What is Analytics Zoo

BigDL

Distributed, High-Performance

Deep Learning Framework

for Apache Spark

https://github.com/intel-analytics/bigdl

ANALYTICS ZOO

Unified Analytics + AI Platform

Distributed TensorFlow, Keras, PyTorch and BigDL
on Apache Spark

https://github.com/intel-analytics/analytics-zoo

Accelerating Data Analytics + AI Solutions At Scale

Cluster Serving

2020/06/12

Overview

What's Analytics Zoo

ANALYTICS ZOO

Analytics + AI Platform

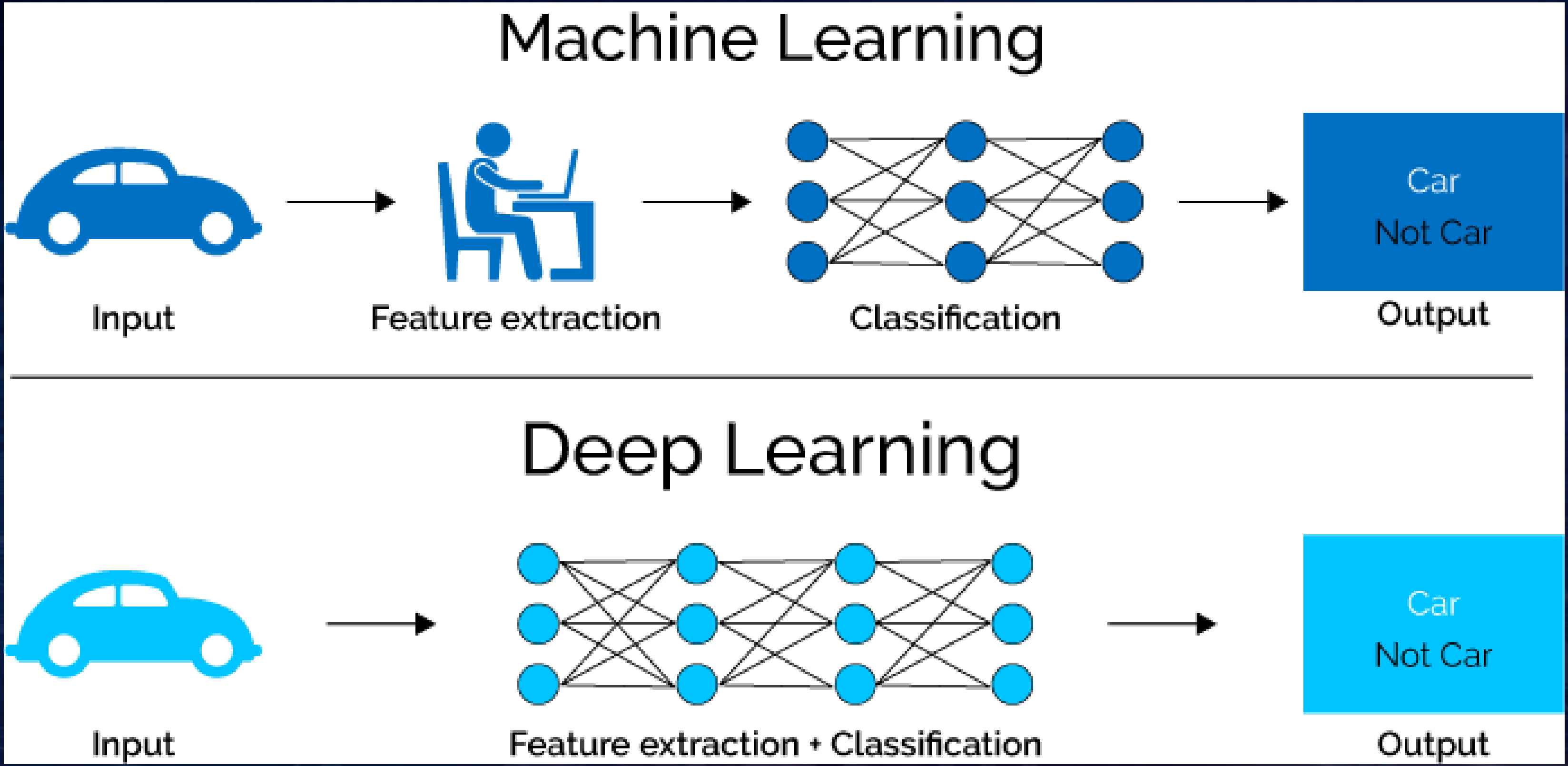
Distributed TensorFlow*, Keras*, PyTorch* and BigDL on Apache Spark*

https://github.com/intel-analytics/analytics-zoo

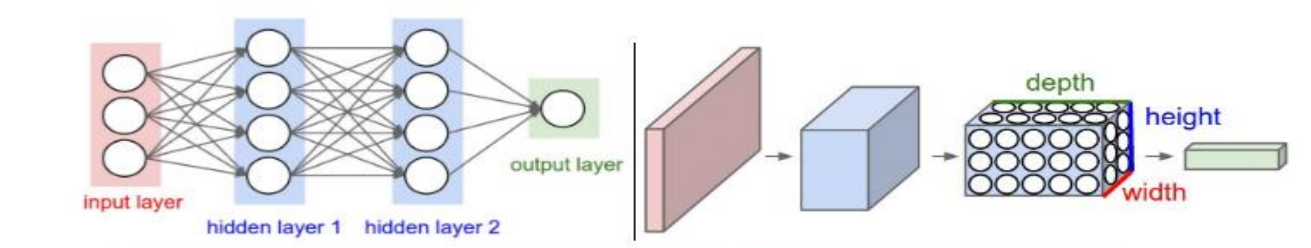
Accelerating Data Analytics + AI Solutions At Scale

