What Every Technologist Should Know About AI and Deep Learning

Alex McDonald

Standards & Industry Associations, NetApp Inc.

The information is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. NetApp makes no warranties, expressed or implied, on future functionality and timeline. The development, release, and timing of any features or functionality described for NetApp’s products remains at the sole discretion of NetApp. NetApp's strategy and possible future developments, products and or platforms directions and functionality are all subject to change without notice. NetApp has no obligation to pursue any course of business outlined in this document or any related presentation, or to develop or release any functionality mentioned therein.

Why is AI & Deep Learning Important?

- AI and Deep Learning is disrupting every industry
- For decades, AI was all about improving algorithms
- **Now** the focus is on putting AI to practical use
  - Critical to leverage well-engineered systems

- This talk will
  - Take you on a broad coherent tour of Deep Learning systems
  - Help you appreciate the role well-engineered systems play in AI disruption
  - Take you a step closer to being a unicorn
    - Systems + AI - something highly desirable, difficult to obtain

WHEN VISITING A NEW HOUSE, IT'S GOOD TO CHECK WHETHER THEY HAVE AN ALWAYS-ON DEVICE TRANSMITTING YOUR CONVERSATIONS SOMEWHERE.

Agenda

- AI Primer
- AI Stacks Overview
- Deep Learning Process
  - Training
  - Inference
- Deep Learning Systems
  - Hardware
  - Software
- Datasets and Data flow
- Future of Systems

Background

AI or ML or DL

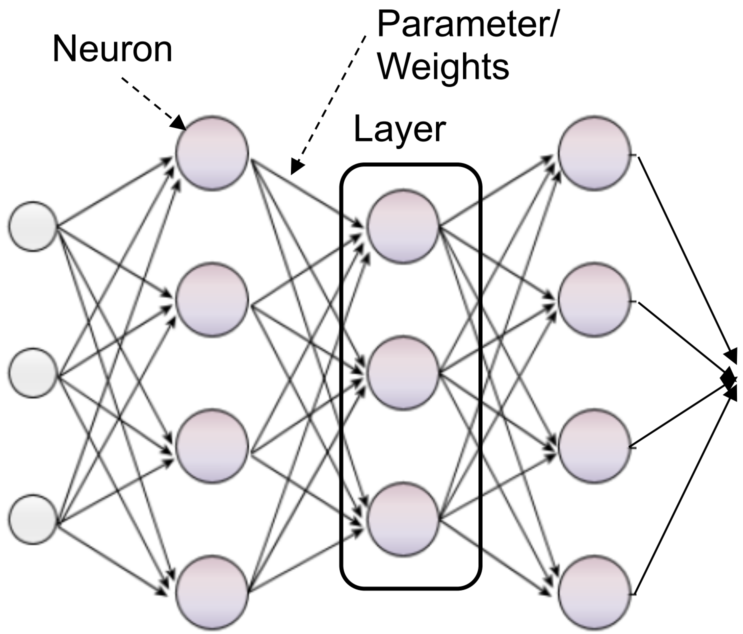
AI – program that imitates human intelligence

ML – program that learns with experience (i.e., data)

DL – ML using >1 hidden layers of neural network

Deep Learning 101

Basic concepts and terminology



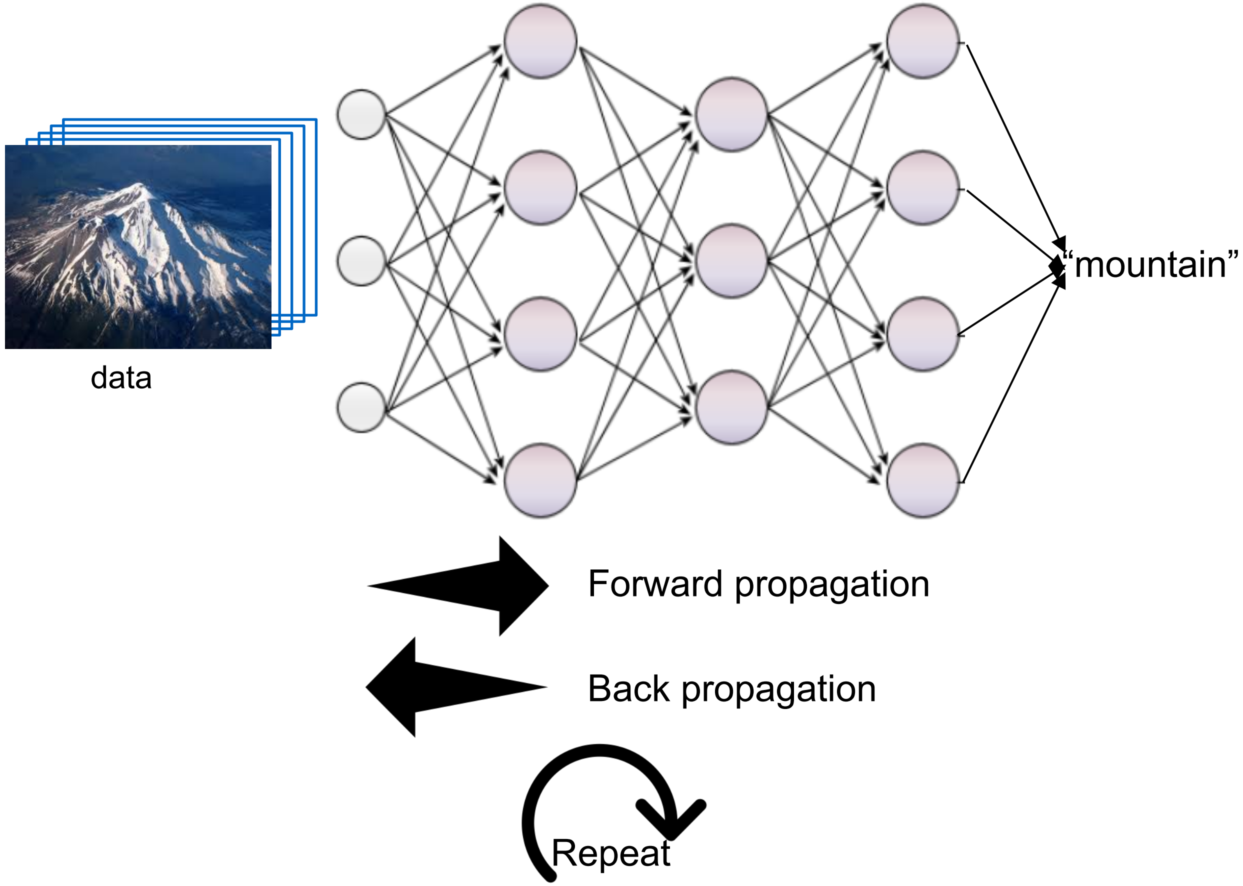
- Neuron: computational unit
- DL Model == type & structure

  More layers => better capture the features in dataset, better performance at task (normally)

  Parameters/Weights

Deep Learning 101

Basic concepts and terminology



- Neuron: computational unit
- DL Model == type & structure

  More layers => better capture the features in dataset, better performance at task (normally)

