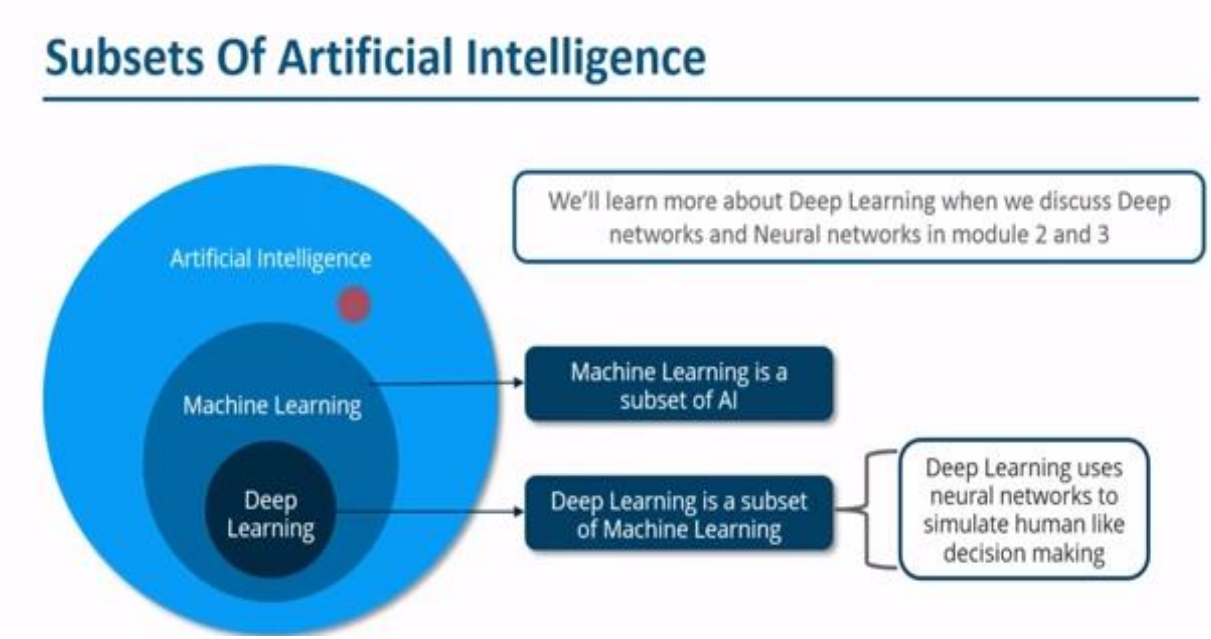
Deep Learning Indaba𝕏Namibia

By Dr. Nalina Suresh

OUTLINE

- Introduction to DL
- Distinction between AI, ML and DL
- Why AI
- Introduction to m/c Learning and its limitations
- What is driving towards DL
- Deep Neural Network (DNN)
- Example/Demo using Sample Dataset
- Work flow for DNN

Relationship between AI, ML and DL



- AI is the A⊇B
- ML is A⊆B of AI—means to achieve AI
- DL A⊆B of ML--- extension of ML to achieve AI

Why AI : Re-engineering human brain or mimicking human brain

What Is Artificial Intelligence?

Artificial Intelligence is the capability of a machine to imitate intelligent human behavior.

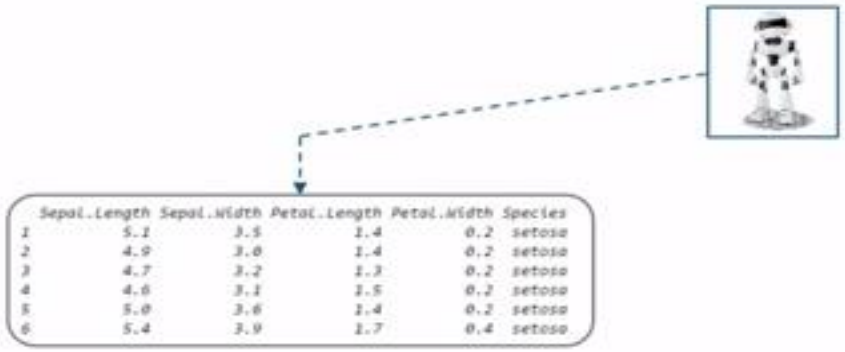
AI is accomplished by studying how human brain thinks, and how humans learn, decide, and work while trying to solve a problem

Outcomes of this study is used as a basis of developing intelligent software and systems.

