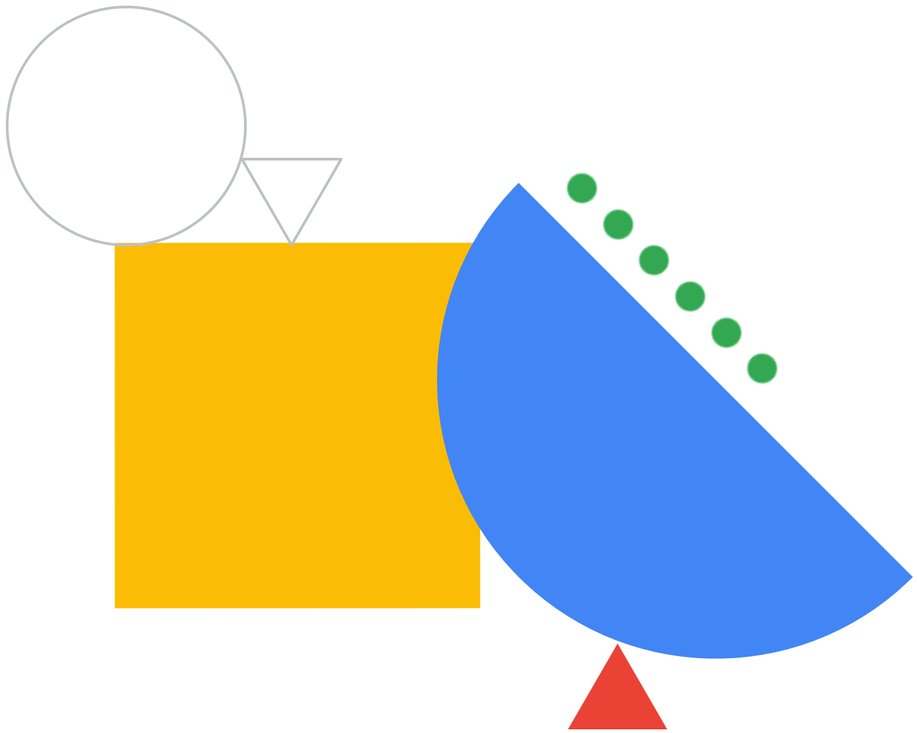
Google Cloud

The Machine Learning Workflow with Vertex AI



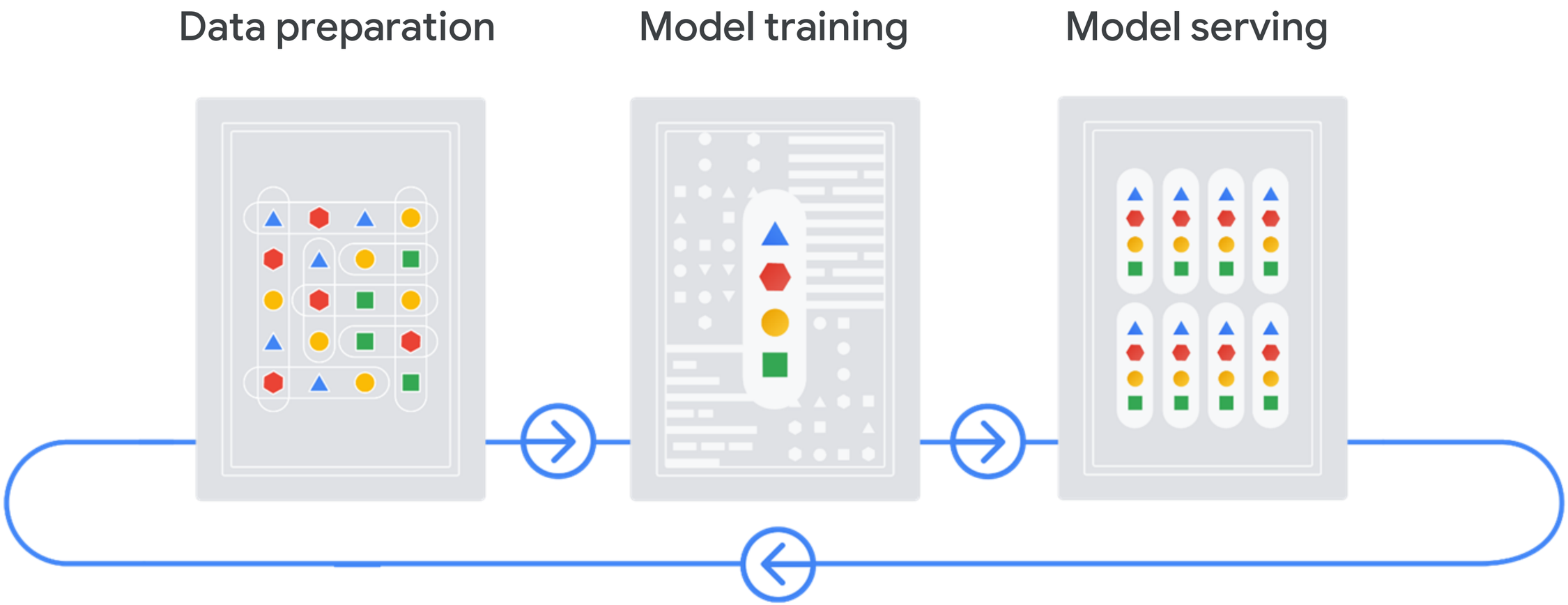
01

Introduction

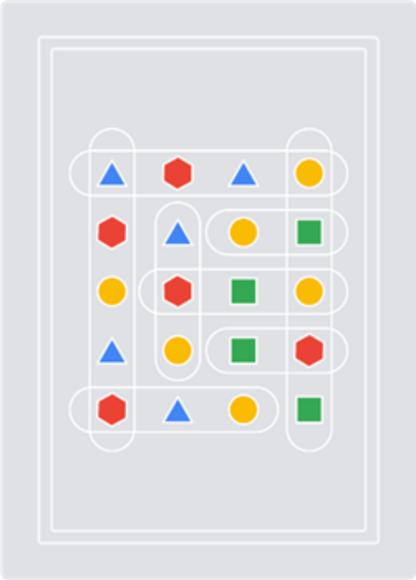
Agenda

- The three stages of the ML workflow
- Data preparation
- Model training
  - Model training
  - Model evaluation
- Model serving
- Hands-on lab

The machine learning workflow



Stage 1: Data preparation

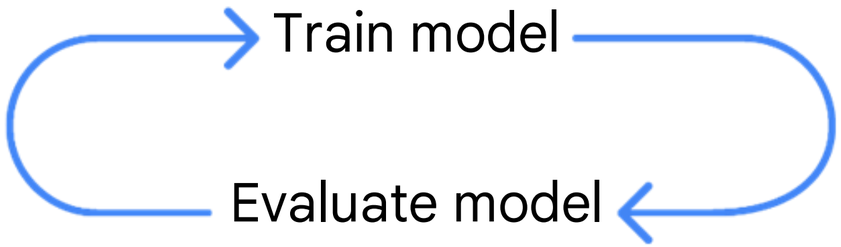


A model needs a large amount of data to learn from.

- Data collection
- Exploratory Data Analysis
- Data Cleaning and Preprocess
- Feature Engineering

Stage 2: Model training

A model needs a tremendous amount of iterative training.



Stage 3: Model serving

A model needs to actually be used in order to predict results.

- Deployed
- Monitored
- Managed

It’s similar to serving food in restaurant

Prepare raw ingredients

Experiment with recipes

Serve the meal

Data preparation

Model training

Model serving

Vertex AI is Google's unified AI platform

Vertex AI

1. AutoML: No-code solution
2. Custom training: Code-based solution

02

Data preparation

Data preparation

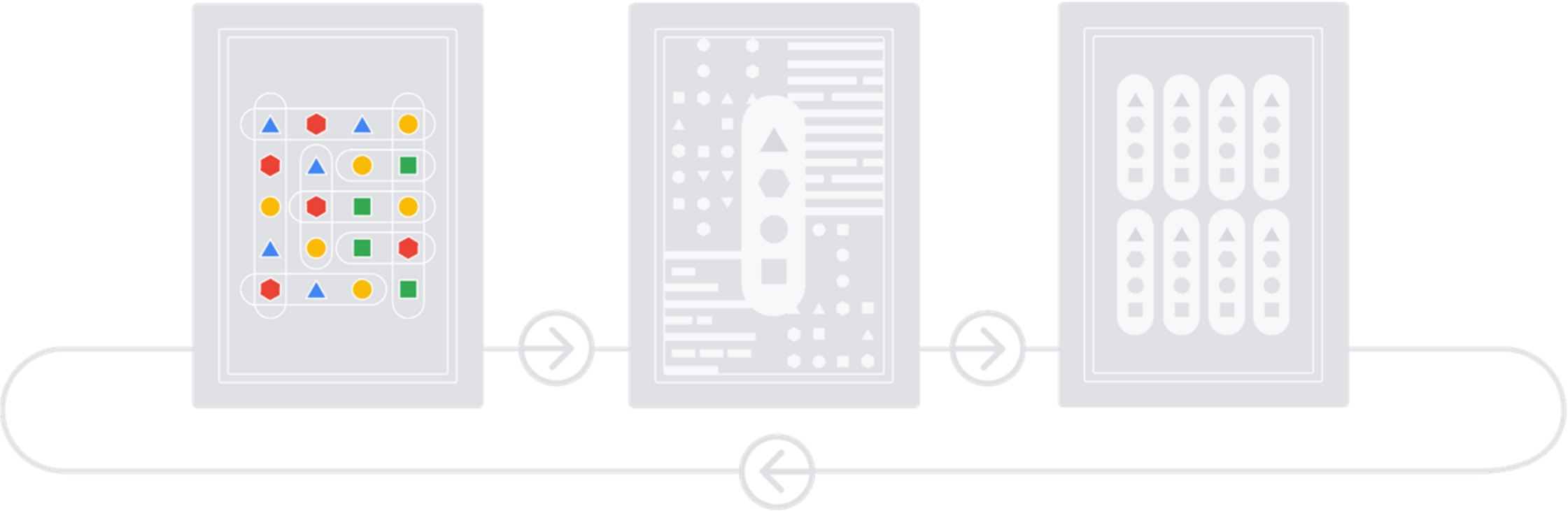
Prepare raw ingredients

01 Data Collection

02 EDA

03 Data Cleaning

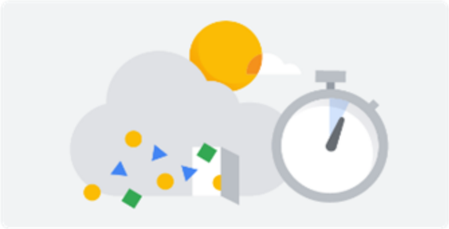
04 Feature Engineering



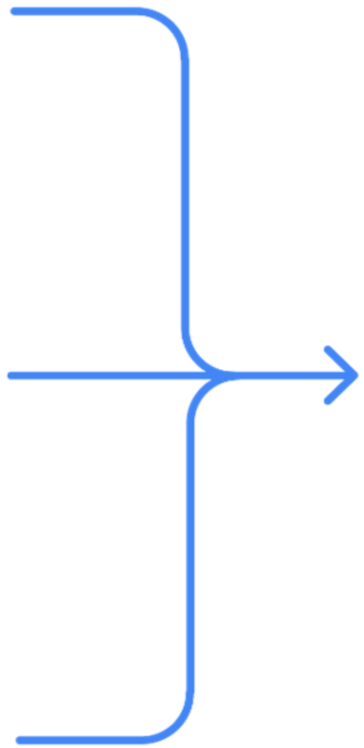
01 Data Collection (Big Query)

