Hands on with Apache MXNet

On the Amazon Deep Learning AMI

Ric Harvey - Technical Developer Evangelist
@ric__harvey

AI and ML

- Artificial Intelligence: design software applications which exhibit human-like behavior, e.g. speech, natural language processing, reasoning or intuition
- Machine Learning: teach machines to learn without being explicitly programmed
- Deep Learning: using neural networks, teach machines to learn from complex data where features cannot be explicitly expressed

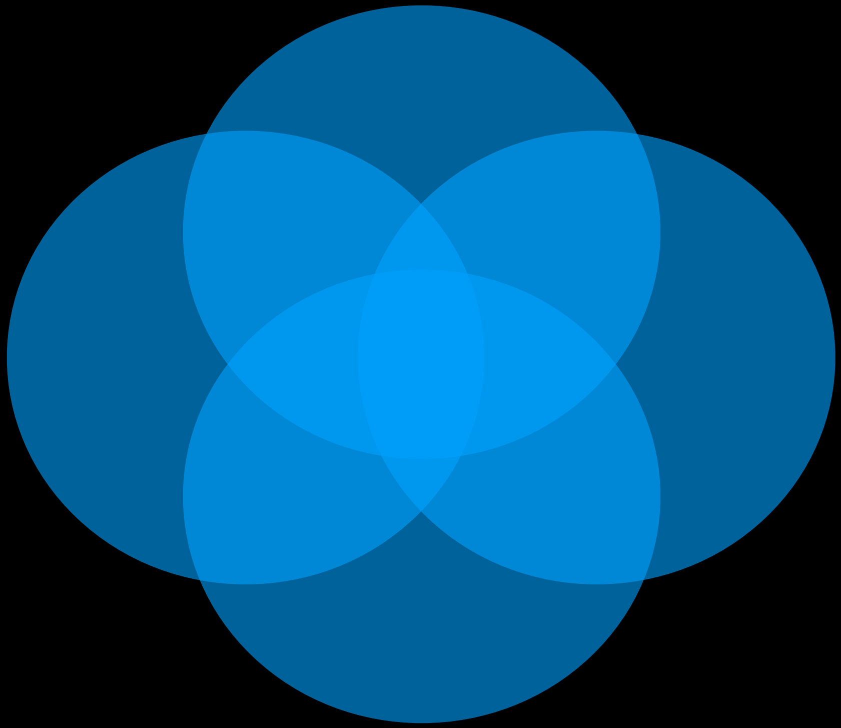
Deep learning

Deep Learning is a subfield of machine learning concerned with algorithms inspired by the structure and function of the brain called artificial neural networks.

Data is passed through multiple non-linear transformations to generate a prediction, models use a cascade of multiple layers of nonlinear processing units for feature extraction and transformation. Each successive layer uses the output from the previous layer as input.

Objective: Learn the parameters of the transformations that minimize a cost function

The advent of Deep Learning



Algorithms

Programming Models

Data

Acceleration (GPU’s)

Uses of Deep Learning

Image understanding

- Expedia have 10M+ images from 300K+ hotels
- Images boost the conversation around a hotel
- So having the best images matter
- They used Keras and EC2 GPU instances and fine tuned a retrained model

Speech recognition

- Key word trigger

Natural language processing

- Intents....

Autonomy

- Object Segmentation
- Last June, tuSimple drove an autonomous truck for 200 miles from Yuma, AZ to San Diego