  Parameters/Weights
- Training: build a model from dataset

  Epoch: a pass over entire dataset

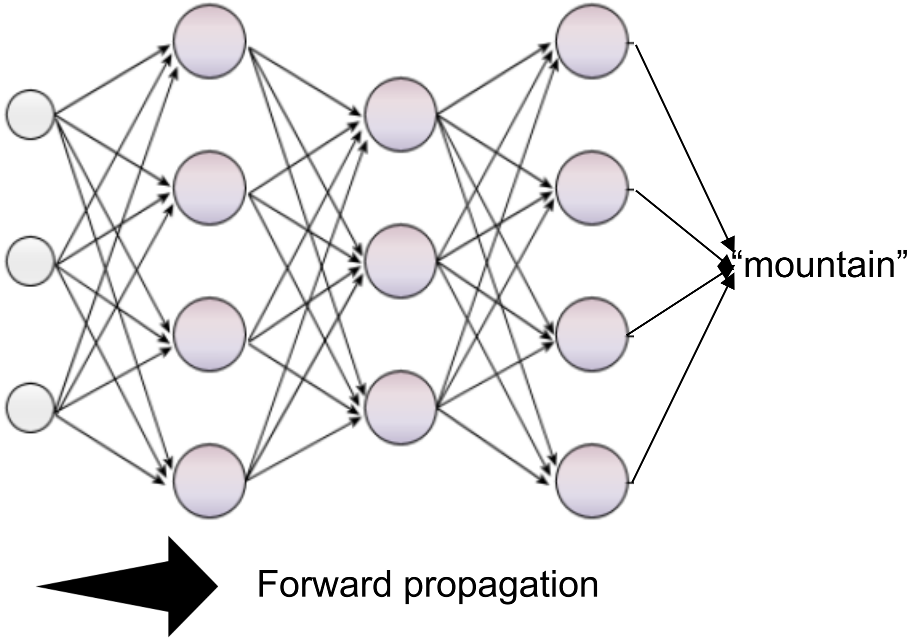
  Batch: a chunk of data

  Preprocessing/preparation: ready data to train

Deep Learning 101

Basic concepts and terminology

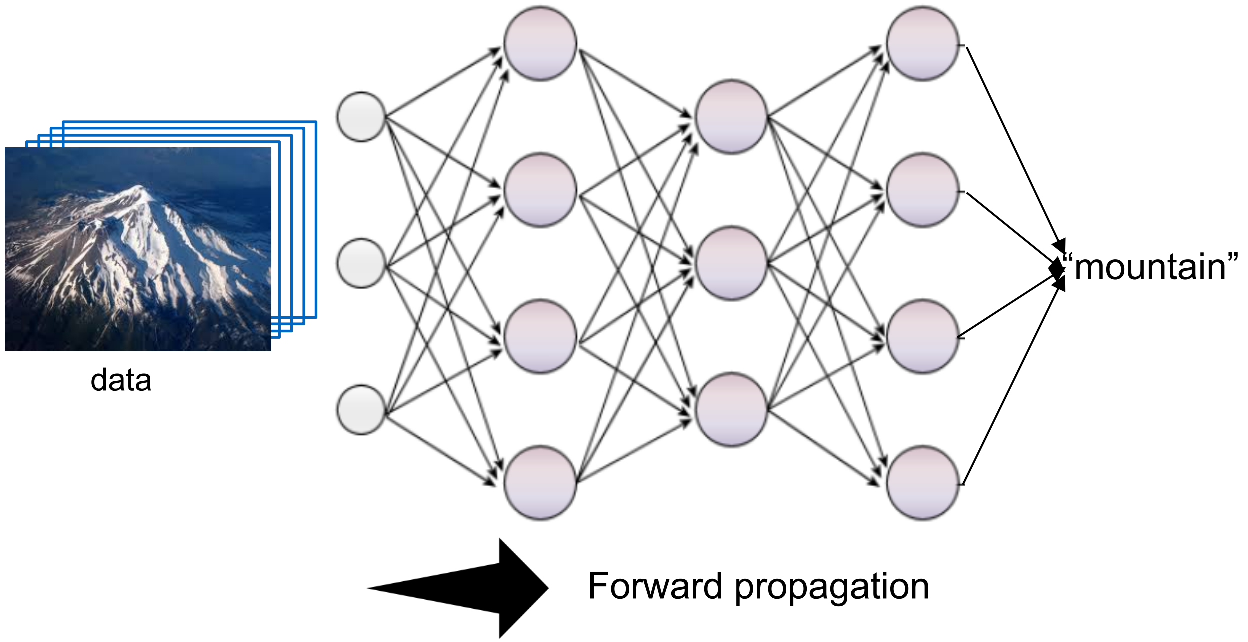
New data



- Neuron: computational unit
- DL Model == type & structure
  More layers => better capture the features in dataset, better performance at task (normally)
  Parameters/Weights
- Training: build a model from dataset
  Epoch: a pass over entire dataset
  Batch size: a chunk of data
  Preprocessing/preparation: ready data to train
- Inference: using a trained model

Deep Learning 101

Basic concepts and terminology



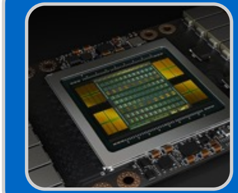
- State-of-the-Art DL is large scale
  - 100s of layers
  - Millions of parameters
  - 100s of GBs to TBs of data
  - Hours/days to train

AI Stack Overview

AI Stack Layers
AI PaaS, End-to-end solutions

AI Stack

Layers



Modern Compute

GPUs, TPUs, FPGAs

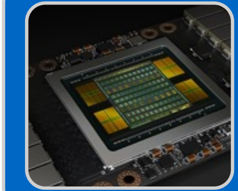
- Optimized hardware to provide tremendous speed-up for training, sometimes inference
- More easily available on cloud for rent

AI Stack

Layers

TensorFlow

Software



Modern Compute

Tensorflow, PyTorch, Caffe2, MxNet, CNTK, Keras, Gluon

- Library that implements algorithms, provides execution engine and programming APIs
- Used to train and build sophisticated models, and to do predictions based on the trained model for new data

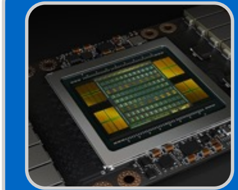
AI Stack

Layers

NetApp

Platform

Software



Modern Compute

Laptop, Cloud compute instances, H2O Deep Water, Spark DL pipelines

- Hardware accelerated platforms, supporting common software frameworks, to run the training and/or inference of deep neural networks
- Typically optimized for a preferred software framework
- Can be hosted on-prem or cloud
- Also offered as fully-managed service (PaaS) by cloud vendors like Amazon SageMaker, Google Cloud ML, Azure ML

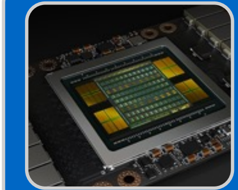
AI Stack

Layers

API-based service

Platform

Software



Modern Compute

Amazon Rekognition, Lex & Polly; Google Cloud API; Microsoft Cognitive Services;

