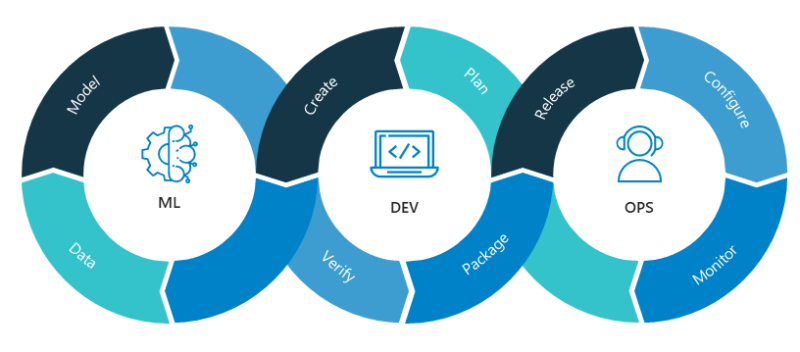
**MLOps**

# **What is MLOps?**

MLOps is a set of practices **to unify ML systems development(dev) and ML systems deployment(ops) to be able to use high accuracy models during production.** It is a continuous execution of certain steps(aka pipeline) in a systematic manner which simplify the management process, and automates the process of deploying ML and DL models for large scale deployments.



Source: <https://blogs.nvidia.com/wp-content/uploads/2020/09/MLOps-Neal-Analytics.png>

# **Why do we need MLOps?**

## **Delay in deployment:**

Applying machine learning models into production has proven even more difficult than finding resourceful data and figuring out how to train the models. Even the most talented engineers find it difficult to deploy their sophisticated softwares.

## **Weak communication:**

To bring a product to production it requires the work of three main teams of people namely Data engineers, data scientists and software engineers. These three teams seem to sit worlds apart from operations teams. Even if we are able to sneak out of development stages, there is rarely sufficient, streamlined support to bring our work to full production.

## **Lack of Foresight:**

The data which are experienced in production might be way different from the data used to train models. The model should be future ready( continuously trained and monitored)

## **High expectations:**

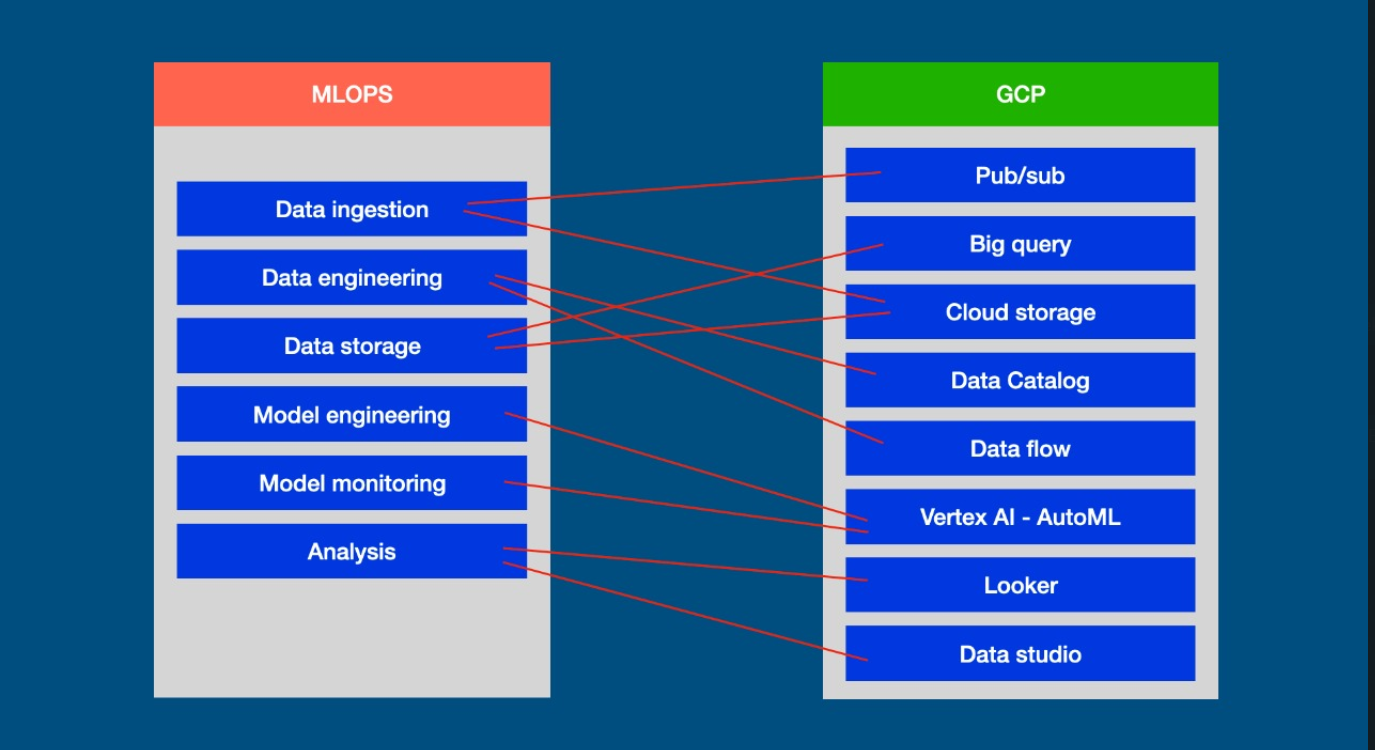
There is so much hype around data science and machine learning due to which expectations are generally very high. They require complex technologies and a lot of resources along with an enormous amount of time and effort to implement even a single machine learning model in the real world.

