Scaling Deep Learning Algorithms on Extreme Scale Architectures

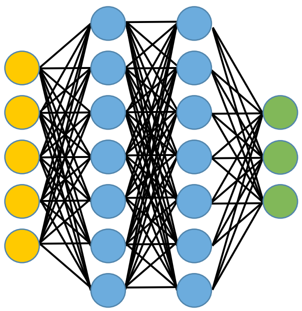
ABHINAV VISHNU

Team Lead, Scalable Machine Learning, Pacific Northwest National Laboratory

MVAPICH User Group (MUG) 2017

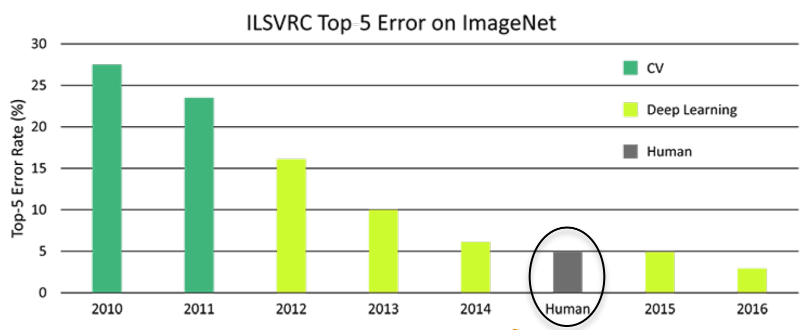
The rise of Deep Learning!

Pacific Northwest
NATIONAL LABORATORY
Proudly Operated by Battelle Since 1965



FeedForward

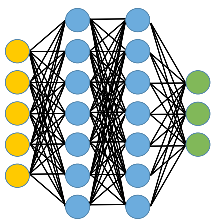
Back-propagation



Human accuracy!

Several scientific applications have shown remarkable improvements in modeling/classification tasks !!

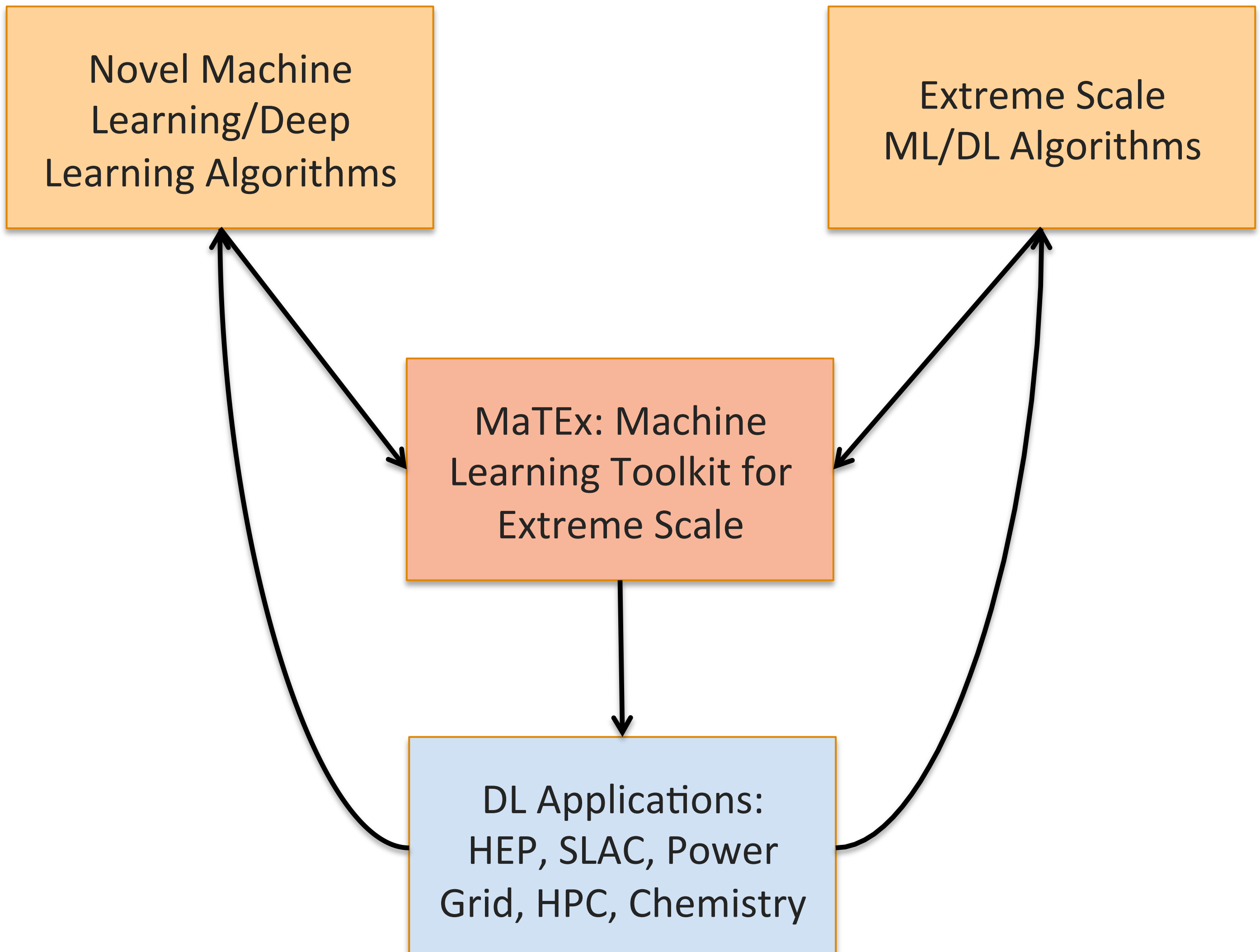
Challenges in Using Deep Learning



- How to design DNN topology?
- Which samples are important?
- How to handle unlabeled data?