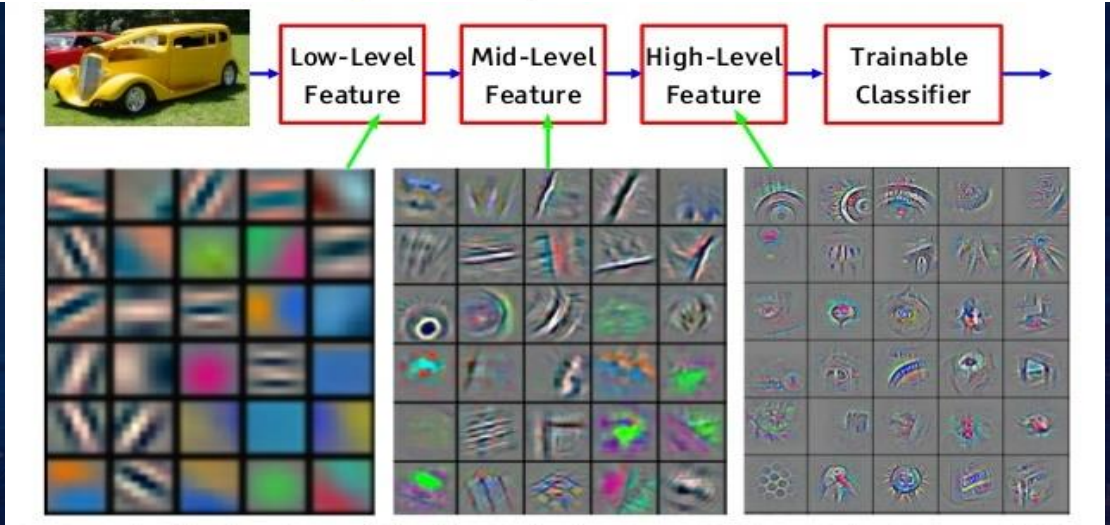
*Other names and brands may be claimed as the property of others.

Machine Learning VS Deep Learning



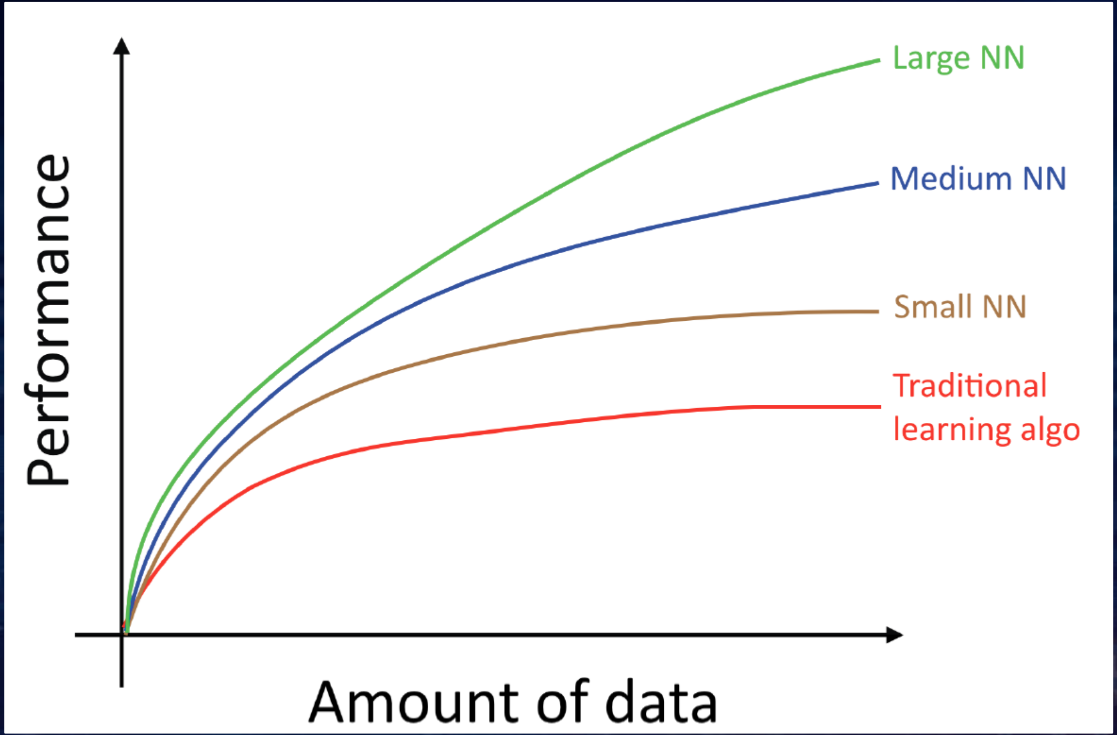
Feature Visualization





Feature visualization of convolutional net trained on ImageNet from [Zeiler & Fergus 2013]