## **Collecting data:**

This is the most important step which is not released until it's too late. About 60% of the work is done if we collect the right data. Knowledge of data engineering will prove helpful in collecting the right data.

Data scientists primarily focus on developing complex algorithms, neural network architectures and transformation of data, not the process of deploying microservices, network security, or other critical aspects of real-world implementations. MLOps lays the groundwork for people from multiple disciplines who have varied expertise to come together to help deploy machine learning into real world applications.

# **Building blocks of MLOps and its respective GCP Tool/Service**



## **MLOPS Component #1:**

**MLOPS Component Name:** Problem statement

**MLOPS Component Functionality:**

A simple clear understanding of the problem faced, why it needs to be solved and a description of the solution. These three together make the problem statement.

## **MLOPS Component #2:**

**MLOPS Component Name:** Data ingestion

**MLOPS Component Functionality:**

Moving data from different sources to a storage from which data can be used or analyzed according to its requirements.

**MLOPS Component Implementation GCP Tool/Service Name:**

* IOT core
* Cloud storage
* Pub/sub

**MLOPS Component Implementation GCP Tool/Service Functionality:**

**IOT core is a fully** managed service for securely connecting and managing IoT devices, from a few to millions.

**Cloud storage** is a reliable and highly durable object store. In this environment, Cloud Storage is used as an artifact store to save the outputs of ML pipeline steps.

**Pub/Sub** allows services to communicate asynchronously, with very less latency. Pub/Sub is used for streaming analytics and data integration pipelines to ingest and distribute data. It is equally effective as messaging-oriented middleware for service integration or as a queue to parallelize tasks. Pub/Sub enables you to create systems of event producers and consumers, called publishers and subscribers. Publishers communicate with subscribers asynchronously by broadcasting events.

## **MLOPS Component #3:**

**MLOPS Component Name:** Data Engineering

**MLOPS Component Functionality:**

Set of methods to make raw data useful. Without data engineering, it would be impossible to make sense of the huge amounts of data.

**Data engineering pipeline:**

**Data cleaning:** Handling missing, duplicate and inaccurate data

**Data analysis:** Analyzing data to arrive at meaningful conclusions to make better decisions.

**Data validation:** Validation is done to create data that is consistent, accurate and complete, to prevent data loss and errors during a move.