Batch load

Streaming



Generated data



BigQuery

01 Data Collection (Big Query)

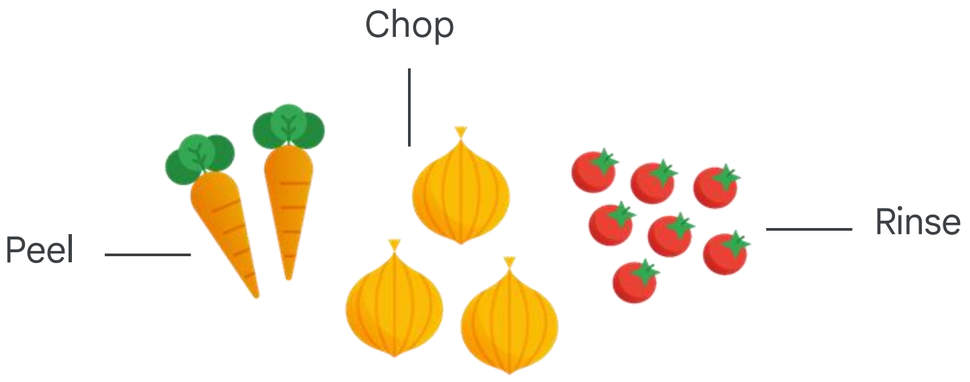
02 Exploratory Data Analysis (Vertex AI Workbench)

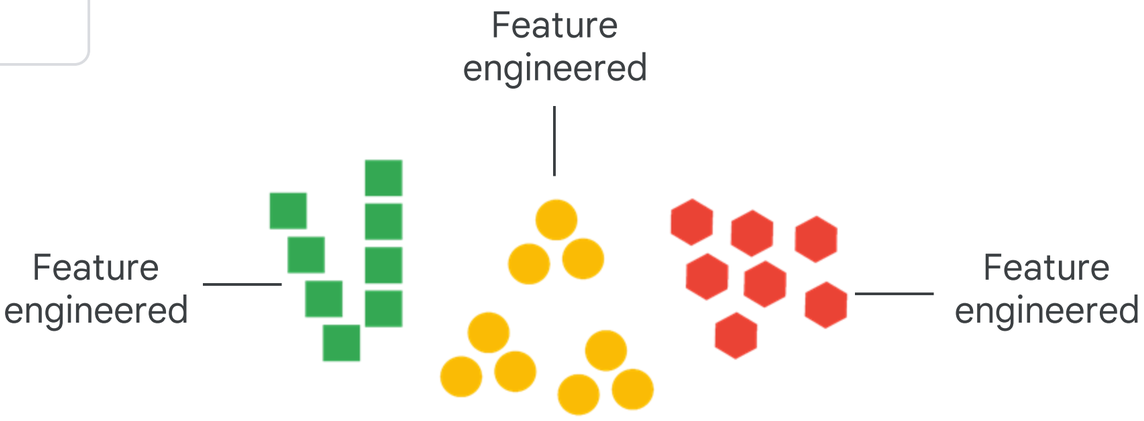
03 Data Cleaning and Preprocess (Vertex AI Workbench)

04 Feature Engineering (Vertex AI Workbench)

A feature is a factor that may contribute to the prediction.

- An independent variable in statistics
- A column in a table





03

Model training

Model training

Data preparation

Prepare raw ingredients

Model training

Experiment with recipes

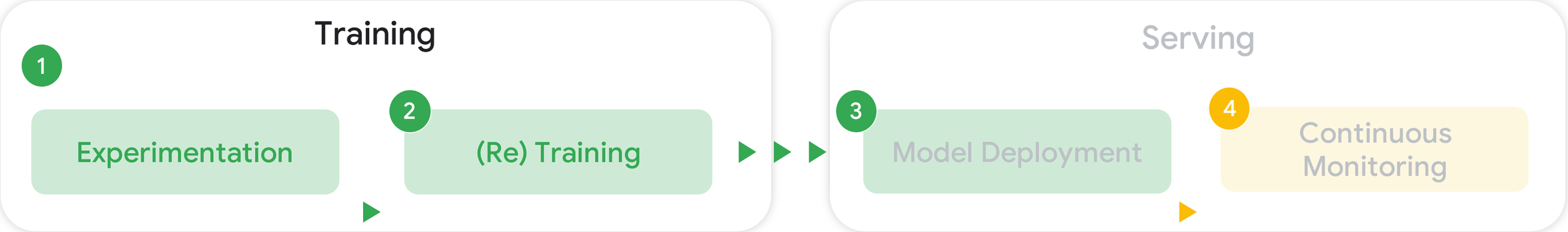
01 Model training

02 Model evaluation

05 Model Training and Evaluation

06 Model Pipeline and Deployment

07 Model Monitoring (Looker Studio)



ML models

Supervised learning

Task-driven and identifies a **goal**

Past data to predict future trends

Classification

Predicts a categorical variable

Use an image to tell the difference between a cat and a dog

Regression

Predicts a continuous number

Use past sales of an item to predict a future trend

Unsupervised learning

Data-driven and identifies a **pattern**

Group customers together

Clustering

Groups data points together

Use customer demographics to determine customer segmentation

Association

Identifies underlying relationships

Correlation between two products to place them closer in a grocery store

Dimensionality reduction

Reduces the number of dimensions

Combining characteristics to create a quote

Which ML options need to specify ML models?

