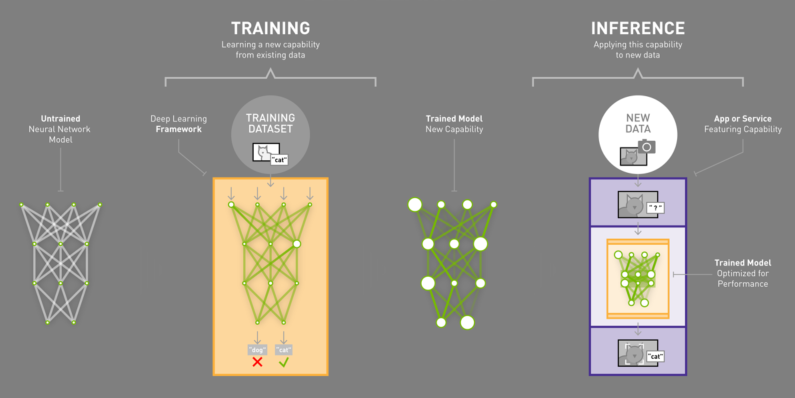
**Fundamentals of Deep Learning for Computer Vision- Course Report**

1. Training Deep Neural Networks
   1. Deep Neural Networks: GPU Task 1

Deep Neural Networks are flexible algorithms inspired by the human brain that allow practitioners to use training strategies inspired by human learning. The input of an image generated an output of the network's confidence that the image belonged to one of two classes. Something clearly changed between the first epoch and the 100th. It indicates algorithms would learn the experience from the huge dataset. A neural network changes when exposed to data to create an accurate map between inputs and outputs.

