DEEP LEARNING INSTITUTE

Neural Network Deployment with DIGITS and TensorRT

Twin Karmakharm

Certified Instructor, NVIDIA Deep Learning Institute

DEEP LEARNING INSTITUTE

DLI Mission

Helping people solve challenging problems using AI and deep learning.

- Developers, data scientists and engineers
- Self-driving cars, healthcare and robotics
- Training, optimizing, and deploying deep neural networks

TOPICS

- Caffe
- NVIDIA’S DIGITS
- Deep Learning Approach
- NVIDIA’S TensorRT
- Lab
    - Lab Details
    - Launching the Lab Environment
- Review / Next Steps

CAFFE

Frameworks

Many Deep Learning Tools

Caffe

PaddlePaddle

TensorFlow

theano

torch

...

WHAT IS CAFFE?

An open framework for deep learning developed by the Berkeley Vision and Learning Center (BVLC)

caffe.berkeleyvision.org
http://github.com/BVLC/caffe

- Pure C++/CUDA architecture
- Command line, Python, MATLAB interfaces
- Fast, well-tested code
- Pre-processing and deployment tools, reference models and examples
- Image data management
- Seamless GPU acceleration
- Large community of contributors to the open-source project

CAFFE FEATURES

Deep Learning model definition

Protobuf model format

- Strongly typed format
- Human readable
- Auto-generates and checks Caffe code
- Developed by Google
- Used to define network architecture and training parameters
- No coding required!

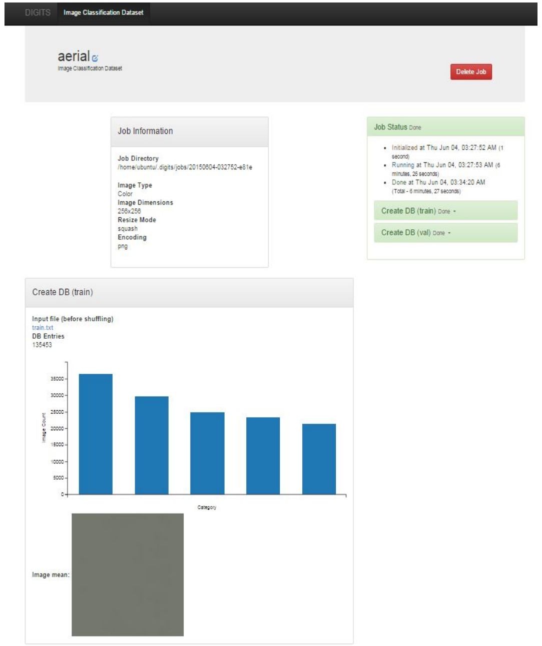
```
name: "conv1"
type: "Convolution"
bottom: "data"
top: "conv1"
convolution_param {
    num_output: 20
    kernel_size: 5
    stride: 1
    weight_filler {
        type: "xavier"
    }
}
```

