios y servicios incluídos por Google.
Estos datos nos aportarán información de alto valor relativa al sentimiento del usuario acerca de cada comercio de interés para nuestro análisis, pero además, mediante el procesamiento del lenguaje natural, esperamos extraer patrones que nos permitan determinar aquellos servicios y características que sean más valorados o menos valorados por los clientes, y que nos permitan cumplir con el objetivo de asesorar con respecto a servicios a incluir o a qué tipo de comercio apostar para una ampliación de las unidades de negocios.
También podremos obtener tendencias y evaluar tiempos de respuesta de los comercios a las críticas de sus clientes.

Estos archivos contienen 9 columnas, que se detallan a continuación:

1. “user_id”: identifica al usuario generador de la review dentro de la plataforma de Google Maps, ej.: "115503702309973535230".
2. “name”: nombre del usuario que generó la review, ej.: “Travone Laffitte”.
3. “time”: fecha y hora en la que el usuario realizó la review. Se encuentra expresado en milisegundos transcurridos desde el 01/01/1970 a las 00:00 horas, ej.: “1623021007311”
4. “rating”: puntaje otorgado por el usuario al comercio en cuestión, siendo 1 el mínimo y 5 el máximo, ej.: “4”.
5. “text”: string en el cual el usuario brinda los detalles que justifican su puntuación, ej.; “I got what I wanted. All of my questions were answered. I'm just waiting for the delivery”.
6. “pics”: campo no obligatorio, se utiliza para los casos en los que el usuario incorpore fotografías del comercio a su review,
7. “resp - time”: fecha y hora en la que el comercio efectuó la devolución a la crítica recibida (en caso de haberlo hecho). Se encuentra expresado en milisegundos transcurridos desde el 01/01/1970 a las 00:00 horas, ej.: “1623077492807”. Este campo nos permitirá evaluar si las unidades de negocio de nuestro cliente responden a las reviews de sus usuarios (algo que consideremos una buena práctica) y los tiempos de respuesta, comparándolos con los de sus competidores.
8. “resp - text”: string conteniendo la respuesta del comercio a la review de su cliente, ej.: “Thanks Travone for your visit and business!!!”.

9. “gmap_id”: representa el identificador único de un sitio en Google Maps, ej.: “0x88f16e41928ff687:0x883dad4fd048e8f8”. Es el campo clave para vincular los datasets de metadata y reviews y conectar con la API.

Entregable - Semana 2

1 - Flujo de trabajo

1 - EDA
2 - Limpieza de datos [ETL no automatizada]
3 - Subida de datos
4 - ETL

2- Stack elegido y fundamentación

3 - Diccionario de datos

4 - Corrección de datos (local)

Documentación del proceso ETL

Introducción:
El siguiente documento describe el proceso ETL realizado para la obtención de información de opiniones de los usuarios de Google Maps y nuestros negocios "Target". El objetivo es proporcionar información clara y concisa sobre cómo se han extraído, transformado y cargado los datos en una base de datos de CGT.
Tengamos en cuenta que nuestro set está subdividido en dos sets "metadata" y "reviews".

Set "reviews"

Pipeline de ETL de los archivos del Set "reviews" para exportación de archivos finales filtrados por categorías(usando "gmap_id"), eliminando la columna "Pics" por ser irrelevante y poseer más del 80% de nulos. Pasos:

1. Importar librerías.

```
import pandas as pd
import json
import os
```

2. Crear filtro usando archivo "gmap_id_metadata_filtrado_category_final.csv" para hacer un data frame llamado "gmap_app".

```
gmap_app =
pd.read_csv("./Dataset/auxi/gmap_id_metadata_filtrado_category_final.csv")
```

3. Importar archivos Json por estados ej: "reviews_Alabama/1.json" y concatenar todos en un solo data frame llamado "df_reviews_estado".
Este paso importa 11 archivos json con toda la metadata de los negocios del SET original.

```
df_reviews_Hawaii = pd.DataFrame()
for i in range(1, 12):
    filename = f'./Dataset/Google_Maps/reviews-estados/review-Hawaii/{i}.json'
    df = pd.read_json(filename, lines=True, dtype={'user_id': str})
    df_reviews_Hawaii = pd.concat([df_reviews_Hawaii, df], ignore_index=True
df_reviews_Hawaii
```

4. Eliminar columna "Pics".
Este paso elimina la columna "pics" del set

```
df_reviews_Hawaii.drop("pics", axis=1, inplace=True)
```

5. Eliminar filas que no correspondan con nuestro filtro "gmap_app".
Este paso elimina toda la data que no está involucrada con nuestro mercado utilizando los "gmap_id"s que obtuvimos del filtrado de nuestros negocios por categoría en el set de "metadata"

```
df_reviews_Hawaii =
df_reviews_Hawaii[df_reviews_Hawaii['gmap_id'].isin(gmap_app['gmap_id'])]
```

6. Exportar nueva versión de los archivos unificados por estado.
En este paso creamos los json finales para ingestar

df_reviews_Hawaii.to_json("./Dataset/reviews_filtered/reviews_Hawaii_final.json", orient="records")

