

Embedded Machine Learning for Edge Computing

ML Pipeline, Model Evaluation and Deployment

Sahan Hemachandra

Aug/2024



- About me

- Graduated from Department of Electronic and Telecommunications Engineering in 2021, specializing in computer vision.
- Authored papers several papers in computer vision conferences and a journal.
- Previously worked as a Senior Software Engineer at WSO2
- Currently working as a Research Assistant at Mohamed bin Zayed University of Artificial Intelligence, Abu Dhabi



What will you Learn?

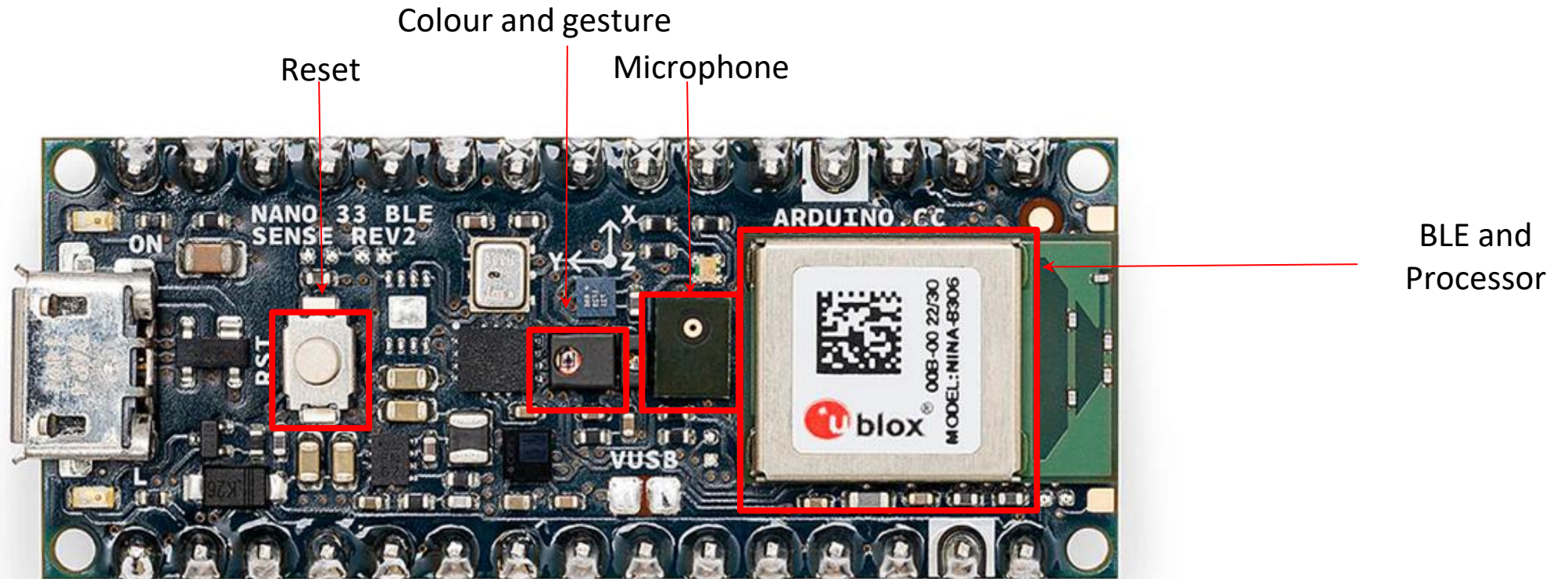
1. How can we deploy Embedded AI solutions (Theory)?
 - A. Required devices
 - B. Training process for a model
 - C. How to optimize models for embedded devices
 - D. Porting and running models



