SOFE 4630U: Cloud Computing

Dr. Mohamed El-darieby

Date: March 29th, 2022

Project Milestone #4 -Data Processing

Github: Yusuf-shaik/Cloud-Lab-4 (github.com)

COURSE GROUP #1	
Shanjay Kailayanathan	100624670
Jana Kanagalingam	100603975
Yusuf Shaik	100655451
Yakhneshan Sivaperuman	100703400

1. Google Cloud has another processing service called DataProc. Name another processing service that is usually used in the cloud environment (not necessarily GCP). Compare between it and both Dataflow and DataProc. Your comparison may include but is not limited to the major differences, advantages, disadvantages, and limitations.

An alternative to Google Cloud DataProc is Amazon EMR. Amazon EMR is another big processing service that is usually used in the cloud environment and is a managed cluster platform that simplifies running big data frameworks.

Differences

- Amazon EMR has a greater market share with about 11x more customers and is used in about 2x more countries.
- Google Dataflow can automatically run jobs on a cluster such as balancing work or Scaling the number of workers for a job which can be very time consuming on other systems.

Similarities

- Resizing
- Autoscaling
- Flexible virtual machine types

Advantages

- Serverless simplicity-it doesn’t require software installation,additional licensing costs and overhead
- Fast exploration and anomaly detection-data is explored instantly
- Easy and powerful data prep-predicts next data transformation

Disadvantages/Limitations

- API access limited
- Don't have access to data quality rules
- No scheduling/plan management
- Support is limited

2. Suggest a practical application using both stream and batch processing that can be applied to a given dataset.

The practical application in my opinion would be a smart home control system that would control temperature of houses in an area. The application would contain sensors inside and outside the house to measure windspeed, temperature, humidity, weather conditions, and other environmental factors. The sensors inside would measure the temperature within the houses and there would be an actuator that regulates the humidity, and temperature within the house based on the conditions outside.

The application would function by creating a "subnetwork" of a city which would be selection of various houses in the same area, temperature data would be accumulated from outside each house to ensure a variety of datapoints for the application which would result in a more accurate determination of the weather conditions. As mentioned before, there would be two sources of data, so kafka would require two separate schemas for the data ingestion module:

Sensors within the house [home sensors] would follow the following schema:

- {temperature: float, lighting: int, humidity: float, number of people: int, area of house to regulate: float}

Sensors external to the house:

- {temperature: float, time of day: datetime, humidity: float, precipitation: float, windspeed: float, natural disaster: Boolean}

Data would be constantly streamed every second, for multiple houses, from both outdoor and indoor sensors. This would generate a large volume of data after a short while.

This application will follow the lambda architecture which uses a batch layer and speed layer, also known as a cold and hot path. The hot path follows quick analytics to make real time decisions, while the cold path usually caters to longer term analytics and storage. Before entered into the paths, and right after ingestion, preprocessing is needed to ensure the data is clean, there are no missing fields, etc. This application will use dataprep cluster for this since it follows the preprocessing mechanisms defined in the previous answer. The "removal of redundant data" feature that is offered by dataprep however, will not be utilized since we want all data to be stored in a data lake, and schema on read availability when the data is queried.

In this application, the cold path will constantly feed data to a machine learning model using HDInsight to update the model, and improve its efficiency at determining the best possible temperature based on current conditions. In this path, the data will also be fed into Azure

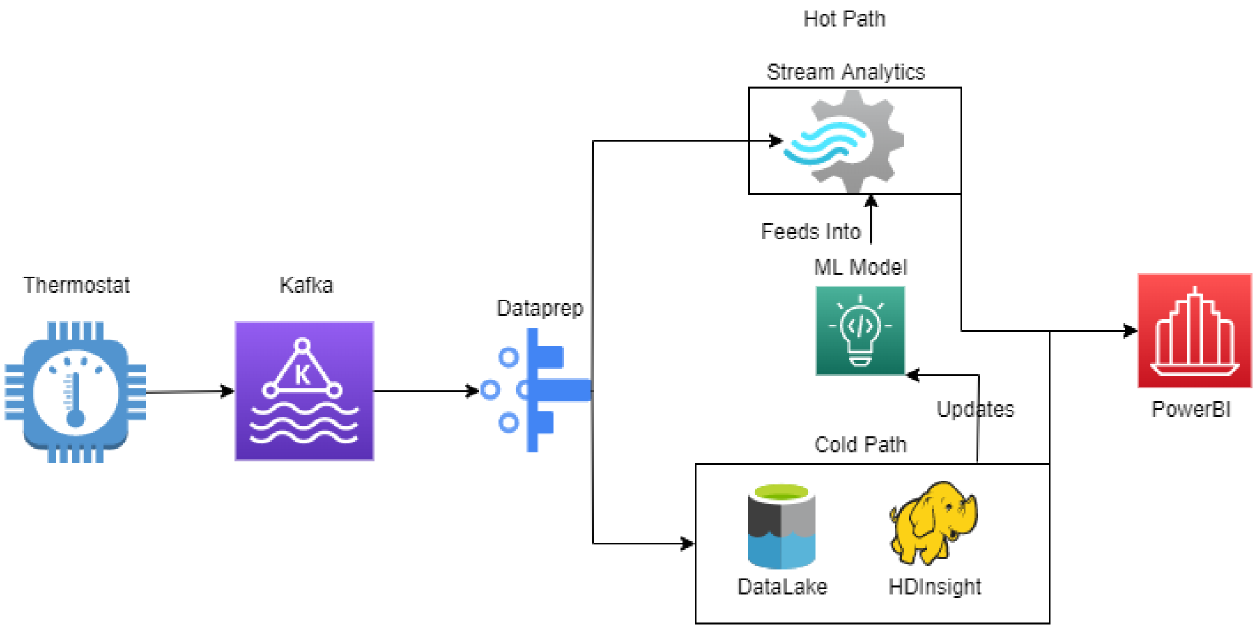
DataLake that will store the data for a long period of time, allowing the user to benefit from the schema-on-read ability when the data is used for further processing.

The hot path will use Azure stream analytics to make real time decisions based on the current conditions in the area. It will use the constantly updated machine learning model to get the best possible temperature for a given house based on the conditions in that area from the given house, and its surrounding areas.

On the business end, a powerbi user interface will be used to display the distribution of data in given areas, and understand how the temperature is regulated over a given area. This will be useful in case the ML model contains any obvious visual bias that needs to be fixed, and it would cause the ML model to need to be retrained.

With the rising cost of energy, and the need to aim all users to be more energy efficient, this application will allow for smart controls of a users home to ensure that energy is not wasted. If a user leaves their home to go on vacation, or simply leaves for the workday, there will be no wastage of energy since the system will be able to determine if a user is home, and regulate the energy that way.

As explained, here is a graphical representation of the pipeline:


Hot Path
Stream Analytics
Feeds Into
ML Model
Thermostat
Kafka
Dataprep
Updates
Cold Path
PowerBI
DataLake
HDInsight