Empirical Study on BigQuery Machine Learning Budge towards Google BigQuery

T.Logeswari
Associate Professor, New Hoirzon College, Bangalore, Karnataka, India

Abstract

Storing and querying massive datasets can be time overwhelming and costly without the right hardware and infrastructure. Google BigQuery is an enterprise data depot that solves this problem by enabling super-fast SQL queries using the processing authority of Google's infrastructure. BigQuery ML, novel software that helps users to build some machine learning models inside the Google BigQuery cloud data warehouse with standard SQL commands. BigQuery Machine Learning (ML) eliminates the need to move cloud-based data sets from Google BigQuery to a separate tool to develop and train analytical models.

KEYWORDS: GoogleQuery, GoogleQuery ML

Introduction

Google BigQuery is a web Service planned for analyzing data on the order of billions of rows, using a SQL-like syntax. It runs on the Google Cloud Storage communications and can be accessed with a REST-oriented application program interface (API). By using a web UI or a command-line tool, or by making calls to the BigQuery REST API using a variety of client libraries such as Java, .NET, or Python or variety of third-party tools that can use to interact with BigQuery, such as visualizing the data or loading the data[3]. It is fully-managed to deploy any resources, such as disks and virtual machines to get started by running a web query or using the command-line tool. BigQuery ML enables users to create and execute machine learning models in BigQuery using SQL queries.

The goal is to democratize machine learning by enabling SQL practitioners to build models using their existing tools and to increase development speed by eliminating the need for data movement. BigQuery ML enables data scientists and data analysts to construct and operationalize ML models on planet-scale planned or semi-structured data, straight inside BigQuery, using simple SQL in a little bit of the time.

Literature Review

BigQuery, which was released as V2 in 2011, is what Google calls an "externalized version" of its home-brewed Dremel query service software[1]. Dremel and BigQuery employ columnar storage for fast data scanning and a tree architecture for dispatching queries and aggregating results across huge computer clusters. BigQuery in its Dremel form has been used inside Google to track device installation data, create crash reports and analyze spam. Since its inception, BigQuery features have continually been improved.

In early 2013, data joins and time stamps were added to the service. Later in the year, stream data insert capabilities were added. BigQuery ML likely won't convince many data scientists who analyze data stored in BigQuery to change how they build models, said Daniel Mintz, chief data evangelist at software vendor Looker Data Sciences Inc[2]., which has teamed up with Google to enable its data modeling and analytics platform to function as a front-end tool for BigQuery ML users[1]. With BigQuery ML, Campo-Rembado added, his team was able to build a linear regression model in just 30 seconds to analyze movie trailers to help pinpoint audiences that should be targeted in promoting the latest Maze Runner movie released in January.

II. BigQuery Features

a) **Serverless** - data warehousing gives the resources to focus on data and analysis, rather than in service and sizing computing possessions.
b) **Real-time Analytics** -BigQuery's high-speed streaming insertion API provides a controlling base for real-time analytics.
c) **Automatic High Availability**-Free data and compute reproduction in several locations means data is available for query even in the case of excessive failure modes.

d) **Standard SQL**- BigQuery supports a standard SQL dialect which is ANSI:2011 compliant, reducing the need for code rewrite and allowing you to take advantage of advanced SQL features. BigQuery provides free ODBC and JDBC drivers to ensure your current applications can interact with BigQuery's powerful engine.
e) **Federated Query and Logical Data Warehousing**-BigQuery breaks down data silos to analyze all your data assets from one place. Through powerful federated query, BigQuery can process data in object storage (Cloud Storage), transactional databases (Cloud Bigtable), or spreadsheets in Google Drive — all without duplicating data. One tool lets you query all your data sources[4].
f) **Storage and Compute Separation**-BigQuery provides with fine-grained control of cost and access. With BigQuery's separated storage and compute, you pay only for the resources you use. The option to choose the storage and processing solutions that make sense for your business and control access for each.
g) **Automatic Backup and Easy Restore**- BigQuery automatically replicates data and keeps a seven-day history of changes, reducing worries about unexpected data changes. This allows you to easily restore and compare data from different times.
h) **Geospatial Datatypes and Functions**-BigQuery GISBETA brings SQL support for the most commonly used GIS functions right into data warehouse. With support for arbitrary points, lines, polygons, and multi-polygons in WKT and GeoJSON format, simplify your geospatial analyses, see your location-based data in new ways, or unlock entirely new lines of business with the power of BigQuery.
i) **Data Transfer Service**-BigQuery makes it easy to get started with data warehousing, even data is in a SaaS application. The BigQuery Data Transfer Service automatically transfers data from external data sources, like Google

Marketing Platform, Google Ads, and YouTube, to BigQuery on a scheduled and fully managed basis.