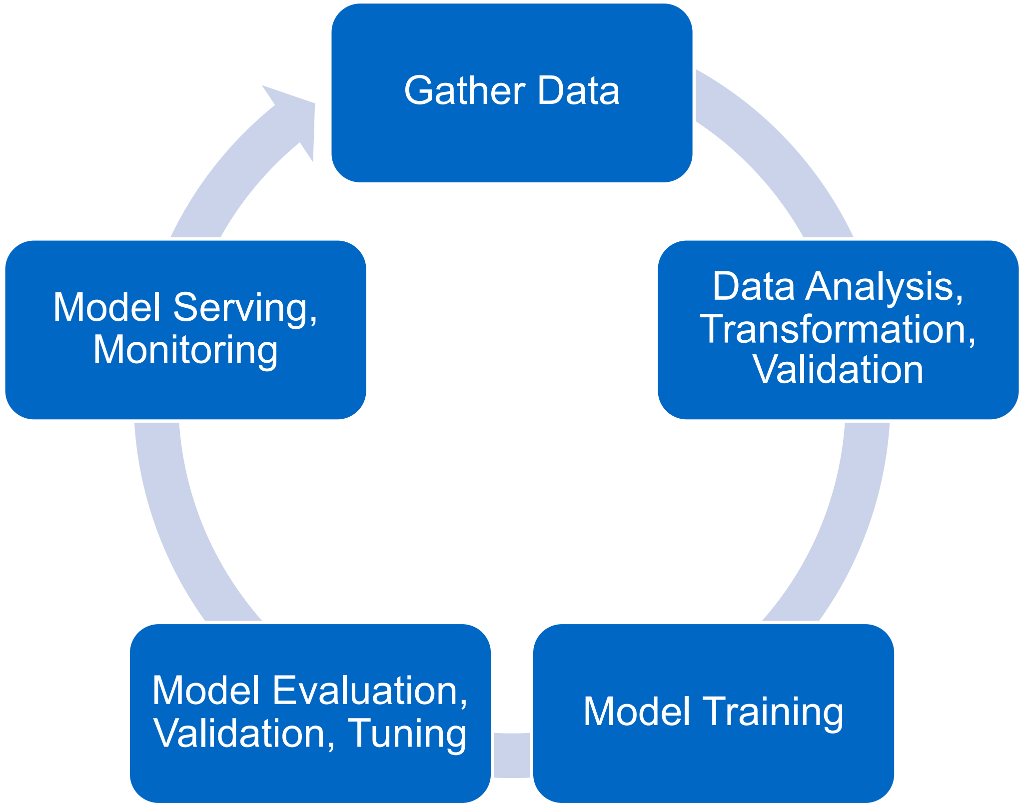
- Allows query based service access to generalizable state of art AI models for common tasks
  - Ex: send an image and get object tags as result, send mp3 and get converted text as result and so on
- No dataset, no training of model required by user
- Per-call cost model
- Integrated with cloud storage and/or bundled into end-to-end solutions and AI consultancy offerings like IBM Services Watson AI, ML & Cognitive consulting, Amazon's ML Solutions Lab, Google's Advanced Solutions Lab

Deep Learning Process

Training
Inference

DL Process and Data Lifecycle

DL lifecycle is very unlike traditional systems software development

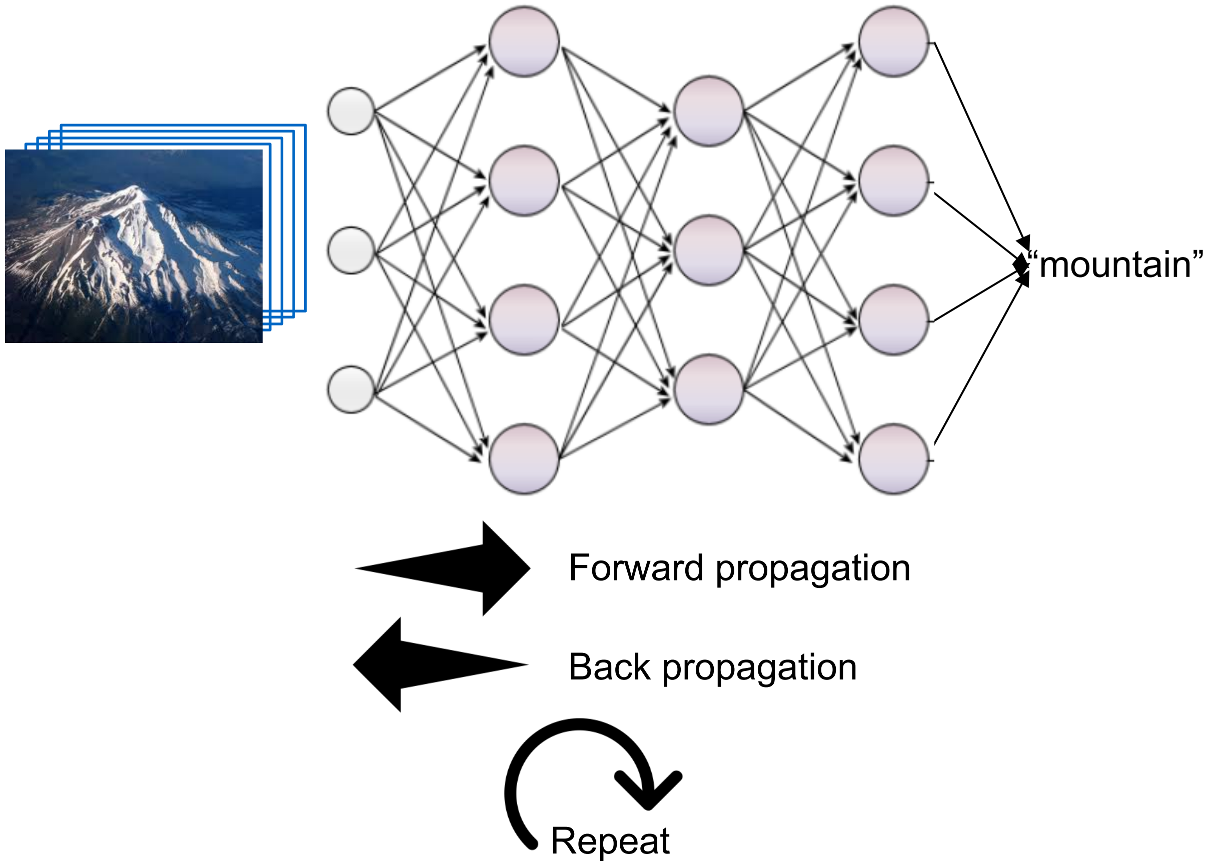


- Gathering and curating quality datasets and making them accessible across org
- Diverse tools and flexible infrastructure needed
- Evaluation criteria is critical but hard
  - Comparing algorithms is not straightforward
- Tracking artifacts like dataset transformations, tuning history, model versions, validation results more important than code
- Debugging, interpretability and fairness is limited
- Tension/friction: Data security and privacy; IT