Data & Performance Relationship



“Machine Learning Yearning”,
Andrew Ng, 2016

Real-World ML/DL Applications Are Complex Data Analytics Pipelines

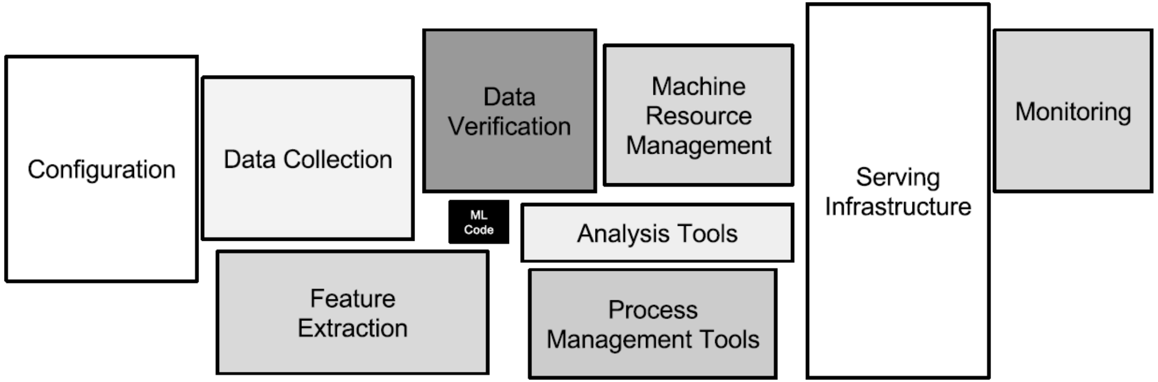
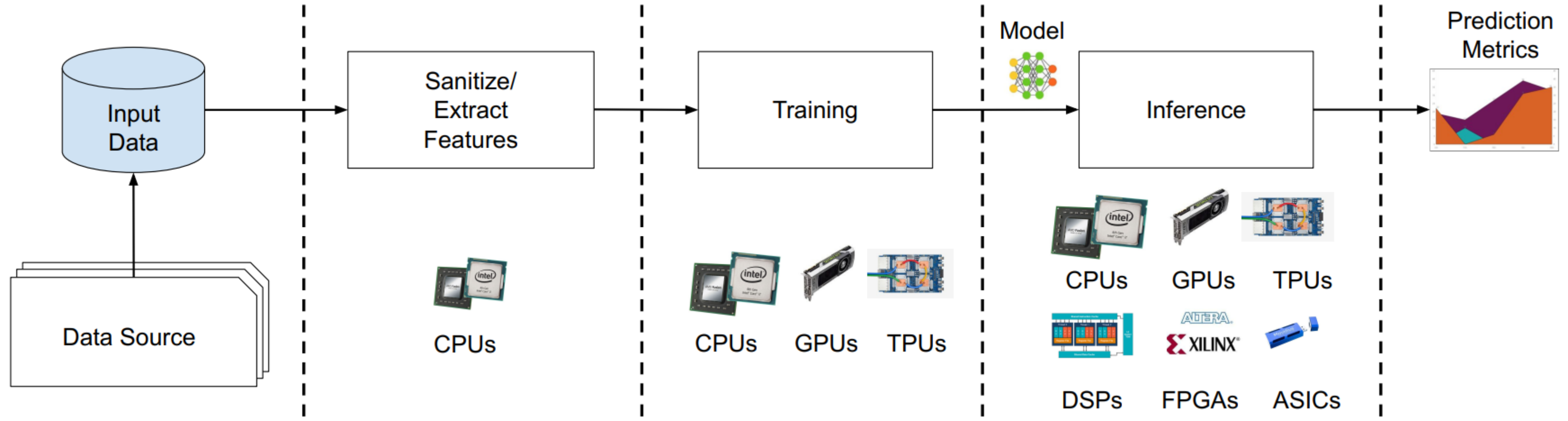


Figure 1: Only a small fraction of real-world ML systems is composed of the ML code, as shown by the small black box in the middle. The required surrounding infrastructure is vast and complex.

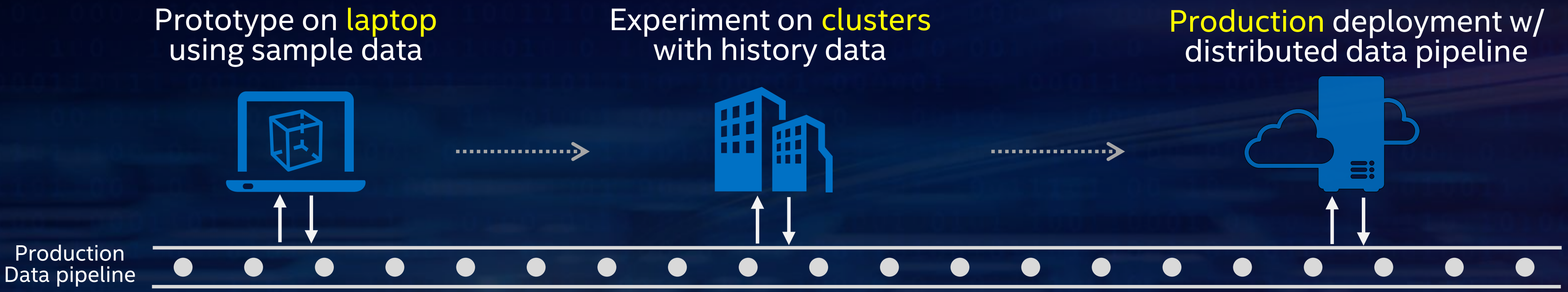
"Hidden Technical Debt in Machine Learning Systems",
Sculley et al., Google, NIPS 2015 Paper