Machine Learning

- Machine Learning is a type of artificial intelligence (AI) that provides computers with the ability to learn without being explicitly programmed.

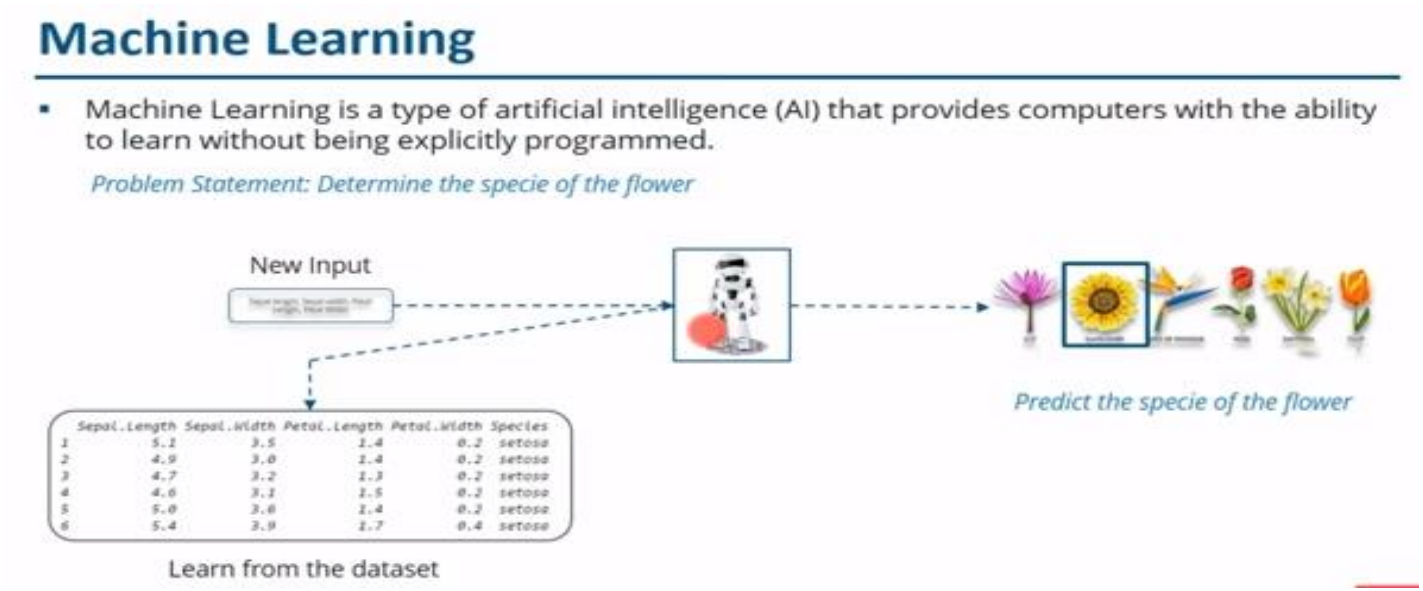
Problem Statement: Determine the specie of the flower



	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3.0	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa
5	5.0	3.6	1.4	0.2	setosa
6	5.4	3.9	1.7	0.4	setosa