NVIDIA’S DIGITS

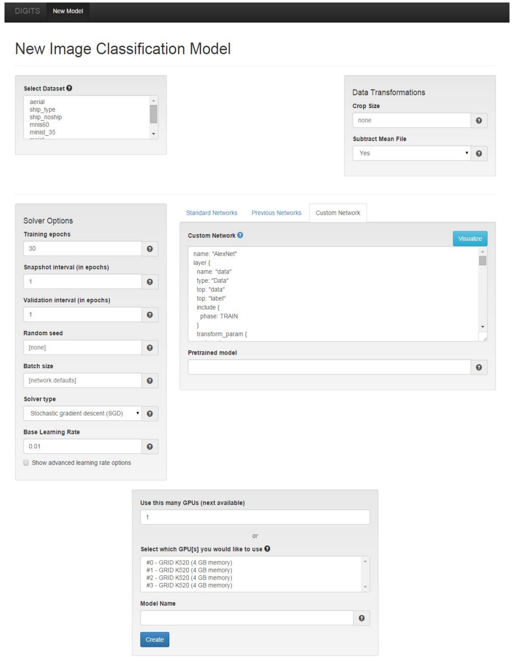
NVIDIA'S DIGITS

Interactive Deep Learning GPU Training System

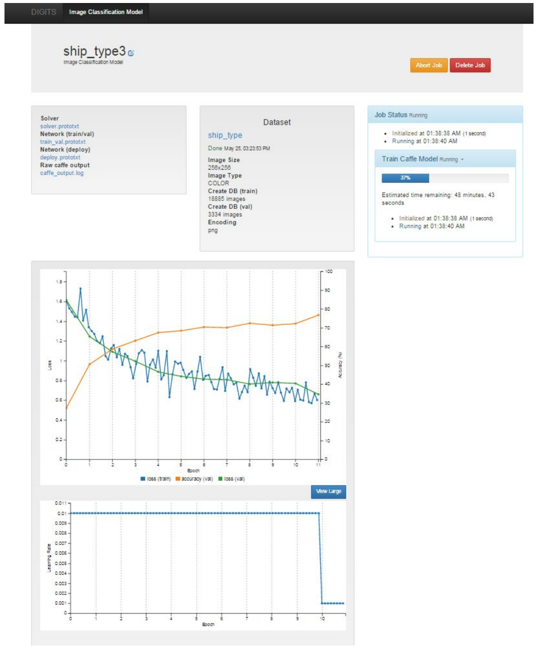
Process Data



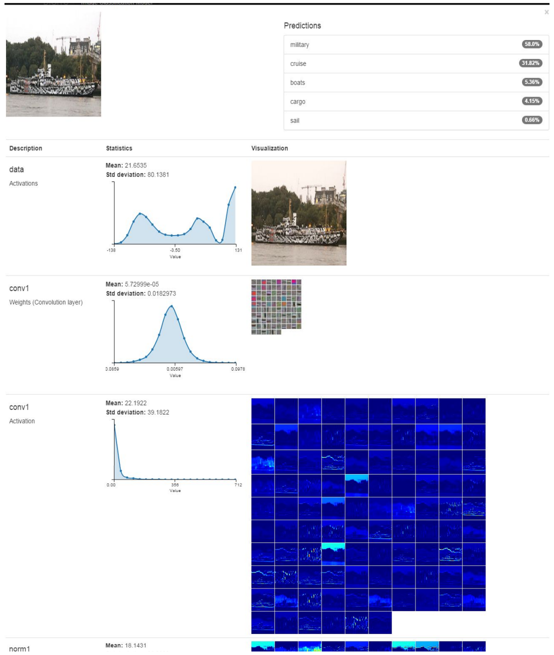
Configure DNN



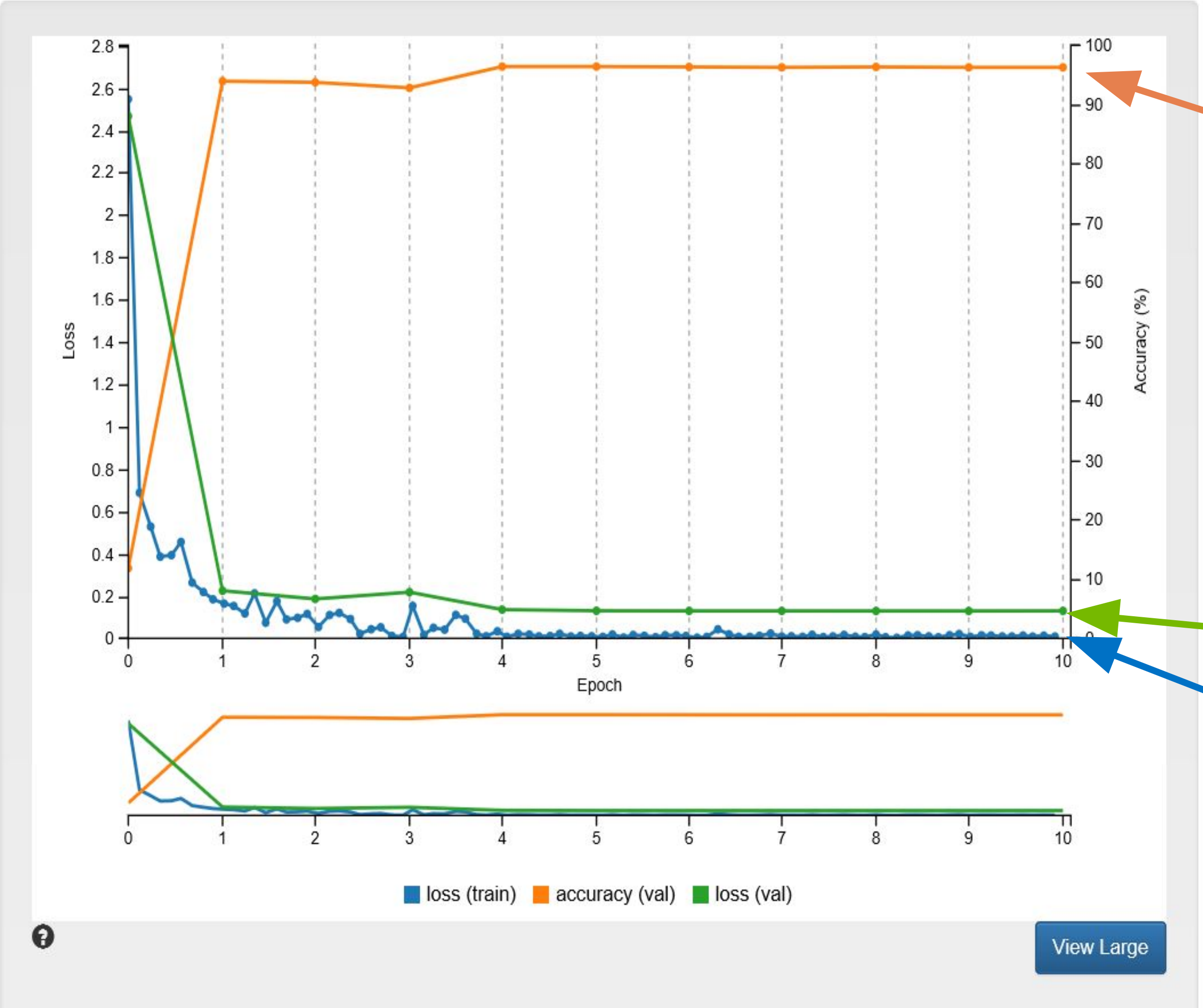
Monitor Progress



Visualization



NVIDIA'S DIGITS



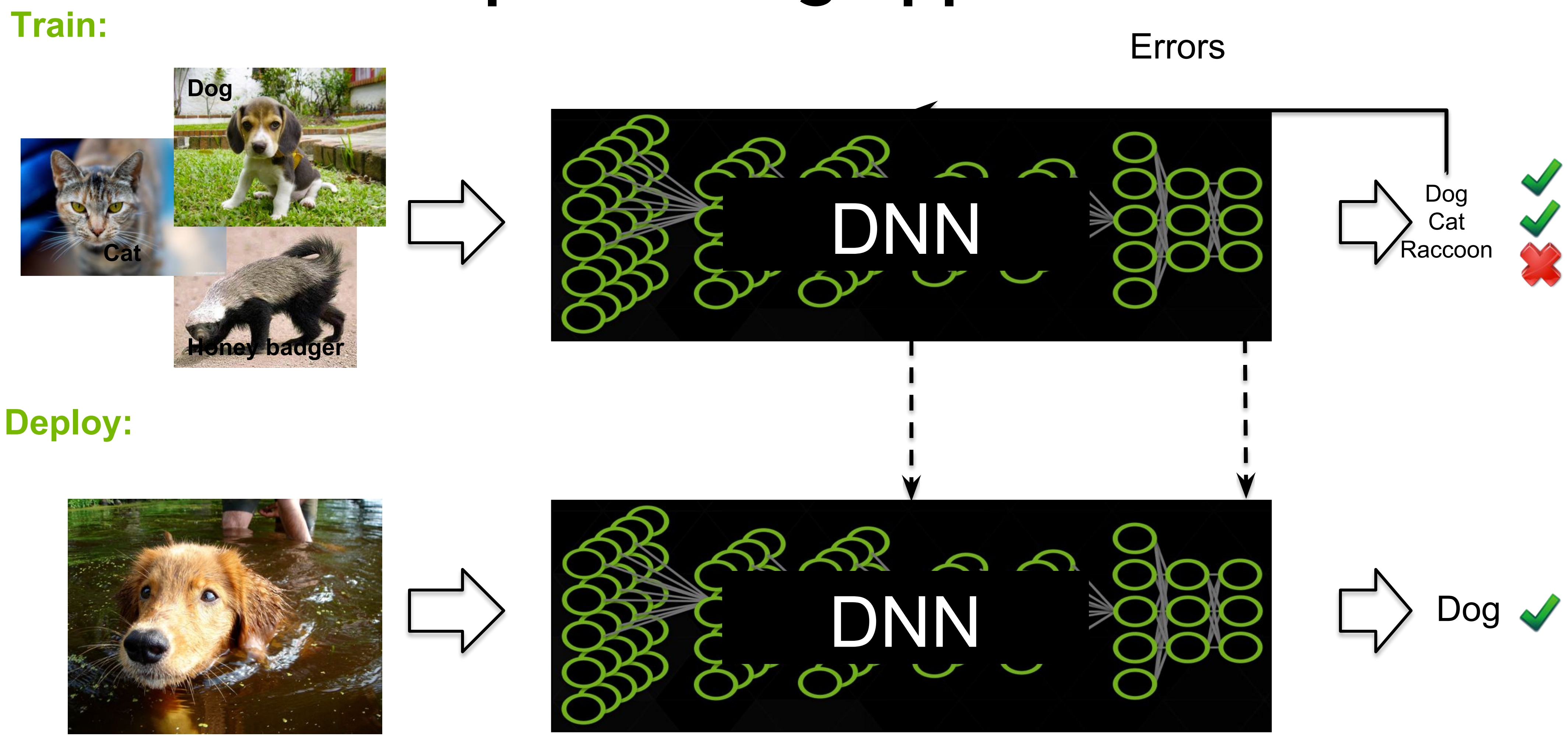
Accuracy obtained from validation dataset

Loss function (Validation)

Loss function (Training)