Data Analytics Pipeline from production perspective



End-to-End Big Data Analytics and AI Pipeline

Seamless Scaling from Laptop to Production with ANALYTICS ZOO



Prototype on laptop using sample data

Experiment on clusters with history data

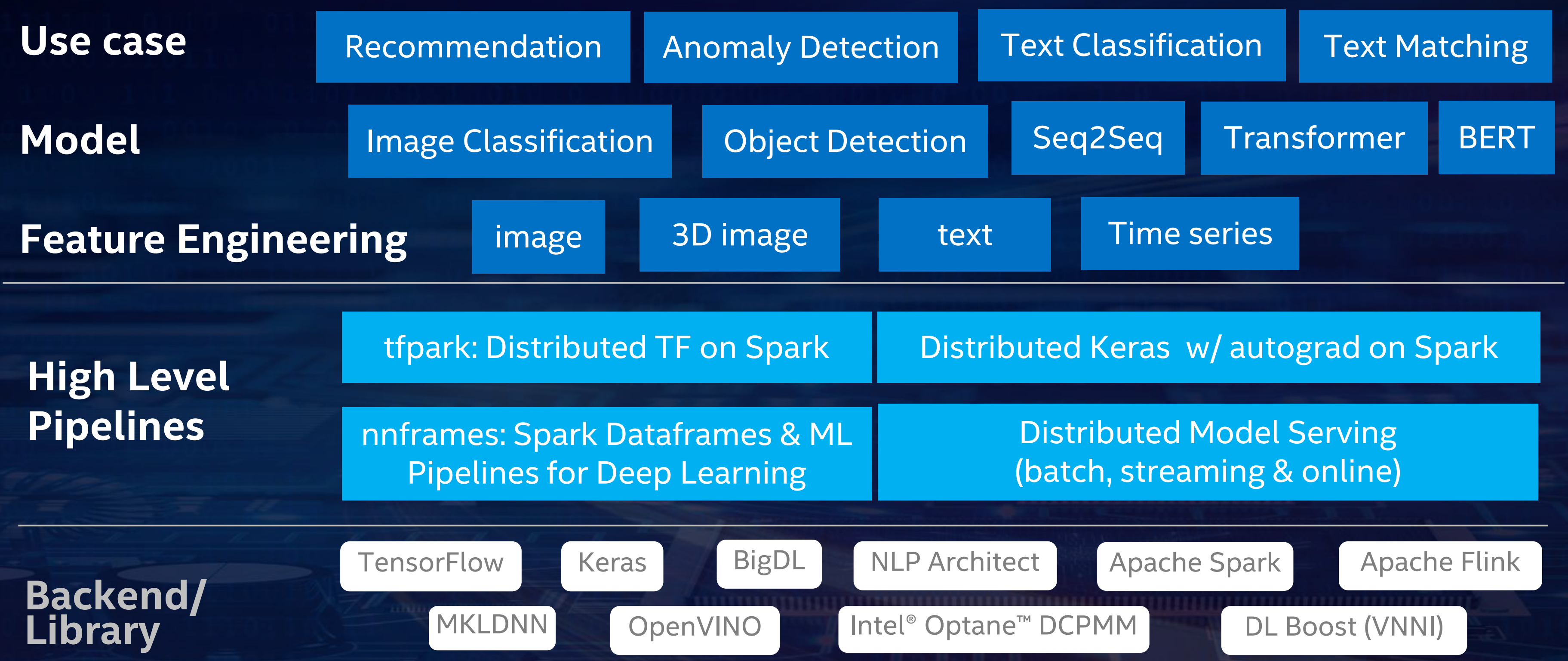
Production deployment w/ distributed data pipeline

Production Data pipeline

- **“Zero” code change from laptop to distributed cluster**
- **Directly access production data (Hadoop/Hive/HBase) without data copy**
- **Easily prototype the end-to-end pipeline**
- **Seamlessly deployed on production big data clusters**

Analytics Zoo

Unified Analytics + AI Platform for Big Data



Use case

Recommendation

Anomaly Detection

Text Classification

Text Matching

Model

Image Classification

Object Detection

Seq2Seq

Transformer

BERT

Feature Engineering

image

3D image

text

Time series

High Level Pipelines

tfpark: Distributed TF on Spark

Distributed Keras w/ autograd on Spark

nnframes: Spark Dataframes & ML Pipelines for Deep Learning

Distributed Model Serving (batch, streaming & online)

Backend/ Library

TensorFlow

Keras

BigDL

NLP Architect

Apache Spark

Apache Flink

MKLDNN

OpenVINO

Intel® Optane™ DCPMM

DL Boost (VNNI)

https://github.com/intel-analytics/analytics-zoo

What's Analytics Zoo

ANALYTICS ZOO

Analytics + AI Platform

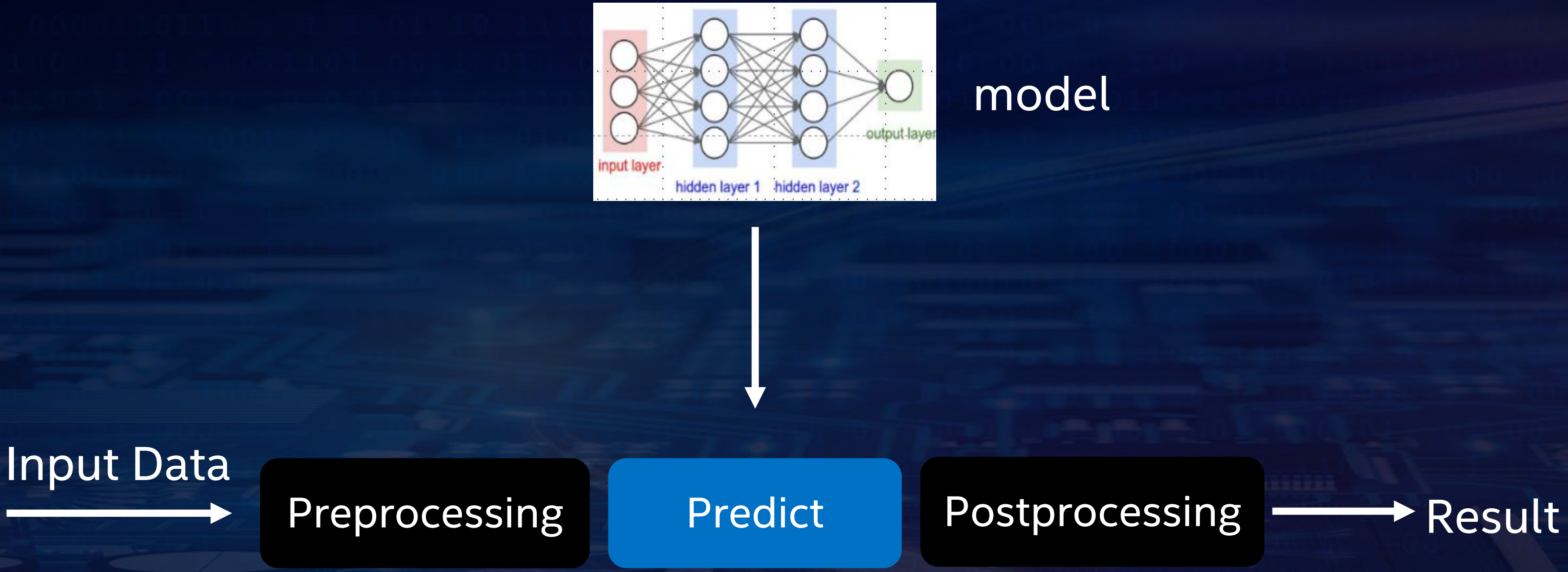
Distributed TensorFlow*, Keras*, PyTorch* and BigDL on Apache Spark*