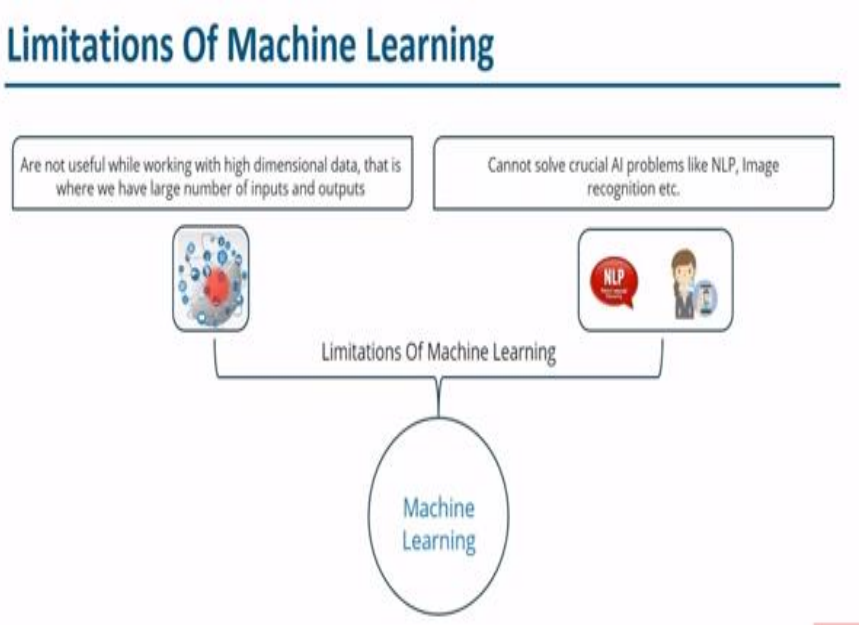
Learn from the dataset

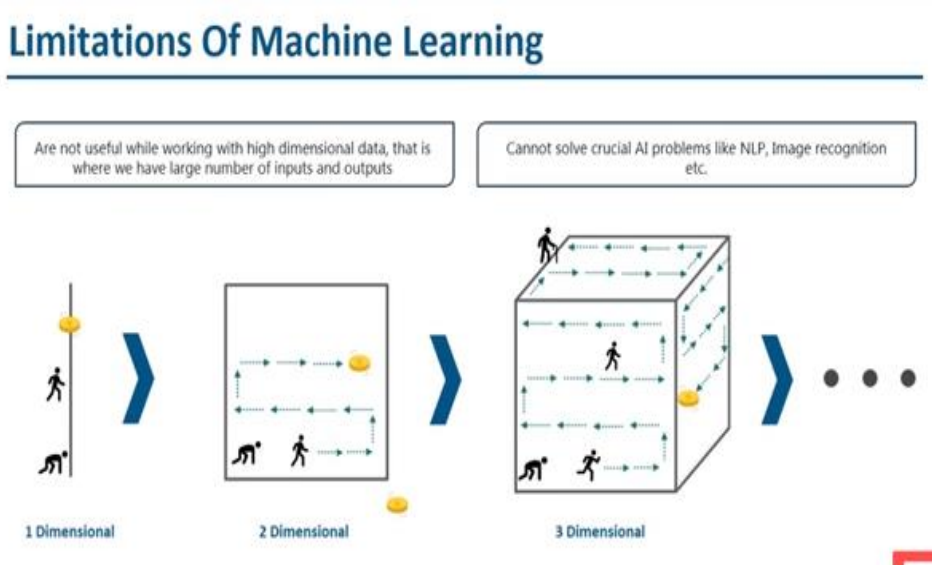
- We don’t have to define all the steps or conditions like any other programming language applications
- However, We train the m/c on a Input data set large enough to create a model which helps m/c to take decision based on the learning