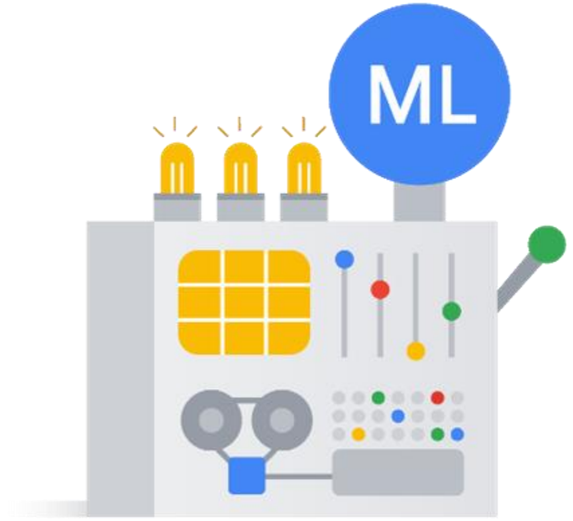
No need to specify

AutoML

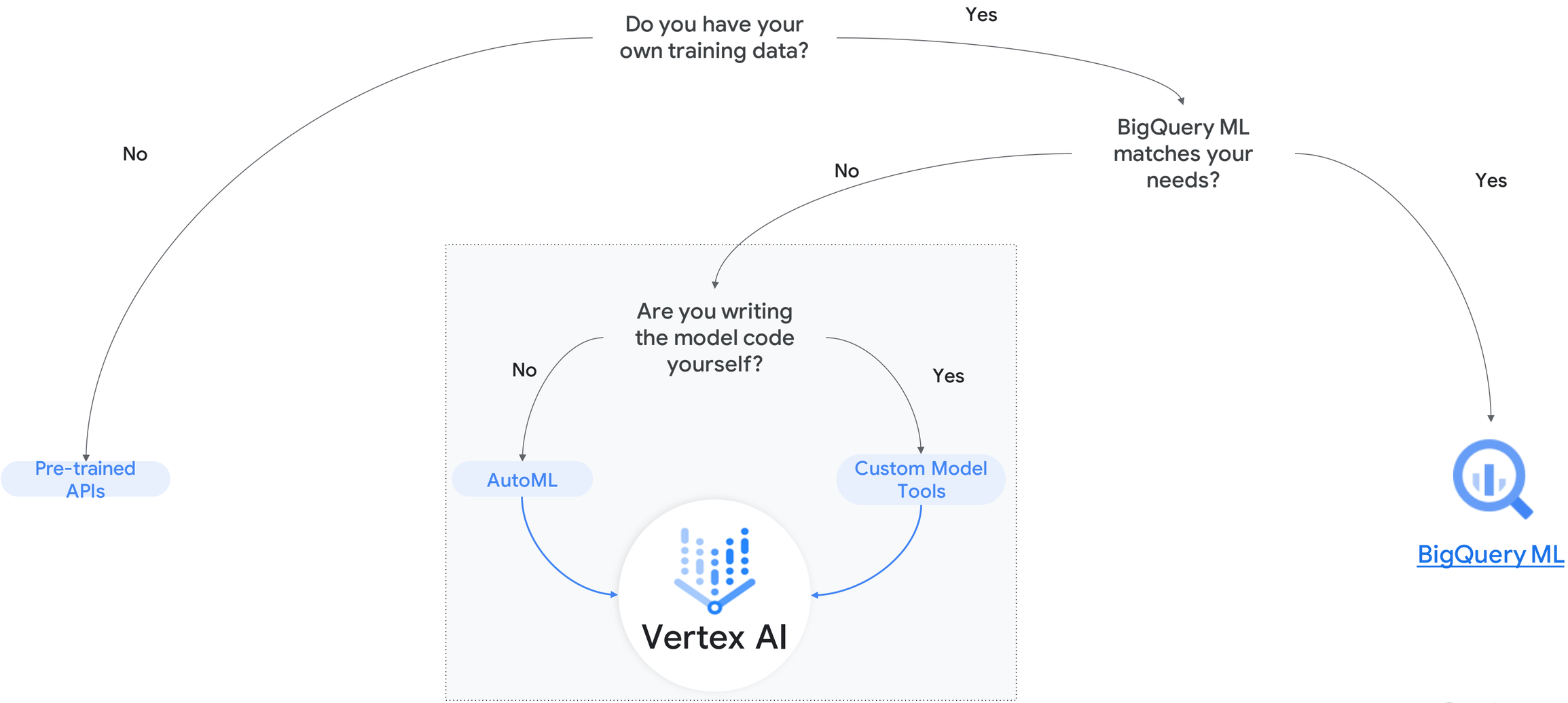
Pre-built APIs

Need to specify

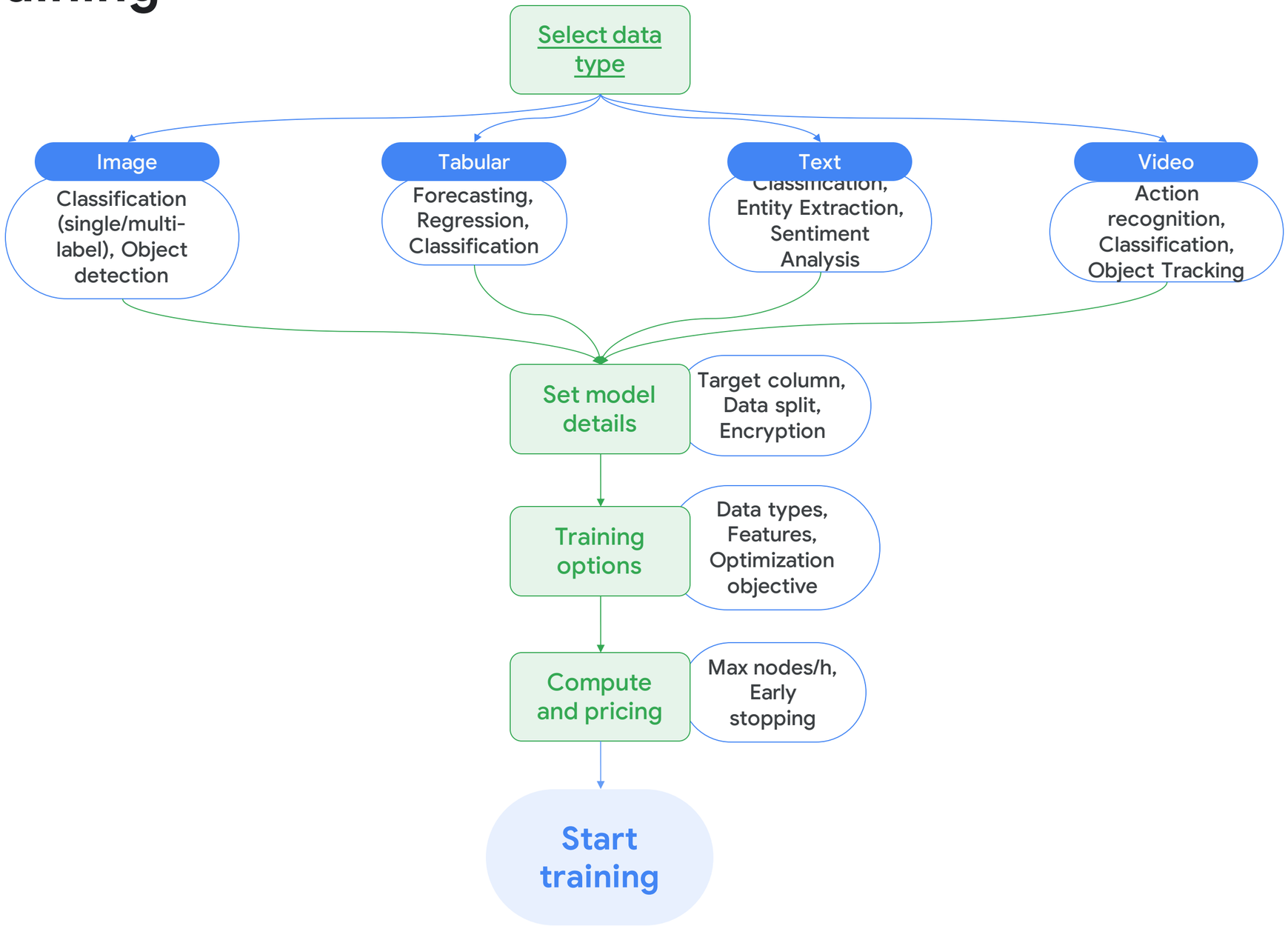
BigQuery ML
Custom training



Which training tool is right for you?

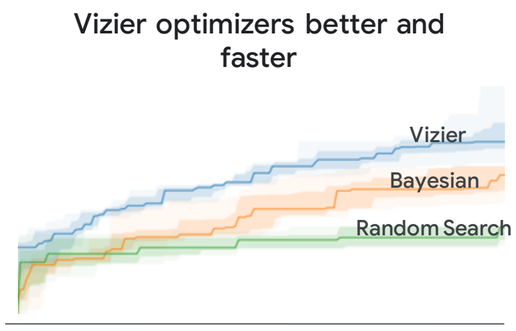


AutoML training



How to train a custom model on Vertex AI

```
├── setup.py   # Package setup script
└── trainer    # Model trainer package
    ├── __init__.py
    └── task.py  # Model trainer entry point
```



Training Code

Select ML framework

Prebuilt Containers

TensorFlow
learn
PyTorch
dmlc XGBoost

Custom containers

Model artifacts Cloud Storage

Hyperparameter tuning (optional)

Automatic Hyperparameter tuning

Vizier Optimizer

Compute Resources

Worker Pool Specs

Accelerators

Boot Disk (optional)

Create a custom training job

Supported regions

04

Model evaluation

Model evaluation

Data preparation

Prepare raw ingredients

Model training

Experiment with recipes

01 Model training

02 Model evaluation

Evaluation metrics

Vertex AI

Evaluation metrics

- Confusion
- Recall
- Precision
- Feature importance

Confusion matrix

Actual values \ Predicted values	Positive (cat)	Negative (dog)
Positive (cat)	**True positive**  This is a cat.	**False negative**  This is a dog. Type 2 error
Negative (dog)	**False positive**  This is a cat. Type 1 error	**True negative**  This is not a cat.

Recall and precision

Actual values	Predicted values: Positive	Predicted values: Negative
Positive	True positive (TP)	False negative (FN)
Negative	False positive (FP)	True negative (TN)

$$\text{Recall} = \frac{TP}{TP+FN}$$

Recall

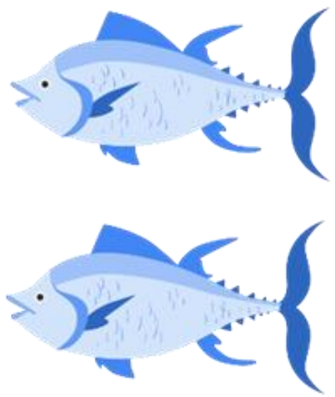
Refers to all the positive cases, and looks at how many were predicted correctly.

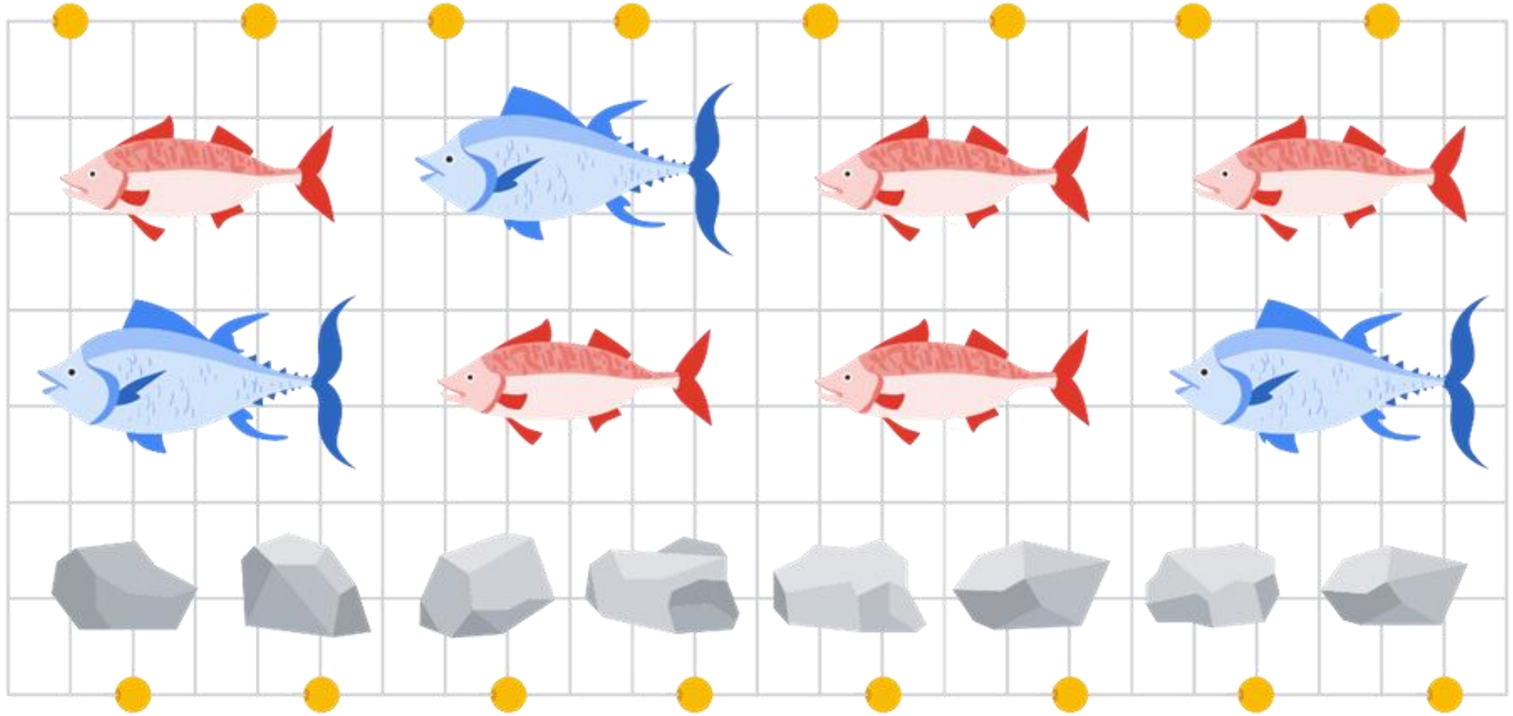
Precision

Refers to all the cases predicted as positive, and how many are actually positive.