Deep Learning Training

Training: build a model from a dataset

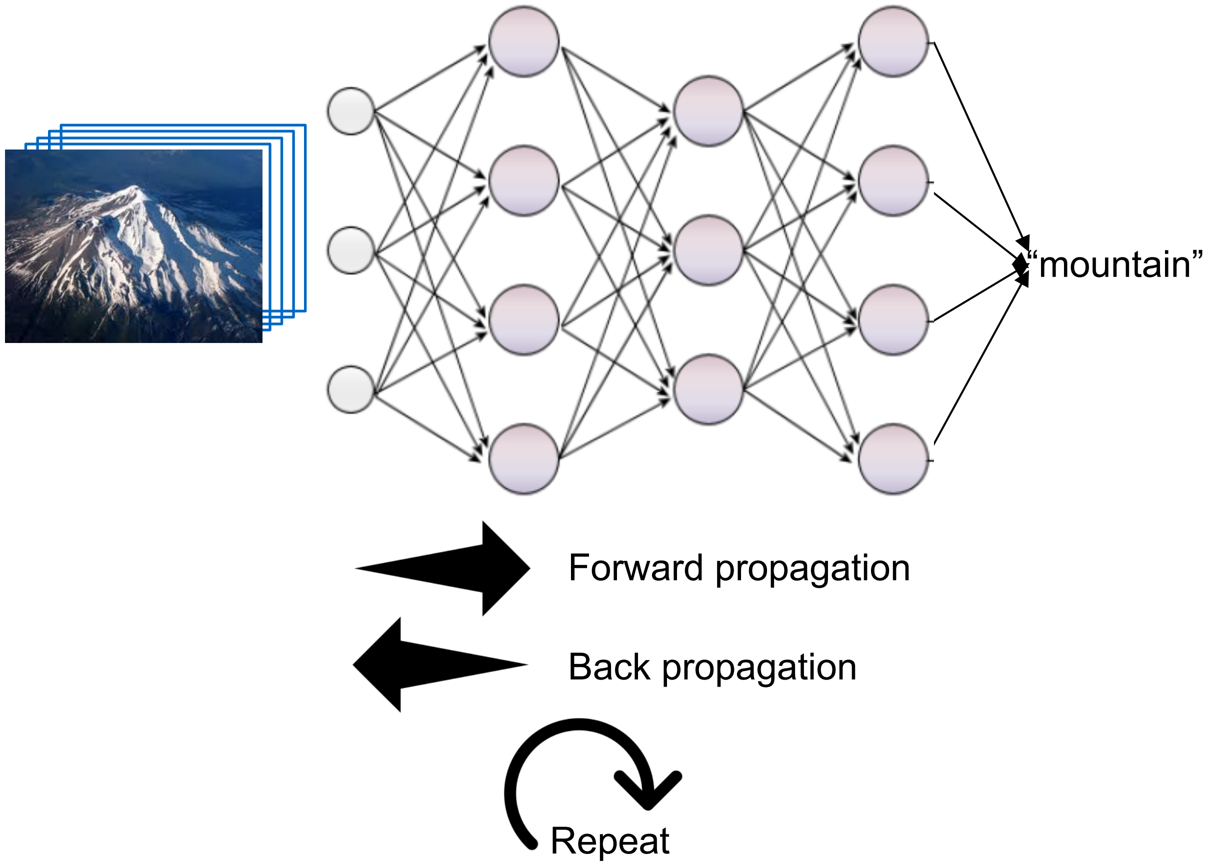
- Is memory and compute bound
  - Big datasets, complex math operations
- Is highly parallelized/distributed – across cores, across machines
  - Partition data, or model, or both
  - Scale Up before Scale Out
  - Communication to computation ratio
  - Speed vs Accuracy tradeoff
  - Federated learning
- Leans on enhancements to data quality
  - Augmentation, randomness
  - Transformations
  - Efficiently fit in memory



Deep Learning Training

Training: build a model from a dataset

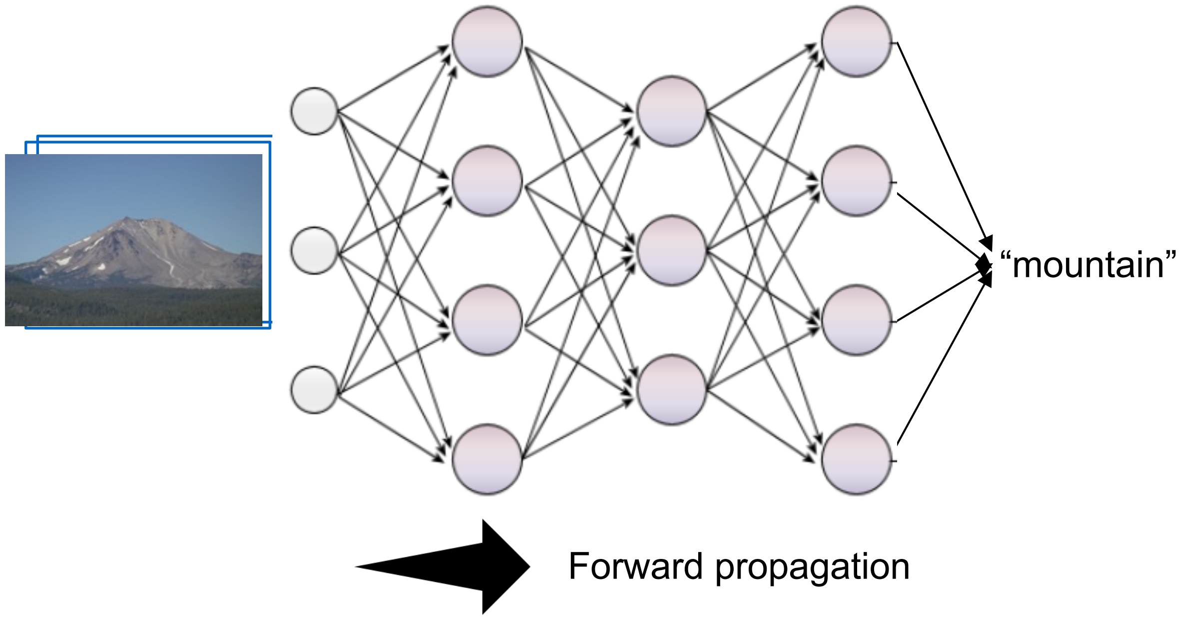
- Supervision - rely on labeled data
  - Transfer learning: a pre-trained model, train few layers
  - Learning label
- Involves a lot of hyperparameter tuning
  - Example: #layers, #neurons, batch size …
  - Multi-model training on same data set
  - Trial and error search - easier to automate
- Rise of AutoML
  - Learn how to model given data – no modeling/tuning expertise required
  - Example: AmoebaNet beat ResNet Image Classification



Deep Learning Inference

Inference*: use a trained model on new data

- Is computationally simpler
  - single forward pass
- Typically a containerized RPC/Web server
  - with pre-installed DL software + NN model
- Multiple inputs are batched for better throughput
  - But much smaller than training batch
  - Low latency

