Design and Coding Project: Data Engineer

Problem Statement

An e-commerce company needs to create targeted marketing campaigns by segmenting its customers based on behavior, demographics, and purchase history.

Design and implement a configurable ETL pipeline to process and segment customer data from an e-commerce platform. The pipeline should handle data extraction from various sources, cleaning, feature engineering, segmentation, and loading the results into a reporting database for analysis.

NOTE:

1. When working on the assignment, if you are unsure about a part of the problem, make an assumption, add the assumption to your README, and move ahead.
2. Will not be evaluated based on, specific tooling. Preferable to have experience with AWS services, but want someone who is agnostic of the cloud and can learn new platforms quickly

Steps

1. **Generation:** Generate data points with Customer profile information (name, age, income, mobile, gender) and store this file on disk. Generate customer purchase history (mobile, date, amount, store) from a REST API endpoint at runtime.
2. Build a pipeline that achieves the following steps,
    a. **Extraction:** Extract data from both the above sources and join them based on the primary keys.
    b. **Transformation:**
        i. **Cleaning:** Remove duplicate records, Handle missing values (e.g., impute missing ages, fill missing purchase history with zero), and Correct any inconsistencies in data (e.g., standardize date formats).
        ii. **Feature Engineering:**
            1. Calculate customer lifetime value (CLV) based on purchase history.
        iii. **Normalization:**
            1. Normalize numerical features (e.g., scale age, income).
            2. Standardize categorical features (e.g., encode gender as binary).
        iv. **Segmentation:**
            1. Apply a clustering algorithm such as scikit K-means to segment customers into distinct groups (e.g., high-value customers, frequent shoppers, occasional buyers).
            2. Use a predefined number of clusters
        v. **Aggregation:**
            1. Aggregate metrics per segment - total spend per segment, and total number of customers per segment.
3. **Data Loading:**
    - **Schema Design:** Design a schema for the reporting database to store customer segments and metrics.
    - **Load Data:** Load the transformed data into the reporting database in tables like `customer_segments`, `segment_metrics`, etc.

Expected Deliverables

1. **ETL Pipeline Code** which performs data extraction, transformation, and loading.
2. **Data Warehouse Schema** for the reporting database, including table structures and relationships.
3. **Documentation:**
    - A readme explaining the pipeline design, and transformation logic.

    - Instructions for setting up, configuring, and running the pipeline.

4. **Test Results** demonstrating the pipeline’s functionality with sample data, including validation of segmentation accuracy and data integrity.

Evaluation Criteria

1. **Functionality:**
    - Does the pipeline correctly handle data extraction, transformation, and loading?
    - Are customer segments accurately created and loaded into the reporting database?
2. **Efficiency:**
    - Is the pipeline optimized for performance, both in terms of processing time and resource usage?
    - Are transformations and loading processes efficient?
3. **Robustness:**
    - How well does the pipeline manage errors, handle missing data, and process edge cases?
    - Are there adequate logging and error-handling mechanisms?
4. **Documentation:**
    a. Is the code and configuration well-documented and understandable?
    b. Are setup and configuration instructions clear and complete?