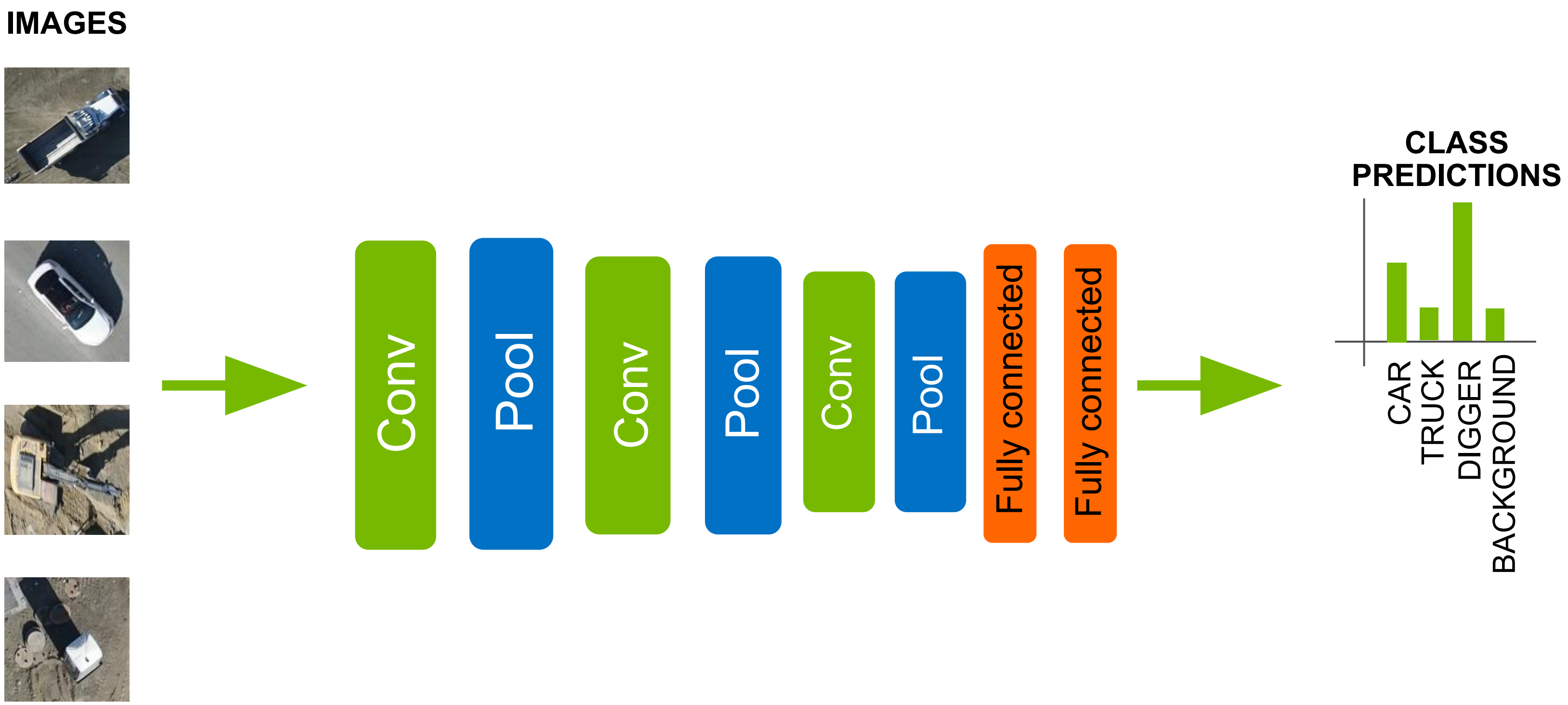
DEEP LEARNING APPROACH

Deep Learning Approach



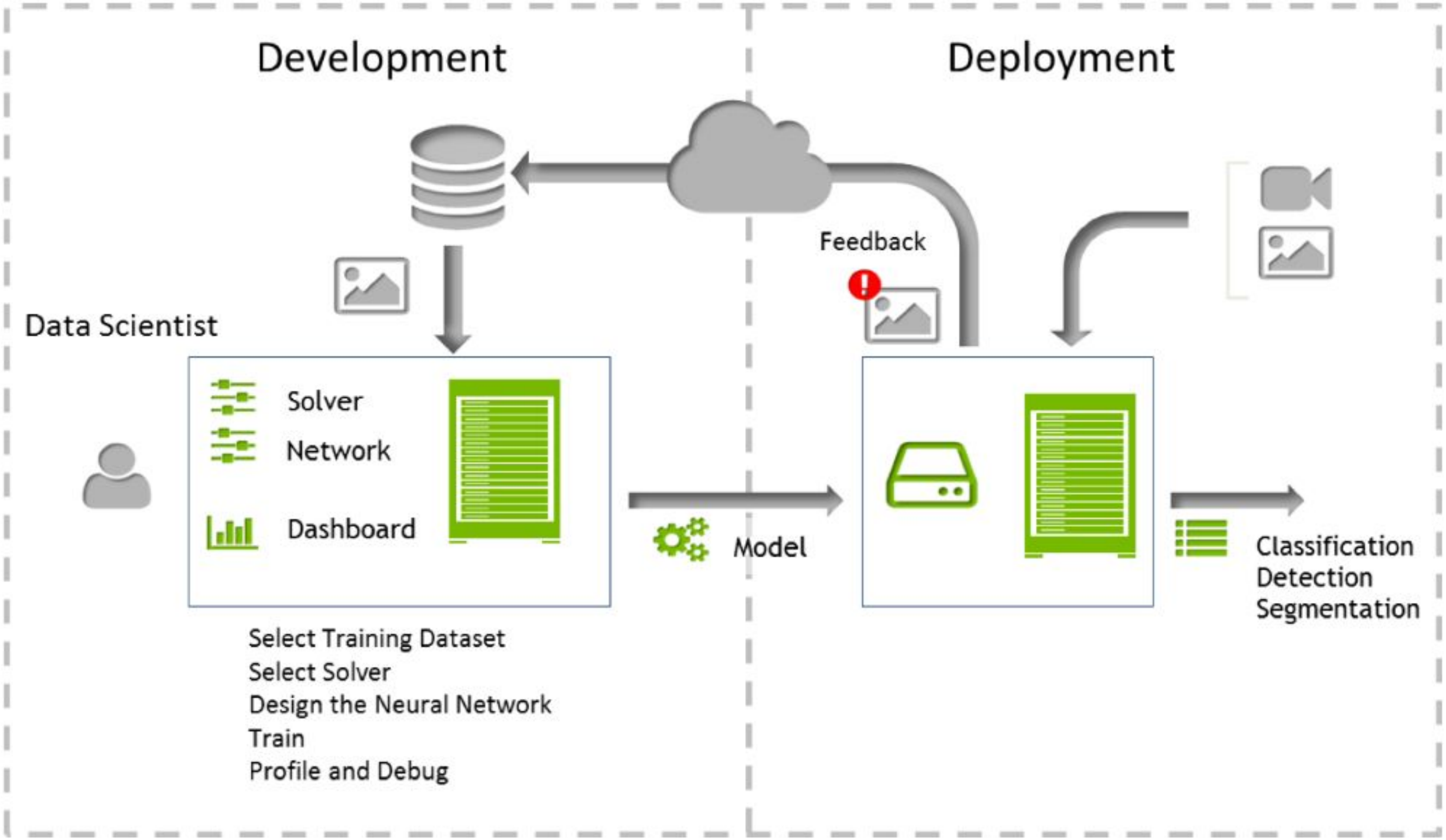
Deep Learning Approach

Convolutional Neural Network



Deep Learning Approach

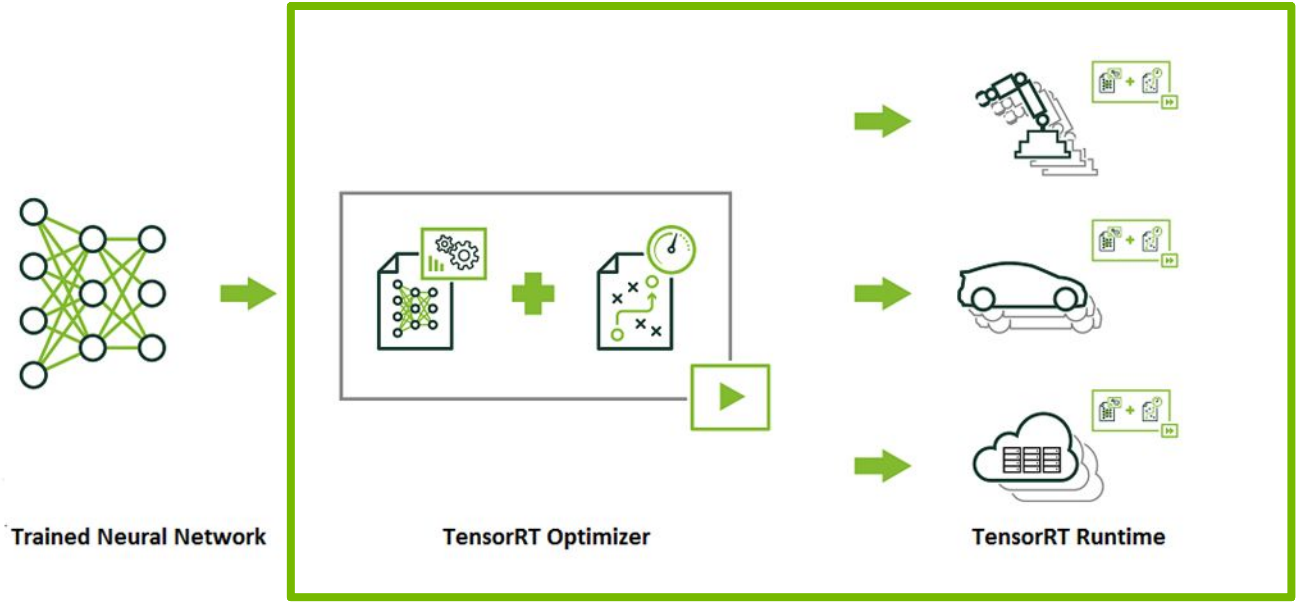
Neural network training and inference



NVIDIA’S TENSORRT

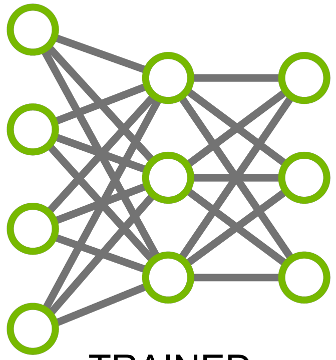
TensorRT

- Inference engine for production deployment of deep learning applications



- Allows developers to focus on developing AI powered applications
    - TensorRT ensures optimal inference performance

