Prueba técnica – Ingeniera de datos – Nequi

Andrea Elneser Tejeda

Para la siguiente prueba técnica, se establecieron pasos y criterios a seguir, que se presentan a continuación:

Paso 1: Alcance del proyecto y captura de datos

En este primer paso, se establece el alcance de la prueba de ingeniería de datos y se capturarán los datos necesarios para llevar a cabo las tareas.

Alcance del proyecto:

El caso de uso principal posible con estos datos sería realizar un análisis de canales de YouTube utilizando un conjunto de datos que contiene información sobre diversos aspectos de los canales. Este análisis permitirá entender mejor la dinámica de los canales de YouTube, identificar tendencias en la creación de contenido, y explorar la relación entre diferentes variables como el número de suscriptores, la cantidad de videos publicados y el rendimiento de los videos en términos de vistas.

Caso de uso específicos:

Identificación de canales destacados: Permite identificar los canales más populares en función del número de suscriptores y vistas, lo que proporciona información valiosa sobre la popularidad y el impacto de estos canales en la plataforma.

Análisis geográfico de los canales: Facilita el análisis de la distribución geográfica de los canales, lo que puede revelar patrones regionales de actividad y audiencia, así como insights sobre la diversidad cultural de los creadores de contenido en YouTube.

Análisis de palabras clave: Permite identificar las palabras clave más utilizadas por los canales exitosos, lo que proporciona información sobre los temas y enfoques de contenido que resuenan con la audiencia.

Otro caso de uso posible puede ser análisis de imágenes usando las columnas que entregan el link del banner y el link del avatar.

Todo esto puede llevar a obtener un mejor entendimiento de los canales y cómo estos pueden lograr el éxito.

Conjunto de Datos:

Nombre: 2024 Youtube Channels

Fuente: https://www.kaggle.com/datasets/asaniczka/2024-youtube-channels-1-million

Descripción: Conjunto de datos con información detallada sobre diversos canales de YouTube, abarcando aspectos como el tamaño de la audiencia, la actividad del canal y el rendimiento de los videos.

Número de filas: 1095243

Paso 2: Explorar y evaluar los datos, el EDA.

Se realizó una exploración EDA de los datos en la cual se entienden los datos y su naturaleza. Se hizo una revisión inicial para entenderlos, identificar los tipos de datos iniciales, validar qué nulos y duplicados pueden existir y se entendió una descripción inicial de estos.

Luego se hizo una limpieza eliminando duplicados y nulos donde fuera necesario, se cambiaron formatos y se hizo imputación de datos en la columna posible.

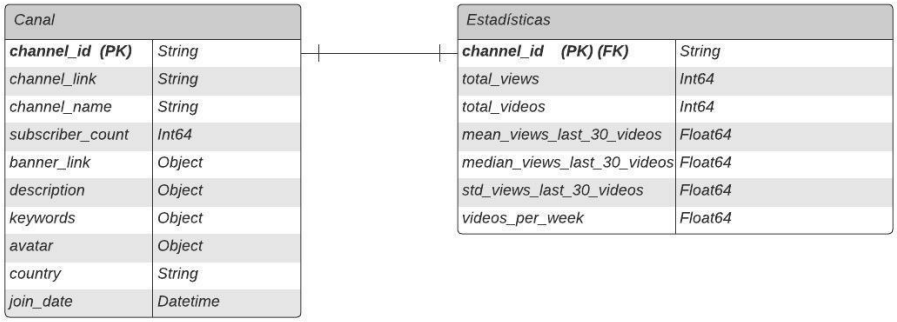
Por último, se hizo un análisis exploratorio visual.

Este proceso permitió conocer los datos y reconocer cómo se deben manejar en los siguientes pasos.

Paso 3: Definir el modelo de datos

- **Trazar el modelo de datos conceptual y explicar por qué se eligió ese modelo.**

El modelo de datos conceptual elegido consta de dos entidades principales: "Canal" y "Estadísticas". Estas entidades están relacionadas entre sí a través del identificador único del canal (channel_id), estableciendo una relación uno a uno.