- Supercomputers are typically used for simulation – effective for DL implementations?
- How much effort required for using DL algorithms?
- Will it only reduce time-to-solution or improve baseline performance of the model?

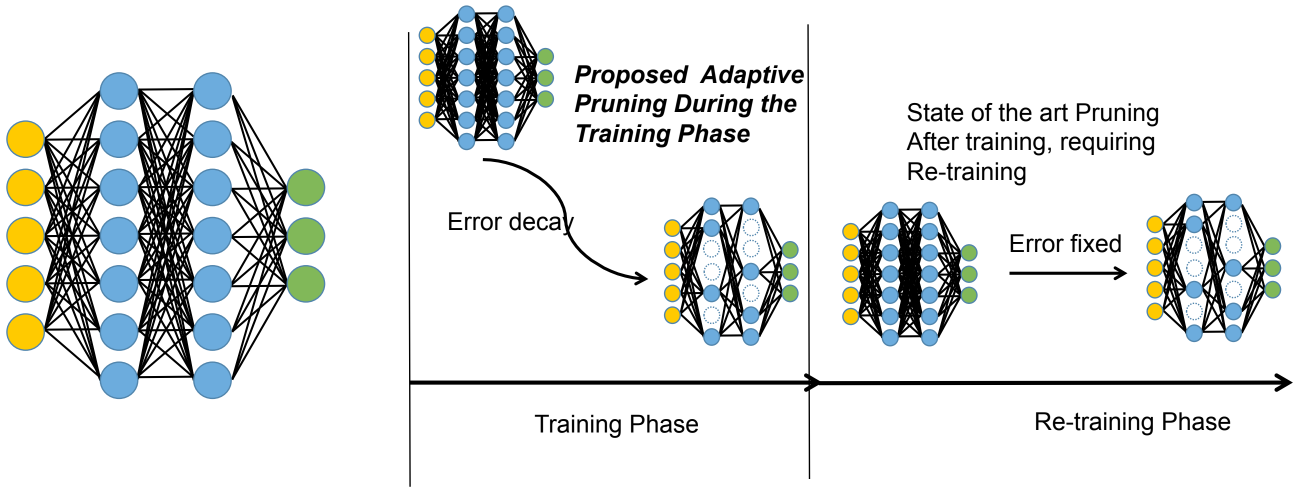
Vision for Machine/Deep Learning R&D



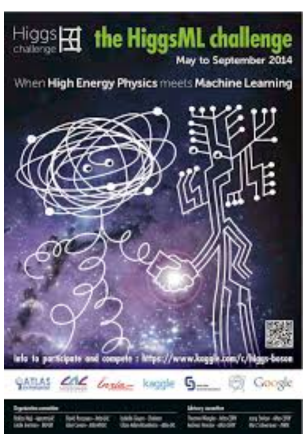
Novel ML/DL Algorithms: Pruning Neurons

Which neurons are important?

Adaptive Neuron Apoptosis for Accelerating DL Algorithms

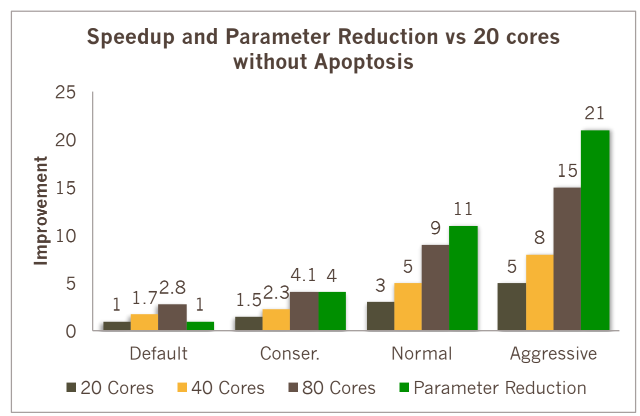


(a) (b)



Area Under Curve - ROC:

1) **Improved from 0.88 to 0.94**
2) **2.5x speedup in learning time**
3) **3x simpler model**