$$\text{Precision} = \frac{TP}{TP+FP}$$

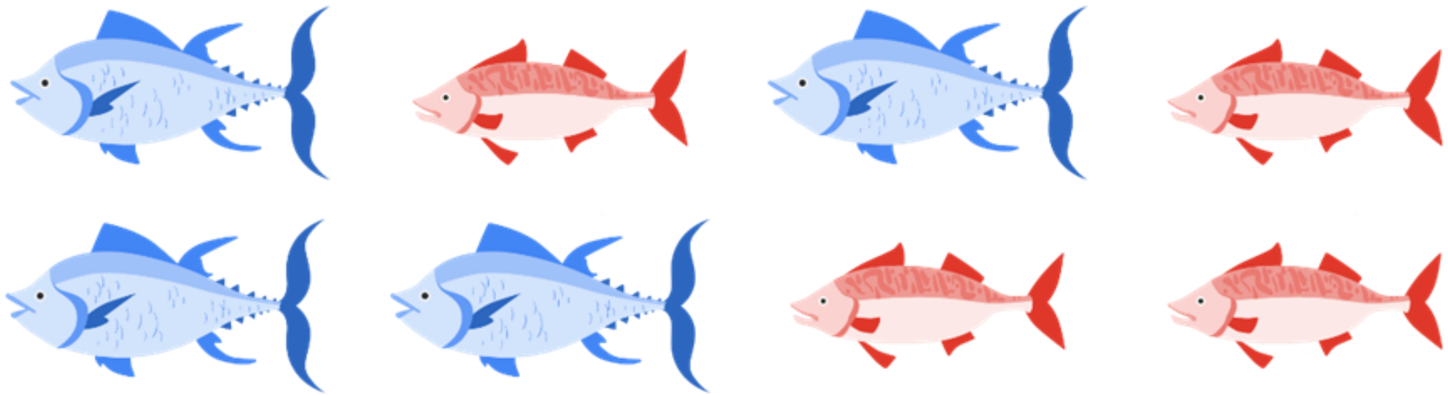
Example: Fishing with a wide net

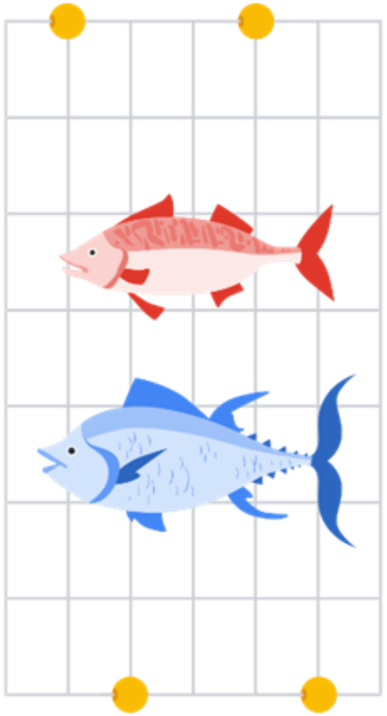




Wide net	Total fish in the lake: 100 Caught: 80 fish + 80 rocks	Recall: 80%	Precision: 50%

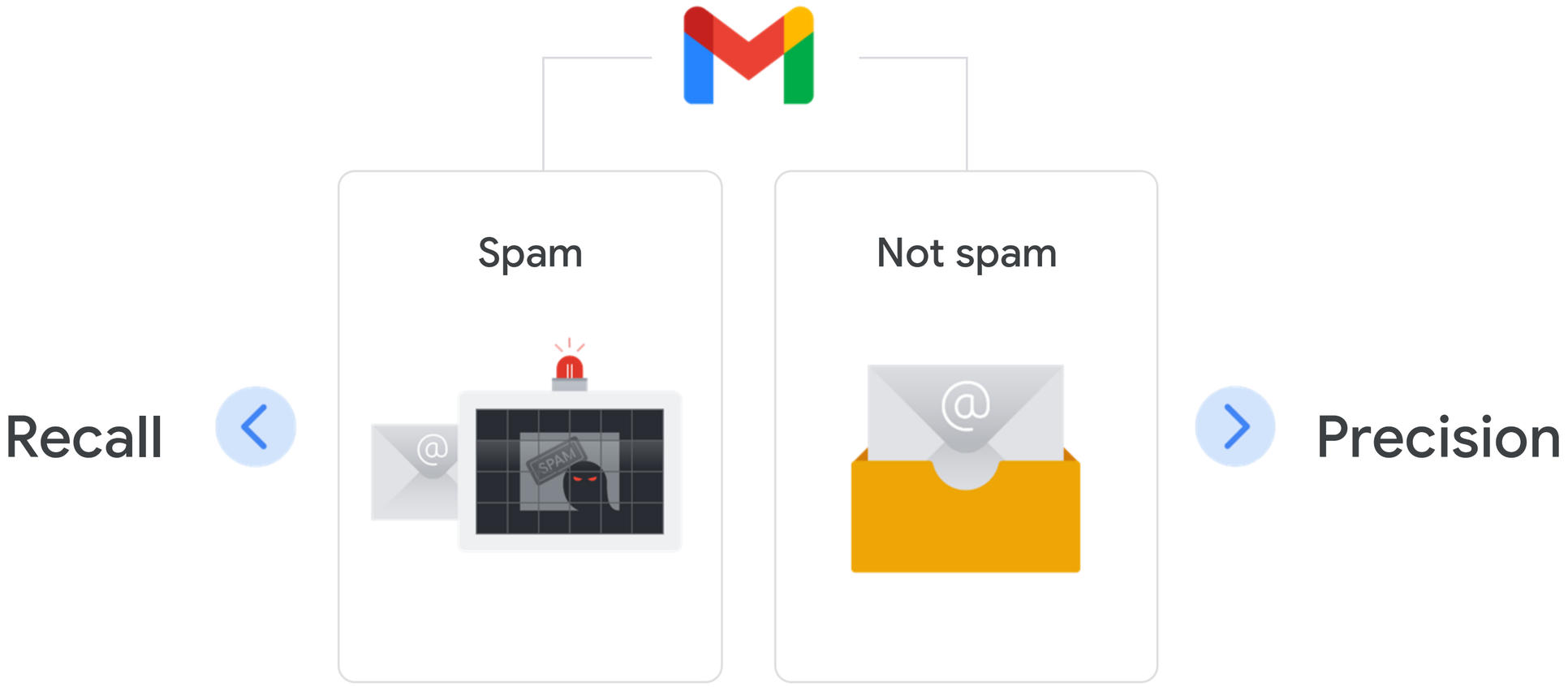
Example: Fishing with a smaller net





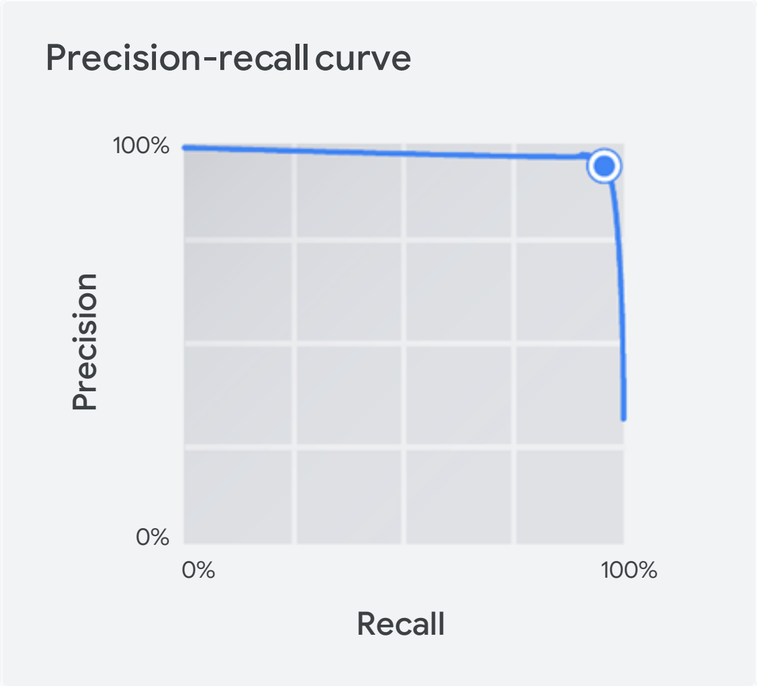
Smaller net	Total fish in the lake: 100 Caught: 20 fish	Recall: 20%	Precision: 100%