TensorRT Optimizer



TRAINED NEURAL NETWORK

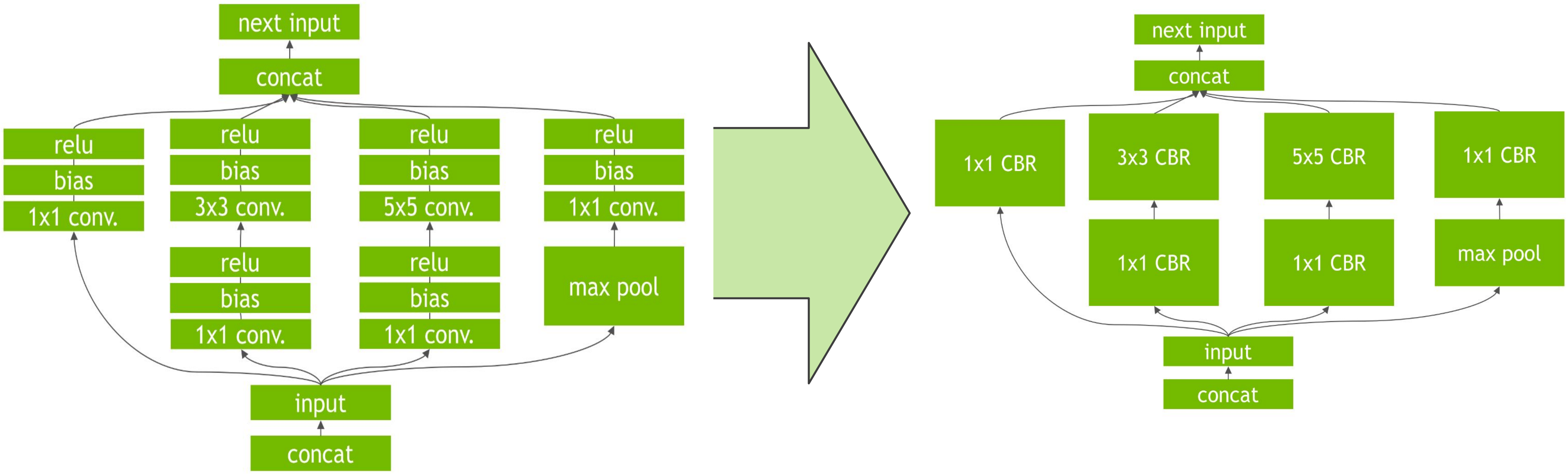
- **Fuse network layers**
- **Eliminate concatenation layers**
- **Kernel specialization**
- **Auto-tuning for target platform**
- **Select optimal tensor layout**
- **Batch size tuning**

OPTIMIZED INFERENCE RUNTIME

developer.nvidia.com/tensorrt

TensorRT Optimizer

Vertical Layer Fusion

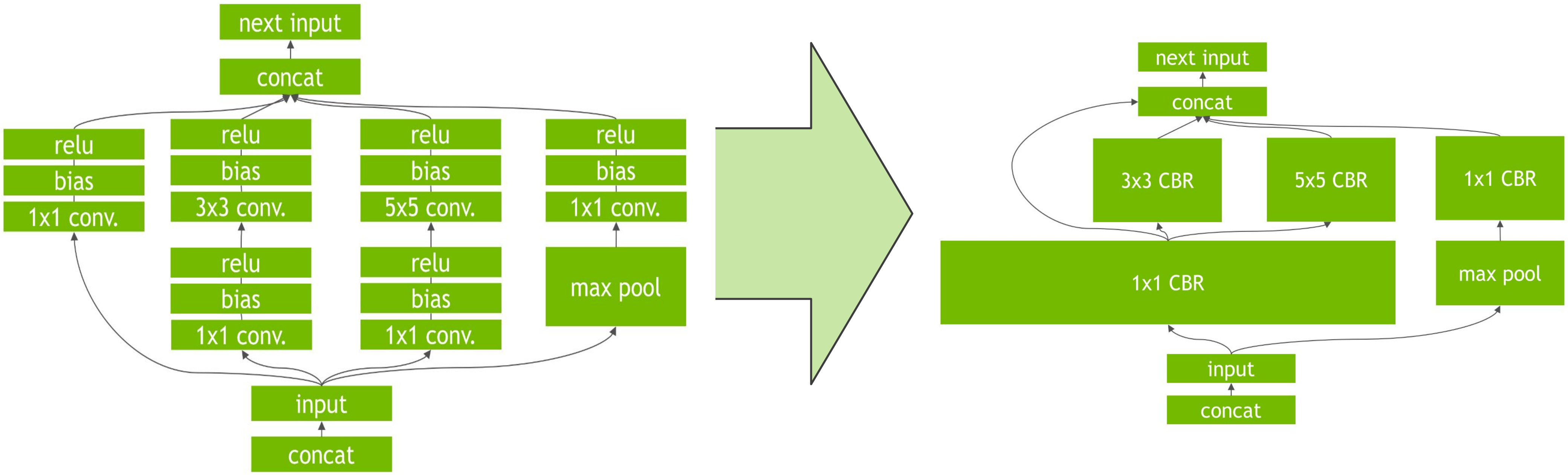


CBR = Convolution, Bias and ReLU

developer.nvidia.com/tensorrt

TensorRT Optimizer

Horizontal Layer Fusion (Layer Aggregation)