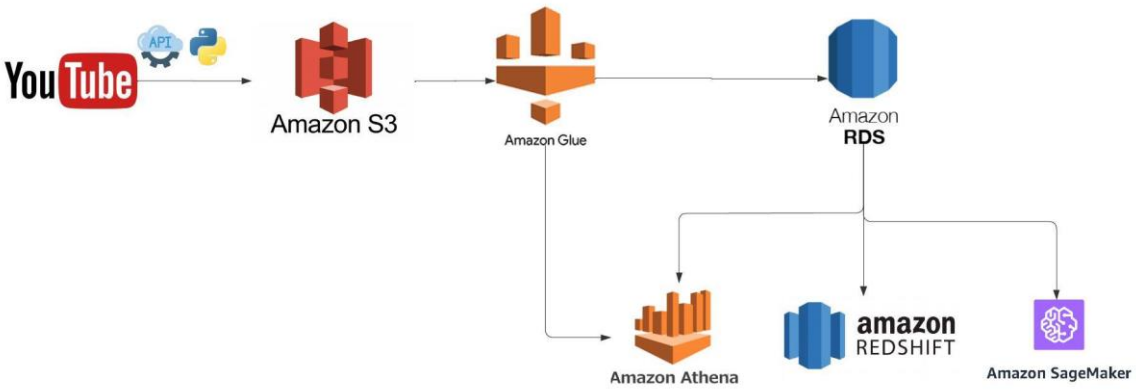
Razones para elegir este modelo:

Simplicidad: El modelo es simple y directo, lo que facilita su comprensión y mantenimiento.

Separación de preocupaciones: Separar la información básica del canal de las estadísticas permite una mejor organización de los datos y evita la redundancia.

Escalabilidad: El modelo permite agregar fácilmente más atributos y funcionalidades relacionadas con el canal y sus estadísticas en el futuro sin afectar la estructura principal.

- **Diseñar la arquitectura y los recursos utilizados.**



Extracción de Datos desde YouTube:

Se utiliza la API de YouTube junto con un script en Python para extraer datos de los canales y sus estadísticas desde YouTube. Estos datos se obtienen en formato JSON.

Almacenamiento en Amazon S3:

Los datos extraídos se almacenan en un bucket de Amazon S3, un servicio de almacenamiento escalable y duradero en la nube de AWS. S3 proporciona una ubicación centralizada y segura para los datos, accesible desde cualquier lugar.

Procesamiento con AWS Glue:

Se utiliza AWS Glue, un servicio de ETL (Extract, Transform, Load) totalmente administrado, para procesar y transformar los datos almacenados en S3 según sea necesario. Glue puede manejar la limpieza de datos, la transformación de formatos, la creación de pipelines de datos y la coordinación automática de tareas de procesamiento.

Almacenamiento en Amazon RDS o Directamente en Athena:

Los datos procesados pueden almacenarse en una base de datos relacional en Amazon RDS (Relational Database Service). Alternativamente, los datos pueden almacenarse directamente en Amazon Athena para consultas ad-hoc y análisis interactivo sin la necesidad de una base de datos relacional.

Análisis con Amazon Athena, SageMaker o Redshift:

Los datos almacenados en Amazon RDS y/o Amazon Athena pueden consultarse y analizarse utilizando SQL estándar en Athena. Esto permite realizar análisis ad-hoc y exploratorios sobre los datos almacenados en S3.

Además, los datos pueden cargarse en Amazon Redshift, un almacén de datos optimizado para análisis de datos a gran escala. Redshift proporciona un rendimiento rápido y escalable para consultas complejas y análisis de datos avanzado.

También se pueden utilizar los datos para entrenar modelos de Machine Learning en Amazon SageMaker, que proporciona un entorno completo para el desarrollo, entrenamiento y despliegue de modelos de ML en la nube.

• **Indique claramente los motivos de la elección de las herramientas y tecnologías para el**

proyecto.

Flexibilidad y Escalabilidad: La arquitectura propuesta ofrece flexibilidad y escalabilidad al permitir la integración de diferentes servicios de AWS según las necesidades específicas del análisis. Se puede elegir entre una variedad de opciones de almacenamiento y análisis de datos, desde bases de datos relacionales hasta almacenes de datos columnares y servicios de análisis ad-hoc.

Automatización y Eficiencia: El uso de servicios completamente administrados de AWS, como AWS Glue, Amazon RDS, Amazon Athena y Amazon Redshift, simplifica el proceso de extracción, procesamiento y análisis de datos al eliminar la necesidad de gestionar la infraestructura subyacente. Esto permite una mayor eficiencia y permite a los equipos centrarse en la análisis y generación de insights en lugar de en la gestión de la infraestructura.