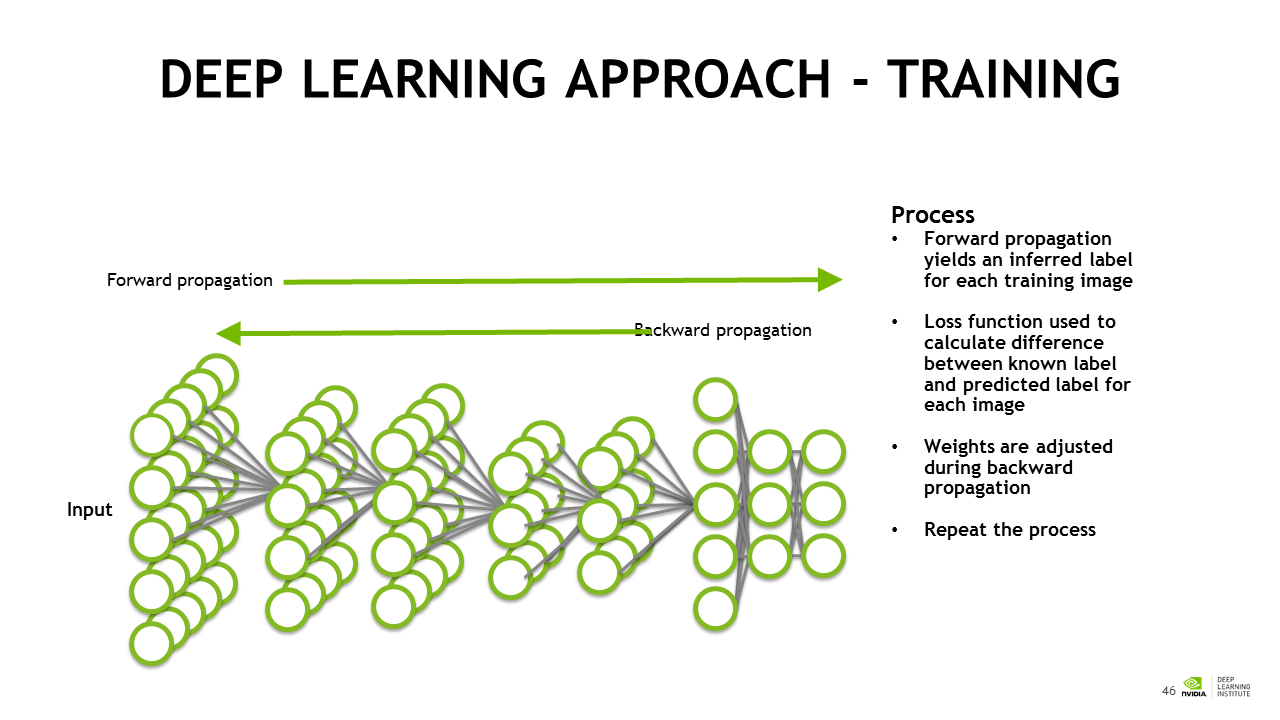
* 1. Deep Neural Networks: GPU Task 2

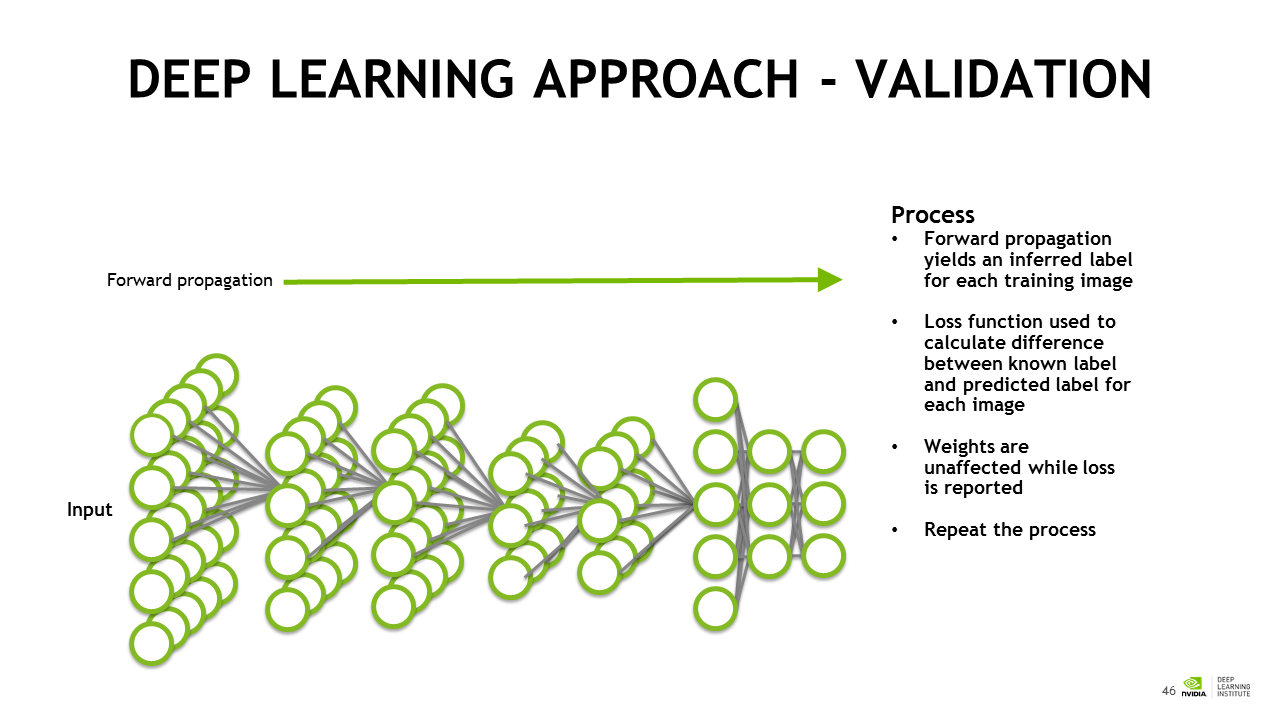
In this task, I mainly learnt the workflow of training the model with data. The following notes are the workflow from course.

Data processing:

1) Standardize them to the same size to match what the network you are training expects. We'll be training AlexNet again which was designed to take an input of 256X256 color images. Expect more on this in the next task.