The trade-off between recall and precision

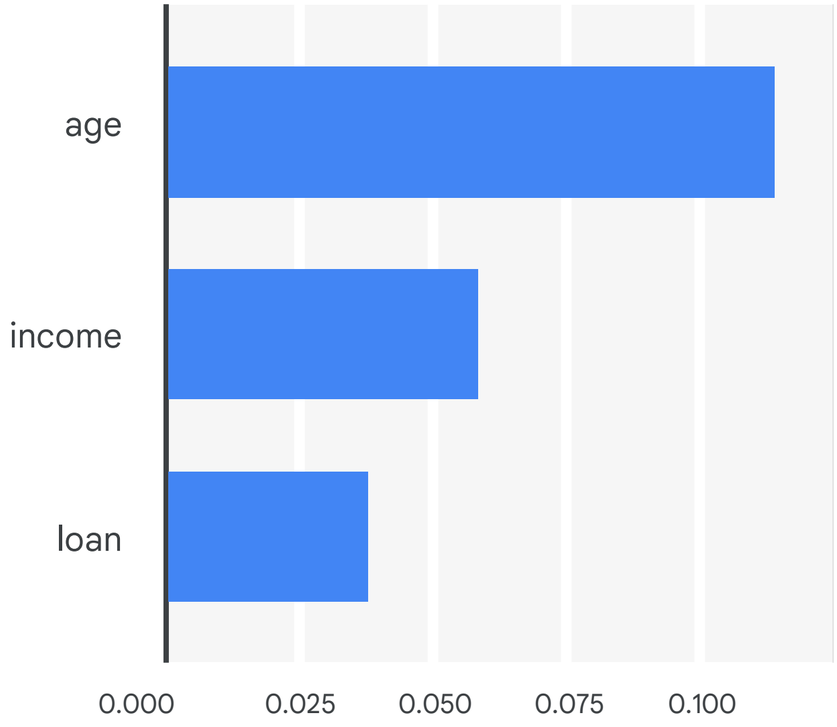


The precision-recall curve

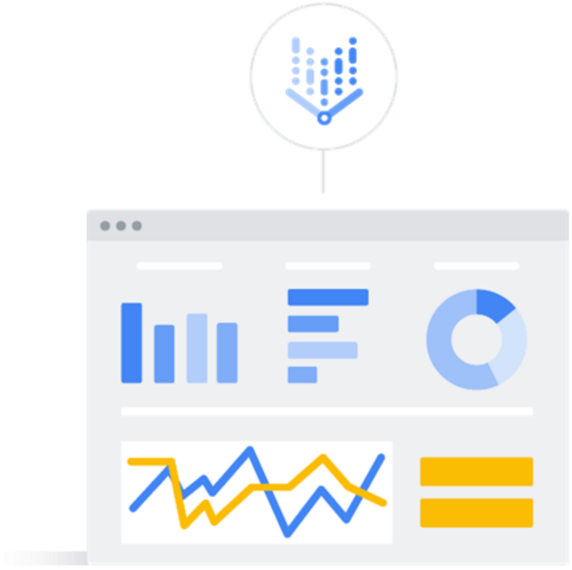
Vertex AI



Feature importance



Feature importance identifies how each feature contributes to a prediction.



Explainable AI

05

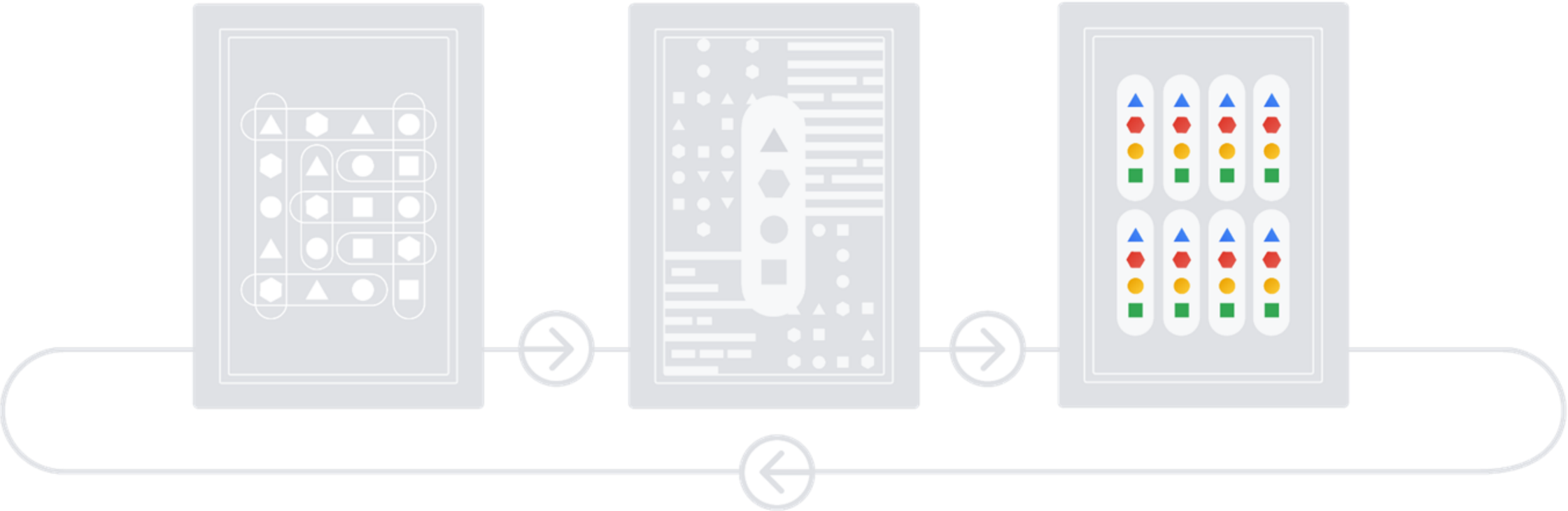
Model deployment and monitoring

Model serving

Serve the meal

01 Model deployment

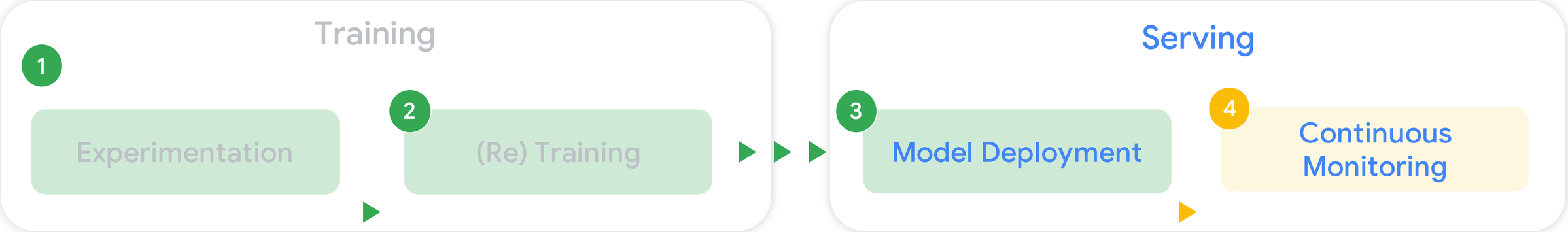
02 Model monitoring



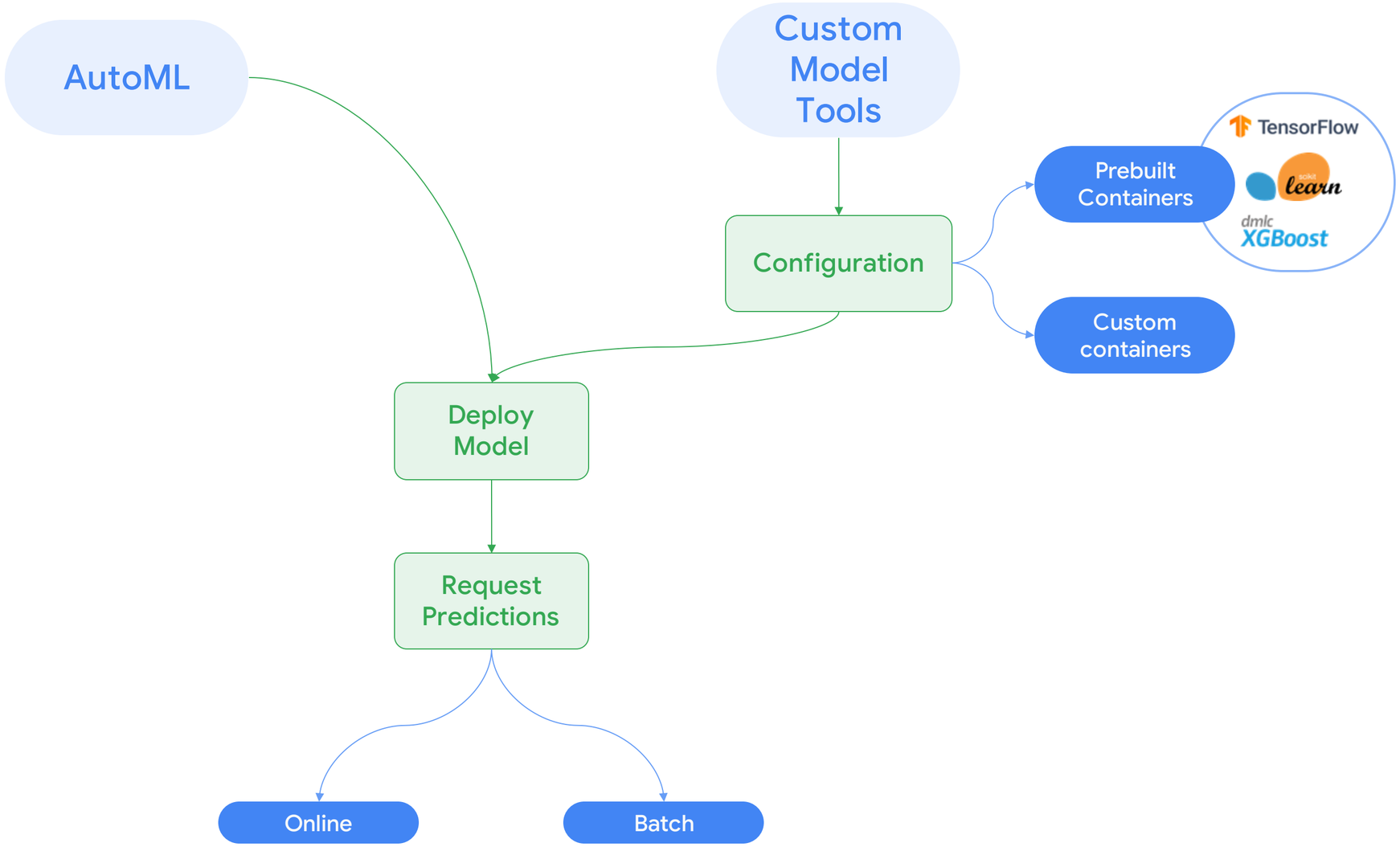
05 Model Training and Evaluation

06 Model Pipeline and Deployment

07 Model Monitoring (Looker Studio)



How to use Vertex AI predictions



Model deployment is when the model is implemented

01

Model deployment

Three ML deployment options

Endpoint

Best when immediate results with low latency are needed.

Must be deployed to an endpoint before that model can be used to serve real-time predictions.

Batch prediction

Best when no immediate response is required, and accumulated data should be processed with a single request.

Offline prediction

Best when the model should be deployed in a specific environment off the cloud.