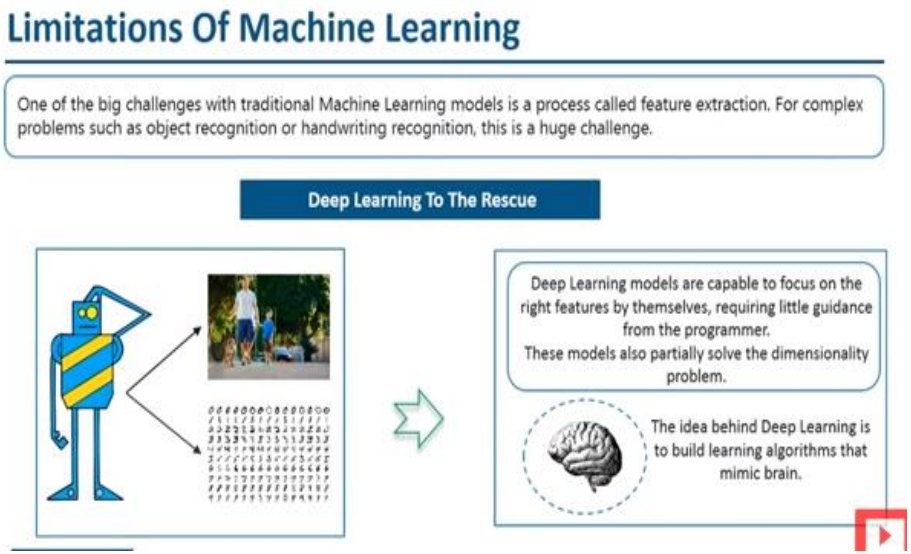


- We Train the m/c with flower data sets which contains Various characters of different flower along with there respective species
- Using this input data set to the m/c to create a model which can be used to classify a flower
- Next, we will pass a set of characteristics as input to the model, and it will output the name of the flower
- This process of training a m/c to create model and using it for decision making is called ML