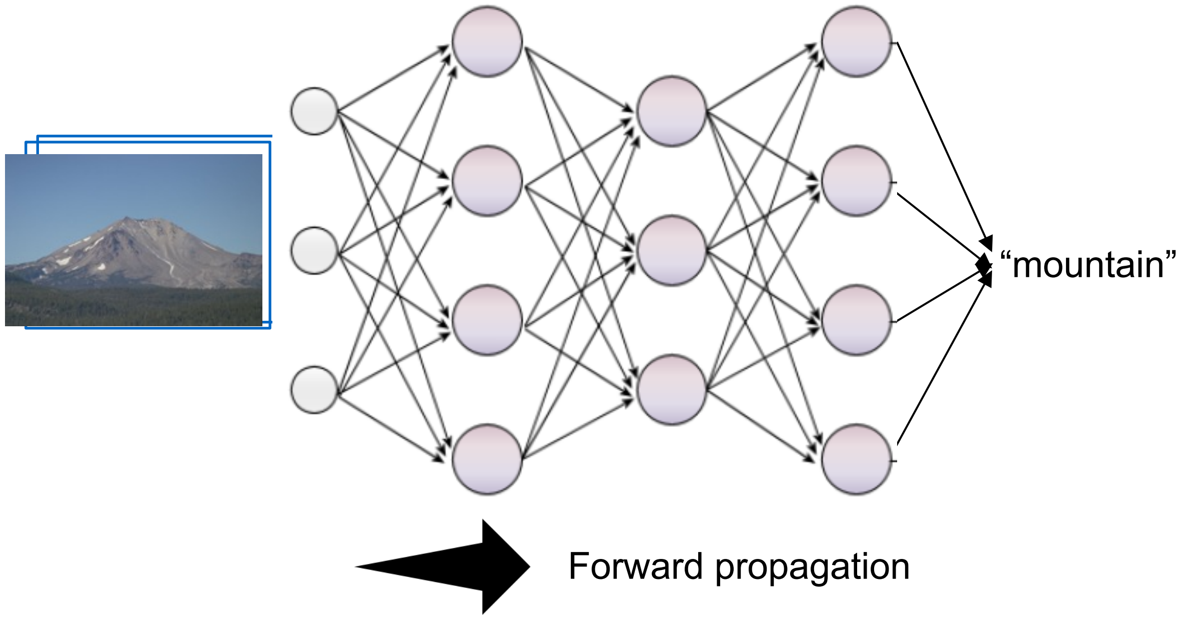


* aka Model Serving, Deployment, Prediction

Deep Learning Inference

Inference*: use a trained model on new data

- DL models can be huge
  - may need hardware acceleration
- On-device/Edge inference is gaining traction
  - Reason: latency & privacy
  - special model optimizations – pruning
  - hardware on-device
- Portability and interoperability of model is important
  - Train any way, deploy anywhere
  - Example: ONNX is a step towards standardizing



* aka Model Serving, Deployment, Prediction

Deep Learning Systems

Hardware Acceleration
Software Frameworks

Role of CPU in AI

- CPUs are still used for ML training
- CPUs are common for inference
  - including certain DL inference
- Struggle to handle DL training
- Data preprocessing are suited for CPUs
- Hybrid hardware of CPUs with other accelerators is common

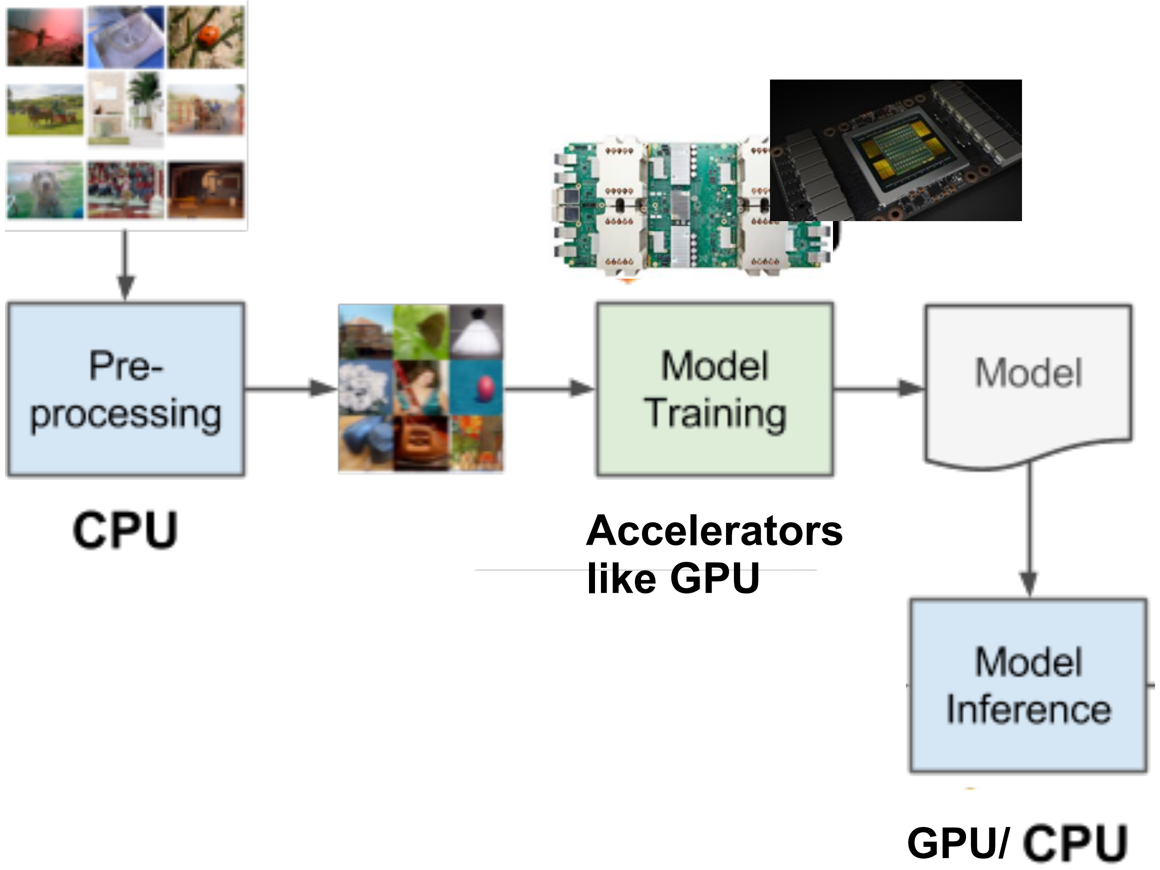


Image source: https://www.oreilly.com. Courtesy Daniel Whitenack.

Hardware Acceleration for DL

GPU (Graphic Processing Unit)

- De facto hardware for AI training
  - Also for large scale inference
- GPU vs CPU : many more cores, parallelization
- Modern GPU architectures used for AI
  - High speed interconnect between CPU/GPUs (NVLink)
  - Bypass CPU for communication (GPUDirect)
  - Efficient message passing (Collective-All-Reduce)
- Available in cloud (EC2 P* instances) and on-premise (DGX)