CBR = Convolution, Bias and ReLU

developer.nvidia.com/tensorrt

TensorRT Optimizer

Supported layers

- Convolution: 2D
- Activation: ReLU, tanh and sigmoid
- Pooling: max and average
- ElementWise: sum, product or max of two tensors
- LRN: cross-channel only
- Fully-connected: with or without bias
- SoftMax: cross-channel only
- Deconvolution

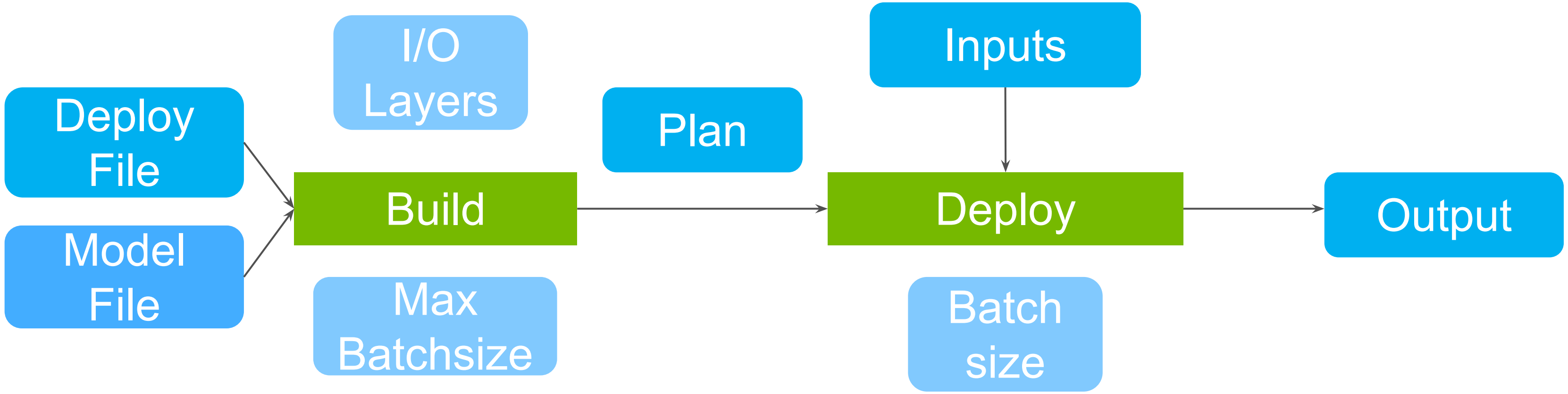
TensorRT Optimizer

- Scalability:
  - Output/Input Layers can connect with other deep learning framework directly
    - Caffe, Theano, Torch, TensorFlow
- Reduced Latency:
  - INT8 or FP16
    - INT8 delivers 3X more throughput compared to FP32
    - INT8 uses 61% less memory compared to FP32

TensorRT Runtime

Two Phases

- **Build:** optimizations on the network configuration and generates an optimized plan for computing the forward pass
- **Deploy:** Forward and output the inference result



TensorRT Runtime

- No need to install and run a deep learning framework on the deployment hardware
- Plan = runtime (serialized) object
  - Plan will be smaller than the combination of model and weights
  - Ready for immediate use
    - Alternatively, state can be serialized and saved to disk or to an object store for distribution
- Three files needed to deploy a classification neural network:
  - Network architecture file (deploy.prototxt)
  - Trained weights (net.caffemodel)
  - Label file to provide a name for each output class

LAB DETAILS

Lab Architectures / Datasets

- *GoogleNet*
  - CNN architecture trained for image classification using the ilsvrc12 Imagenet dataset
  - 1000 class labels to an entire image based on the dominant object present
- *pedestrian_detectNet*
  - CNN architecture able to assign a global classification to an image and detect multiple objects within the image and draw bounding boxes around them
  - Pre-trained model provided has been trained for the task of pedestrian detection using a large dataset of pedestrians in a variety of indoor and outdoor scenes

Lab Tasks

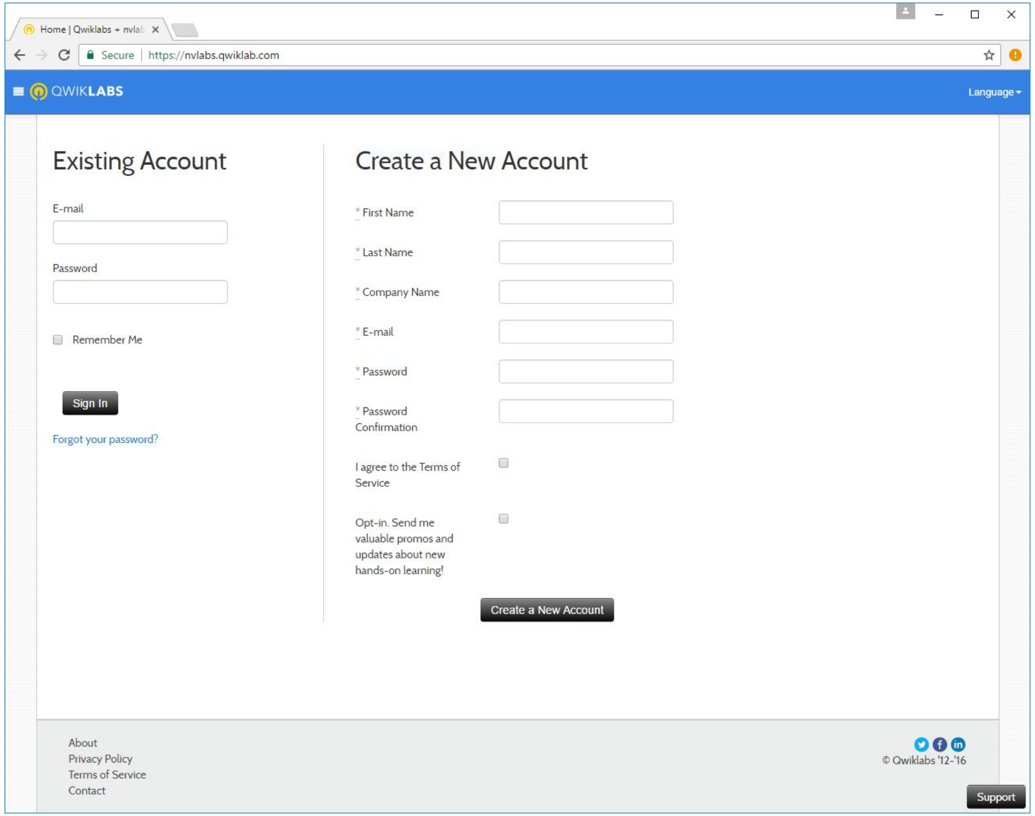
- GPU Inference Engine (GIE) = TensorRT
- Part 1: Inference using DIGITS
    - Will use existing model in DIGITS to perform inference on a single image
- Part 2: Inference using Pycaffe
    - Programming production-like deployable inference code
- Part 3: NVIDIA TensorRT
    - Will run TensorRT Optimizer to build a plan
    - Deploy the plan using TensorRT Runtime