https://github.com/intel-analytics/analytics-zoo

Accelerating Data Analytics + AI Solutions At Scale

*Other names and brands may be claimed as the property of others.

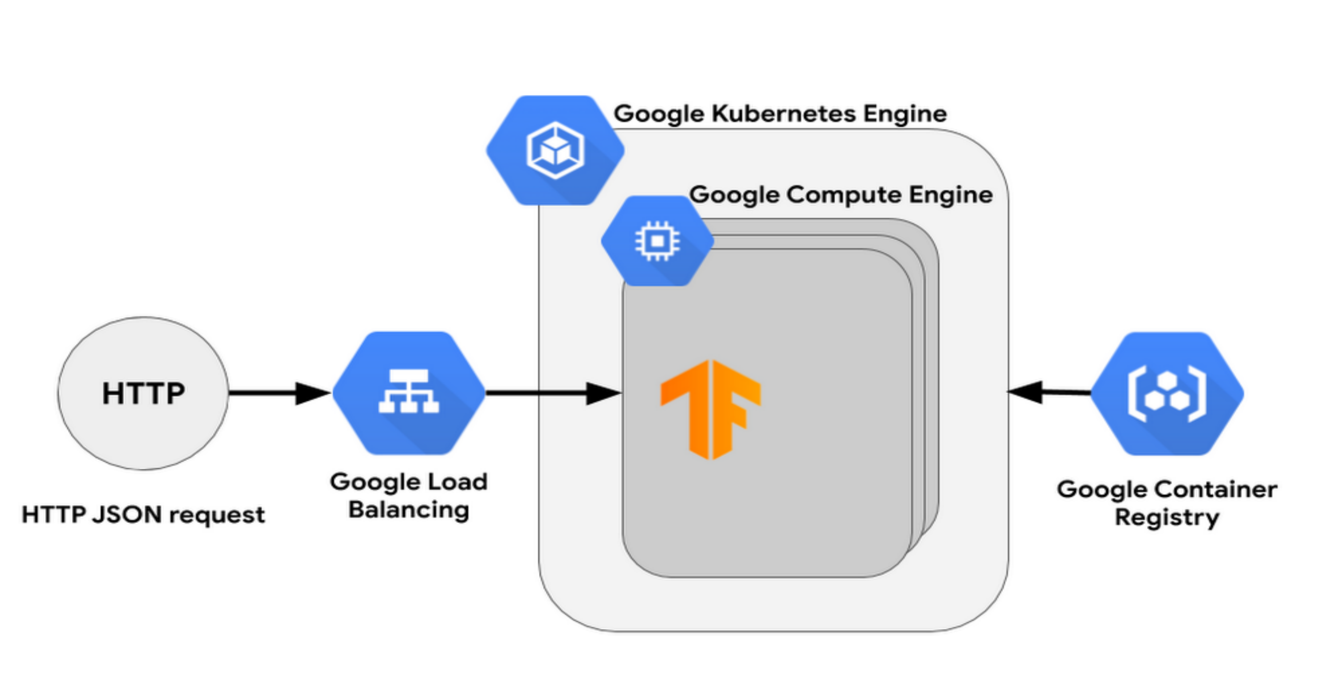
What’s Serving



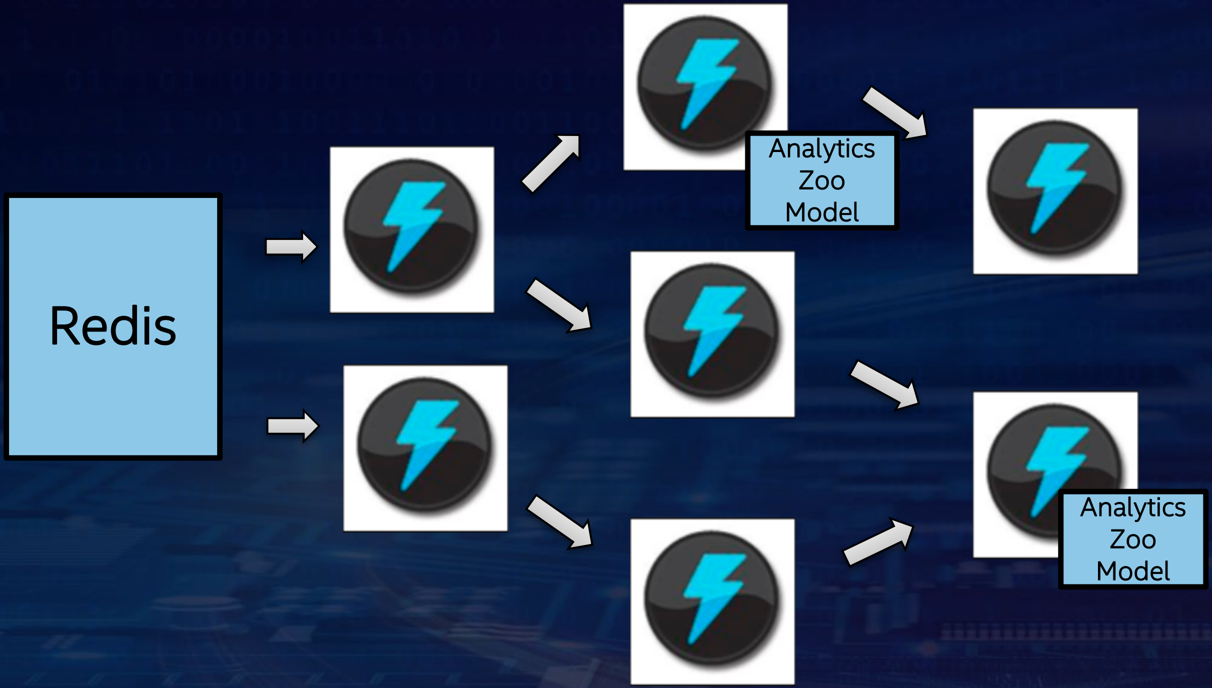