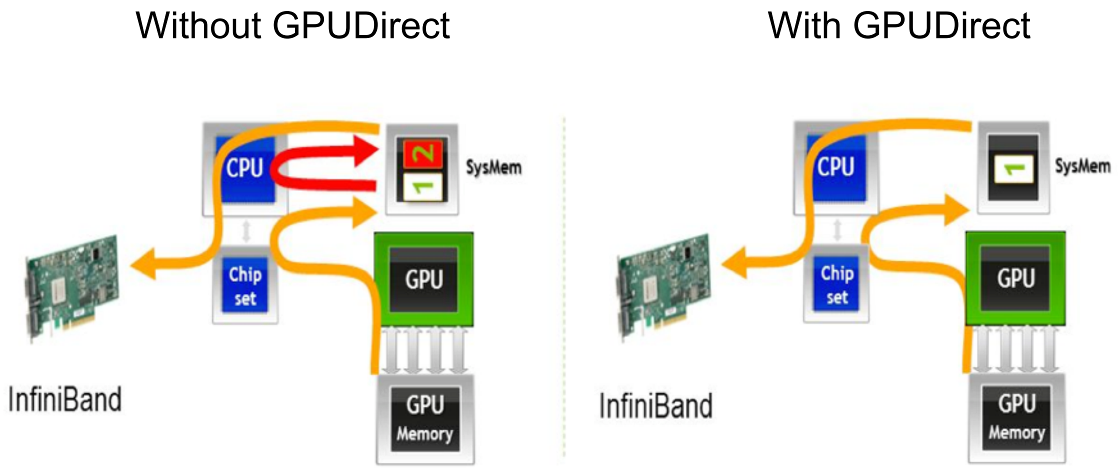


Image source: https://www.nvidia.com

Hardware Acceleration for DL

ASIC (Application Specific Integrated Circuit)

- ASIC designed to speed up DL operations, like Google’s TPU (Tensor Processing Unit)
  - High performance
  - Less flexible
  - Economical only at large scale
- Special optimizations in hardware
  - For example: reduced precision, matmul operator
- Design for inference is different from that for training
  - For example: in 1st generation TPUs fp-units were replaced by int8-units

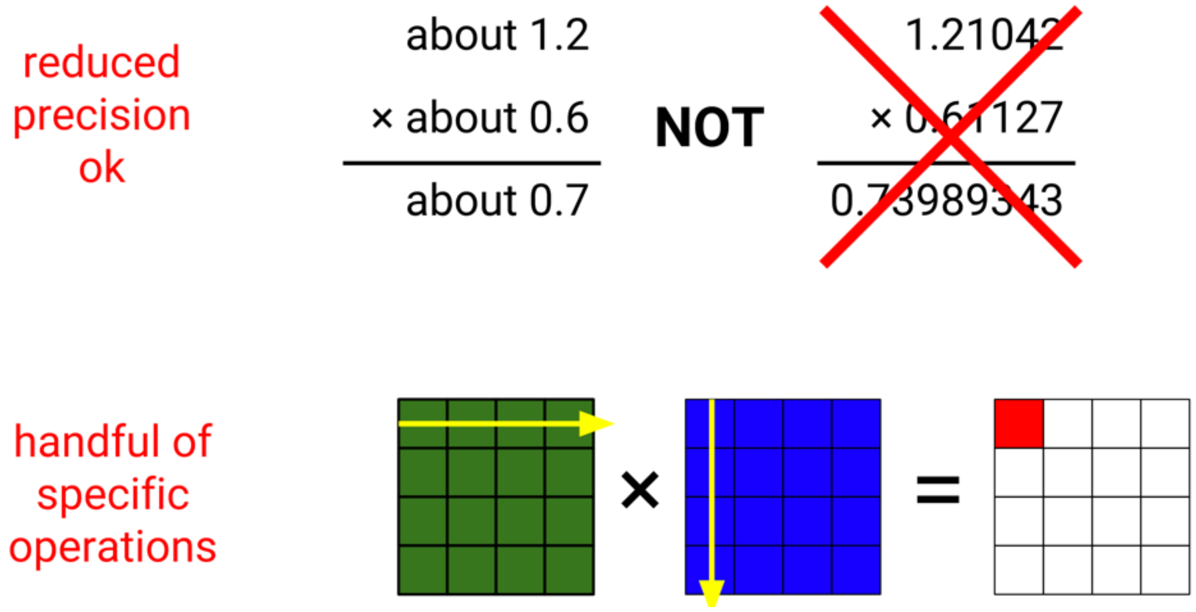
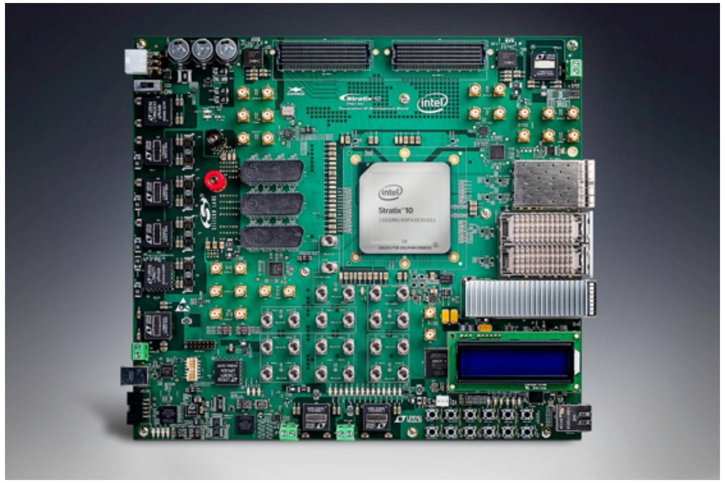


Image source: http://learningsys.org/nips17/assets/slides/dean-nips17.pdf

Hardware Acceleration for DL

FPGA (Field Programmable Gated Arrays)

- Designed to be reconfigurable
  - Flexibility to change as neural networks and new algorithms evolve
- Offer much higher Performance/Watt than GPUs
- Cost effective and excel at inference
- Reprogramming an FPGA is not easy
  - Low level language
- Available on cloud (EC2 F1 instances)



Hardware Acceleration for On-device AI

- Primarily limited to inference-only
- Special SoC design with reduced die space
- Energy efficiency and memory efficiency is more critical
- Special optimization to support specific tasks-only
  - For example: speech-only, vision-only
- Examples: Apple’s Neural Engine, Huawei’s NPU

Software Frameworks

Frontend

- Abstracts the mathematical and algorithm implementation details of Neural Networks
- Provides a high level building blocks API to define neural network models over multiple backends
- A high level language library

Backend

- Hides hardware-specific programming APIs from user
- Optimizes and parallelizes the training and inference process to work efficiently on the hardware
- Makes it easier to preprocess and prepare data for training
- Supports multi-GPU, multi-node execution