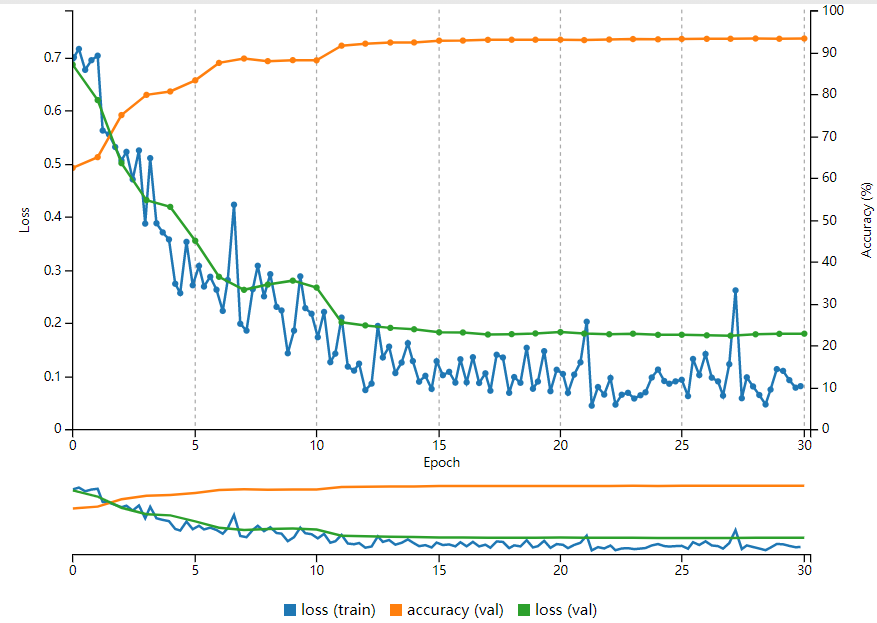
2) Split them into two datasets, where 75% of each class is used for training and 25% is set aside for validation.

We could use training accuracy, validation loss, training loss to measures the performance of model. 

The validation dataset will be used to assess performance on new data using a technique that human learning can not. Validation data is fed through the network to generate an output, but the network does not learn anything from the data. 

And I also did the experiment on cats and dogs dataset. Here is the result of the experiment.

I used alexnet as my model, here is the performance graph of alexnet.