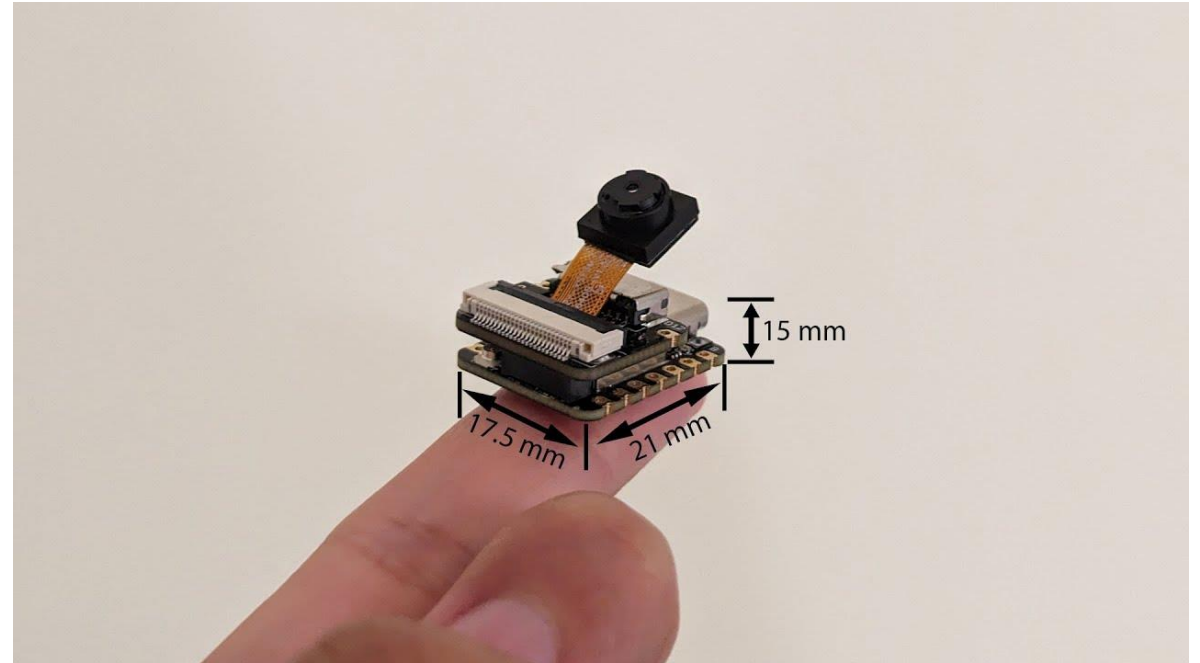
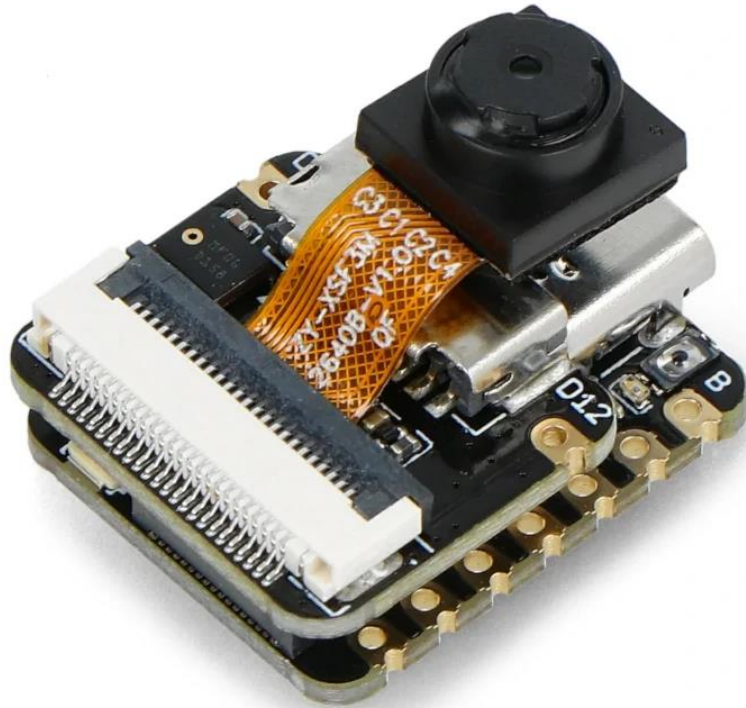
RECAP