NetApp

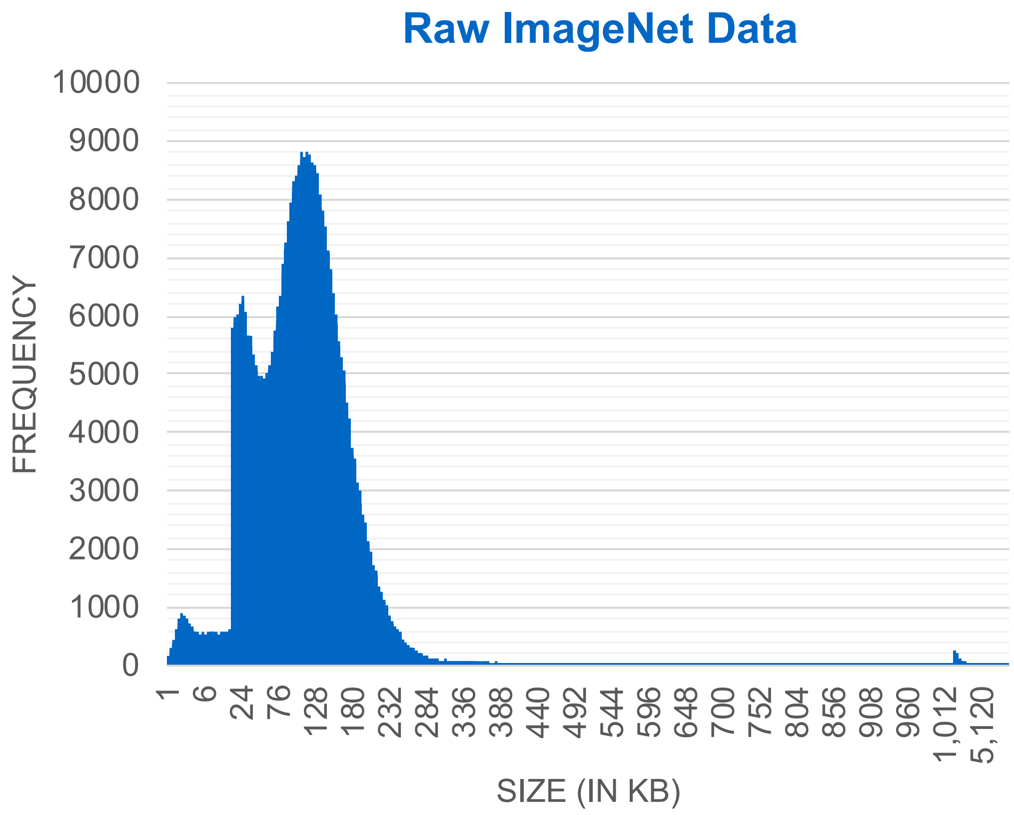
Dataset & Data Flow

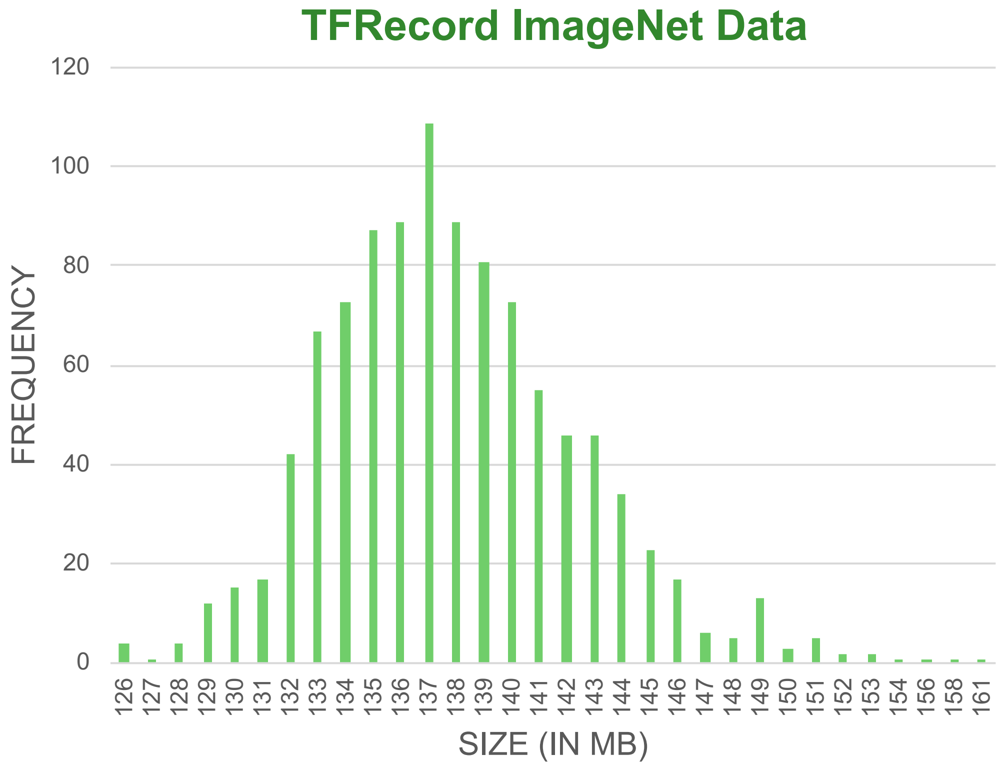
Using Tensorflow as reference

Dataset Transformation – ImageNet Example

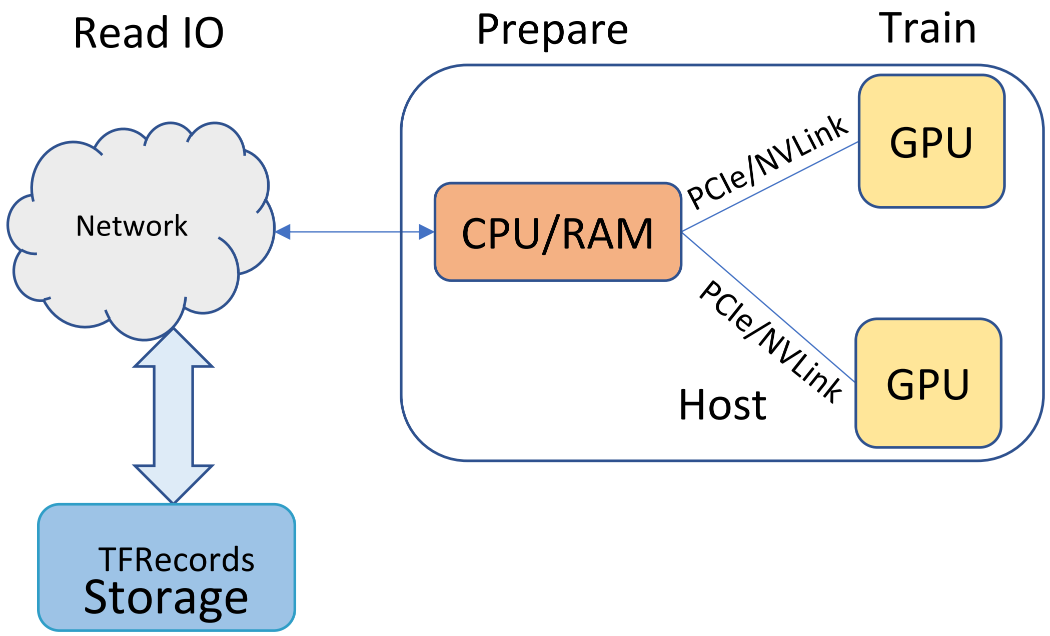
Rawdata vs TFRecords

- Raw data is converted into packed binary format for training called TFRecord (One time step)
  - 1.2 M image files are converted into 1024 TFRecords with each TFRecord 100s of MB in size





TensorFlow Data Pipeline



1. **IO**: Read data from persistent storage
2. **Prepare**: Use CPU cores to parse and preprocess data
   - Preprocessing includes Shuffling, data transformations, batching etc.
3. **Train**: Load the transformed data onto the accelerator devices (GPUs, TPUs) and execute the DL model

Compute Pipeling

- Without pipelining

- With Pipelining (using prefetch API)