Model monitoring

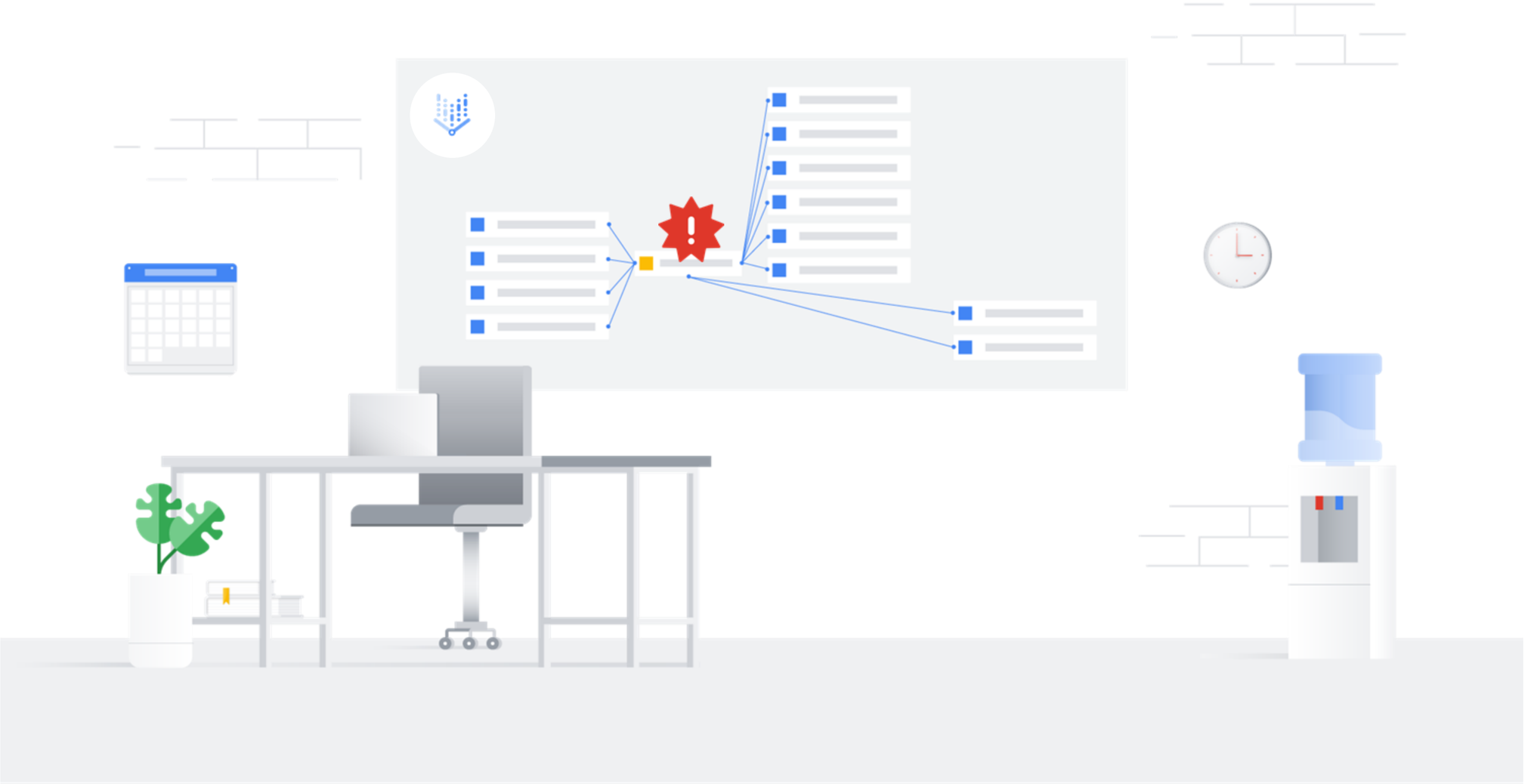
02

Model monitoring

Vertex AI Pipelines

- Automate
- Monitor
- Govern

Vertex AI Pipelines is similar to a production line



You can define pipelines using Vertex AI Workbench

Vertex AI Workbench

Define your own pipeline with prebuilt pipeline components

Vertex AI workflow

01 Data Collection (Big Query)

02 Exploratory Data Analysis (Vertex AI Workbench)

03 Data Cleaning and Preprocess (Vertex AI Workbench)

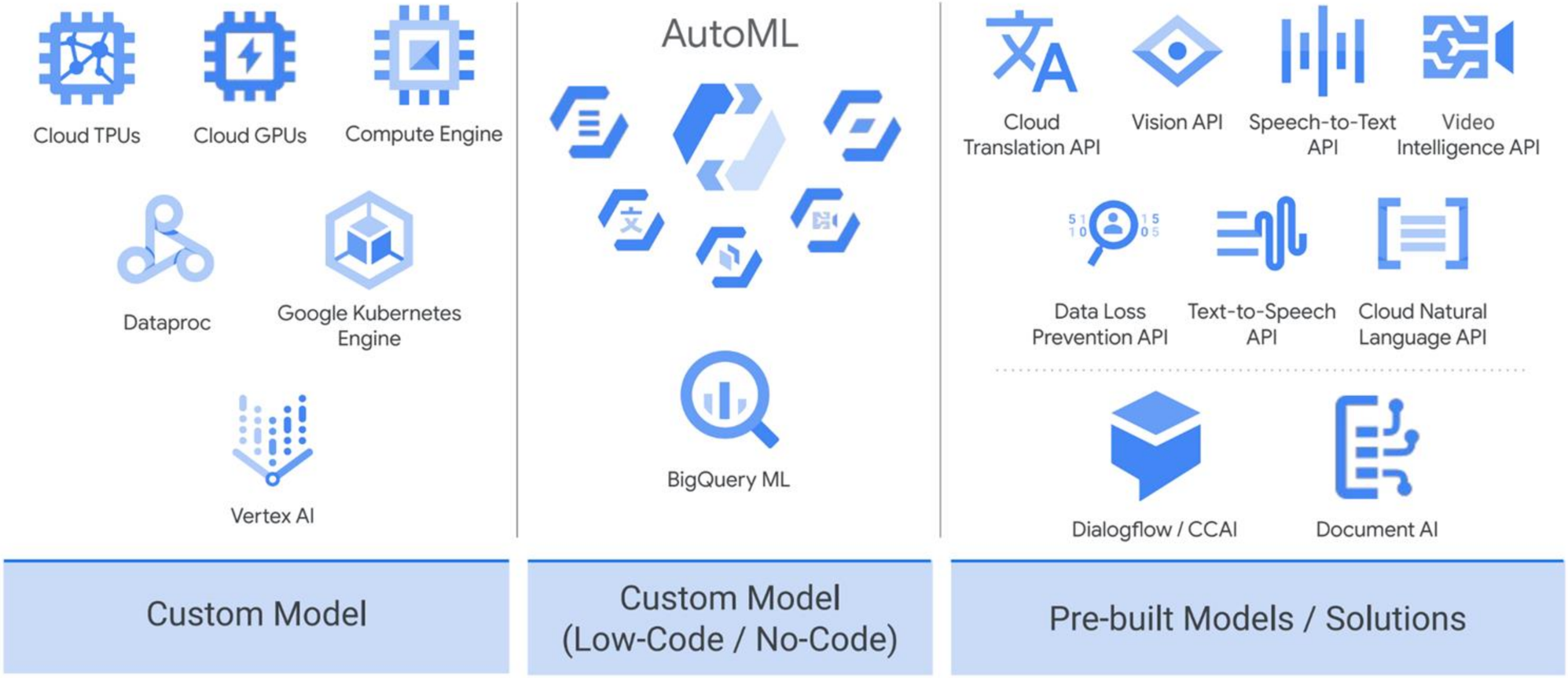
04 Feature Engineering (Vertex AI Workbench)