LAUNCHING THE LAB ENVIRONMENT

NAVIGATING TO QWIKLABS

1. Navigate to: https://nvlabs.qwiklab.com

2. Login or create a new account

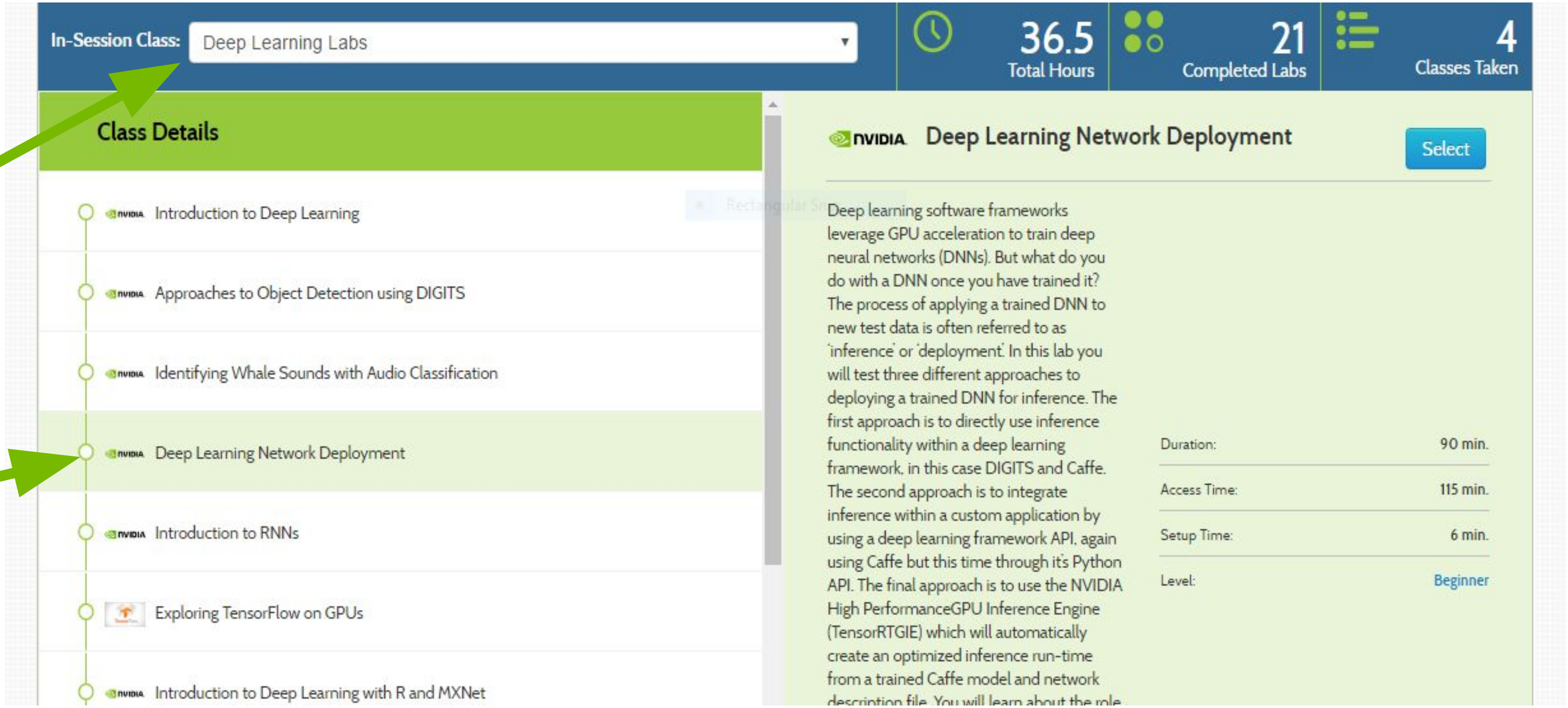


Please use the email address used to register for session

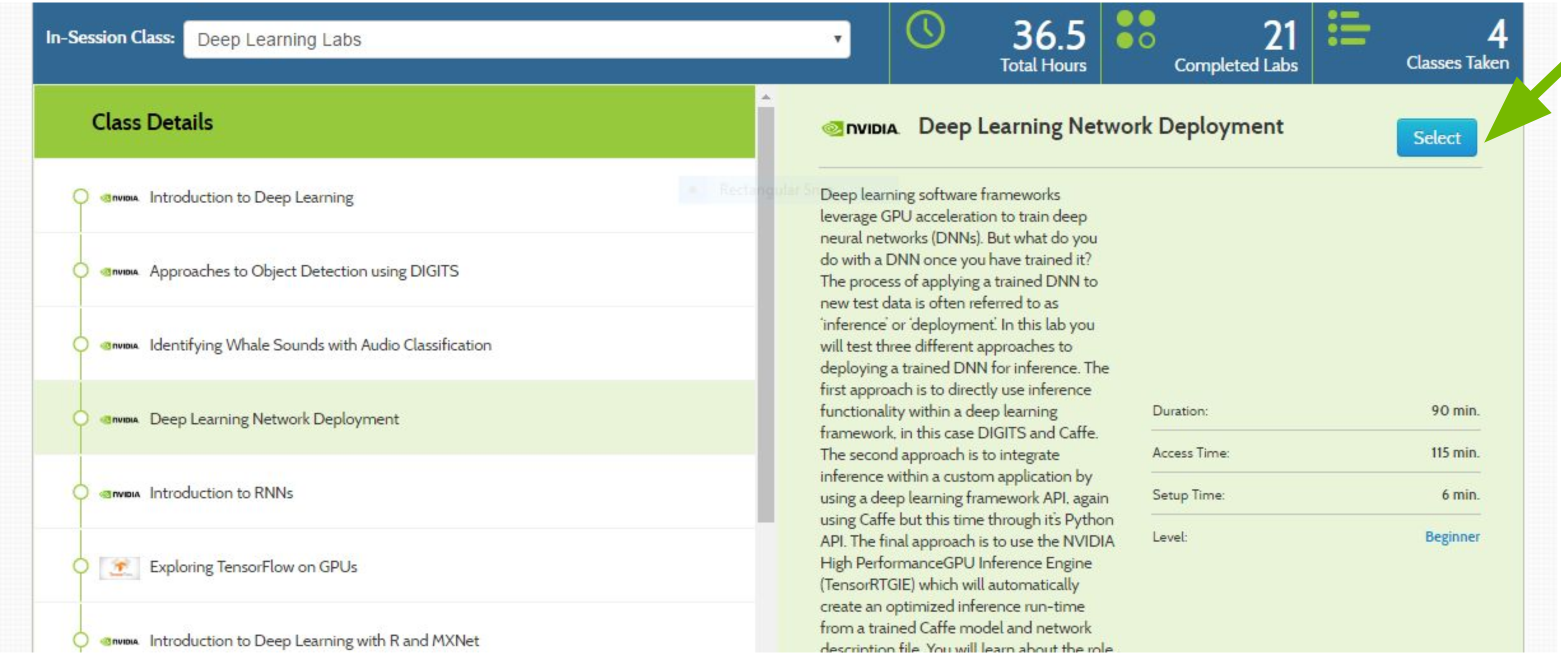
ACCESSING LAB ENVIRONMENT

3. Select the event specific In-Session Class in the upper left

4. Click the "Deep Learning Network Deployment" Class from the list



LAUNCHING THE LAB ENVIRONMENT



5. Click on the Select button to launch the lab environment

- After a short wait, lab Connection information will be shown
- Please ask Lab Assistants for help!