Proceso automatizado

En los pasos anteriores describimos el ETL manual o un siclo de nuestro ETL automatizado.
A continuación el ejemplo de la implementación de nuestro sistema automatizado de ETL para la ingesta a la base de datos de GCP

```
# Alternativa para la subida automatizada de los archivos Json al GCP
carpeta= "./datasets/"
listado= os.listdir(carpeta)
#listado=["./Dataset/Google_Maps/reviews-estados"]
for lista in listado:
    otralista=[]

    if os.path.isdir(carpeta+lista) and "review-" in lista:
        otralista=os.listdir(carpeta+lista)
        df2 = pd.DataFrame()

        for ol in otralista:
            df = pd.read_json(carpeta+lista+"/"+ol, lines=True)
            df2 = pd.concat([df2, df], ignore_index=True)

        df2.drop("pics", axis=1, inplace=True)
        df2 = df2[df2['gmap_id'].isin(gmap_app['gmap_id'])]
        df2.to_json(f'{carpeta}reviews_filtered/{lista}_final.json', orient="records")
```

Set "metadata"

Pipeline para ingesta de datos en Dataflow del set "metadata". Pasos:

1. Importar librerías

```
import pandas as pd
```

2. Importar archivo "category_final"
Este archivo lo utilizaremos en el data frame "categorias_con_filtro" para filtrar toda la data que no corresponde a las mismas

```
categorias_con_filtro = pd.read_csv("./Dataset/auxi/category_final.csv")
```

3. Concatenación de todos los archivos Json del set "metadata" en un único data frame llamado "df_metadata_all".

```
df_metadata_all = pd.DataFrame()
for i in range(1, 12):
    filename =
f'./Dataset/Google_Maps/metadata-sitios/metadata-sitios/{i}.json'
    df = pd.read_json(filename, lines=True)
    df_metadata_all = pd.concat([df_metadata_all, df],
ignore_index=True)
```

4. Creación de un dataframe copia del original

```
df_metadata_tras=df_metadata_all
```

5. Eliminar caracteres. En este paso eliminamos corchetes en nuestra columna "category".
- Transformamos el DataType de Object a String
- Sacamos el primer paréntesis con el método .str.replace
- Sacamos el segundo paréntesis con el método .str.replace

Nota: Este paso es necesario para poder utilizar nuestro filtro de categorías de manera correcta.

```
df_metadata_tras['category'] = df_metadata_tras['category'].astype(str)
df_metadata_tras['category'] = df_metadata_tras['category'].str.replace("[",
"").str.replace("]", "")
df_metadata_tras['category'] = df_metadata_tras['category'].str.replace("'",
"").str.replace("'", "")
df_metadata_tras
```

Comprobación: tenemos 3025011 rows × 15 columns

6. Filtrado del Set "metadata". Este paso utiliza una máscara sobre el dataframe "df_metadata_tras" filtrando todas las categorías ausentes en el dataframe "categorias_con_filtro" utilizando la columna "category".

```
#filtro de df_metadata por categorías
mask = df_metadata_tras['category'].isin(categorias_con_filtro['category'])
df_metadata_tras = df_metadata_tras[mask]
df_metadata_tras.shape
```

Comprobación: tenemos 331966 rows x 15 columns
Nota: En este paso también comprobamos como la data decrece aproximadamente a un 10% de la data original (predicción hecha por nuestro departamento de DS).

7. Eliminación de duplicado por "gmap_id"
En este paso eliminamos todos la data duplicada

```
#drop duplicated
df_metadata_tras.drop_duplicates(subset='gmap_id', inplace=True)
df_metadata_tras.shape
```

Comprobación: (329537, 15)

8. Eliminación de la columna “state”

9. Creación de un .csv para filtrar el Set "reviews". Este paso es el que nos proporciona la máscara para el filtro de las reviews de nuestros locales target.

```
# export gmap_id in csv
df_metadata_tras[['gmap_id']].to_csv("./Dataset/auxi/gmap_id_metadata_filtrado_category_final.csv", index=False)
```

10. Exportar datos en Json

```
df_metadata_tras.to_json("./Dataset/metadata_filtered/metadata_final", orient="records")
```

Proceso automatizado

El proceso automatizado consiste en la extracción, transformación y carga de los datos de las reseñas de Google Maps en un conjunto de negocios seleccionados previamente. El proceso se realiza para todos los archivos de reseñas contenidos en la carpeta "reviews-estados_inprocess". El proceso genera un archivo final para cada archivo de reseñas en la carpeta "reviews-estados_inprocess".

Pasos del proceso:

1. Se carga un archivo CSV "gmap_app" que contiene los gmap_id de los negocios objetivo.

2. Se establecen dos variables "carpeta" y "destino" que representan las rutas de la carpeta de entrada y la carpeta de salida, respectivamente.