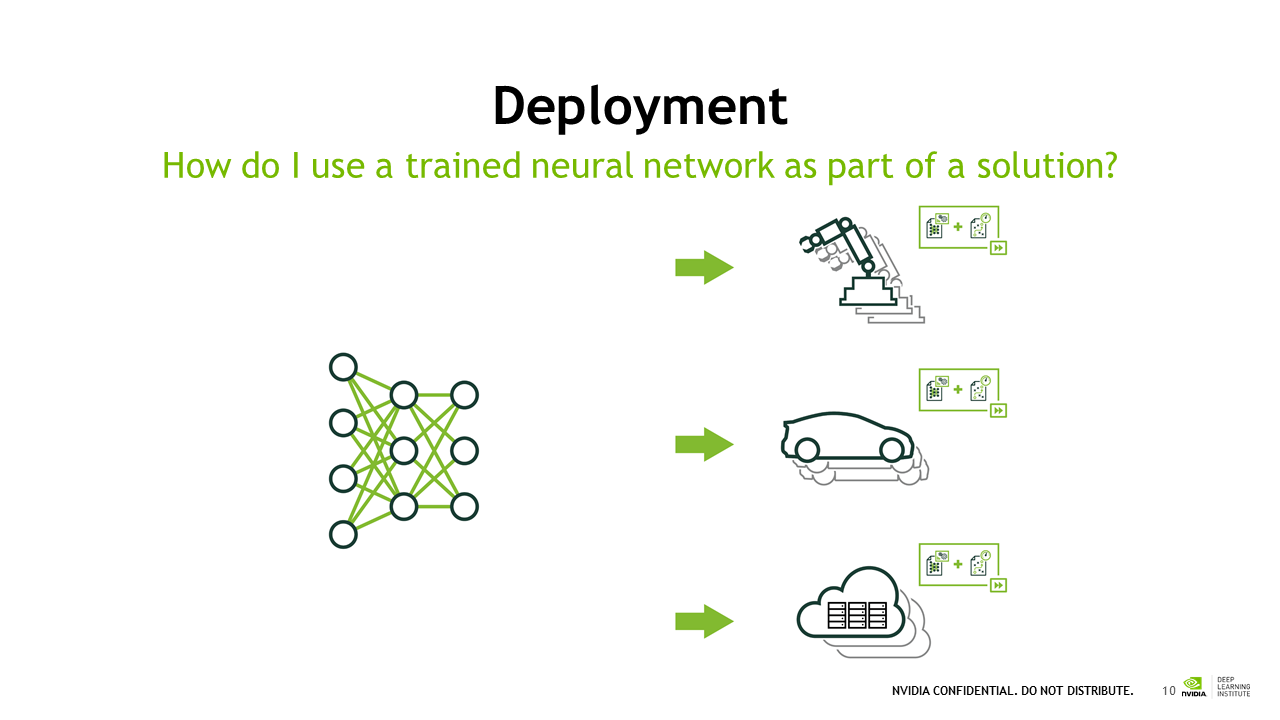


Its validation accuracy is 93%, training loss is 8% and validation loss is 18%.

Its performance is very good.

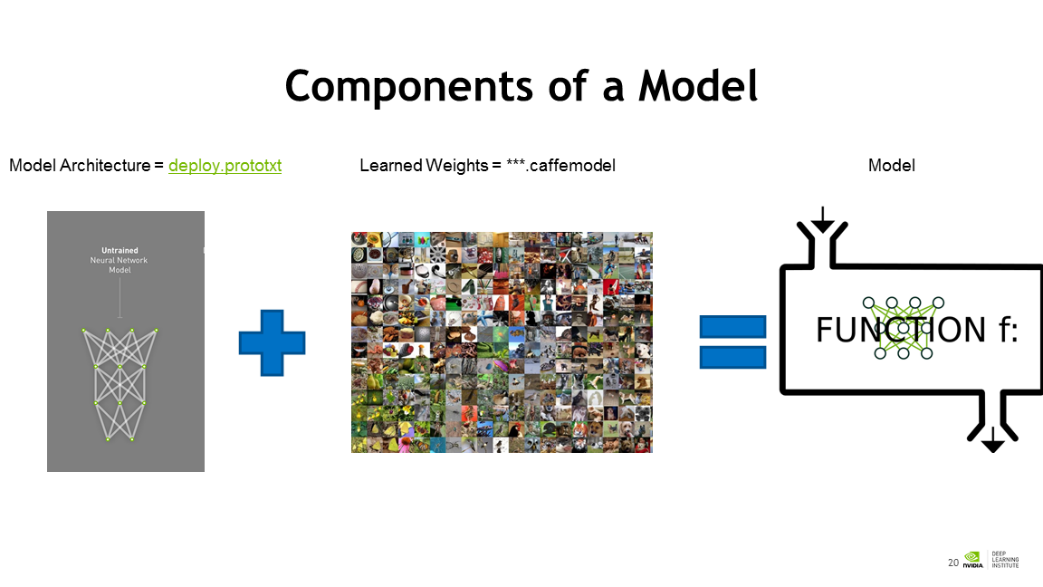
1. Deploying Trained Neural Networks
   1. Deploying our Model: GPU Task 3

In this task, I mainly learnt knowledge about how to deploy trained networks into applications to solve problems in the real world. The following are my notes.

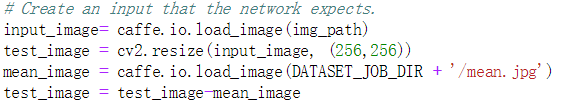
Deployment is the work of taking a trained model and putting it to work as a part of an application. You can deploy to edge devices such as robots or autonomous vehicles. You can also deploy to a server in order to play a role in any piece of software. 