05 Model Training and Evaluation (Pipelines + Workbench)

06 Model Pipeline and Deployment (Vertex AI Pipelines)

07 Model Monitoring (Looker Studio)

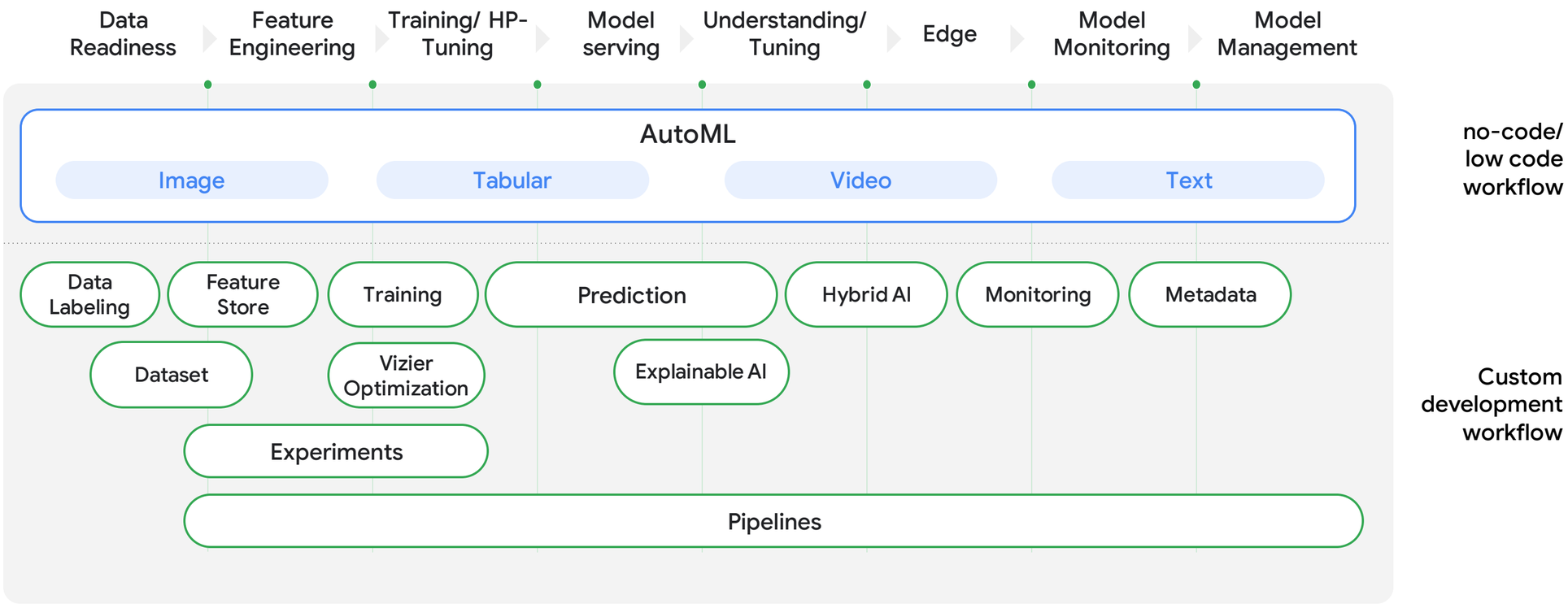
Machine Learning on Google Cloud



Proprietary + Confidential

One comprehensive end-to-end platform for everything AI

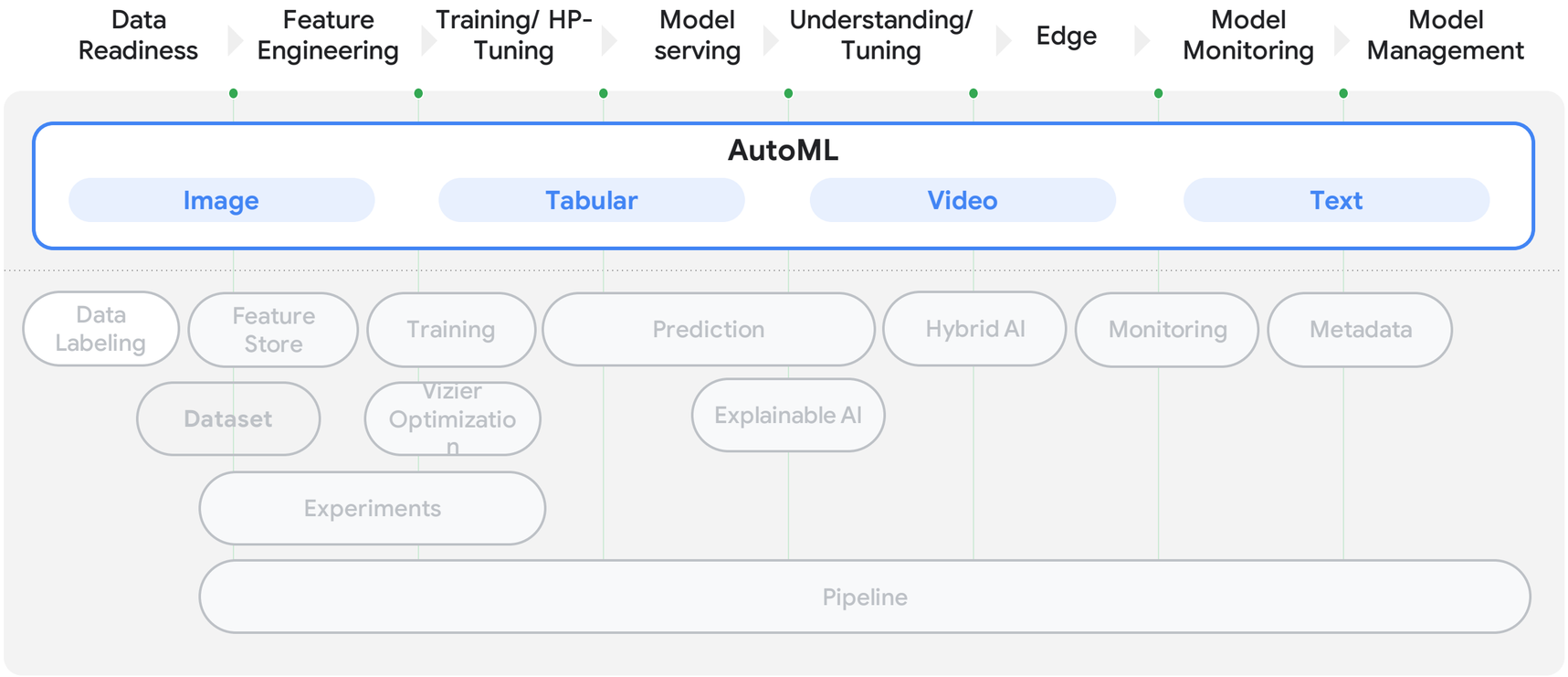
Vertex AI



Proprietary + Confidential

AutoML

Vertex AI



no-code/ low code workflow

Custom development workflow

Use case

Train models with 4 data types:
- Image
- Text
- Table
- Video

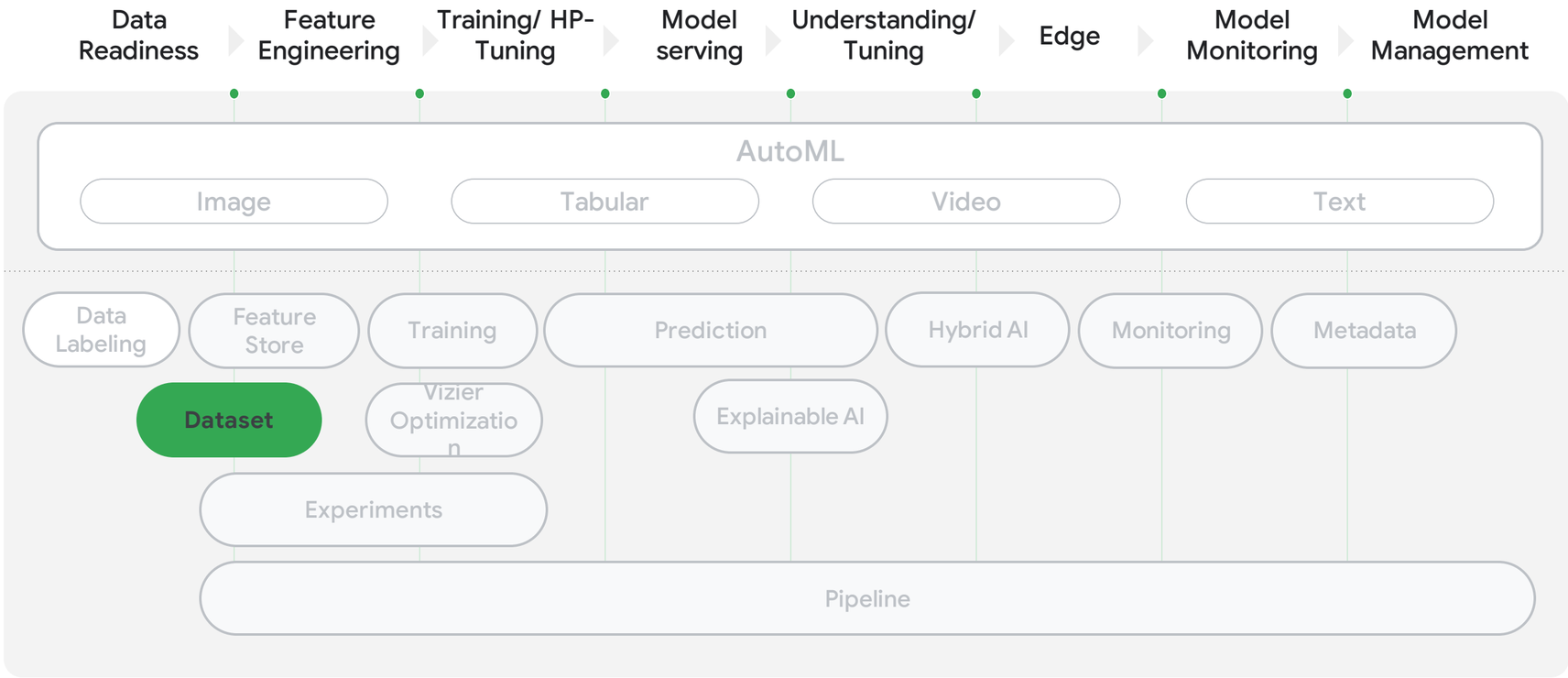
Features

- Automatic architecture search with Google's SOTA algorithms

Proprietary + Confidential

One comprehensive end-to-end platform for everything AI

Vertex AI



no-code/ low code workflow

Custom development workflow

Use case

Managed dataset for AutoML

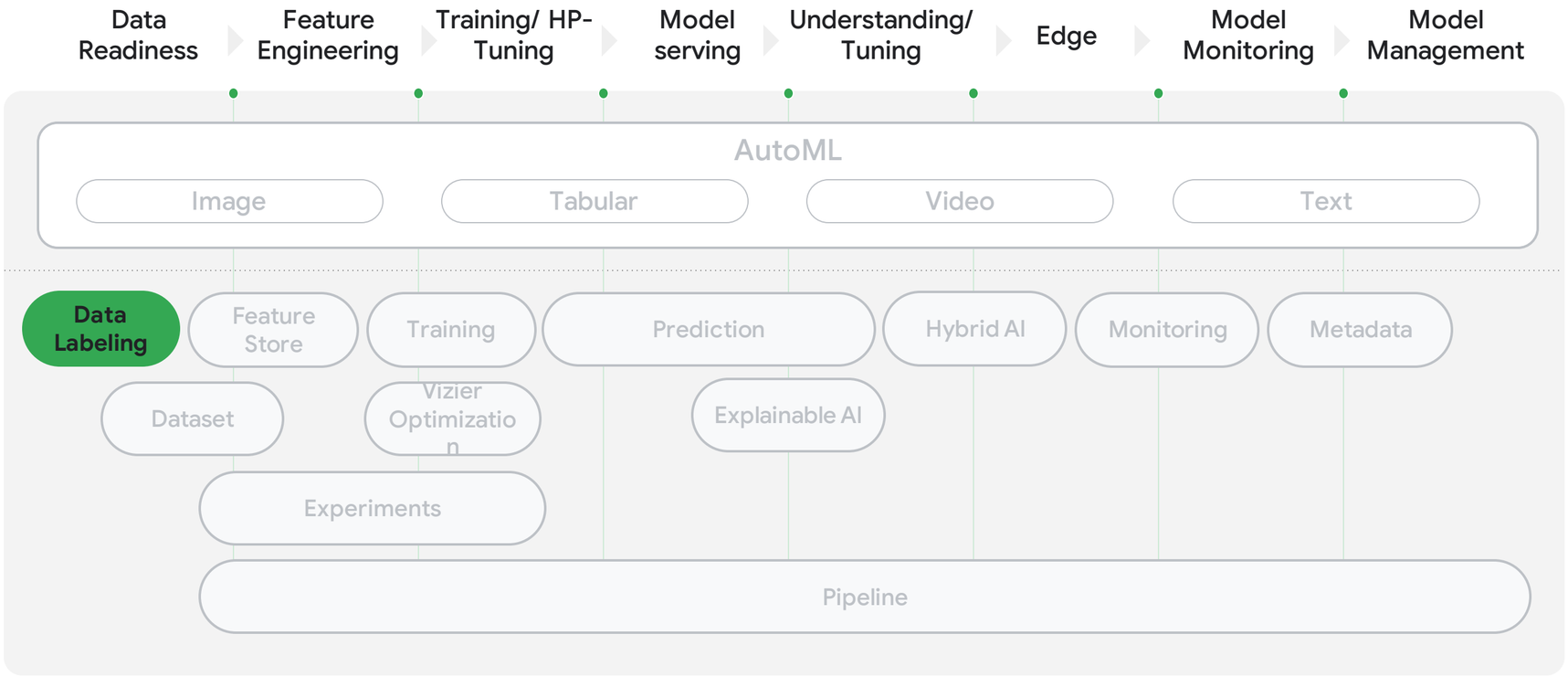
Features

- Import batch data from GCS
- 4 types of data:
  - Image
  - Text
  - Table
  - Video