**Feature Engineering:** Selecting the required features(Variables) from raw data by using knowledge developed from previous steps to create a predictive model.

**Data splitting**: Dividing data into generally two parts Trains and Test for developing and evaluating the predictive models respectively.

**MLOPS Component Implementation GCP Tool/Service Name:**

* Google cloud Dataflow

**MLOPS Component Implementation GCP Tool/Service Functionality:**

**Google Cloud Dataflow** is a cloud-based data processing service for both batch and real-time data streaming applications. It enables developers to set up processing pipelines for integrating, preparing and analyzing large data sets, such as those found in Web analytics or big data analytics applications.

## **MLOPS Component #4:**

**MLOPS Component Name:** Data storage

**MLOPS Component Functionality:**

Storing data in the form of interpretable databases or just simply storing data in the cloud.

**MLOPS Component Implementation GCP Tool/Service Name:**

* Cloud Storage
* BigQuery

**MLOPS Component Implementation GCP Tool/Service Functionality:**

**Cloud storage** A reliable and highly durable object store. In this environment, Cloud Storage is used as an artifact store to save the outputs of ML pipeline steps.

**BigQuery** stores data using a columnar storage format that is optimized for analytical queries. BigQuery presents data in tables, rows, and columns and provides full support for database transaction semantics (ACID).

## **MLOPS Component #5:**

**MLOPS Component Name:** Model engineering

**MLOPS Component Functionality:**

Steps involving Writing, executing and evaluating machine learning algorithms to build a suitable model for the problem statement.

**Model engineering pipeline:**

**Model training:** Applying machine learning algorithms to the data prepared after data engineering to train the model.

**Model evaluation:** Using different metrics to evaluate the performance of the model

**Parameter tuning:** Hyperparameters like number of training steps, learning rate, initialization values and distribution are tuned for improving performance

**Model testing:** Testing the model with the test set to know how the model is performing in the real world.

**Model packaging:** Packing all the requirements to host the model as a usable tool to make it ready for production.

**MLOPS Component Implementation GCP Tool/Service Name:**

* Vertex AI - AutoML
* AI Platform

**MLOPS Component Implementation GCP Tool/Service Functionality:**

**Vertex AI - AutoML** Train models from labeled images and evaluate their performance.

**AI Platform** is a suite of services on Google Cloud specifically targeted at building, deploying, and managing machine learning models in the cloud

## **MLOPS Component #6:**

**MLOPS Component Name:** Model monitoring and deployment

**MLOPS Component Functionality:**

After training the model, it has to be deployed in the form of a web application or mobile app. Monitoring involves observing the model’s performance based on real and previously unseen data. This helps us in fine-tuning and making improvements to the model to make it useful in the real world.

**MLOPS Component Implementation GCP Tool/Service Name:**

* Vertex AI - AutoML Vision Edge

**MLOPS Component Implementation GCP Tool/Service Functionality:**

**Vertex AI - AutoML Vision Edge:** Feature attribution skew or drift detection for models deployed to Vertex AI online prediction. It allows you to train and deploy low-latency, high accuracy models

## **MLOPS Component #7:**

**MLOPS Component Name:** Analysis

**MLOPS Component Functionality:**

This step involves arriving at the final results and the impacts that the model has produced in the real world. Take action if required, like sending alerts.

**MLOPS Component Implementation GCP Tool/Service Name:**

* Google Datastudio
* Looker

**MLOPS Component Implementation GCP Tool/Service Functionality:**

**Google Datastudio** is used to connect to data from different tools (sheets, excel, Google Analytics etc.) and then visualize that data into beautiful reports and share it with others.

**Looker** is a big data analytics platform that helps you explore, analyze and share real-time business analytics easily leading to insightful insights.

# **What are the benefits of MLOps?**

We use MLOps to increase productivity and create usable, enterprise-grade models. From innovative new businesses to large-scale public transportation divisions, teams are using MLOps to create impactful change in their fields.