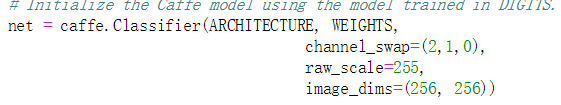
To successfully \*deploy\* a trained model, we have two initial jobs.

1) Our first job is to provide our model an input that it expects.

2) Our second job is to provide our end user an output that is useful. 

Also, I deployed the model according to the requirement.

First, we should preprocessing the data, change the format for the expected input. 

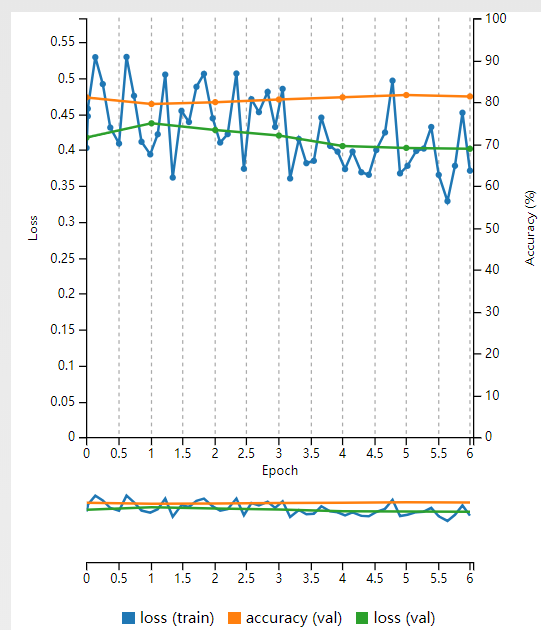
And then loading the model 

Finally, make a prediction 

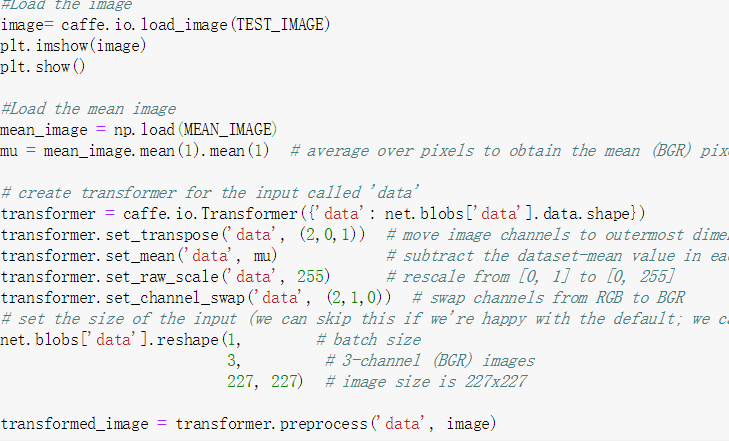
1. Measuring and Improving Performance
   1. Performance

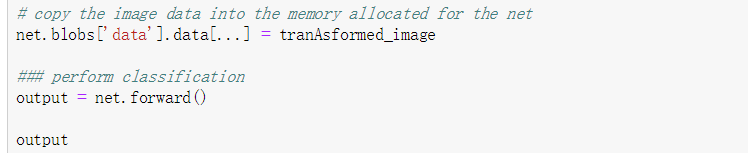
GPU Task 4

In this task, the most important thing that I learnt is the reason of why learning rate decrease. The learning rate decreases throughout the training session because the network is getting closer to its ideal solution.

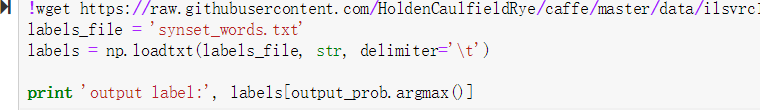
According to requirement, I select cat vs dog model as pretrained model and train it again. 

This is the graph shows that the performance of the new model.