Proprietary + Confidential

Vertex AI Data Labelling

Vertex AI



no-code/ low code workflow

Custom development workflow

Use case

Label Vertex AI managed dataset

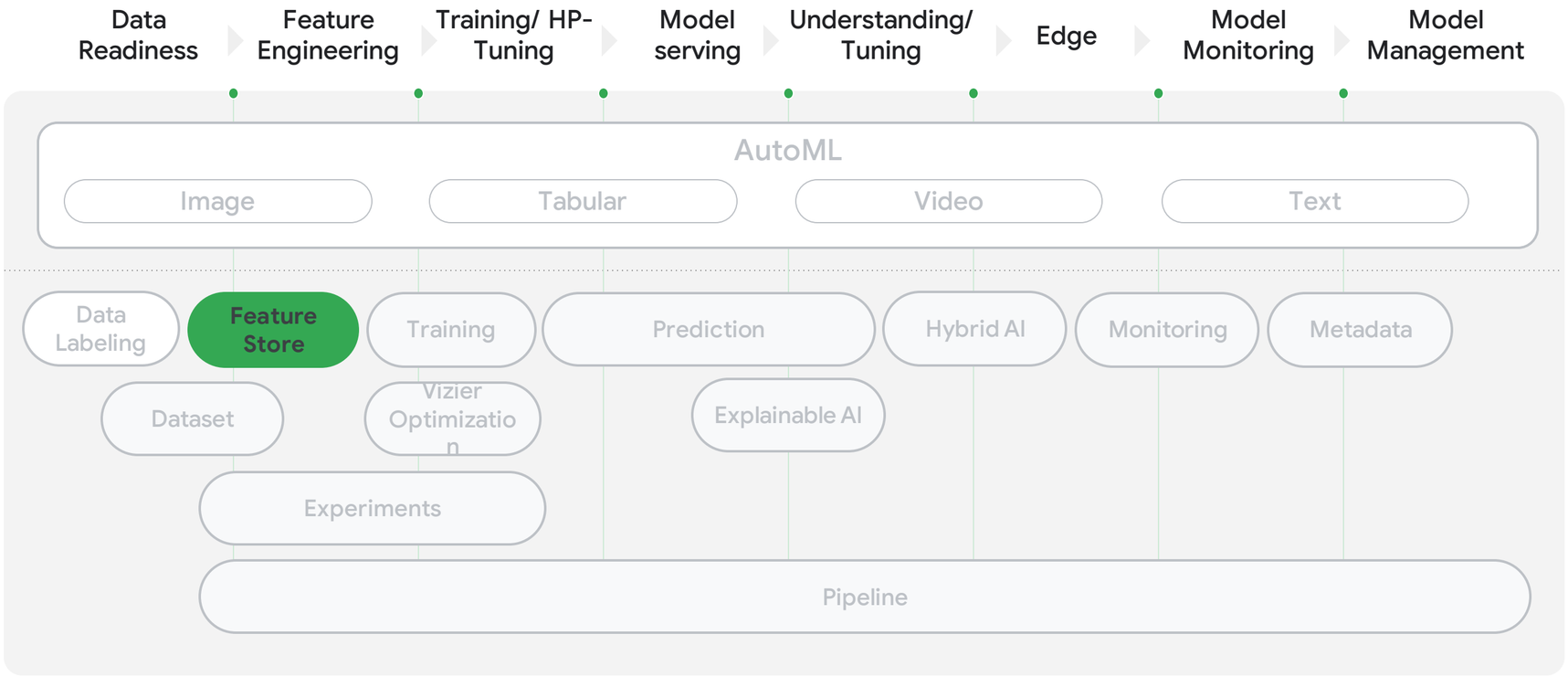
Features

- Manage data labeling tasks
- Assign for Google's labelers

Proprietary + Confidential

Vertex AI Feature Store

Vertex AI



no-code/ low code workflow

Custom development workflow

Use case

Centrally store feature values (tabular data)

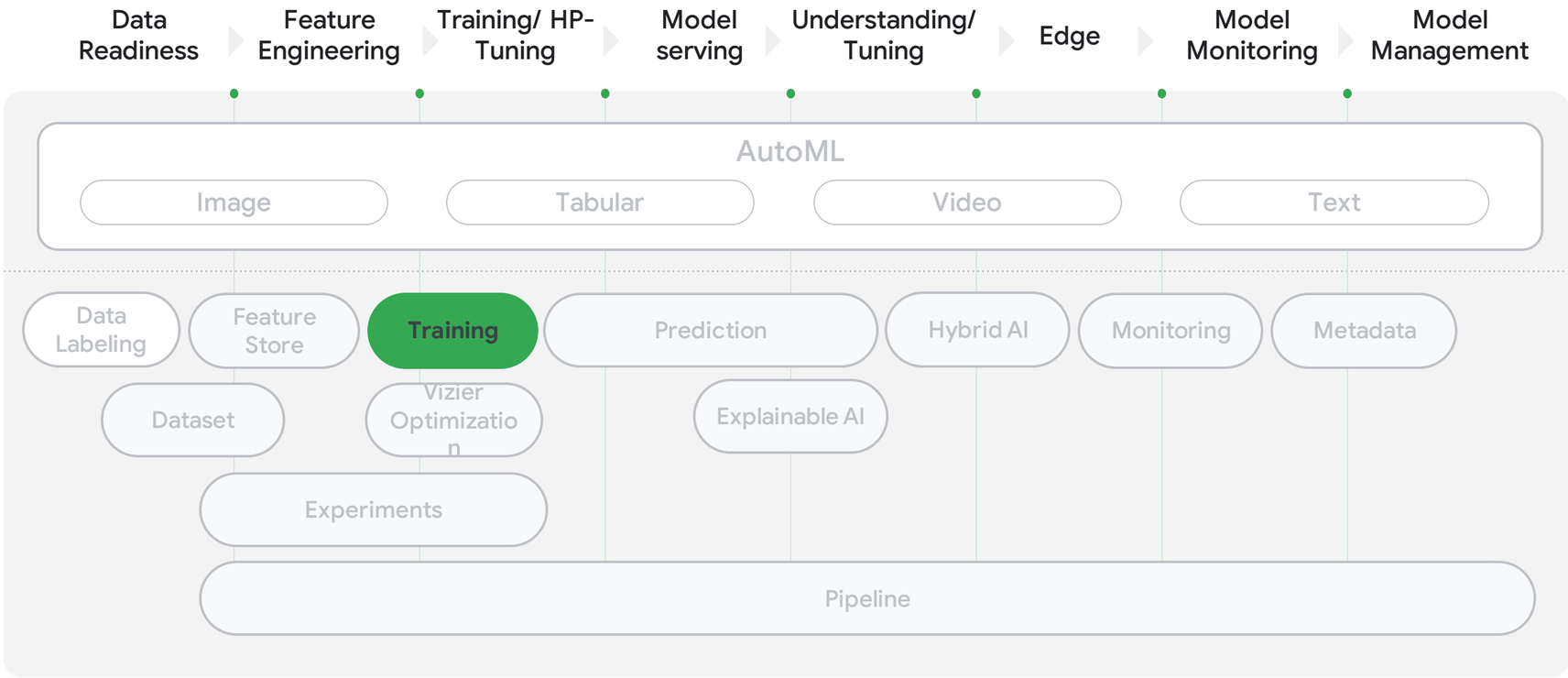
Features

- Batch serving for training
- Streaming serving for online prediction
- Export to archive feature values

Proprietary + Confidential

Vertex AI Training

Vertex AI



no-code/ low code workflow

Custom development workflow

Use case

Train models

Features

- Train with pre-built containers:
  - TensorFlow
  - PyTorch
  - scikit-learn
  - XGBoost
- Train with custom containers

06

Lab: Deploy a BigQuery ML Customer Churn Classifier to Vertex AI for Online Predictions

07

Summary

The three stages of the ML workflow

Prepare raw ingredients

Data preparation

Experiment with recipes

Model training

Serve the meal

Model serving

Summary

1 Data preparation

- Data Collection
- EDA
- Data Cleaning

2 Feature engineering

Model training

3 Model training

- Model evaluation

Model serving

- Model deployment
- Model monitoring

08

AI/ML usecase

Customer segment | Customer Lifetime Value

Purpose	Divide the customer base into distinct groups based on their characteristics, behaviors, needs, and preferences to better understand and serve customers
Algorithms	K-means, Hierarchical Clustering, Gaussian Mixture Models
Feature Engineering	Customer Demographic, Customer Trading Transaction.
Metrics	Silhouette Score
Customer Lifetime Value (CLV)	Assess the potential value of each customer segment by estimating their CLV. Identify high-value segments that contribute significantly to revenue and prioritize resources accordingly

Recommendation Engine - Association Rule

Purpose	**Customer like you also buy** - The recommendation engine can suggest securities to investors based on the preferences and investment decisions of similar investors
Algorithms	Collaborative filtering, matrix factorization, and deep learning models
Feature Engineering	Customer Demographic, Customer Transaction History, Stock Data
Metrics	Recall , Precision , Hit rate
Real-Time Recommendation	Integrate real-time market data, user data into the recommendation engine to ensure improving their accuracy and timeliness

Churn prediction

Purpose	Churn prediction refers to the process of using data analytics and predictive modeling techniques to identify customers or clients who are at a high risk of discontinuing their business relationship with the company
Algorithms	Gradient Boosting Tree, Logistic Regression, Tree based method
Feature Engineering	Customer Transaction History, Customer Demographic, Customer Service Account History (optional).
Metrics	Recall, Precision, F1- score
Retention Strategies	Implement targeted retention strategies for high-risk customers identified through churn prediction