Arduino Nano BLE

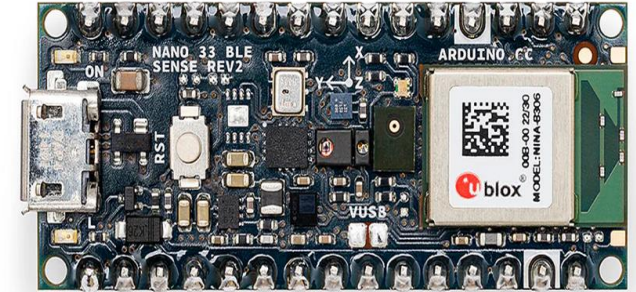


XIAO ESP32S3

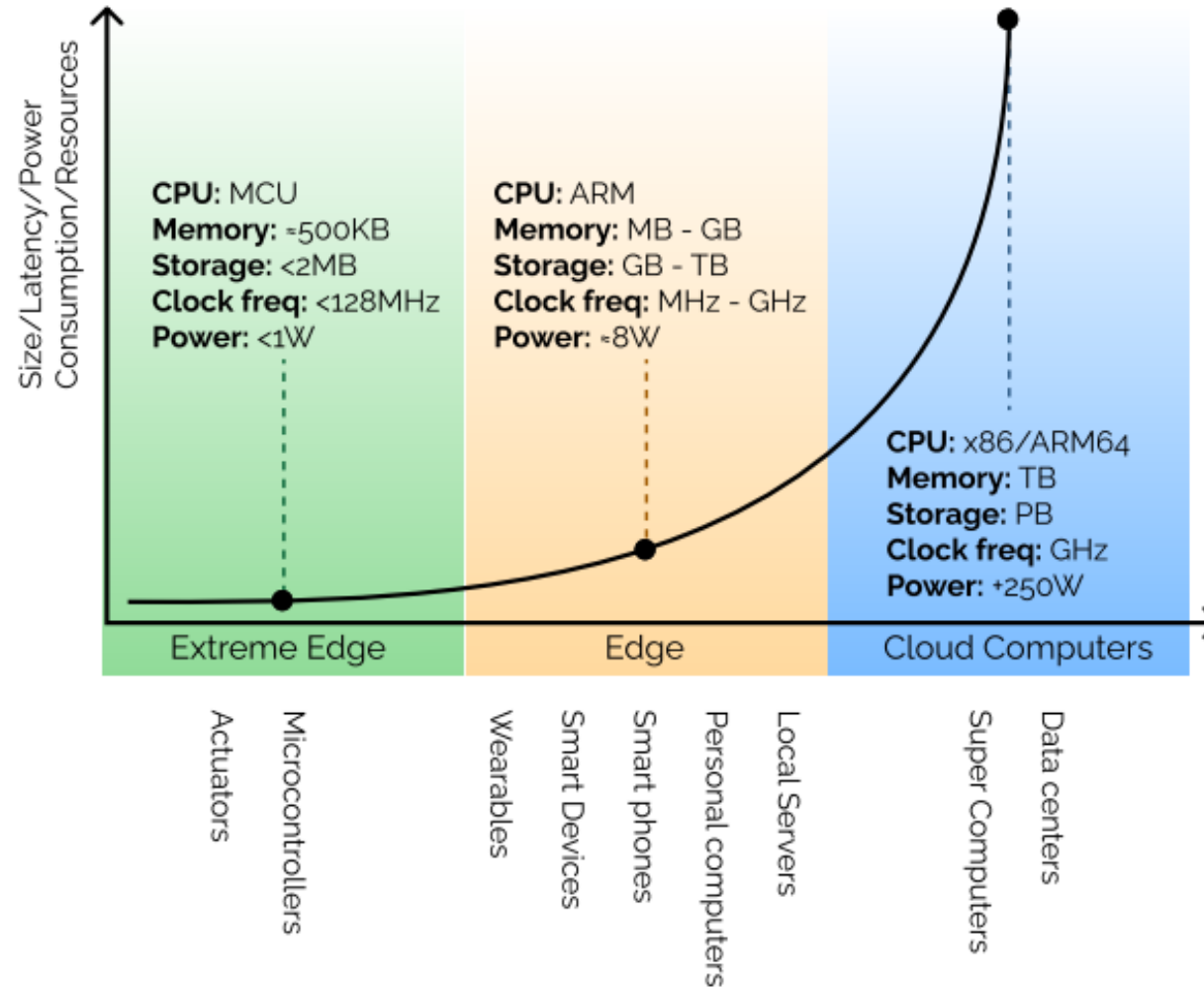


Required Devices

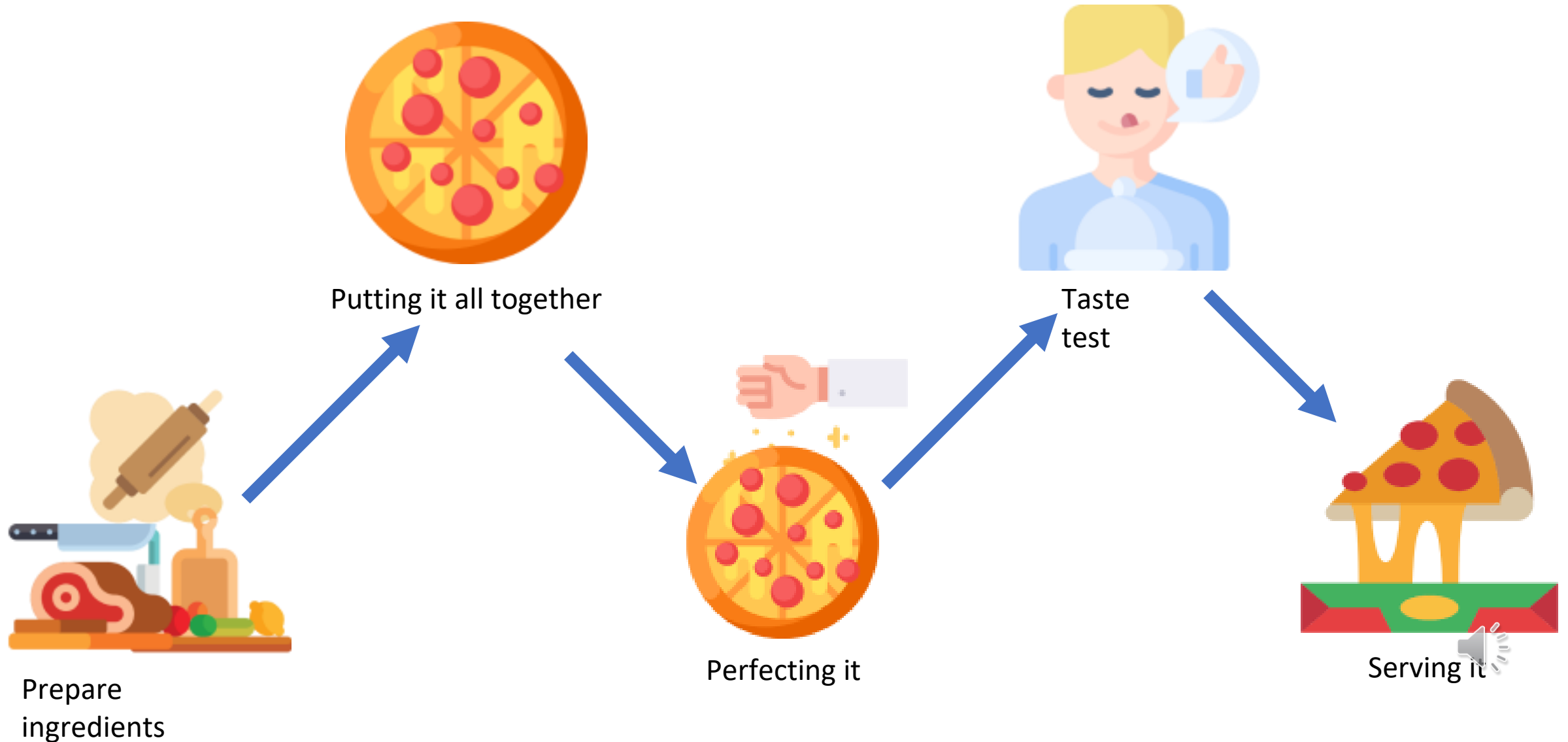
- A resource rich device for training
- An embedded system for inference