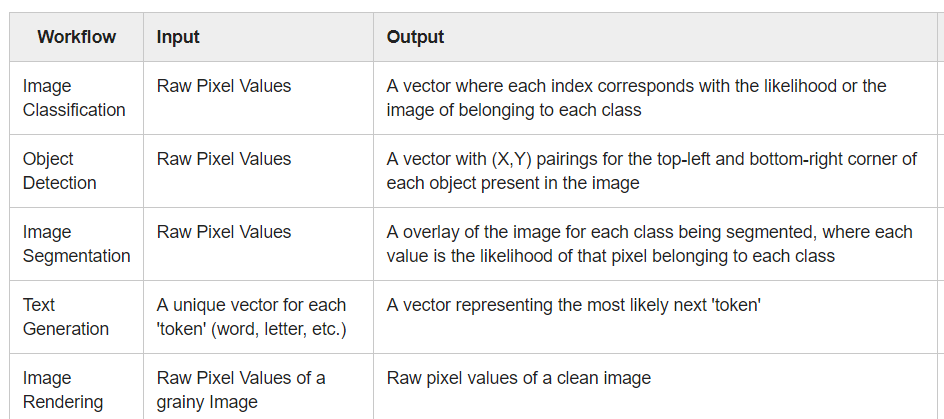
After that, I followed the instruction and downloaded the award winning models. The winning model is trained on ImageNet. I used the wget command to download the pretrained model and dataset. Then, I loaded the dataset. 

And then I made prediction with the following code. 

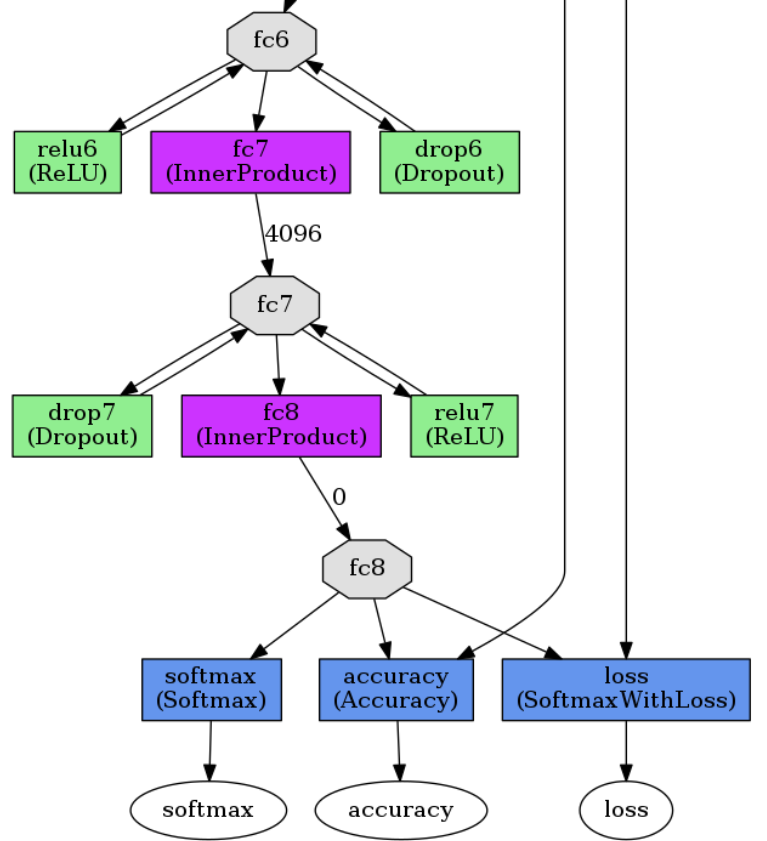
At last, I made output useful for users.



GPU Task 5: Object Detection

Here are my notes from the course. 

In this task, our goal is to detect and localize objects within images. One of methods is to slide window with non-overlapping grid squares. The other approach is to rebuild from an existing neural network. First, we should know the architecture of AlexNet. According to jupyter notebook, we visualize the architecture of AlexNet.



The picture above is part of architecture. And this method is to covert AlexNet to fully convolutional new work because convolutional network could be fed by various size of images without first splitting them into grid squares.