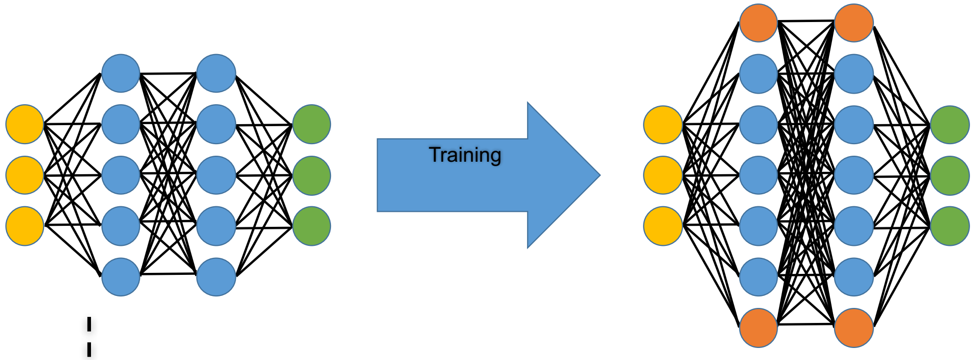


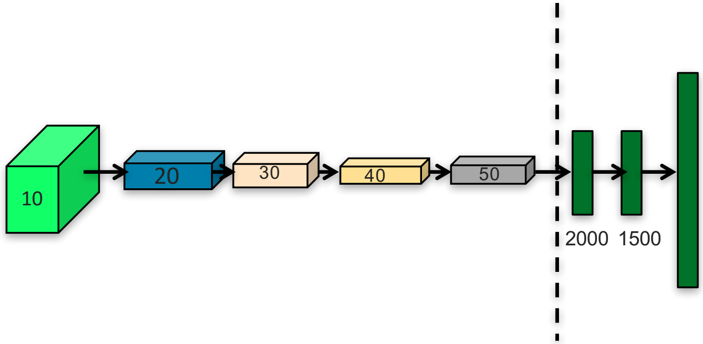
Novel ML/DL Algorithms: Neuro-genesis

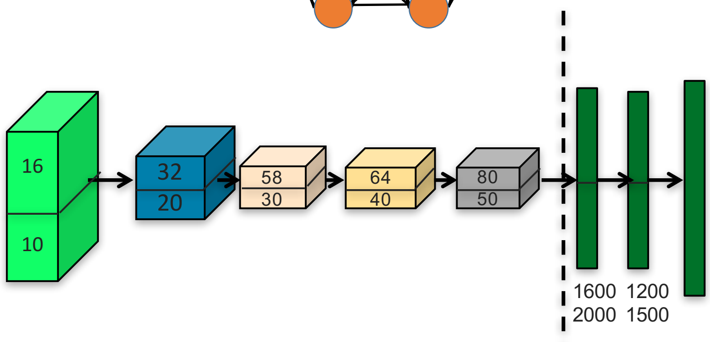
Pacific Northwest
NATIONAL LABORATORY
Proudly Operated by Battelle Since 1965

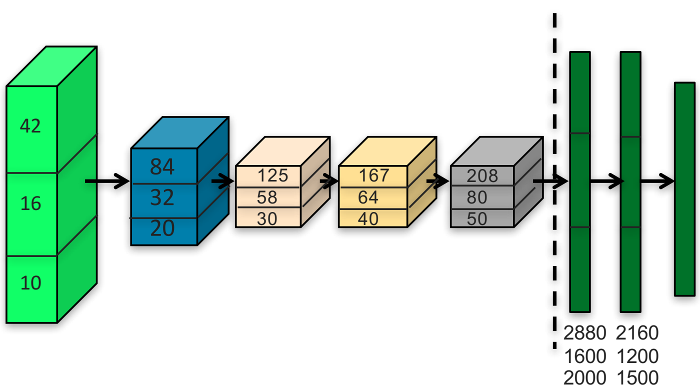
Can you create neural network topologies semi-automatically?

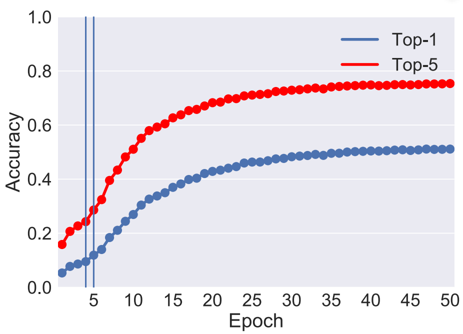
Generating Neural Networks from BluePrints