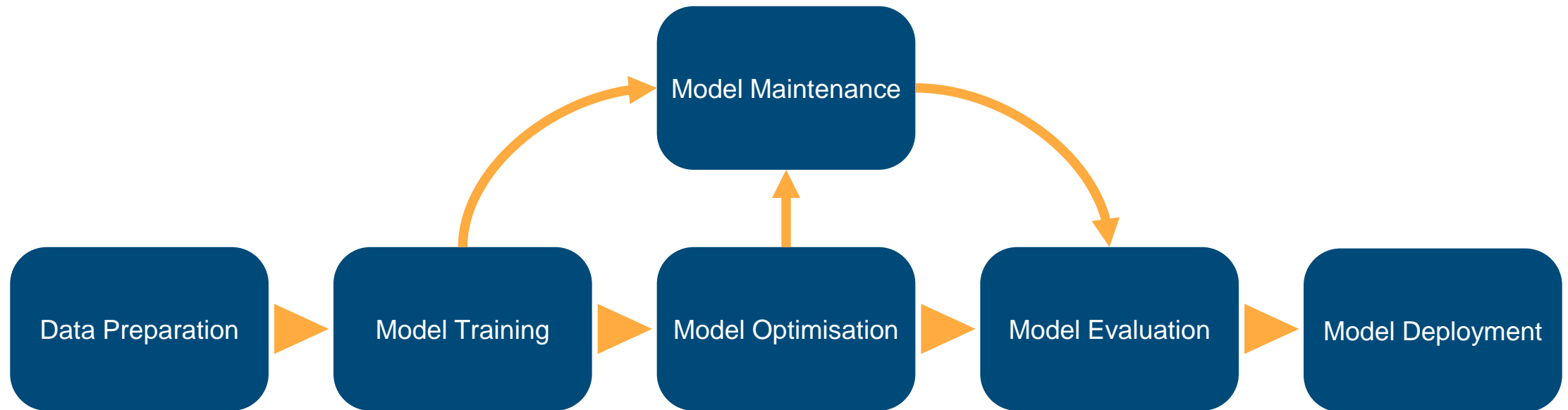
Required Devices



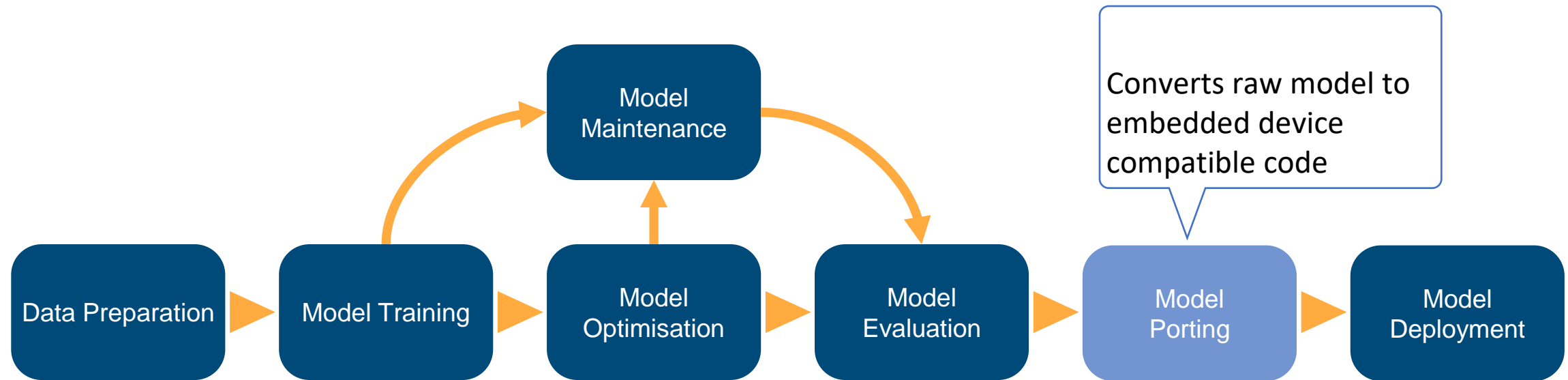
Pipeline: Traditional ML



Pipeline: Traditional ML

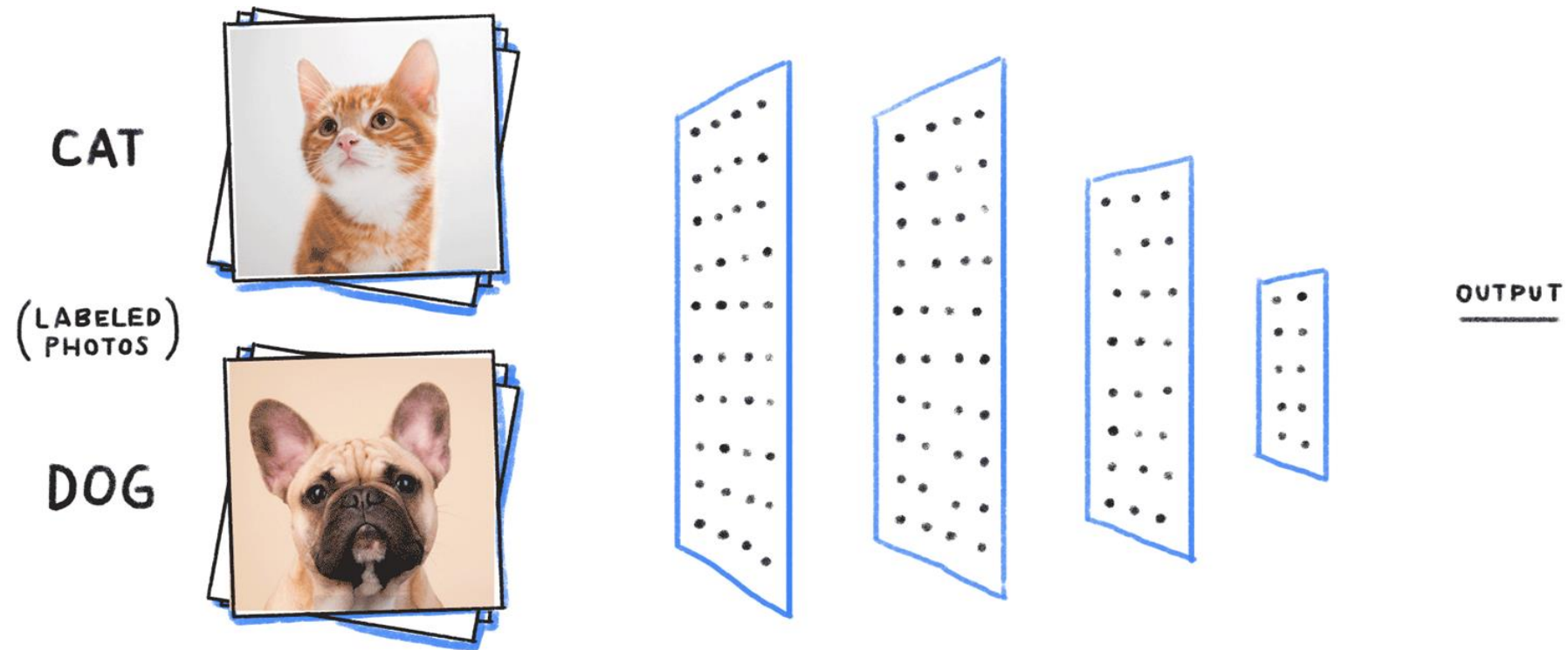


Pipeline: Embedded ML

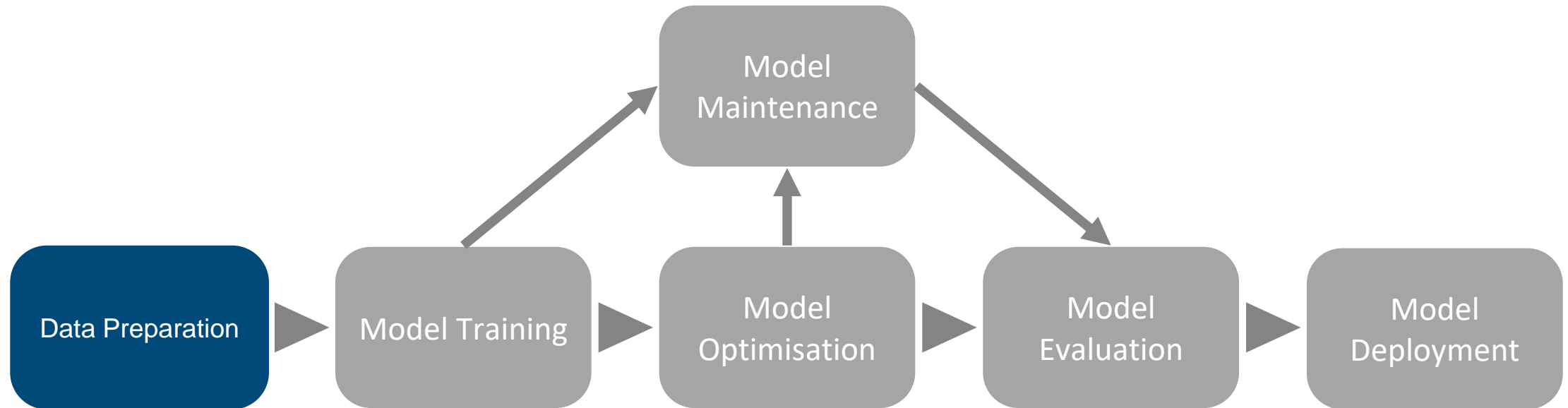


Running Example

Cat-Dog detection model



Pipeline: Traditional ML



Capturing the data



Google
Dataset Search Beta

Better data = Better model