Dimensionality problem

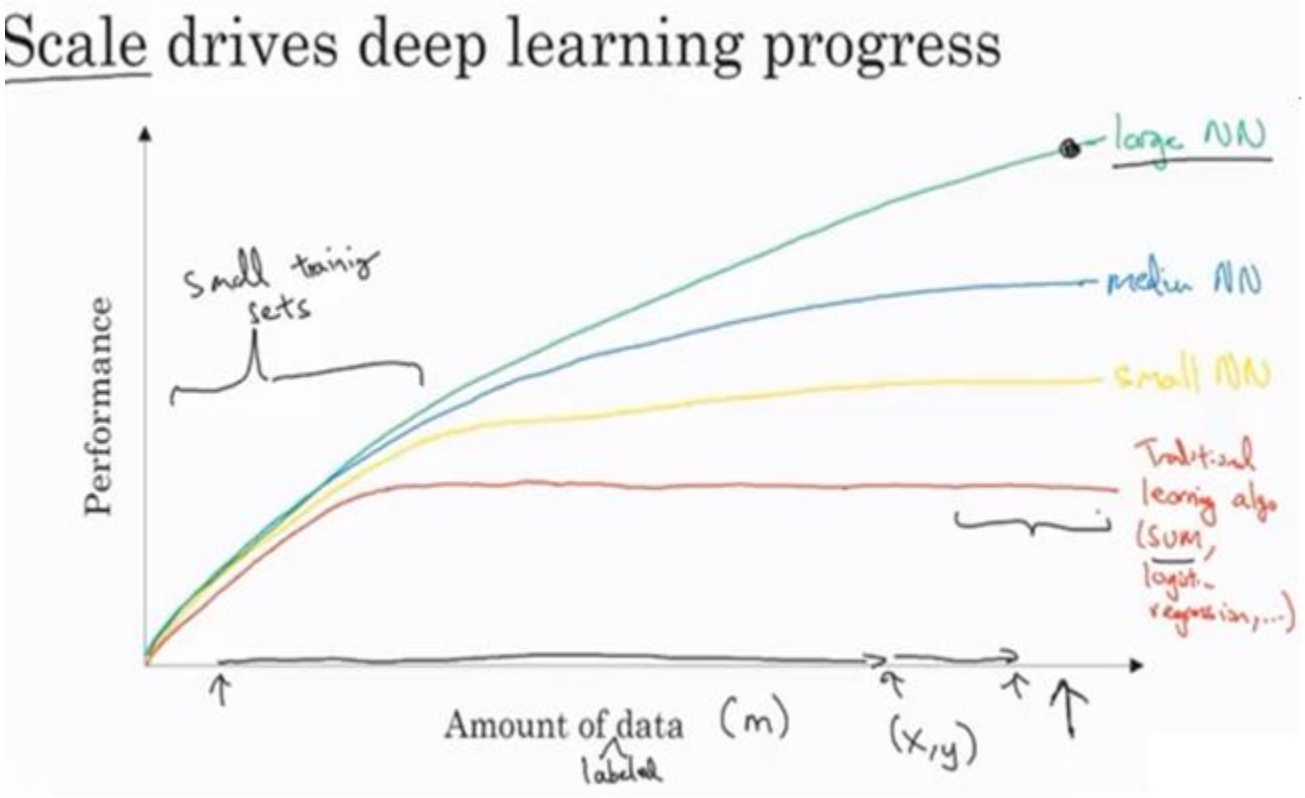




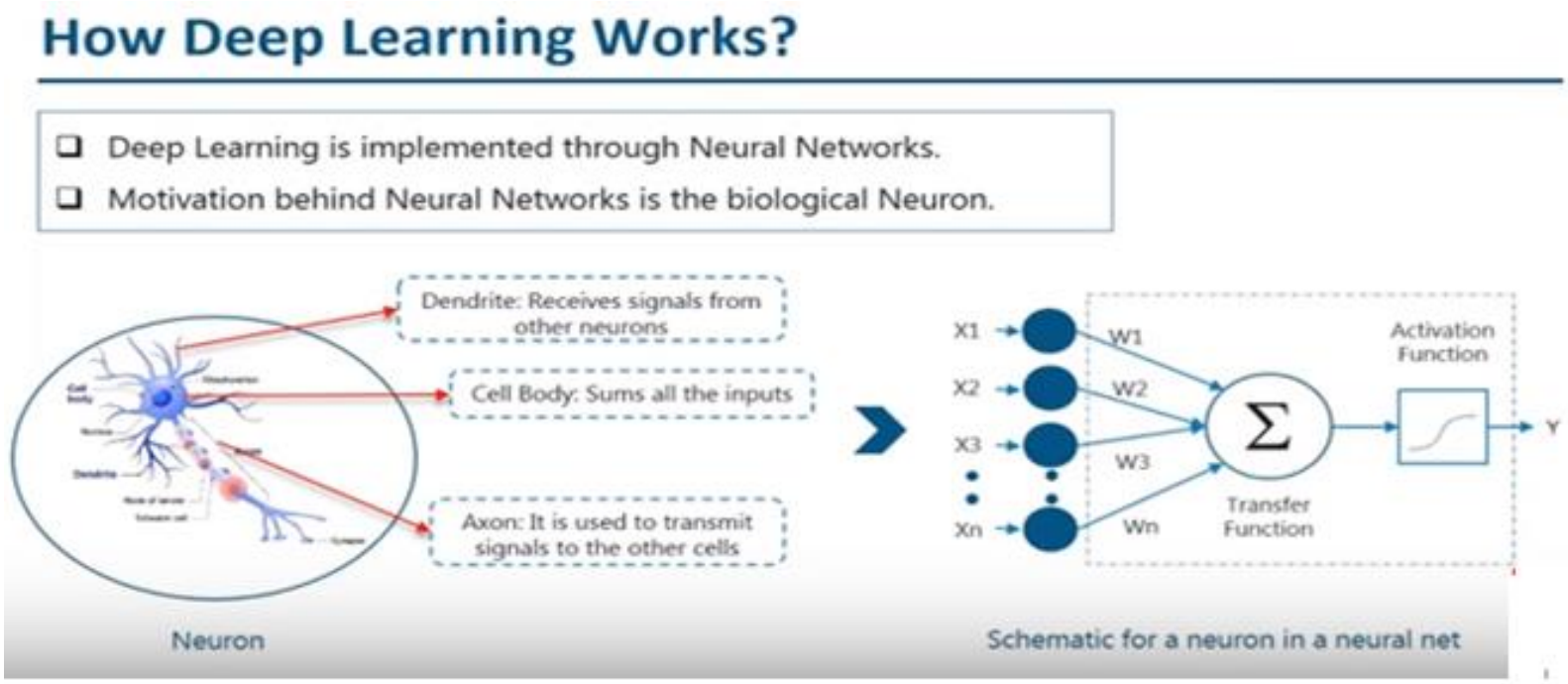


- Number of dimension is high complexity to do the task or consume that input increases as dimension increases