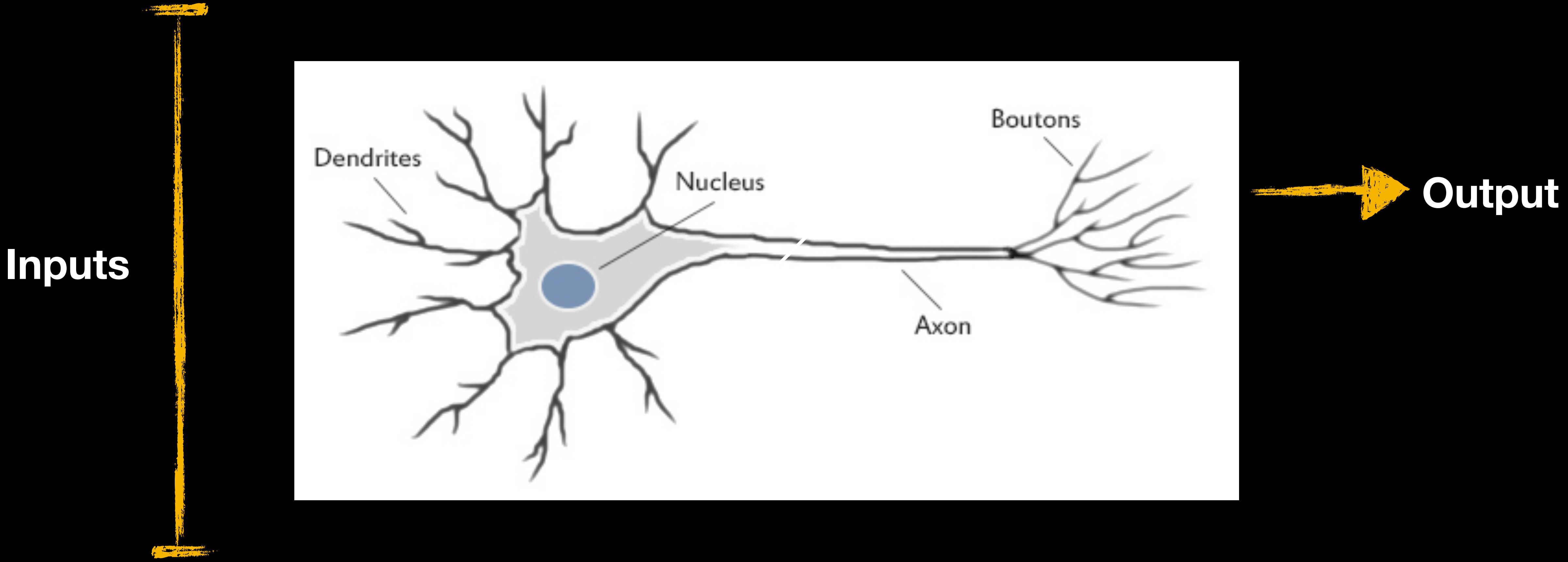
Customers Running AI on AWS

NETFLIX
The Washington Post
realnetworks
Expedia
Stanford
图森 tu Simple
HAPPY SNAP
zmags
HubSpot
OhioHealth
RNIB
duolingo
Pinterest
iTranslate
intel
slack
NVIDIA
WOLFRAM
Carnegie Mellon
American Heart Association
MapBox
Zillow
NASA
SAMSUNG SmartThings
FINRA
GoAnimate
W UNIVERSITY of WASHINGTON
Capital One
C-SPAN

