UNIDAD DE INGENIERÍA EN COMPUTACIÓN
Sede Central Cartago
Bases de Datos 2
Profesor: Kenneth Obando Rodríguez

TEC | Tecnológico de Costa Rica

Proyecto 1 (%) Fecha de Entrega: 22 de noviembre, 2024

Red Social de Viajes

Descripción

Este proyecto tiene como objetivo que los estudiantes apliquen conceptos de integración de datos utilizando Neo4J para almacenar y consultar datos, Spark para procesar grandes volúmenes de información y SQL para almacenar y visualizar resultados. Usaremos el **Customer Transaction Dataset** de Kaggle, que contiene datos de transacciones de clientes en una tienda. Los estudiantes realizarán análisis de los patrones de transacción y gasto de los clientes.

Dataset: Customer Transaction Dataset

El dataset incluye las siguientes características:

- **Customer_ID**: Identificador único de cada cliente.
- **Product_ID**: Identificador único de cada producto.
- **Transaction_Amount**: Monto de la transacción.
- **Transaction_Date**: Fecha de la transacción.

El dataset simula las transacciones de los clientes en una tienda, y se usará para analizar el comportamiento de compra.

Objetivos y Pasos

1. Carga de Datos en Neo4J

Los estudiantes deben cargar el dataset en Neo4J, representando:

- **Clientes**: Cada cliente será un nodo con propiedades como el `Customer_ID`.
- **Productos**: Cada producto será un nodo con propiedades como el `Product_ID`.
- **Transacciones**: Cada transacción se representará como una relación entre un cliente y un producto, con propiedades como `Transaction_Amount` y `Transaction_Date`.

Instrucciones:

- Transformar el archivo CSV en consultas Cypher para crear los nodos y relaciones en Neo4J.
- Crear índices en `Customer_ID` y `Product_ID` para optimizar las consultas.

2. Extracción de Datos con Spark

Utilizando Spark, extraer datos desde Neo4J para analizarlos. Los estudiantes deben utilizar consultas Cypher dentro de Spark para cargar los datos necesarios y realizar cálculos.

Ejemplos de Consultas y Análisis en Spark:

- **Gasto Total por Cliente**: Calcular el monto total gastado por cada cliente en todas sus transacciones.
- **Productos Más Comprados**: Contar la cantidad de veces que cada producto fue comprado.
- **Promedio de Gasto por Cliente**: Calcular el gasto promedio de cada cliente por transacción.
- **Frecuencia de Compra por Cliente**: Determinar cuántas transacciones ha realizado cada cliente.

Ejemplo en Spark:

```
from pyspark.sql import SparkSession

spark = SparkSession.builder \
    .appName(”Neo4J_Spark_Project”) \
    .getOrCreate()

# Ejemplo: Leer datos de Neo4J
transactions_df = spark.read \
    .format(”org.neo4j.spark.DataSource”) \
    .option(”url”, ”bolt://localhost:7687”) \
   .option(”query”, ”MATCH (c:Customer)-[t:TRANSACTED]->(p:Product) RETURN c.Customer_ID, p.P
    .load()

# Procesamiento de datos en Spark
total_spent_per_customer = transactions_df.groupBy(”Customer_ID”) \
    .sum(”Transaction_Amount”) \
    .withColumnRenamed(”sum(Transaction_Amount)”, ”Total_Spent”)
```

3. Guardar los Resultados en SQL

Una vez que los estudiantes han realizado los cálculos en Spark, deben almacenar los resultados en una base de datos SQL para facilitar su visualización y análisis posterior.

Instrucciones:

- Crear una tabla SQL para cada análisis realizado:

  - **Total_Spent_Per_Customer**: Gasto total por cliente.
  - **Product_Purchase_Count**: Conteo de cada producto comprado.
  - **Average_Spend_Per_Customer**: Gasto promedio por cliente.
  - **Transaction_Count_Per_Customer**: Cantidad de transacciones por cliente.

- Usar un conector JDBC para guardar los resultados desde Spark a una base de datos SQL como PostgreSQL o MySQL.

Ejemplo de Guardado:

```
total_spent_per_customer.write \
    .format("jdbc") \
    .option("url", "jdbc:postgresql://localhost:5432/mydb") \
    .option("dbtable", "Total_Spent_Per_Customer") \
    .option("user", "myuser") \
    .option("password", "mypassword") \
    .save()
```

4. Visualización de Resultados (Puntos extra)

Para visualizar los datos almacenados en SQL, los estudiantes pueden usar herramientas como Tableau, Power BI o incluso gráficos en Python, tome en cuenta que estas herramientas tienen un tiempo de prueba limitado, opciones gratuitas puede ser Superset o Streamlit pero requieren una configuración más compleja. Deben realizar visualizaciones básicas que muestren:

- Gasto total por cliente en un gráfico de barras.
- Conteo de productos más comprados.
- Gasto promedio por cliente.

Instrucciones Adicionales

1. Documentación

Cada estudiante debe entregar un breve informe que explique:

- Cómo cargaron los datos en Neo4J y qué consultas utilizaron.
- El código utilizado en Spark para realizar cada análisis.
- Capturas de pantalla de los resultados en SQL y de las visualizaciones.

2. Entrega y Evaluación

- **Estructura y Carga en Neo4J**: 25%

- **Procesamiento en Spark**: 40%
- **Guardado en SQL**: 25%
- **Documentación**: 10%
- **Visualización**: 10% extra

Links

- Customer Transaction Dataset en Kaggle