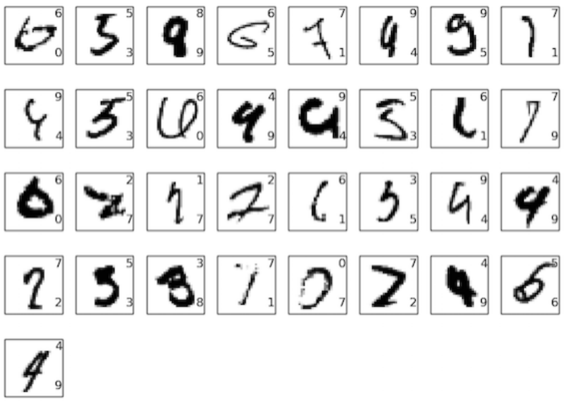
Novel ML/DL Algorithms: Sample Pruning

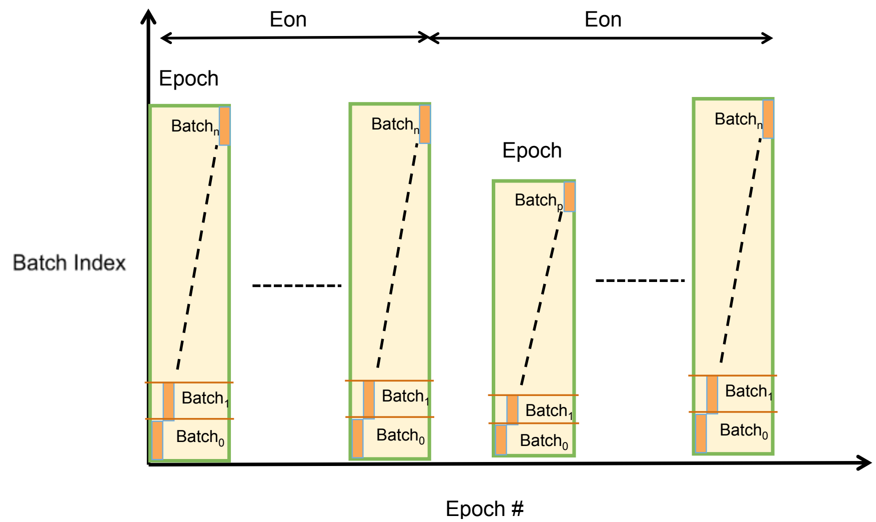
Pacific Northwest NATIONAL LABORATORY

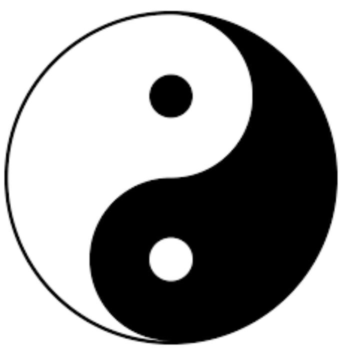
Proudly Operated by Battelle Since 1965

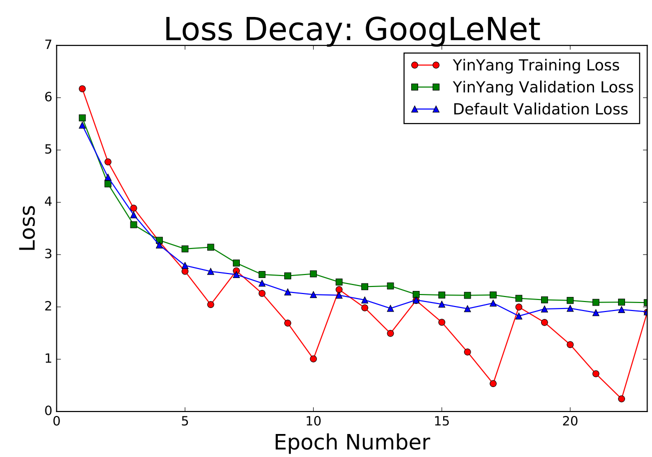
Which Samples are Important?