And many more…

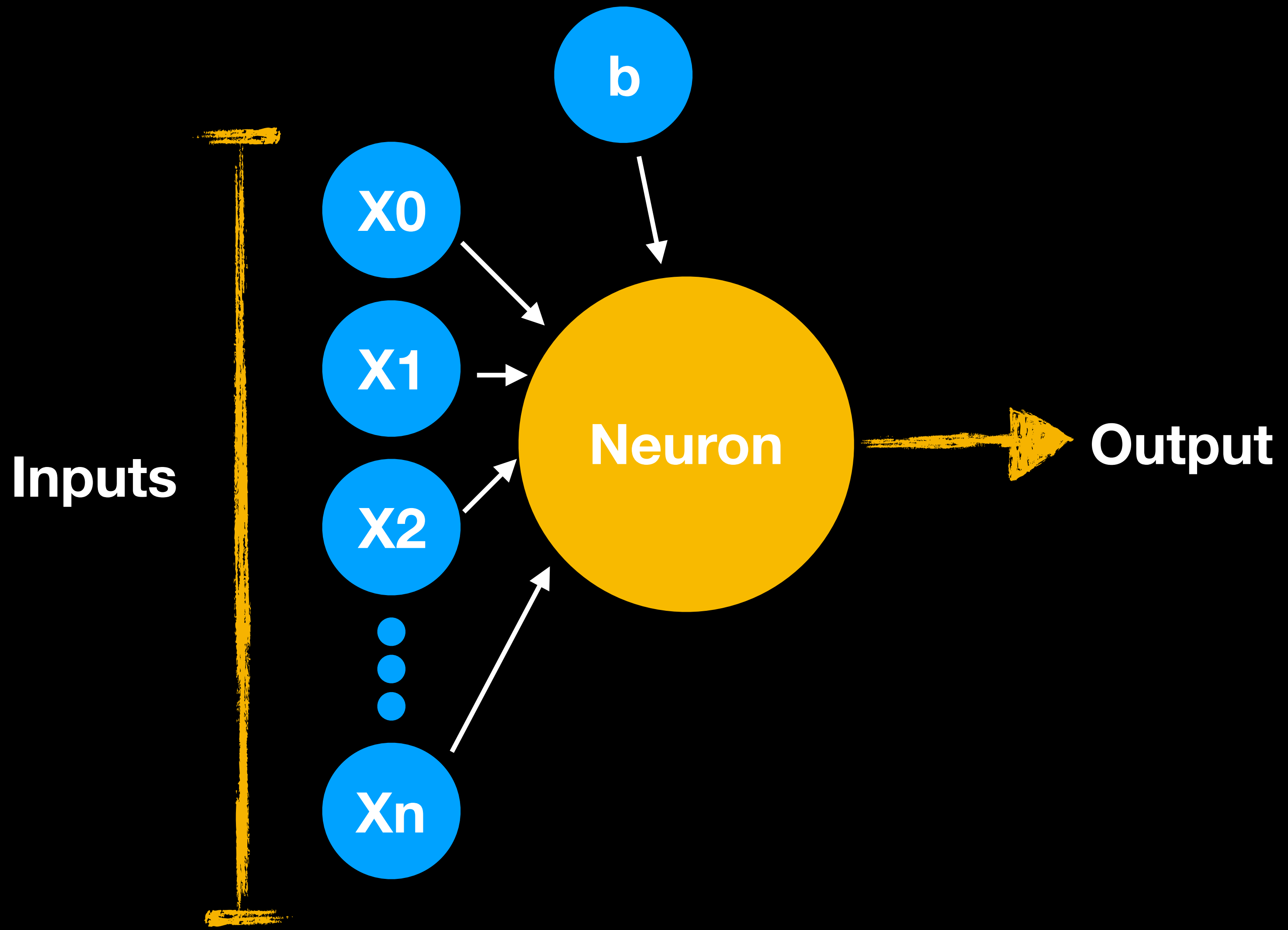
How does Deep Learning work?

Human Neuron



Artificial Neuron and Perception

- Input
  - Vector training of data set (x)
- Output
  - Liner function of input
- Nonlinearity
  - Transformation of output into value range
- Training
  - Learn the weights and bias (b) by minimize loss



$f(x) = \sigma(\langle w, x \rangle + b)$

Models of Neural Networks

Lots of types, feed forward, recurrent neural network, radial based function, regulatory feedback and so on….

- **Convolution Neural Network(CNN)**
  - **Feedforward network**
  - **Inspired by the visual cortex and responds to stimuli in a restricted area**
  - **Good for image processing**
- **Long Short Term Memory(LSTM)**
  - **Propagates data forward and also backwards from later stages to earlier**
  - **LSTM out performed every other RNN model**