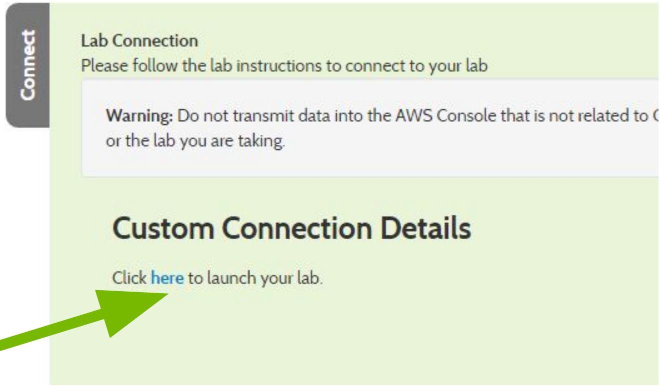
LAUNCHING THE LAB ENVIRONMENT

6. Click on the Start Lab button

LAUNCHING THE LAB ENVIRONMENT

You should see that the lab environment is “launching” towards the upper-right corner

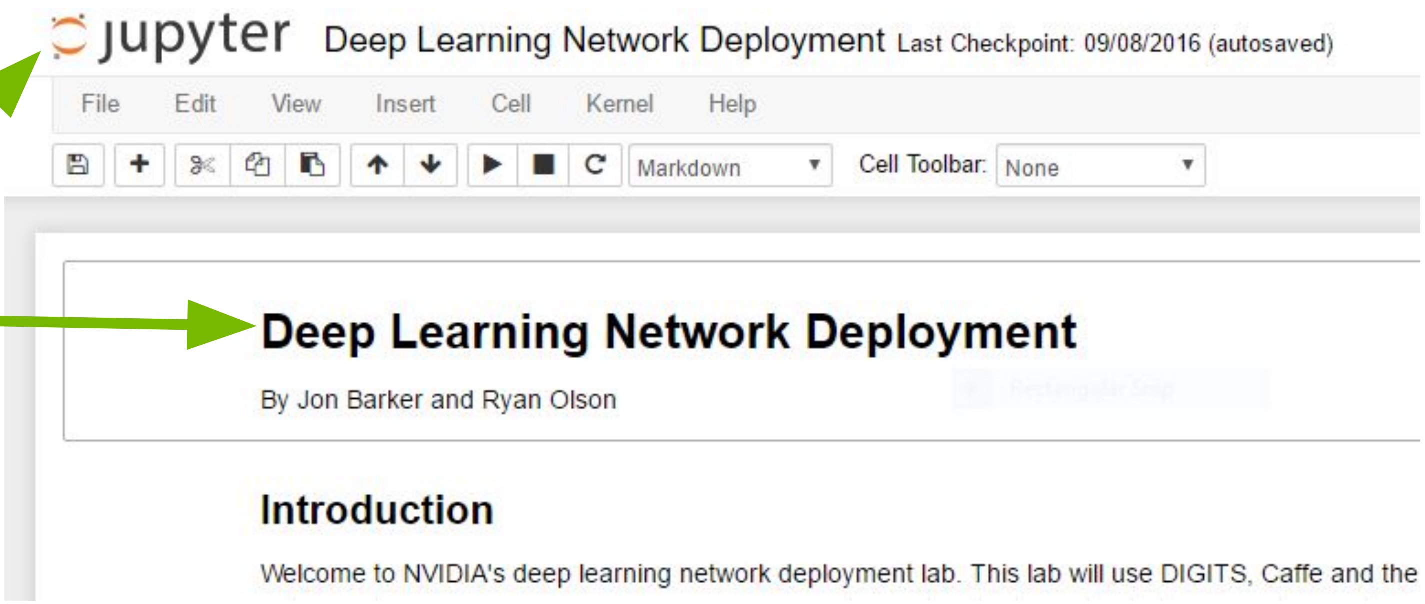
CONNECTING TO THE LAB ENVIRONMENT



7. Click on “here” to access your lab environment / Jupyter notebook

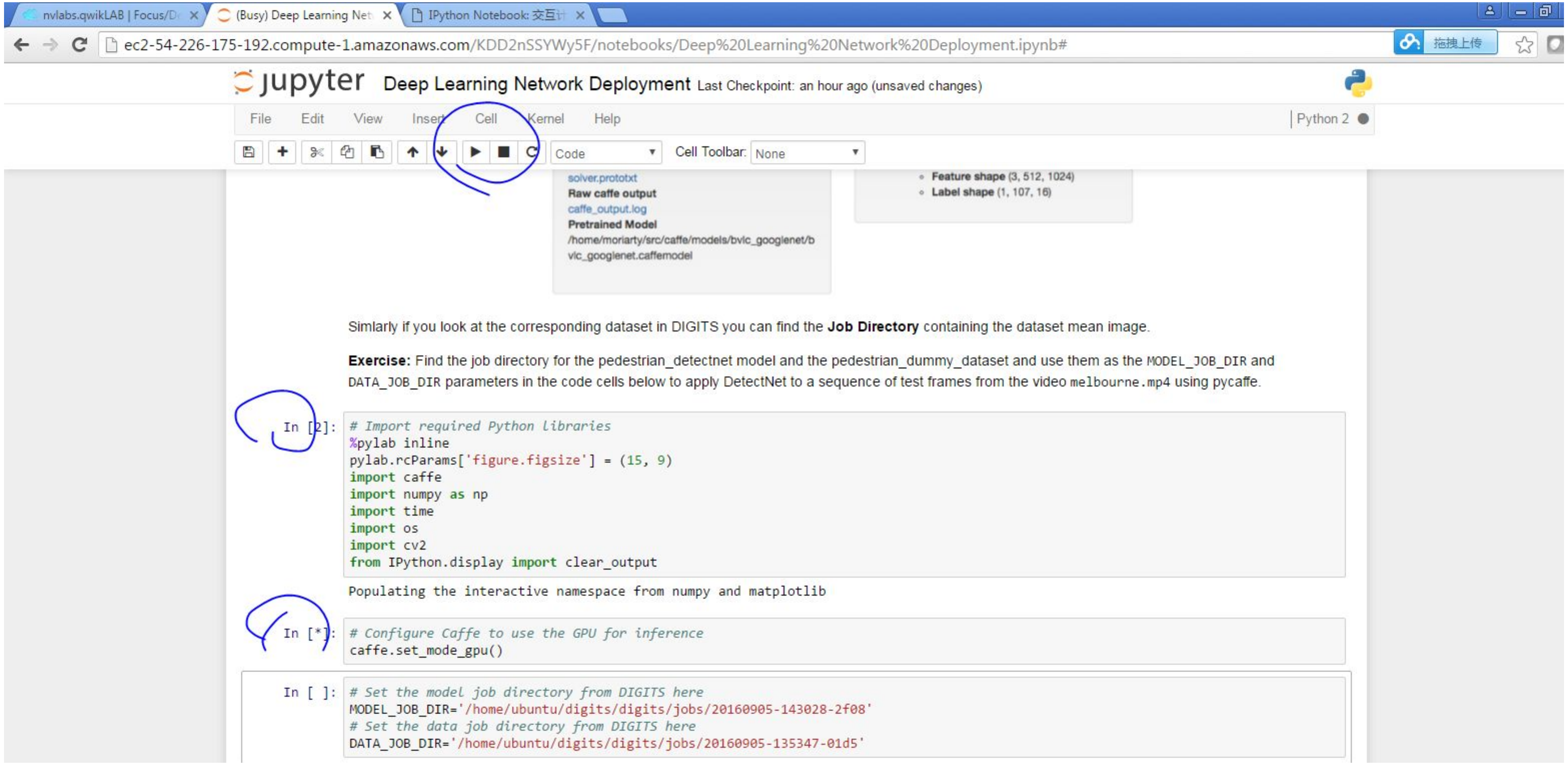
CONNECTING TO THE LAB ENVIRONMENT

You should see your "Deep Learning Network Deployment" Jupyter notebook



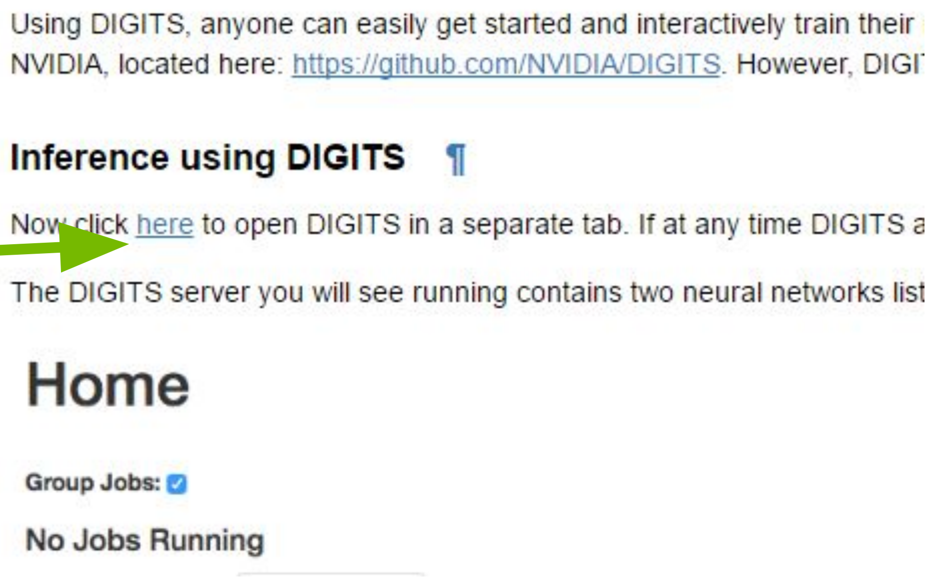
Jupyter Notebook Introduction

Interface: Run



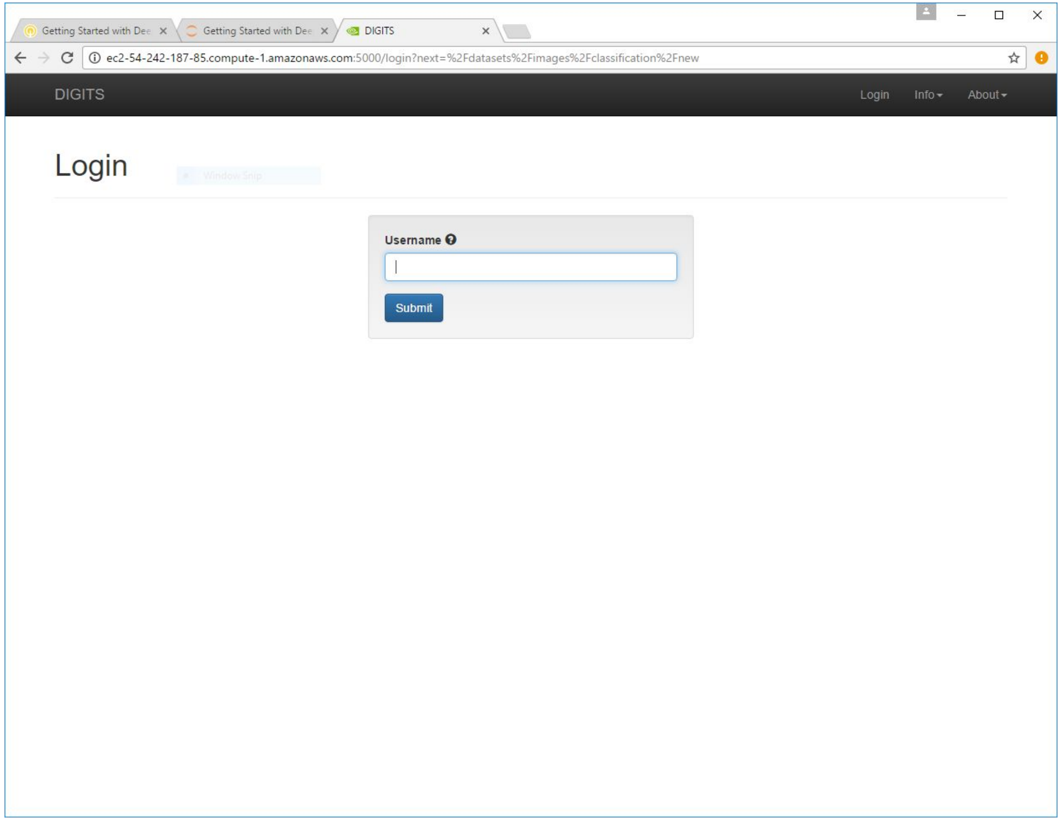
STARTING DIGITS

Instruction in Jupyter notebook will link you to DIGITS



ACCESSING DIGITS

- Will be prompted to enter a username to access DIGITS
    - Can enter any username
    - Use lower case letters



REVIEW / NEXT STEPS

WHAT'S NEXT

- Use / practice what you learned
- Discuss with peers practical applications of DNN
- Reach out to NVIDIA and the Deep Learning Institute
- Look for local meetups
- Follow people like Andrej Karpathy and Andrew Ng

WHAT'S NEXT

TAKE SURVEY

…for the chance to win an NVIDIA SHIELD TV.

Check your email for a link.

ACCESS ONLINE LABS

Check your email for details to access more DLI training online.

ATTEND WORKSHOP

Visit www.nvidia.com/dli for workshops in your area.

JOIN DEVELOPER PROGRAM

Visit https://developer.nvidia.com/join for more.

GTC AROUND THE WORLD

GTC CHINA
BEIJING
SEPTEMBER 25 -27, 2017

GTC EUROPE
MUNICH
OCTOBER 10 - 12, 2017

GTC ISRAEL
TEL AVIV
OCTOBER 18, 2017

GTC DC
WASHINGTON, DC