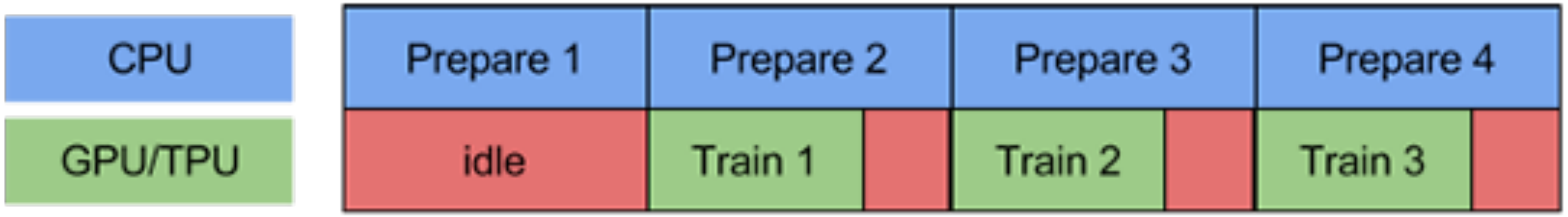
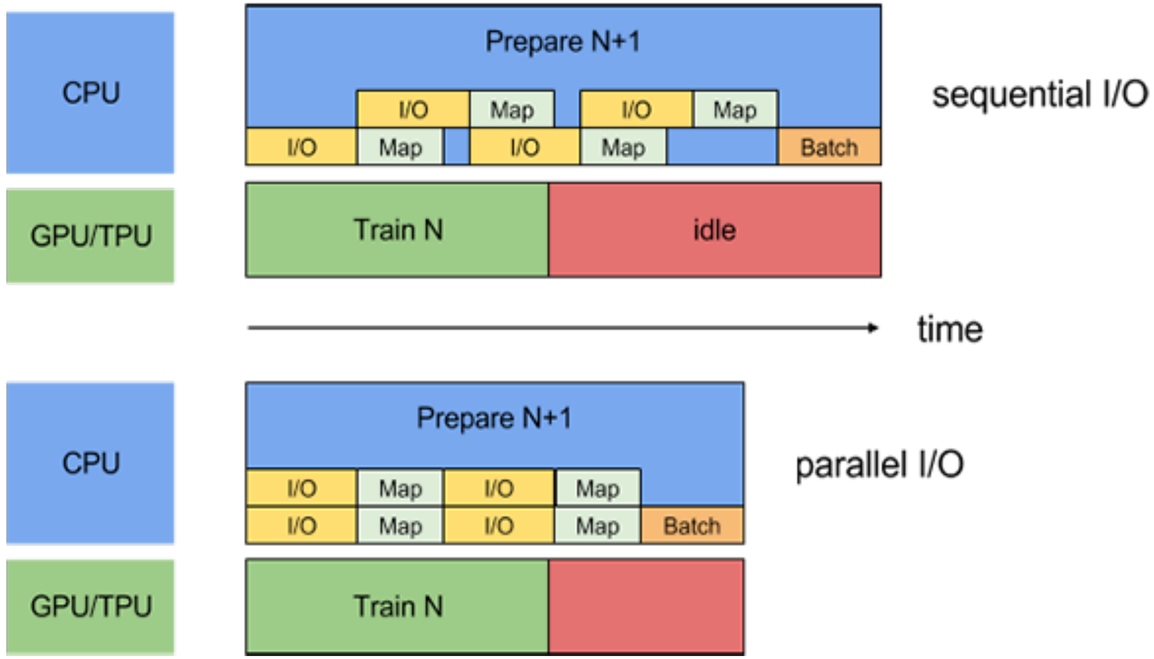


Image source: https://www.tensorflow.org/guide/

Parallelize IO and Prepare Phase

NetApp

- Parallelize IO



- Parallelize prepare

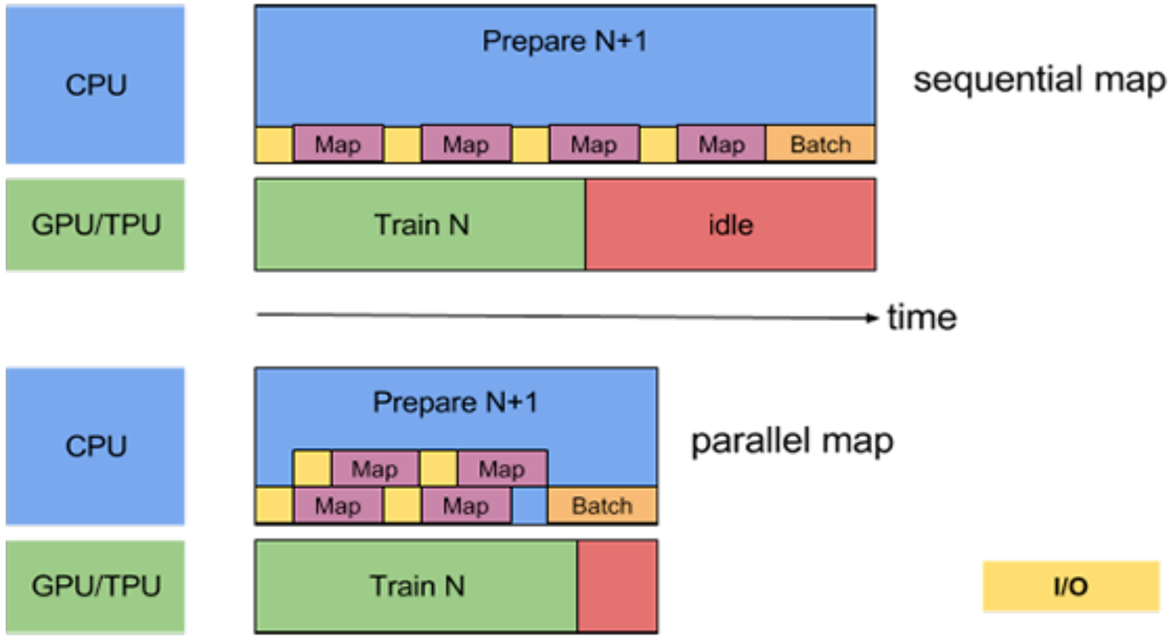


Image source: https://www.tensorflow.org/guide/

Future of Systems

AI for systems

Future of Systems

AI Applied to Systems

- Detect, predict, alert -- assist the experts
- Simplify deployment and operational complexity of software
  - Scheduling
  - Resource management
  - Configuration tuning
- Automatically adapt design tradeoffs within software
  - Replace heuristics – e.g. read/write ratios
  - Replace data structures – e.g. index lookup

Key Takeaways

- Current state of Deep Learning systems
- Critical role well-engineered systems & solutions play to make AI practical
- Impact on future of systems: applied AI in systems software design

Research Directions in AI

Systems related

NetApp

Academic:

- Using DL to replace heuristics-based decision within systems software, or even data structures
- Systems and platforms for DL
- Practical engineering optimizations to improve DL process/lifecycle/performance
- Workload and benchmarking
- Other areas like security, privacy, power etc.

Industry:

- Google Brain: hardware, AutoML
- FAIR: vision, video, AR
- Apple: speech & vision on-device

Carnegie Mellon University

STANFORD UNIVERSITY

Berkeley UNIVERSITY OF CALIFORNIA

Google

amazon