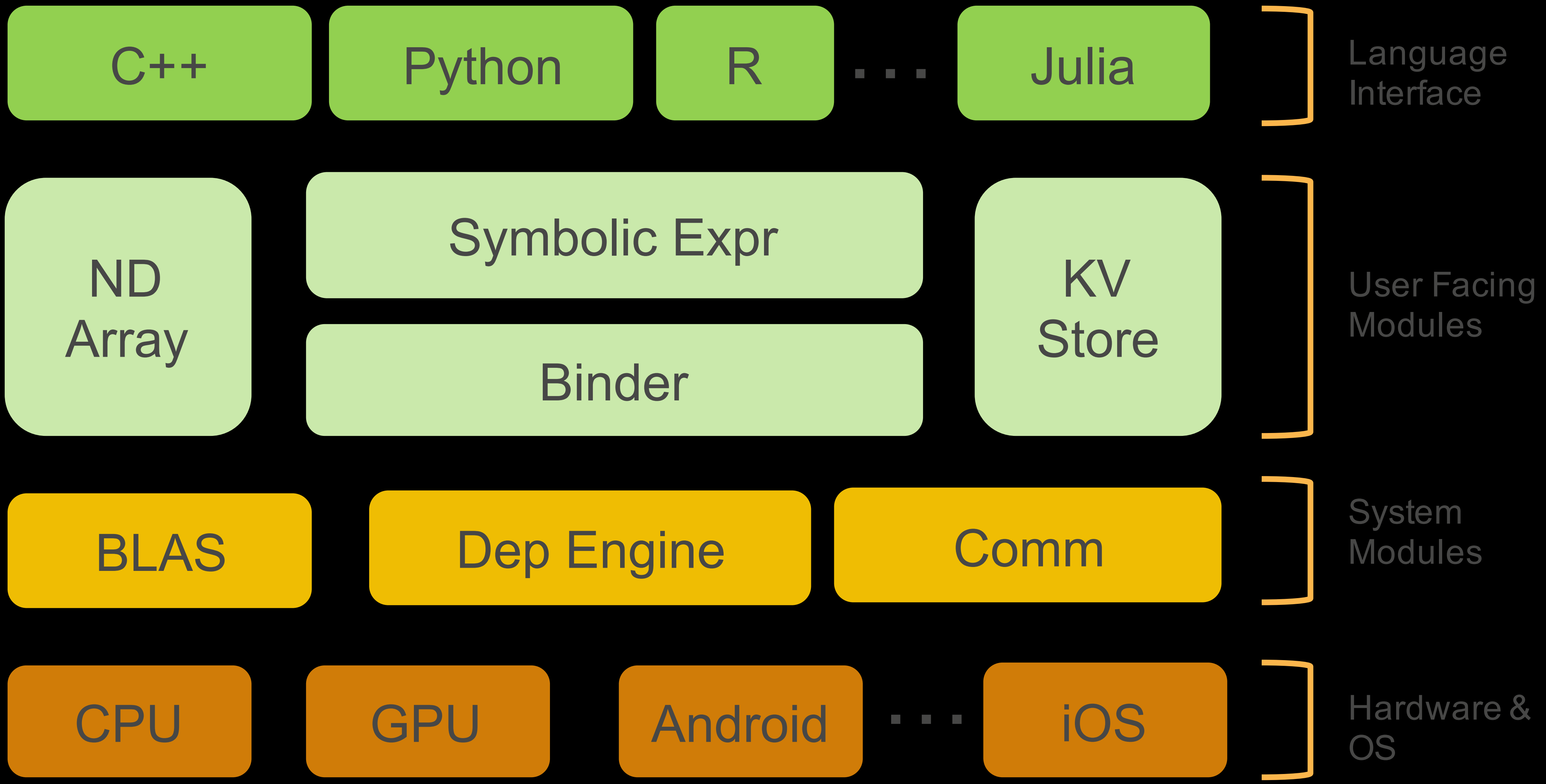
Apache MXNet

https://mxnet.apache.org/

Features

- **Flexible:** Supports both imperative and symbolic programming
- **Portable:** Runs on CPUs or GPUs, on clusters, servers, desktops, or mobile phones
- **Multiple Languages:** C++, Python, R, Scala, Julia, Matlab, Javascript, and Perl
- **Distributed on Cloud:** Supports distributed training on multiple CPU/GPU machines
- **Performance Optimized:** Optimized C++ backend engine parallelizes both I/O and computation
- **Broad Model Support:** CNN, RNN/LSTM