3. Se genera una lista de archivos contenidos en la carpeta de entrada.

4. Se realiza un bucle "for" para cada archivo de la lista.
5. Si el archivo es una carpeta y contiene la cadena "review-" en su nombre, se genera una lista de archivos en la carpeta y se crea un dataframe vacío "df2".
6. Se carga cada archivo de reseña JSON de la carpeta a un dataframe y se concatenan en el dataframe "df2".
7. La columna "pics" se elimina del data frame "df2".
8. Se filtra el data frame "df2" para incluir solo los registros con gmap_id que se encuentran en el archivo CSV "gmap_app".
9. Se eliminan los caracteres ";" y "\n" en la columna "text" del data frame "df2".
10. Se exporta el data frame "df2" a un archivo CSV en la carpeta de salida "destino". El archivo final se nombra con el nombre del archivo de reseñas original con la extensión "_final.csv". El separador de campo se establece en ";" y la primera fila de encabezado se excluye en la salida.

CÓDIGO EJ:

```
import pandas as pd
import json
import os
```

```
        gmap_app =
pd.read_csv("./Dataset/auxi/gmap_id_metadata_filtrado_category_final.csv")

        carpeta= "./Dataset/Google_Maps/reviews-estados_inprocess/"
        destino= "./Dataset/reviews_filtered_csv/"

        listado= os.listdir(carpeta)

        for lista in listado:
            otralista=[]

            print(lista)
            if os.path.isdir(carpeta+lista) and "review-" in lista:
                otralista=os.listdir(carpeta+lista)
                df2 = pd.DataFrame()

                for ol in otralista:
                    df = pd.read_json(carpeta+lista+"/"+ol, lines=True,
dtype={'user_id': str})
                    df2 = pd.concat([df2, df], ignore_index=True)

                df2.drop("pics", axis=1, inplace=True)
                df2 = df2[df2['gmap_id'].isin(gmap_app['gmap_id'])]
                # Replace ";" in the "text" column
                df2['text'] = df2['text'].str.replace(";", "")
                df2['text'] = df2['text'].str.replace("\n", "")
                 df2.to_csv(f'{destino}/{lista}_final.csv', sep=';', index=False,
header=False)
```

Conclusión:

Este proceso ETL permite obtener información de opiniones de los usuarios de Google Maps de manera organizada y estructurada para su posterior análisis. La documentación del proceso garantiza una comprensión clara y sencilla del proceso realizado y facilita su replicación o actualización en el futuro.

Pipeline ETL (infra)

El pipeline comienza con una fuente de datos que es un conjunto de archivos JSON que contienen información de negocios, reseñas y otras entidades de interés en Estados Unidos. Esta fuente de datos es procesada mediante una secuencia de pasos que comienza con la extracción de datos relevantes y la transformación de estos en una forma más estructurada y fácil de usar.

Google Cloud Store

A continuación, los datos transformados se almacenan en Google Cloud Storage, lo que permite su posterior procesamiento. Para llevar a cabo esta tarea, se utiliza el servicio de almacenamiento de Google Cloud, que proporciona una solución escalable y duradera para almacenar datos en la nube.

Cloud Dataflow

Después de que los datos se almacenan en Cloud Storage, se utiliza Cloud Dataflow para llevar a cabo el procesamiento ETL (Extract, Transform, Load) necesario para preparar los datos para el análisis. Cloud Dataflow es un servicio de Google Cloud que permite la creación de pipelines de procesamiento de datos que pueden escalar para manejar grandes cantidades de información.

Metadata

2.1 - Generamos listado de categorías target (local)

2.2 - Importamos y concatenamos todos los archivos

2.3 - Normalización de "category"

2.4 - Filtrado del Set "metadata" por nuestro target de categorías

2.5 - Eliminación de columna “state”

2.6 - Exportamos la data para ingestar en GCP

2.7 - Subimos la data target a CloudStorage

2.8 - Importamos los DataSets a BigQuery

Reviews

1.1 - Importamos y concatenamos archivos por estado

1.2 - Eliminamos columna "pics"

1.3 - Filtramos por "gmap_id"

1.4 - Exportamos la data para ingestar en GCP

1.5 - Subimos la data target a CloudStorage

1.6 - Importamos los DataSets a BigQuery

BigQuery

Finalmente, una vez que los datos se han procesado y se han almacenado en un formato adecuado, se pueden analizar utilizando BigQuery, que es una herramienta de análisis de datos que permite la consulta de grandes volúmenes de información de manera rápida y eficiente. Con BigQuery, los datos se pueden analizar y visualizar de diversas maneras, lo que permite extraer información valiosa para la toma de decisiones empresariales.

Decisiones ejecutivas

A lo largo del proceso de ingesta de datos en la plataforma y calibración para el ETL tomamos una serie de decisiones de modificación de nuestra data para el uso correcto.
Desde el reemplazo de separadores o la eliminación de saltos de línea para una correcta lectura o por formato adecuado para la arquitectura de la data.
Hasta la eliminación de columnas irrelevantes por diferentes razones.

Nuestros criterios buscan eficiencia y evitarnos consumos excesivos

Reviews

- Eliminación de columna pics.

Metadata

- Eliminación de columna state.
- Dividir columna resp en 2 columnas.
- Definimos variables categóricas para ML, correspondiente a columna MISC.

Ambos

- Modificamos separador de los campos de (,) a (;). Información corrupta.
- Eliminamos salto de línea (/n) por problemas de carga.

Diagrama entidad-relación