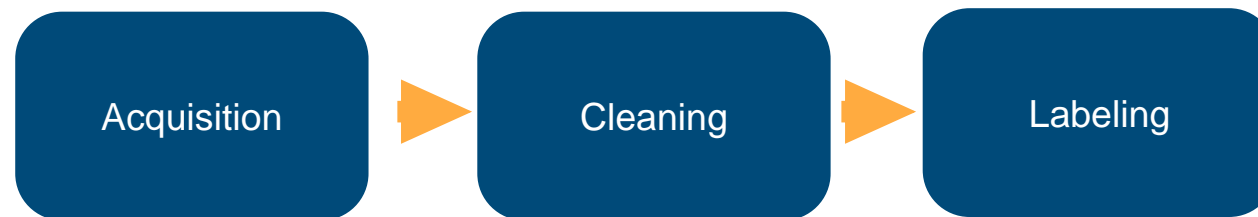
kaggle



Capturing the data - Creating a dataset

Process of creating your own dataset

1. Data **Acquisition**
2. Data **Cleaning**
3. Data **Labeling**



Capturing the data - Creating a dataset

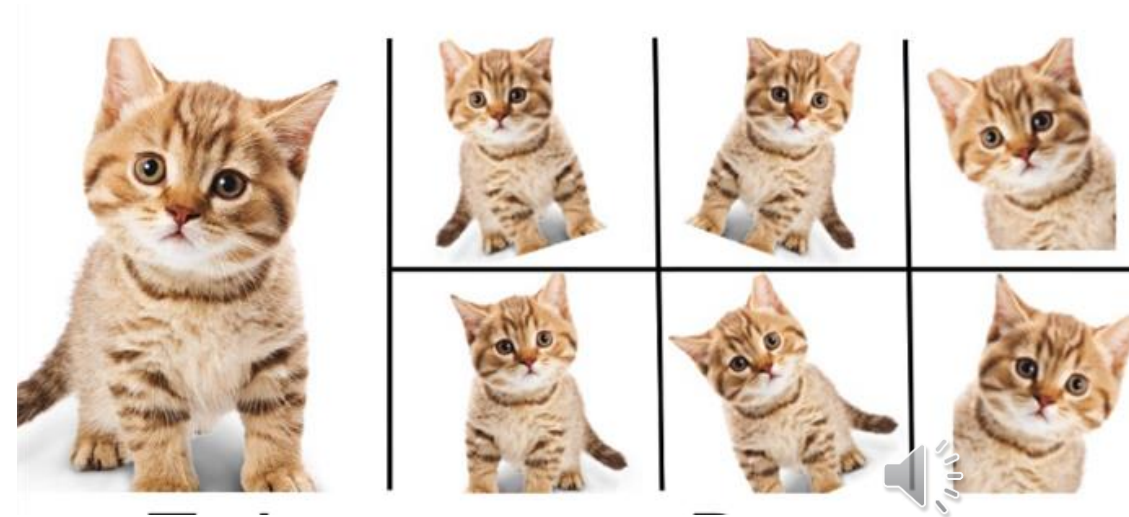
Data Acquisition

1. Generate the data

- Used when **no** dataset for the use case exists
- Can be done via [crowdsourcing](#) or **synthesizing** data