- In reality high dimension data can be found in use cases like image processing, NLP, image translation.
- ML not capable, hence deep learning came to rescue.

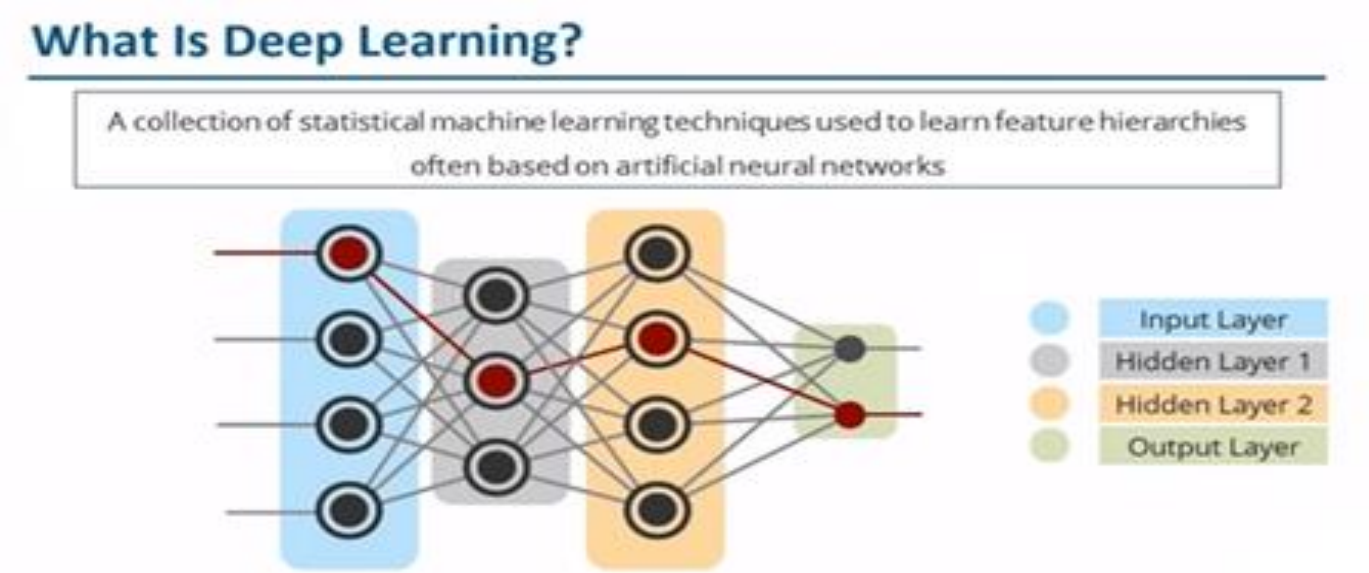
Why Deep Learning(DL)