j) **Big Data Ecosystem Integration**-With Cloud Dataproc and Cloud Dataflow, BigQuery provides integration with the Apache Big Data ecosystem, allowing existing Hadoop/Spark and Beam workloads to read or write data directly from BigQuery. BigQuery allows you to get the most out of structured data by making it easy to analyze in SQL and easy to integrate with your existing Big Data jobs.

k) **Petabyte Scale**-BigQuery is fast and easy to use on data of any size. With BigQuery, great performance on data, while knowing you can scale seamlessly to store and analyze petabytes more without having to buy more capacity.

l) **Flexible Pricing Models**-BigQuery enables you to choose the pricing model that best suits you. On-demand pricing lets you pay only for the storage and compute to use. Flat-rate pricing enables high-volume users or enterprises to choose a stable monthly cost for analysis.

m) **Data Encryption and Security**-The full control over who has access to the data stored in BigQuery. BigQuery makes it simple to keep strong refuge with fine-grained uniqueness and contact organization with Cloud Identity and Access Management, and data is always encrypted at respite and in transit.

n) **Data Locality**-The option to store your BigQuery data in US, Japan, and European locations while continuing to benefit from a fully managed service. BigQuery gives the option of geographic data control, without the headaches of setting up and managing clusters and other computing resources in-region.

III. Introduction to BigQuery ML

BigQuery ML enables users to create and execute machine learning models in BigQuery using standard SQL queries[2]. BigQuery ML democratizes machine learning by enabling SQL practitioners to build models using existing SQL tools and skills. BigQuery ML increases development speed by eliminating the need to move data.

a)BigQuery ML currently supports the following types of models:

- Linear regression models can be used for predicting a numerical value.
- Binary logistic regression models can be used for predicting one of two classes
- Multiclass logistic regression for classification models can be used to predict more than

two classes such as whether an input is "low-value", "medium-value", or "high-value".

b)BigQuery ML functionality is available by using:

- The BigQuery web UI
- The bq command-line tool
- The BigQuery REST API
- An external tool such as a Jupyter notebook or business intelligence platform

Machine learning on huge data sets requires widespread programming and information of ML frameworks. These necessities limit solution growth to a very small set of people within each company, and they exclude data analysts who understand the data but have limited machine learning knowledge and programming expertise. BigQuery ML empowers data analysts to use machine learning through existing SQL tools and skills. Analysts can utilize BigQuery ML to construct and estimate ML models in BigQuery[5]. Analysts no longer need to export small amounts of data to spreadsheets or other applications.

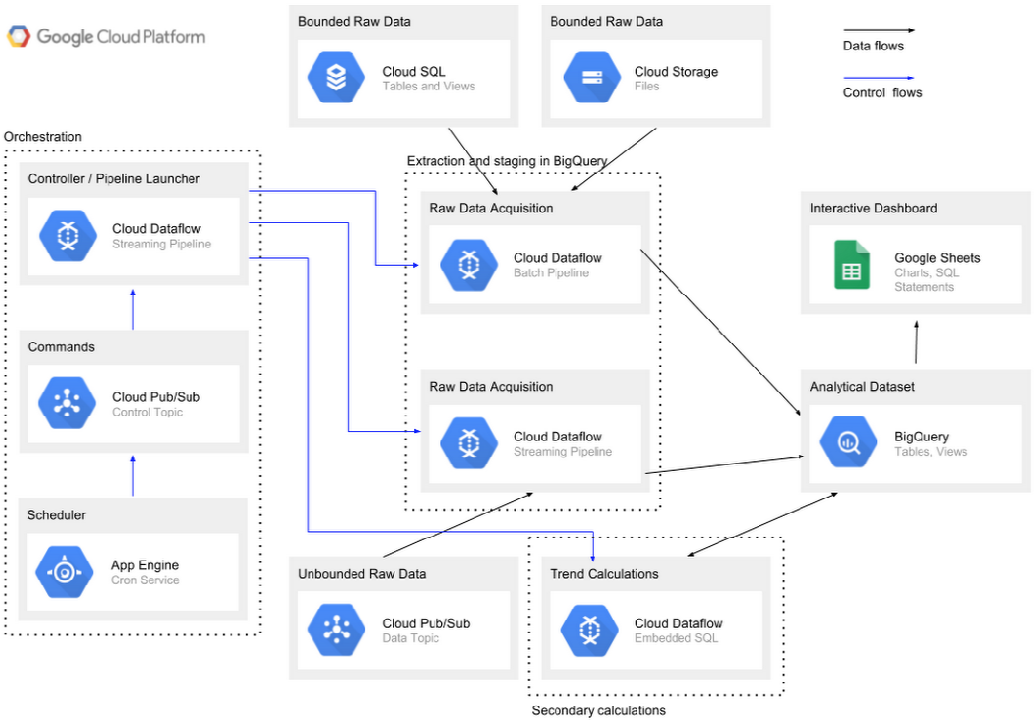
c)Advantages of BigQuery ML

BigQuery ML has the following advantages over other approaches to using ML with a cloud-based data warehouse:

- BigQuery ML democratizes the use of ML by empowering data analysts, the primary data warehouse users, to build and run models using existing business intelligence tools and spreadsheets. This enables business decision making through predictive analytics across the organization.
- There is no need to program an ML solution using Python or Java. Models are trained and accessed in BigQuery using SQL a language data analysts know.
- BigQuery ML increases the speed of model development and innovation by removing the need to export data from the data warehouse. Instead, BigQuery ML brings ML to the data. Exporting and re-formatting the data:
- Increases complexity of multiple tools is required.
- Reduces speed of Moving and formatting large amounts data for Python-based ML frameworks takes longer than model training in BigQuery.
- Requires multiple steps to export data from the warehouse, restricting the ability to experiment on your data.

d**) Architecture components**

The mechanism of this architecture include shows from left to right



:

Figure 1- ETL architecture for cloud-native data warehousing on GCP

- A task orchestrator built using Google App Engine Cron Service, Google Cloud Pub/Sub control topic and Google Cloud Dataflow in streaming mode
- Cloud Dataflow for importing bounded (batch) raw data from sources such as relational Google Cloud SQL databases (MySQL or PostgreSQL, via the JDBC connector) and files in Google Cloud Storage
- Cloud Dataflow for importing unbounded (streaming) raw data from a Google Cloud Pub/Sub data ingestion topic
- BigQuery for storing staging and final datasets
- Additional ETL transformations enabled via Cloud Dataflow and embedded SQL statements
- An interactive dashboard implemented via Google Sheets and connected to BigQuery

All these mechanism are examples of fully-managed services on GCP; with this architecture, there's no infrastructure for you to deploy, manage, secure or scale.

IV NEED OF BIGQUERY ML

In BigQuery ML, a model can be used with data from multiple BigQuery datasets for training and for prediction.

```
CREATE MODEL dataset.model_name
 OPTIONS(model_type='linear_reg', input_label_cols=['input_label'])
AS SELECT * FROM input_table;
```

When you create a model, categorical variables (of type BOOL, STRING, BYTES, DATE, DATETIME, or TIME) are one-hot encoded by default during training and prediction.

TIMESTAMP is not currently one-hot encoded by default. Use the CAST function to cast TIMESTAMP columns to STRING so that BigQuery ML treats the column as categorical.

1. CREATE MODEL syntax

{CREATE MODEL | CREATE MODEL IF NOT EXISTS | CREATE OR REPLACE MODEL}

model_name

[OPTIONS(model_option_list)]

[AS query_statement]

2. ML.EVALUATE Function

ML.EVALUATE function to evaluate model metrics. The ML.EVALUATE function can be used with both linear regression and logistic regression models. You can also use the ML.ROC_CURVE function to evaluate logistic regression models, but ML.ROC_CURVE is not supported for multiclass models.

The output of the ML.EVALUATE function is a single row containing common metrics applicable to the type of model supplied.

3. ML.EVALUATE syntax

ML.EVALUATE(MODEL model_name,

{TABLE table_name | (query_statement)}

[, STRUCT(AS threshold)])

4. ML.PREDICT function

The ML.PREDICT function can be used to predict outcomes using the model. Prediction can be done during model creation, after model creation, or after a failure (as long as at least 1 iteration is finished). The output of the ML.PREDICT function has as many rows as the input table, and it includes all columns from the input table and all output columns from the model. The output column names for the model are predicted_<label_column_name> and (for logistic regression models) predicted_<label_column_name>_probs. In both

columns, label_column_name is the name of the input label column used during training[6].

5. ML.PREDICT syntax

ML.PREDICT(MODEL model_name,

{TABLE table_name | (query_statement)})

The areas to use BIGQUERY ML are includes

1. Predict Basketball Outcomes

2. Predict Birth Weight

Conclusion

Google BigQuery is an analytics examine, inexpensive project data warehouse which has now been rebranded as BigQuery ML. The main advantages of BigQuery is that it transforms SQL queries into complex implementation plans, dispatching them onto carrying out nodes to on time provide insights into the data. It enables developers to execute SQL as a in particular parallel handing out query with hundreds of CPU cores and kind disk storage space, scanning and aggregating terabytes of data in seconds.

However, BIGQUERY ML can improve the following things

- ✓ Clearout and Preprocessing Data in SQL
- ✓ More control to Data Analysts:
- ✓ Democratizes ML
- ✓ Reduced Waiting Time

References

1. https://searchbusinessanalytics.techtarget.com/news/252445723/BigQuery-ML-moves-machine-learning-into-Google-BigQuery
2. https://searchdatamanagement.techtarget.com/definition/Google-BigQuery
3. https://cloud.google.com/products/ai/
4. https://cloud.google.com/bigquery/docs/bigqueryml-analyst-start
5. https://files.eric.ed.gov/fulltext/EJ1137342.pdf
6. https://www.analyticsindiamag.com/how-googles-bigquery-ml-is-empowering-data-analysts/