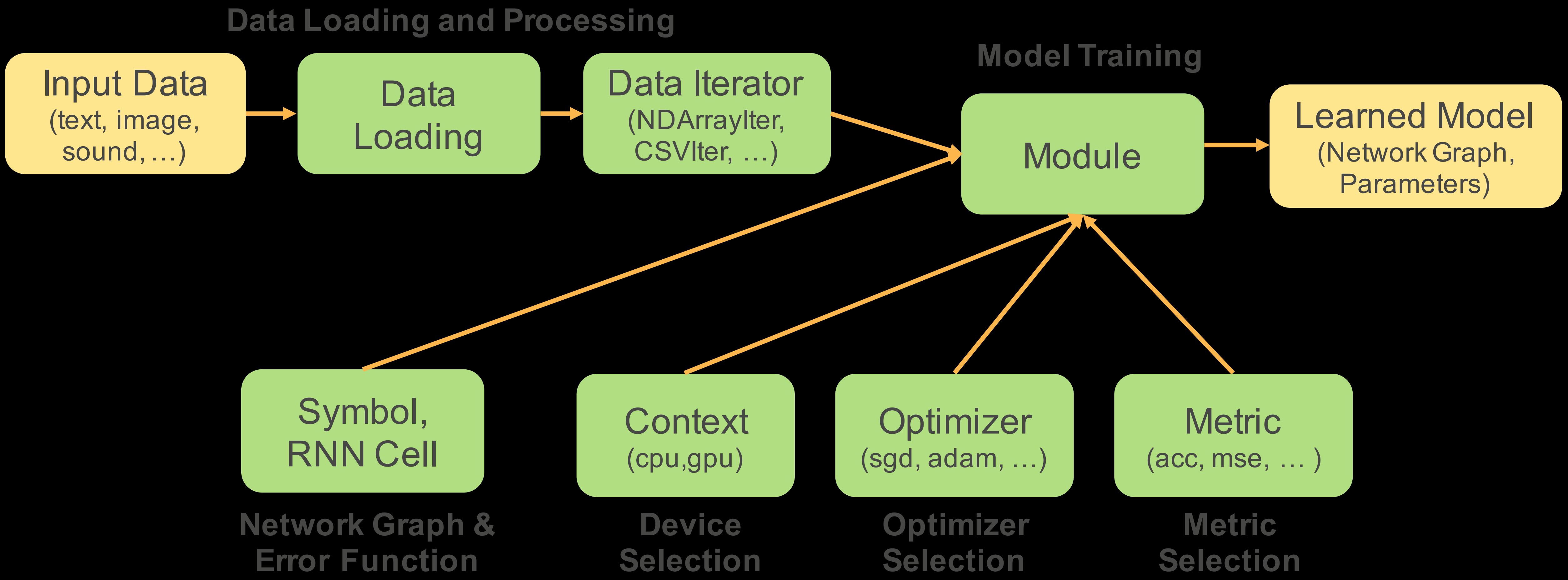
MXNet architecture



MXNet API Components

- **NDArray:** Provides imperative tensor operations
- **Symbol:** Provides neural network graph and auto-differentiation
- **RNN Cell:** Tools for building RNN symbolic graph
- **Module:** Provides interface for performing computation with Symbol
- **Data Loading:** Provides iterators for reading data
- **Metric:** Evaluation metric to evaluate performance of trained model

Model training flow in MXNet



Data Loading and Processing

Input Data (text, image, sound, …)

Data Loading

Data Iterator (NDArrayIter, CSVIter, …)

Model Training

Module

Learned Model (Network Graph, Parameters)

Symbol, RNN Cell

Network Graph & Error Function

Context (cpu,gpu)

Device Selection

Optimizer (sgd, adam, …)

Optimizer Selection

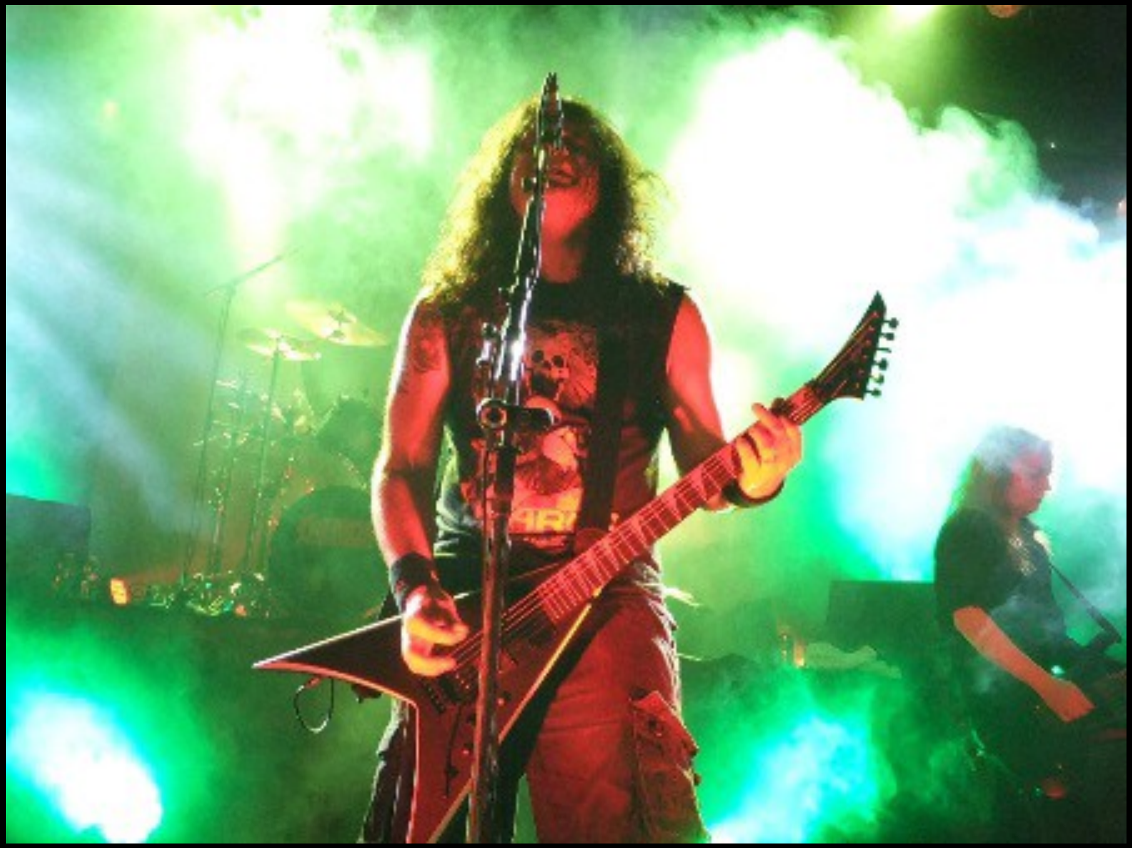
Metric (acc, mse, …)