model

Input Data → Preprocessing → Predict → Postprocessing → Result

Example of TFServing



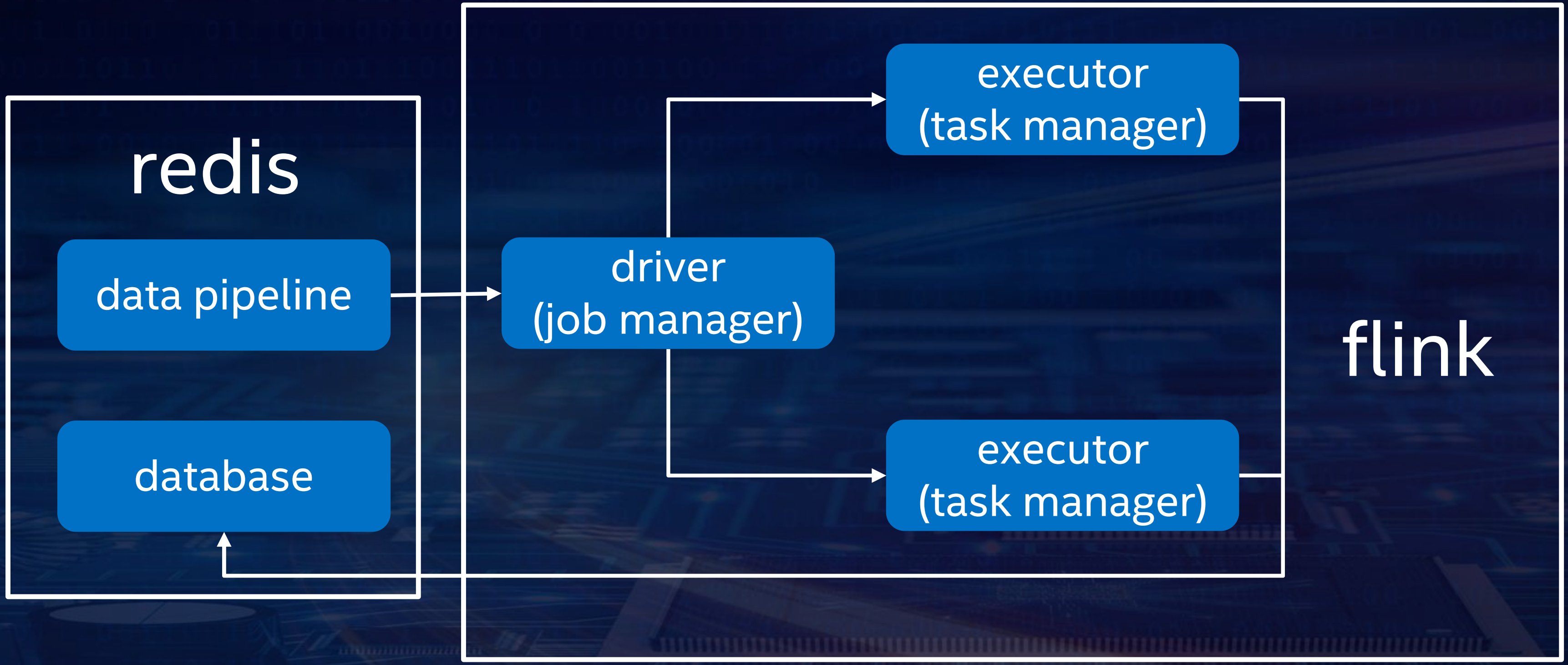
Distributed Model Serving



Distributed model serving in **Web Service**, **Flink**, **Kafka**, **Storm**, etc.