2. Augment the data

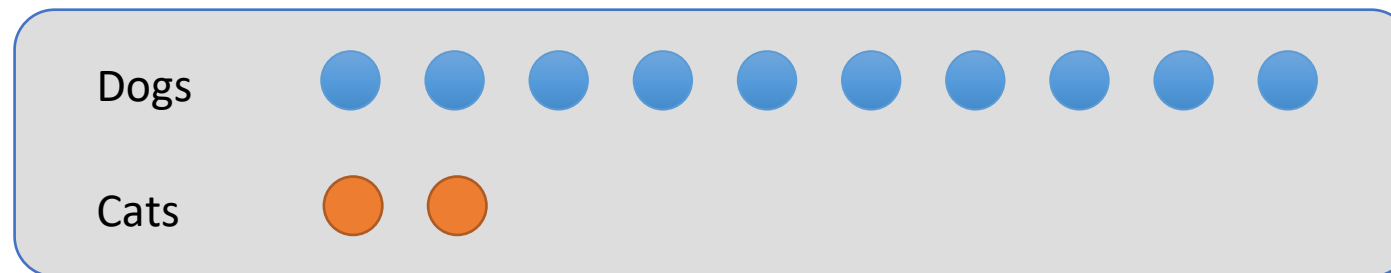
- Uses an existing dataset by “modifying” the existing
 - Cropping
 - Flipping
 - Rotating
 - Adjusting contrast and/or brightness



Capturing the data - Creating a dataset

Data Cleaning

- Cleaning out rogue data (duplicates, noisy, etc.)
- Cleaning out formatting issues (e.g., String to data-time formatting)
- Removing bias' in the data — diversify



Dataset



Capturing the data - Creating a dataset

Data Labeling

- Give more contextual meaning to the data
 - e.g., An image of a dog should be labelled as “Dog”



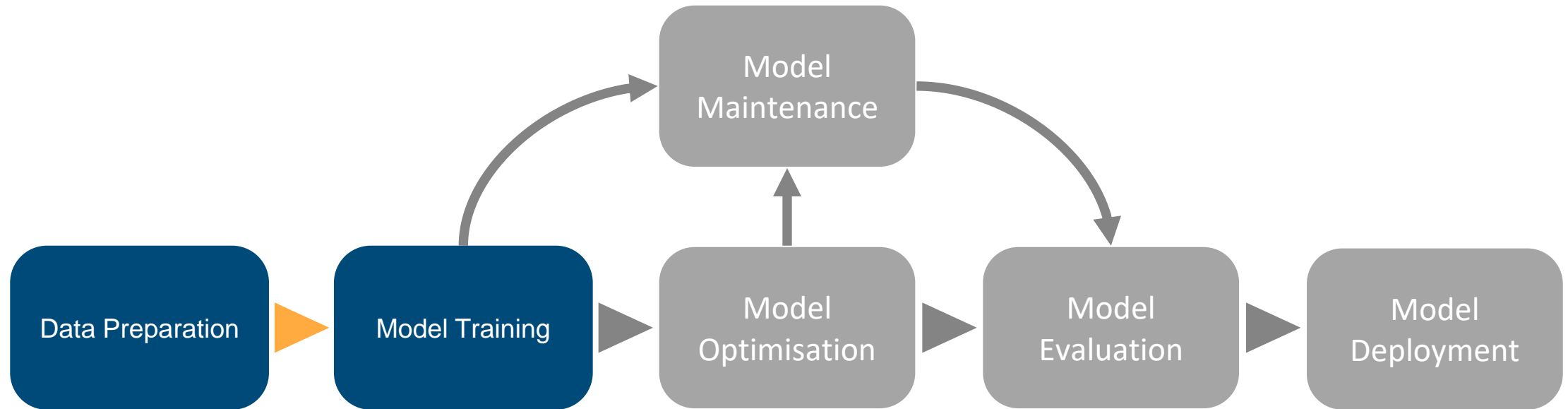
“Dog”



“Cat”