Metric Selection

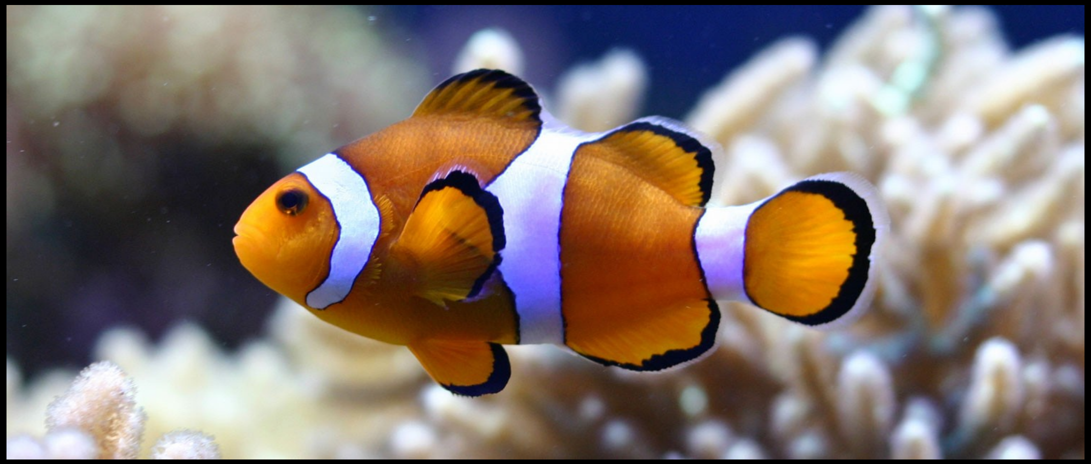
Workshop

MXNet with a pre-trained model

Exercise



Use the Amazon Deep Learning AMI to identify whats in these pictures and compare pre-trained models.



Conclusion

Model Comparison

How much memory does it use?

We can take an educated guess by looking at the size of the parameters file

- VGG16: 528MB (about 140 million parameters)
- ResNet-152: 230MB (about 60 million parameters)
- Inception v3: 43MB (about 25 million parameters)

How fast can it predict?

This is more difficult and can depend on batch size but in our example, lets look at the average over a few calls

```
*** VGG16
Predicted in 0.30 millisecond
*** ResNet-152
Predicted in 0.90 millisecond
*** Inception v3
Predicted in 0.40 millisecond
```

Summary

In these tests

- ResNet-152 has the best accuracy of all three networks (by far) but it's also 2–3 times slower.

- VGG16 is the fastest—due its small number of layers?—but it has the highest memory usage and the worst accuracy.

- Inception v3 is almost as fast, while delivering better accuracy and the most conservative memory usage. This last point makes it a good candidate for constrained environments.