Pago por Uso: La arquitectura propuesta sigue el modelo de pago por uso de AWS, lo que significa que solo se paga por los recursos y servicios que se utilicen, lo que la hace rentable y escalable para proyectos de cualquier tamaño.

• Proponga con qué frecuencia deben actualizarse los datos y por qué.

La frecuencia de actualización de los datos depende de varios factores, incluida la naturaleza de los datos, la velocidad de cambio, los requisitos comerciales y los recursos disponibles. En este caso:

Frecuencia: Se podría considerar una actualización diaria de los datos, ya que los canales de YouTube pueden experimentar cambios frecuentes en sus estadísticas, como el número de suscriptores, vistas y videos publicados.

Razones: La actualización diaria garantiza que los análisis y reportes se basen en datos recientes y precisos, lo que es fundamental para tomar decisiones informadas y mantener la relevancia de las métricas de rendimiento.

La elección final de la frecuencia de actualización debe ser una decisión conjunta entre los equipos técnicos y comerciales, teniendo en cuenta los compromisos de rendimiento, costos y objetivos comerciales del proyecto.

Paso 4: Ejecutar la ETL

Para la creación de la tubería y modelo de datos, se desarrolló un ETL inicial que sube los datos a Amazon S3 después de haberlos transformado y obtenido desde un .csv.

Este ETL cumple con los criterios de:

- Crear las tuberías de datos y el modelo de datos
- Ejecutar controles de calidad de los datos para asegurar que la tubería funcionó como se esperaba
- Control de calidad en los datos con la integridad en la base de datos relacional (por ejemplo, clave única, tipo de datos, etc.)
- Pruebas de unidad para los “Script” para asegurar que están haciendo lo correcto.
- Comprobaciones de fuente/conteo para asegurar la integridad de los datos.
- Incluir un diccionario de datos

• Criterio de reproducibilidad

Paso 5: Completar la redacción del proyecto

• ¿Cuál es el objetivo del proyecto?

El objetivo del proyecto es realizar un análisis exhaustivo de los canales de YouTube utilizando un conjunto de datos que contiene información detallada sobre diversos aspectos de los mismos. Esto incluye comprender la dinámica de los canales, identificar tendencias en la creación de contenido y explorar la relación entre diferentes variables como el número de suscriptores, la cantidad de videos publicados y el rendimiento de los videos en términos de vistas.

• ¿Qué preguntas quieres hacer?

Algunas preguntas que podríamos plantear incluyen:

- ¿Cuáles son los canales más populares basados en el número de suscriptores y vistas?
- ¿Cuál es la distribución geográfica de los canales?
- ¿Qué palabras clave están asociadas con los canales más exitosos?
- ¿Cuál es la relación entre el número de videos publicados y el rendimiento en términos de vistas?

• ¿Por qué eligió el modelo que eligió?

El modelo de datos conceptual seleccionado se basó en la simplicidad, la claridad en la relación, la eficiencia en el acceso a los datos y la flexibilidad para futuras expansiones. Al constar únicamente de dos entidades principales, "Canal" y "Estadísticas", vinculadas a través de un identificador único de canal (channel_id) en una relación uno a uno, se logra una estructura directa y fácil de entender. Esta elección facilita el diseño, la implementación y el mantenimiento de la base de datos, al tiempo que permite la posibilidad de futuras ampliaciones sin alterar significativamente la estructura existente.

• Incluya una descripción de cómo abordaría el problema de manera diferente en los siguientes escenarios:

o Si los datos se incrementaran en 100x: en este caso, consideraría utilizar servicios de procesamiento distribuido como Apache Spark para manejar el aumento en el volumen de datos de manera eficiente.

o Si las tuberías se ejecutaran diariamente en una ventana de tiempo especifica: implementaría un sistema de orquestación de tareas como Apache Airflow o Amazon Data Pipeline para programar y ejecutar las tuberías de manera automatizada y programada.

o Si la base de datos necesitara ser accedido por más de 100 usuarios funcionales: herramientas como AWS Redshift estaría en la capacidad de hacer esto, pero tendría más en cuenta criterios como optimización de consultas, gestión de recursos y accesos y particionamiento de datos.

o Si se requiere hacer analítica en tiempo real, ¿cuáles componentes cambiaria a su arquitectura propuesta?: introduciría componentes de procesamiento de datos en tiempo real como Apache Kafka y Apache Flink o Amazon Kinesis en la arquitectura para permitir el análisis de datos en tiempo real y la generación de insights en tiempo casi real.