YinYang Deep Learning for Large Scale Systems

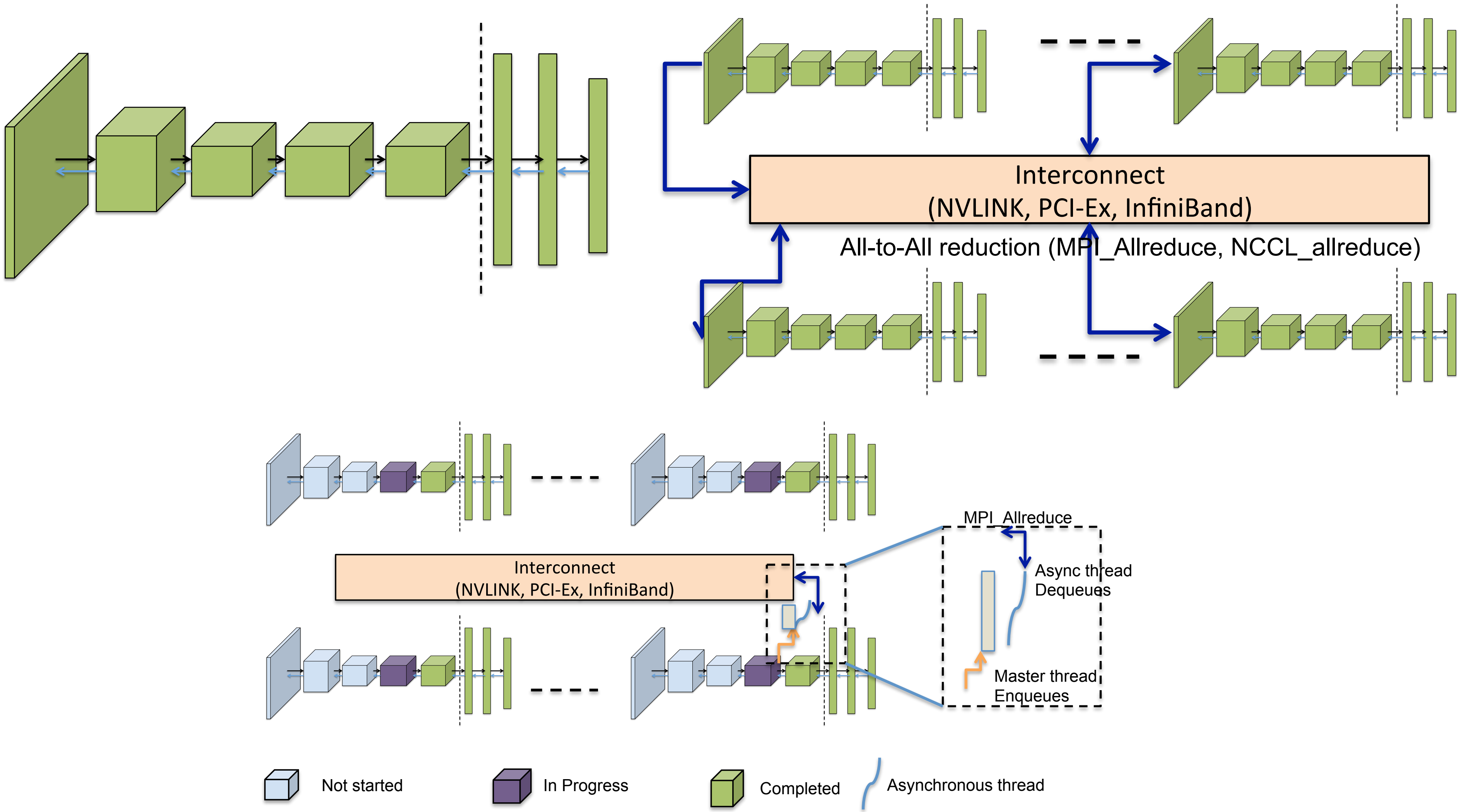




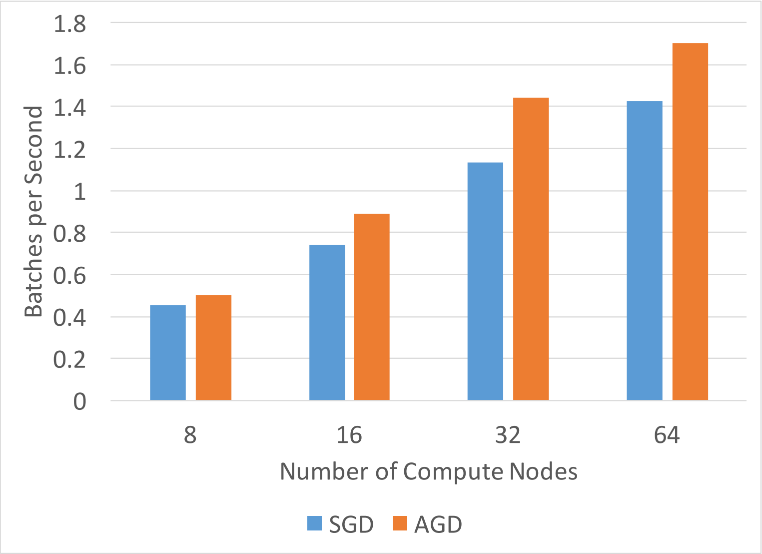




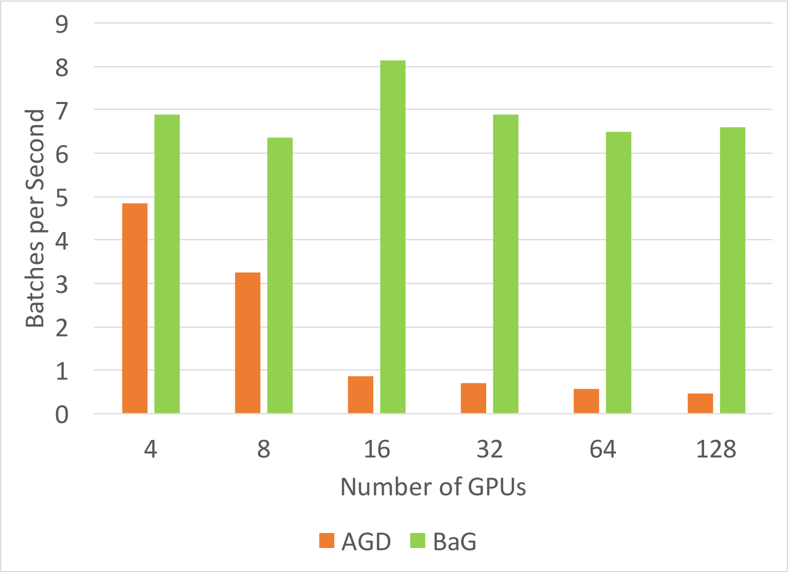
Scaling DL Algorithms Using Asynchronous Primitives



Sample Results



PIC
MVAPICH
Strong Scaling



SummitDev
IBM Spectrum MPI
Weak Scaling

What does Fault Tolerant Deep Learning Need from MPI?

MPI has been criticized heavily for lack of fault tolerance support

1) Existing MPI implementation
2) User-Level Fault Mitigation
3) Reinit Proposal

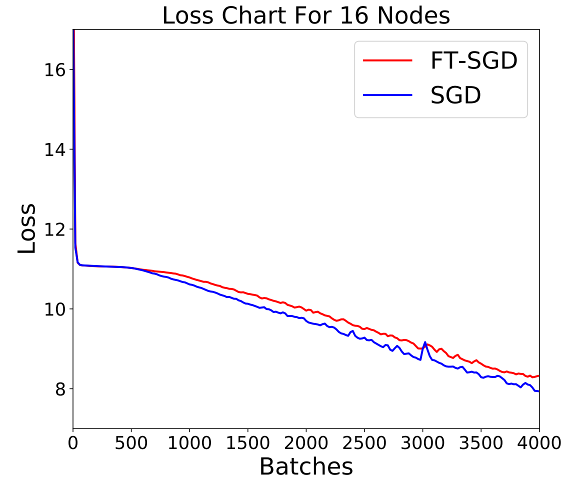
Which proposal is necessary and sufficient?