Pipeline: Traditional ML



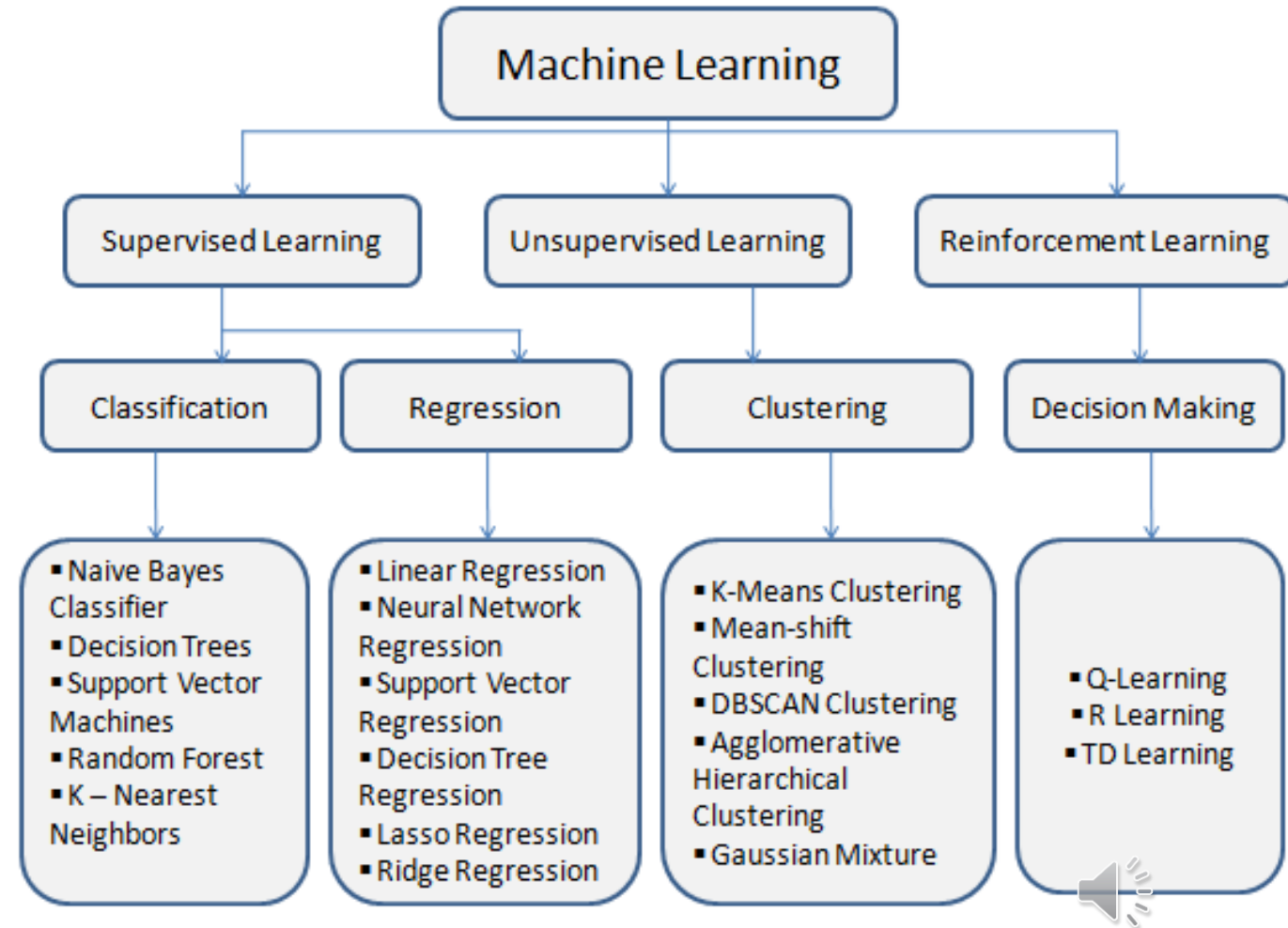
Training the model

1. Selecting the model for the problem

1. Understand the use-case
2. Consider the type of data

2. Creating the model

- Can use popular libraries
 1. TensorFlow
 2. SciKit Learn
 3. Keras





Cont. Cat-dog detection

1. What is the type of data we will feed into the model to get an output?

- Text data
- Image data
- Numerical data

2. What model is most suitable for the job?

- Convolutional Neural Networks
- Decision trees
- K-Nearest Neighbor

3. What framework should be use?

- Keras
- OpenCV
- TensorFlow



Cont. Cat-dog detection

1. What is the type of data we will feed into the model to get an output?

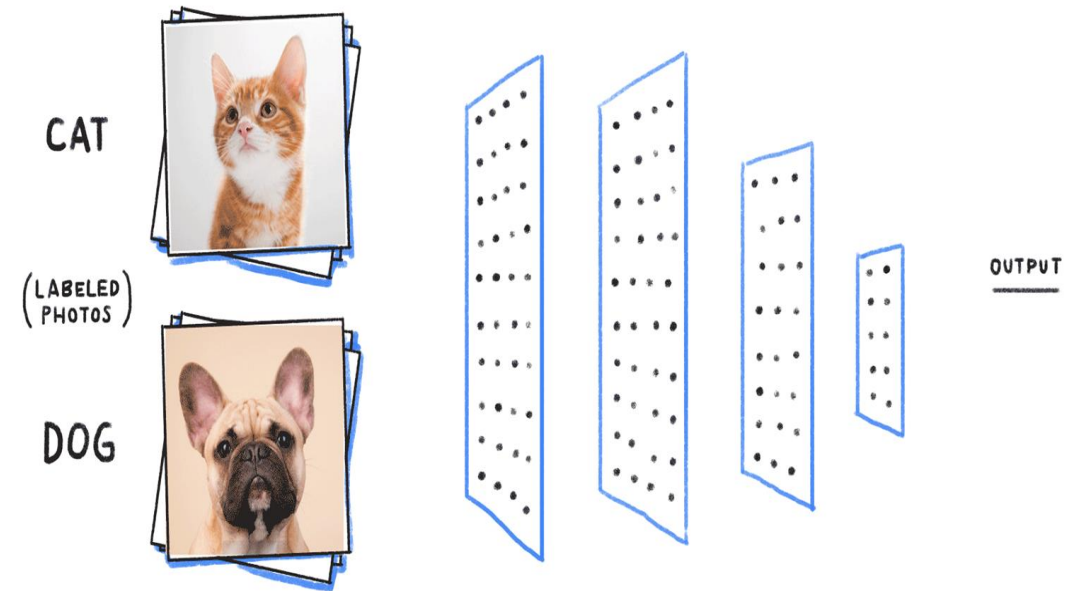
- Text data
- **Image data** ✓
- Numerical data