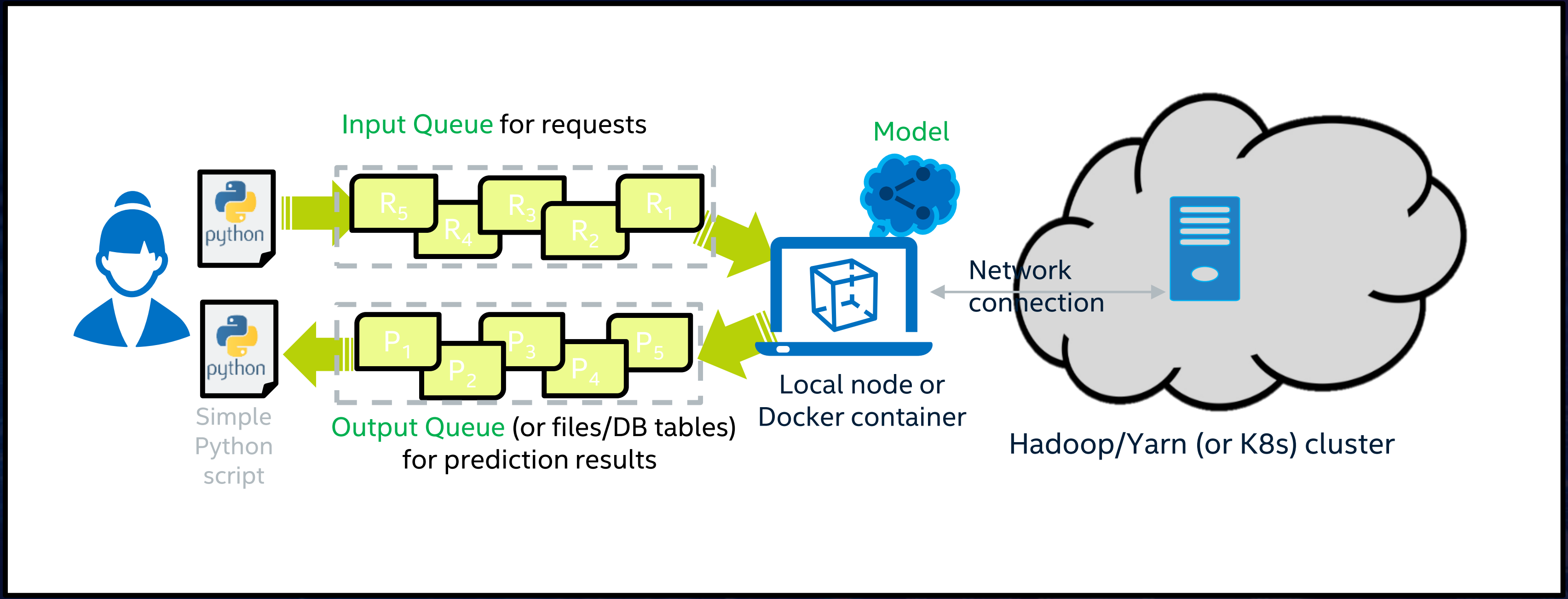
- Plain Java or Python API, with OpenVINO and DL Boost (VNNI) support

Main Version of Cluster Serving



Version based on Spark Streaming is also supported.

Data pipeline User Perspective



Input Queue for requests
python
R5
R4
R3
R2
R1
Model
Network connection
python
P1
P2
P3
P4
P5
Simple Python script
Output Queue (or files/DB tables) for prediction results
Local node or Docker container
Hadoop/Yarn (or K8s) cluster

Deploy Your Own Cluster Serving

One command to pull docker image, and customize your config, then call cluster-serving-start to start your serving

example of config:

```
## Analytics-zoo Cluster Serving
model:
  # model path must be set
  path: resources
data:
  # redis address
  src: XXXXXX:6379


…
```

API Introductions

http API

data are represented by json format, and call http post method to enqueue your data into pipeline
(http API is compatible with TFServing)

python API

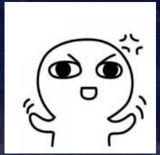
data are represented by ndarray, and call python method to enqueue your data into pipeline

Use Case – Medical Imaging Analysis

Consider a Very large medical image of patient, the mission is to determine if tumor exists

End-to-end pipeline would contain image preprocessing, predict, all reduce analysis

