Data Engineering Capstone Project

- US I94 Immigration Analytics

Author: Fan Yuan

Created: 04/25/2021

# Overview

This project attempts to build an ETL pipeline hosted on S3 with the use of Spark so as to set up efficient analytics framework with I94 immigration, global land temperatures and US demographics data. Within the project, we will load data from S3, process the data into analytics tables (fact and dimension tables which will act as the fundamental of further business cases analysis) using Spark based on EDA (exploratory data analysis) result, and load them back into S3. The Spark process will be deployed on a cluster using AWS. The project breaks down into steps as following:

# Scope the Project and Gather Data

* Project scope

This project is going to utilize I94 immigration, global land temperatures and US demographics data to build up analytics database. By combining the datasets appropriately, analysis can be done on the dataset to explore the pattern and insight of US immigration along the time.

The first step will be loading the data set with Spark, then appropriate processing and cleansing will be made based on EDA (exploratory data analysis) on I94 immigration, demographics and global land temperatures dataset. Then, dimension and fact tables will be created accordingly so that further advanced analysis can be done on the database.

* Data to be used

The project will be working with four datasets that resides in S3. The main dataset will include data on immigration to the United States, and supplementary datasets will include data on airport codes, U.S. city demographics, and temperature data.

End use cases

* Technology and tools

The project will store the data on Amazon S3 and use Apache Spark to read in source data from staging tables, extract necessary columns needed for analysis and populate the fact and dimension tables. Then will use Spark to write the data back to S3 if needed. For the data modeling part, the project will use the dimensional model which will make it easy for business users to work with the data and also improve analytical queries performance. So in this case, we will use Star Schema which fits OLAP (online analytical processing) very well.

* Why S3 and Spark

When dealing with the large dataset in this project, with the combining of both batch and streaming capabilities, Spark can support the use case very well where the data need to be stored and analyzed in real-time. It will have more flexibility when more type and volume of data sources need to be added. Therefore, storing the data on S3 will eliminate need to invest in costly hardware and scale up with full flexibility when needed. And speaking of Parquet files, the columnar format that being used will be a good option to store big data set and for analytics purpures as well.

And Spark can efficiently read data from S3 and process the data with full sets of data analytics and machine learning libraries. Especially when dealing with large dataset, Spark has more capacity to handle the performance and efficiency.

# Explore and Assess the Data

Explore the data to identify data quality issues, like missing values, duplicate data, etc.

Document steps necessary to clean the data

# Define the Data Model

Map out the conceptual data model and explain the reason of choosing that model

List the steps necessary to pipeline the data into the chosen data model.

In this project, we will use Star Schema to convert the data which is a model designed to support OLAP (online analytics processing) at its best. We will have Fact tables which will record business events (like an order, a phone call, or a book review, etc.) recorded in quantifiable metrics (like quantity of an item, duration of a call, or a book rating, etc.). And Dimension tables will record the context of the business event (e.g. who, what, where, why, etc.) with attributes value like location of a store where the item was purchased, or name of the customer who made the call, etc.

Conceptual data model

Steps to map out the data into the model

# Run ETL to Model the Data

Create the data pipelines and the data model

Include a data dictionary

Run data quality checks to ensure the pipeline ran as expected

Integrity constraints on the relational database (e.g., unique key, data type, etc.)

Unit tests for the scripts to ensure they are doing the right thing

Source/count checks to ensure completeness

How often the data should be updated and why

# More application scenarios

If the data was increased by 100x.

If the pipelines were run on a daily basis by 7am.

If the database needed to be accessed by 100+ people.