Code Snippet of Original Callback

```
...
// Original on_gradients_ready
On_gradients_ready(float *buf) {

// conduct in-place allreduce of gradients
 rc = MPI_Allreduce (... ,  ...);

// average the gradients by communicator size

...
```

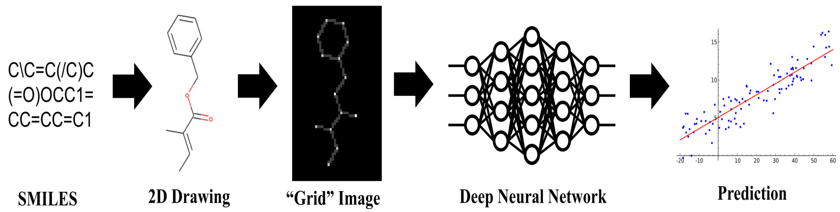
Code Snippet for Fault tolerant DL

```
...
// Fault tolerant on_gradients_ready
On_gradients_ready(float *buf) {

// conduct in-place allreduce of gradients
 rc = MPI_Allreduce (...,  ...);

While (rc != MPI_SUCCESS) {
// shrink the communicator to a new comm.
MPIX_Comm_shrink(origcomm, &newcomm);
rc = MPI_Allreduce(..., ...);
}
// average the gradients by communicator size
...
```



Impact of DL on Other Application Domains

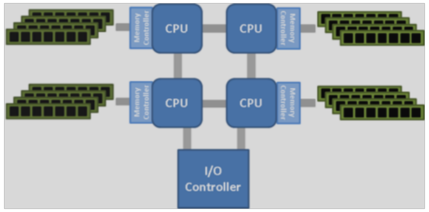
Can molecular structure predict the molecular properties?

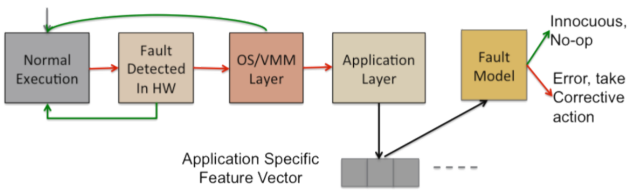


Computational Chemistry

HPC

When multi-bit faults result in application error?





What DL techniques are useful for Energy Modeling of Buildings?