2. What model is most suitable for the job?

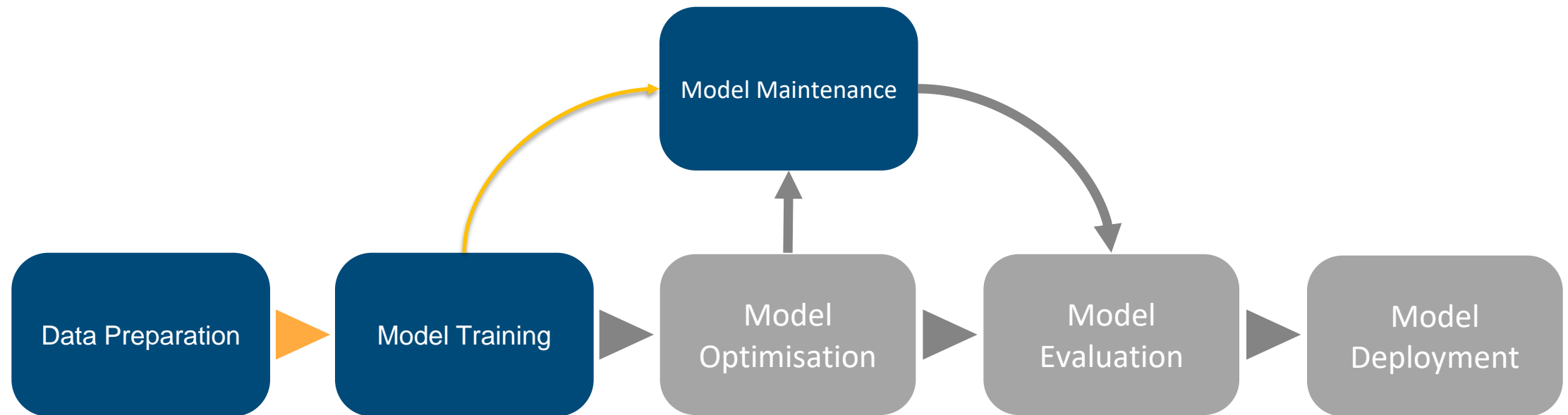
- **Convolutional Neural Networks** ✓
- Decision trees
- K-Nearest Neighbor

3. What framework should be use?

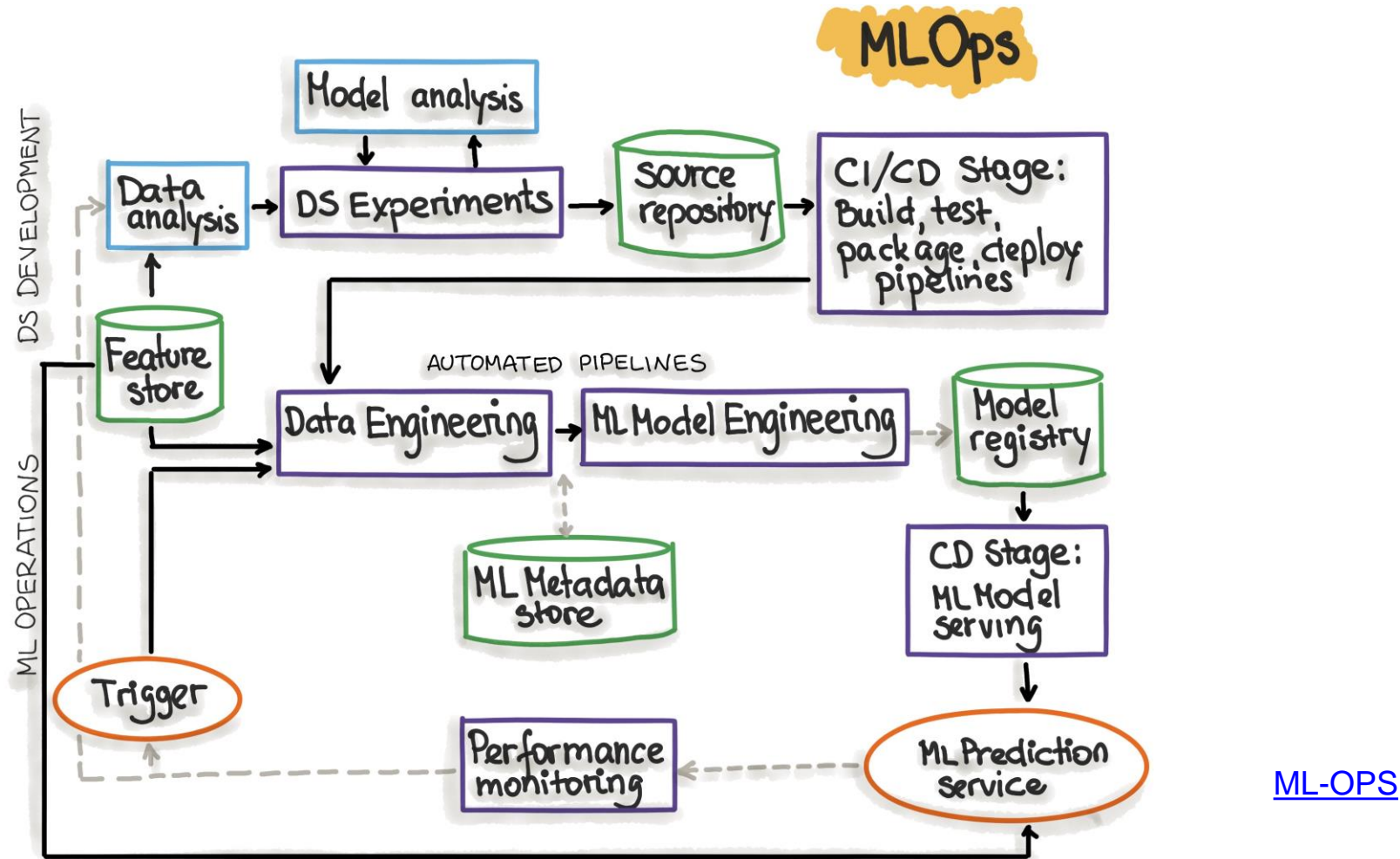
- Keras
- OpenCV
- **TensorFlow** ✓



Pipeline: Traditional ML



Model Maintenance



SUPERB[®]
wallpapers

That's all Folks!