tumor →



→ if num > 1% the condition is bad

Advantages

Wide Range Deep Learning model support

Tensorflow, Caffe, OpenVINO, Pytorch, BigDL

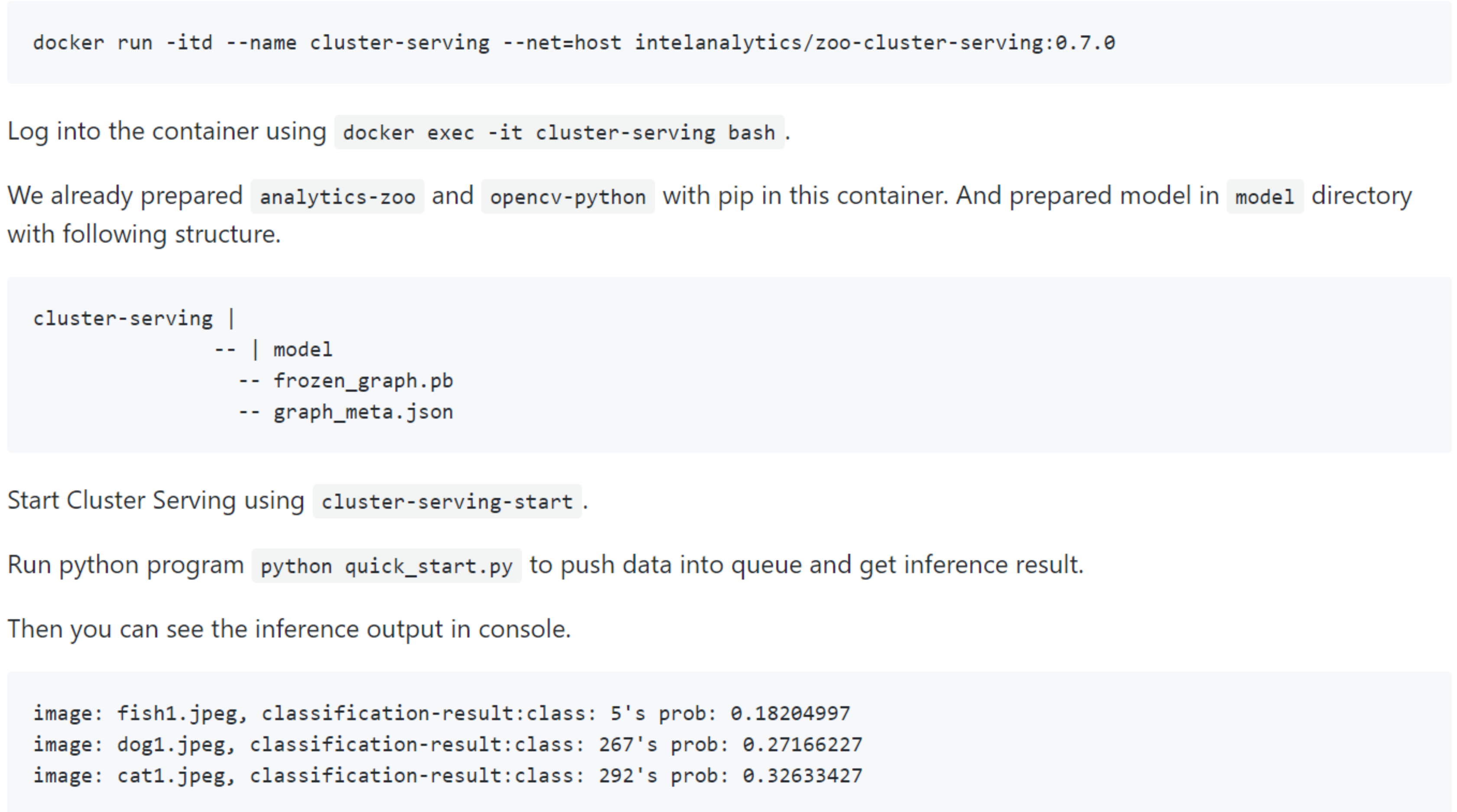
Low Latency

Continuous Streaming pipeline is supported by Apache Flink, also Spark version is supported for users who are more familiar.

High Throughput & Scalability

Optimization of multithread control, and could easily scale out to clusters.

Very Quick Start



```
docker run -itd --name cluster-serving --net=host intelanalytics/zoo-cluster-serving:0.7.0
```

Log into the container using `docker exec -it cluster-serving bash`.

We already prepared `analytics-zoo` and `opencv-python` with pip in this container. And prepared model in `model` directory with following structure.

```
cluster-serving |
               -- | model
                 -- frozen_graph.pb
                 -- graph_meta.json
```

Start Cluster Serving using `cluster-serving-start`.

Run python program `python quick_start.py` to push data into queue and get inference result.

Then you can see the inference output in console.

```
image: fish1.jpeg, classification-result:class: 5's prob: 0.18204997
image: dog1.jpeg, classification-result:class: 267's prob: 0.27166227
image: cat1.jpeg, classification-result:class: 292's prob: 0.32633427
```

End-to-End Big Data and AI Pipelines

Seamless Scaling from Laptop to Production

ANALYTICS ZOO

Unified Analytics + AI Platform

Distributed TensorFlow*, Keras*, PyTorch* & BigDL on Apache Spark*

https://github.com/intel-analytics/analytics-zoo

*Other names and brands may be claimed as the property of others.

https://github.com/intel-analytics/analytics-zoo/blob/master/docs/docs/ClusterServingGuide/ProgrammingGuide.md

Analytics-Zoo

rm

Distributed Te DL on Apache Spark*

https://g

*Other names and brands may be claimed as the property of others.

What is Analytics Zoo

BigDL

Distributed, High-Performance

Deep Learning Framework

for Apache Spark

https://github.com/intel-analytics/bigdl

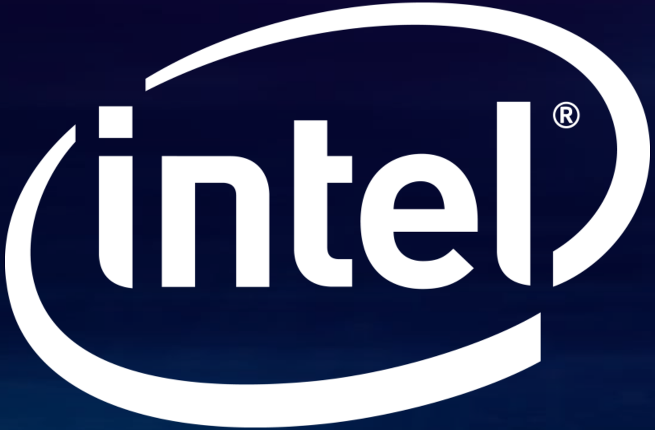
ANALYTICS ZOO

Unified Analytics + AI Platform

Distributed TensorFlow, Keras, PyTorch and BigDL on Apache Spark

https://github.com/intel-analytics/analytics-zoo

Accelerating Data Analytics + AI Solutions At Scale



intel®

LEGAL DISCLAIMERS

- Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Learn more at intel.com, or from the OEM or retailer.
- No computer system can be absolutely secure.
- Tests document performance of components on a particular test, in specific systems. Differences in hardware, software, or configuration will affect actual performance. Consult other sources of information to evaluate performance as you consider your purchase. For more complete information about performance and benchmark results, visit http://www.intel.com/performance.

Intel, the Intel logo, Xeon, Xeon phi, Lake Crest, etc. are trademarks of Intel Corporation in the U.S. and/or other countries.
*Other names and brands may be claimed as the property of others.

What is Analytics Zoo

BigDL

Distributed, High-Performance

Deep Learning Framework

for Apache Spark

https://github.com/intel-analytics/bigdl

ANALYTICS ZOO

Unified Analytics + AI Platform

Distributed TensorFlow, Keras, PyTorch and BigDL on Apache Spark

https://github.com/intel-analytics/analytics-zoo

Accelerating Data Analytics + AI Solutions At Scale