Buildings, Power Grid

MaTEx: Machine Learning Toolkit for Extreme Scale

TensorFlow

K + TensorFlow

Argonne NATIONAL LABORATORY | Leadership Computing Facility

OLCF
OAK RIDGE LEADERSHIP COMPUTING FACILITY

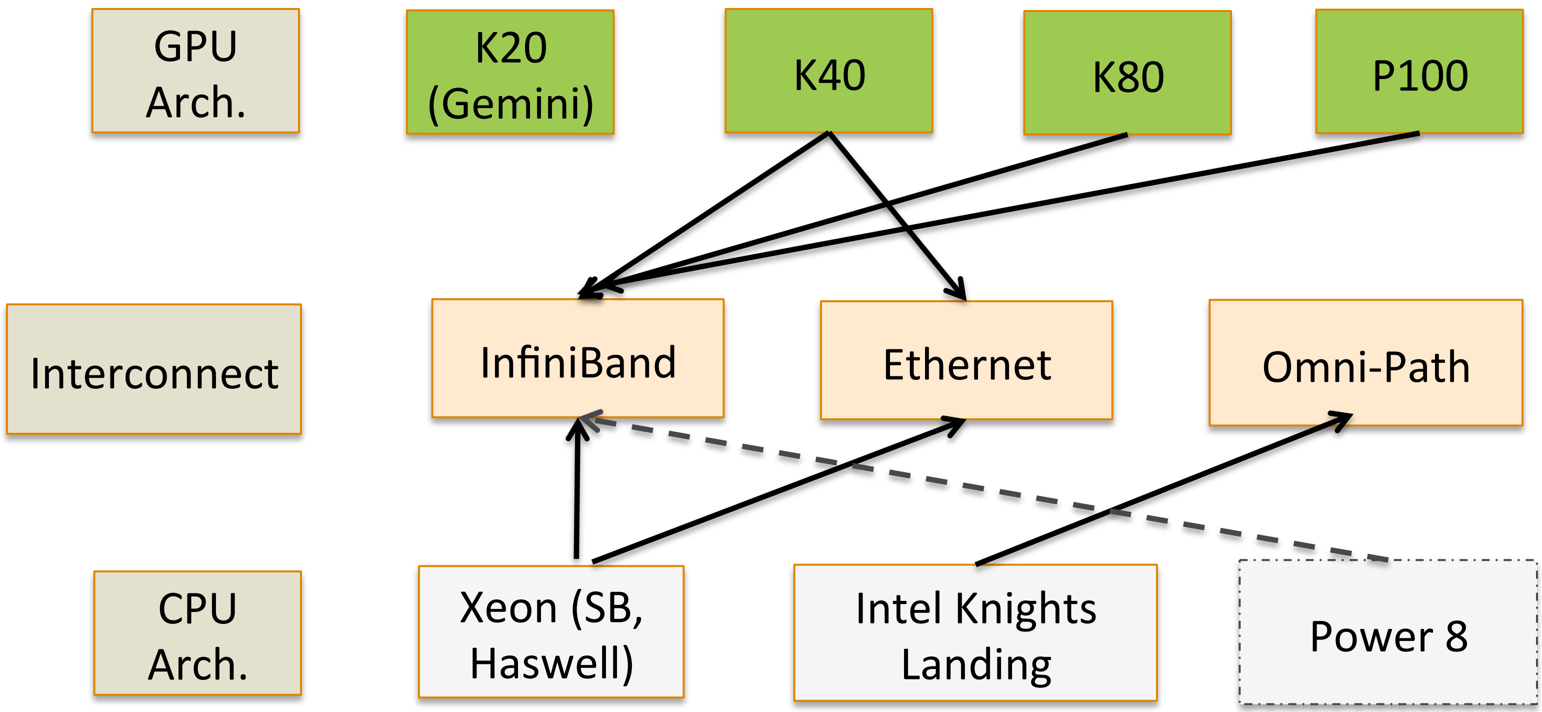
Caffe

NERSC

1) Open source software with users in academia, laboratories and industry
2) Supports graphics processing unit (GPU), central processing unit (CPU) clusters/ LCFs with high-end systems/interconnects
3) Machine Learning Toolkit for Extreme Scale -MaTEx: github.com/matex-org/ matex

Architectures Supported by MaTEx

Pacific Northwest NATIONAL LABORATORY
Proudly Operated by Battelle Since 1965



Comparing the Performance of NVIDIA DGX-1 and Intel KNL on Deep Learning Workloads, ParLearning'17, IPDPS'17

Demystifying Extreme Scale DL

Google-TensorFlow

Data Readers

TF Scripts
(gRPC)

TF Runtime

Architectures

Not attractive for scientists!

MaTEx-TensorFlow

Data Readers

TF Scripts

TF Runtime
(MPI Changes)

Architectures

Requires no TF specific changes for users

Supports automatic distribution of HDF5, CSV, PNetCDF formats

Example Code Changes

MaTEx-TensorFlow Code

```
import tensorflow as tf
import numpy as np
...
from datasets import DataSet
...
image_net = DataSet(...)
data = image_net.training_data
labels = image_net.training_labels
...
# Setting up the network
...
# Setting up optimizer
...
init = tf.global_variables_initializer()
sess = tf.Session()
sess.run(init)
...
# Run training regime
```

Original TF Code

```
import tensorflow as tf
import numpy as np
...
...
data = ...    # Load training data
labels = ... # Load Labels
...
# Setting up the network
...
# Setting up optimizer
...
init = tf.global_variables_initializer()
sess = tf.Session()
sess.run(init)
...
# Run training regime
```

Supports automatic distribution of HDF5, CSV, PNetCDF formats

User-transparent Distributed TensorFlow, A. Vishnu et al., Arxiv'17

User-Transparent Distributed Keras

```
import tensorflow as tf
import numpy as np
# Keras Imports
...
dataset = tf.DataSet(...)
data = dataset.training_data
labels = dataset.training_labels
...
# Defining Keras Model
...
# Call to Keras training method
...
```

```
import tensorflow as tf
import numpy as np
# Keras Imports
...

data = ...    # Load training data
labels = ... # Load Labels
...
# Defining Keras Model
...
# Call to Keras training method
...
```

1) **Distributed Keras with MPI available on github.com/matex-org/matex**
2) **Currently the only Keras implementation that does not require any MPI specific changes to code**
3) **Tested on NERSC architectures**

Use-Case: SLAC Water/Ice Classification

Reducing the time to new science **-** From Experiment to Publication

Typical Experiment:

1) ~100 images/sec
2) ~100 TB of data
3) Problem further exacerbated for upcoming LCLS-2 (up to 1M images/sec)
4) Several domains exhibit these characteristics

Typical Problems:

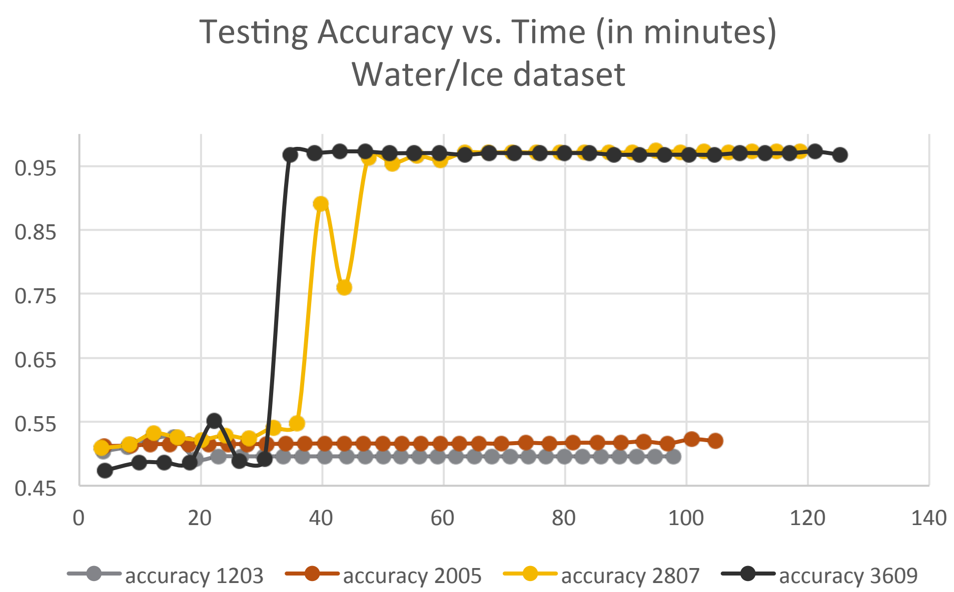
1) Too many images – can we find the important ones?
2) Unknown whether the experiment is on the "right track":
   1) Results not known till post-hoc data analysis
3) If the experiment succeeds:
   1) Exorbitant time spent (several man days) in data cleaning/labeling
   2) Several man days spent in manual data analysis (such as generating probability distribution functions)