* Rapid innovation and effective management through well planned machine learning lifecycle management
* Optimises the productivity of teams
* Reproducible Machine learning pipelines and models
* Easy deployment of high precision models in any location
* Machine learning resource management system and control

# **Who needs MLOps?**

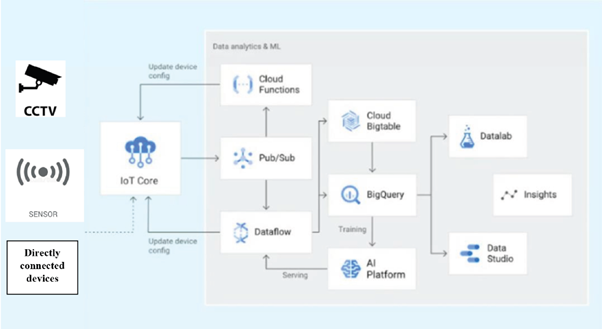
Any product based company or person looking to build and deploy machine learning models into the real world for production will need MLOps to help them achieve their targets effectively and efficiently.

# 

# 

# **MLOps Architecture in GCP:**

**(Specific to Computer vision solutions)**



# **MLOPs: An intuitive case study on ENERGY EFFICIENCY AND ENERGY BENCHMARKING.**

## **Problem statement:**

Energy consumed in buildings is a significant fraction of that consumed in all end-use sectors. Although percentages vary from country to country, buildings are responsible for about 30 to 45% of the global energy demand. For

instance, buildings consume approximately 41% of total primary energy use by sector

in the US (United States Department of Energy [USDOE] 2012) and particularly, 18%(17.4 out of 97.4 quadrillion Btu) from commercial buildings. Energy consumption from the HVAC(**Heating, Ventilation, and Air Conditioning)** system and lighting requirements in a typical building range between 45 to 55% and 20 to 35% from its operational energy consumption respectively.

Despite having simple energy saving techniques, the world has taken a move towards advanced building energy management systems. Accordingly, the implementation of A combination of various technologies under the title of “ Building Automation”, “Smart buildings” or “Building Management System”(BMS) are increased. BMS is an integrated management system, which is able to display information and arrange all smart controlling systems in a short time. In recent times researchers have suggested that CCTV systems as an improved way of object detection, tracking and counting in real-time for energy conservation in open building spaces despite the common goal of using CCTV systems for the prevention of crime and disorder by tracking and observation. However, this approach still has not been converted into a system design or implemented within buildings for the purpose of energy conservation. Occupant behaviour is a crucial factor in determining energy use in buildings. With this motivation, we have introduced a novel building energy conservation system based on real-time object detection, tracking and counting using CCTV cameras in order to tackle several of these needed advancements. On detecting the absence of human faces, the building manager is notified or the BMS automatically turns off the electrical appliances in the building.

Depending on the count of people in the room, if the energy used is more than required, reducing the number of appliances in use is a way to reduce energy consumption( eg. For 10 people in a room 2AC’s and 4 lights are required but if there are only 4 people 1Ac and 2 lights is more than enough) .Implementation of this not only helps in saving energy but also significantly reduces the cost spent on energy consumption.

## **Data engineering:**

Data ingestion takes place from either of the two data sources, namely YouTube URL or camera streaming URL. According to the selection, the URL is verified for its correctness.

If the given URL is valid, then it starts streaming the video using OpenCV.

Then we need to upload a CSV file which contains the details of the customers whose name can be identified by the model.

## **Data storage:**

Simultaneously, the captured frames are encoded and stored in the MySQL database.

After processing(which is explained below), the details of the people matched with the csv details and detected face are stored in the database.

## **Model engineering:**

We used DeepFace open-source library to load required models. Now the captured frames are processed to identify the name, race and gender count in each frame.

## **Analysis:**

Now it checks whether any frame doesn’t contain people, if so an alert mail or SMS would be sent to the corresponding Mail ID or phone number.