NOVEMBER 1 - 2, 2017

GTC JAPAN
TOKYO
DECEMBER 12 - 13, 2017

GTC 2018
SILICON VALLEY
MARCH 26 - 29, 2018

WWW.GPUTECHCONF.COM

Instructor: Twin Karmakharm

DEEP
LEARNING
INSTITUTE

www.nvidia.com/dli

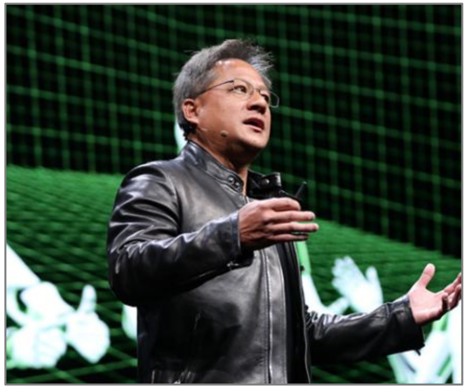
Join the Conversation
#GTC18

CONNECT

Connect with technology experts from NVIDIA and other leading organisations.

LEARN

Gain insight and valuable hands-on training through hundreds of sessions and research posters.



DISCOVER

Discover the latest breakthroughs in fields such as autonomous vehicles, HPC, smart cities, VR, robotics, and more.

INNOVATE

Hear about disruptive innovations as startups and researchers present their work.

APPENDIX

Lab Debug

Can’t display Ipython Notebook?

IPython Notebook

- Chrome/Firefox/Safari recommended. IE will work but not as well
- Websockets are required - you can test at websocketstest.com
  - Look for this result:

WebSockets (Port 80)	
Connected	Yes ✓
Data Receive	Yes ✓
Data Send	Yes ✓
Echo Test	Yes ✓
Server time	2016/8/24 02:42:20

- Execute cells with ctrl+enter or pressing play button

Lab Debug

Don’t know if cell is running??

You should see In[*] and not In[ ] or In[<some number>].

Solid grey circle in the top-right of the browser window

If you only see #1 and not #2, then you need to try the following in order:

- Press the stop button on the toolbar. Try again.
- Click Kernel -> Restart. Try again.
- Save the Notebook and refresh the page. Try again.
- End the lab from the qwikLABS page and start a new instance. All work will be lost. (Please let me know before you do this)

Lab Debug

Reverse to some checkpoint