Can we do better?

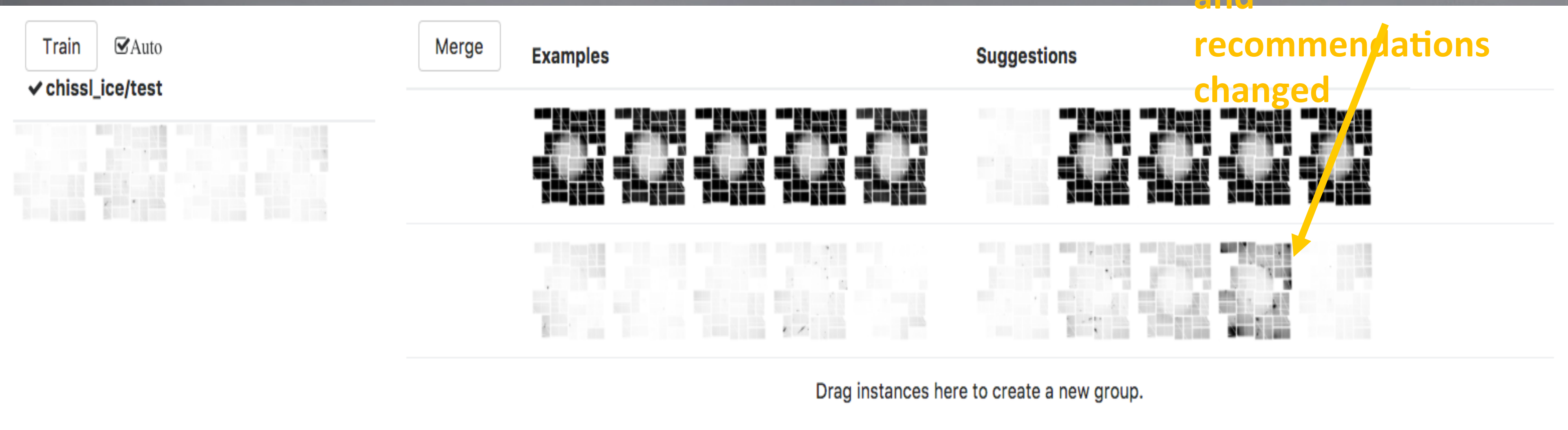
Sample Proof: Distinguishing Water from Ice

Dataset Specification:

1) ~68GB of data consisting of images with Water and Ice crystals
2) Scientists spent 17 man days labeling each image as representing Water or Ice
3) Objective – can we reduce the labeling time, while achieving very high accuracy?
    1) We take 4000 samples and consider following data splits:
        1) Label 1200 to 2800 samples using Deep Learning (Convolutional + Deep Neural architectures) and see the accuracy on remaining samples (2800 – 1200)
    2) **Observation:** With 2800 samples, we can accurately classify ~97% of remaining samples correctly
4) Conclusion: major reduction in labeling time with results matching human labeling
    1) Potential for significant reduction in time for scientific discovery
    2) Labeling only "boundary" samples would further reduce the human effort



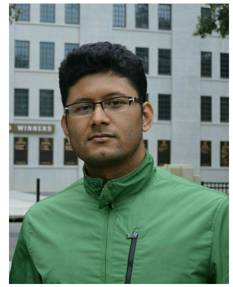
Prototype for Semi-Supervised Learning



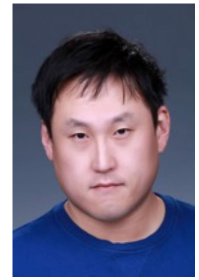
Collaborators

Jeff Daily

Charles Siegel



Vinay Amatya



Leon Song

Ang Li

Joseph Manzano

Garrett Goh

Malachi Schram

Vikas Chandan

Thomas J Lane@SLAC

Thanks!

TensorFlow

Argonne NATIONAL LABORATORY | Leadership Computing Facility

K + TensorFlow

OLCF
OAK RIDGE LEADERSHIP COMPUTING FACILITY

Caffe

NERSC

Contact: abhinav.vishnu@pnnl.gov
MaTEx webpage: https://github.com/matex-org/matex/
Publications: https://github.com/matex-org/matex/wiki/publications