Analogy Biological Neuron ≡ Artificial Neural Network



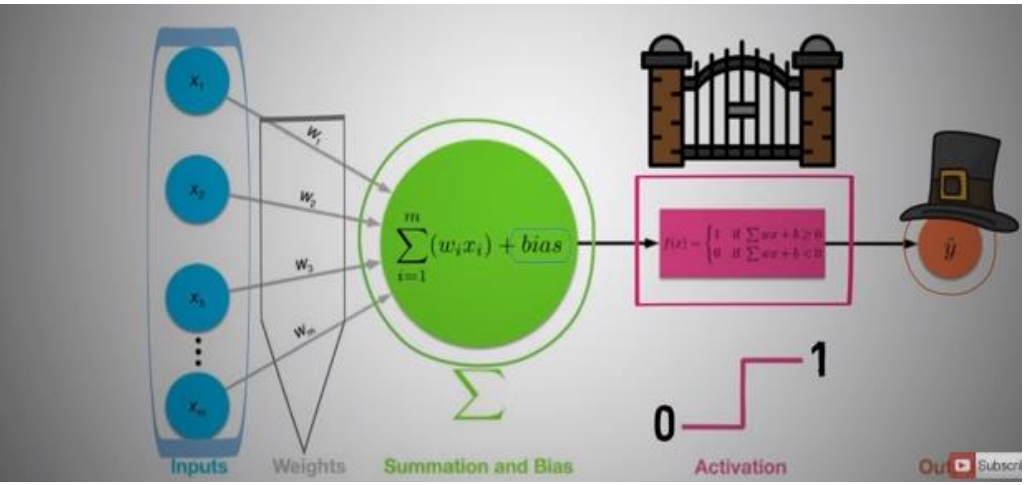
- Dentries—which are used to take the = inputs Perceptron receives Multiple inputs
- Inputs are summed in the cell body and finally passed into the next neuron through Axon = perceptron receives multiple inputs, then applies various transfer functions and transfers to next perceptron.
- Brain →network of neurons
  Artificial brain-network of perceptron's to create Deep NN



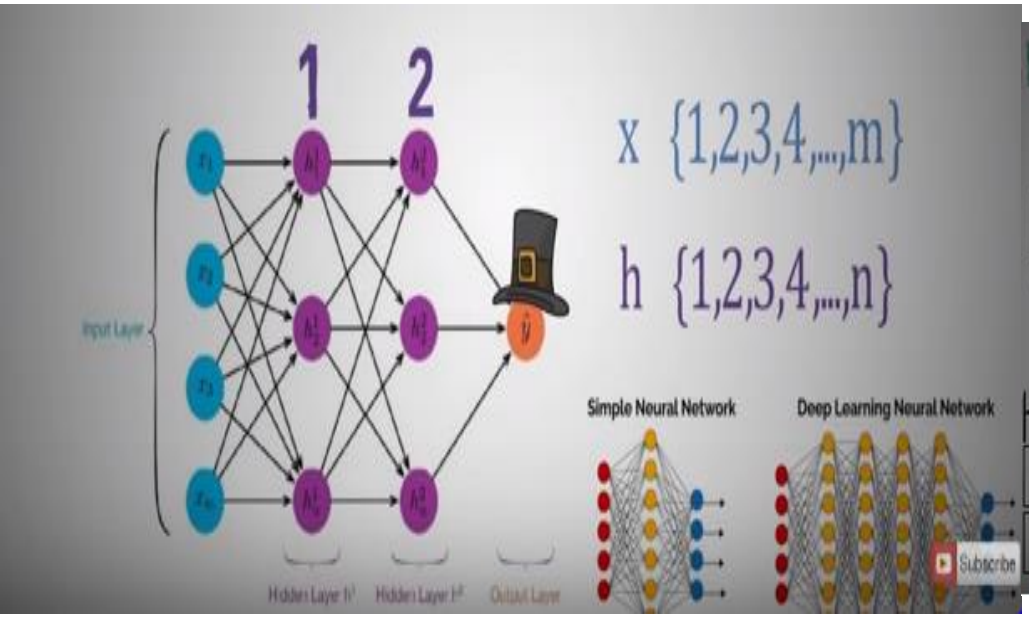
there are three types of layers

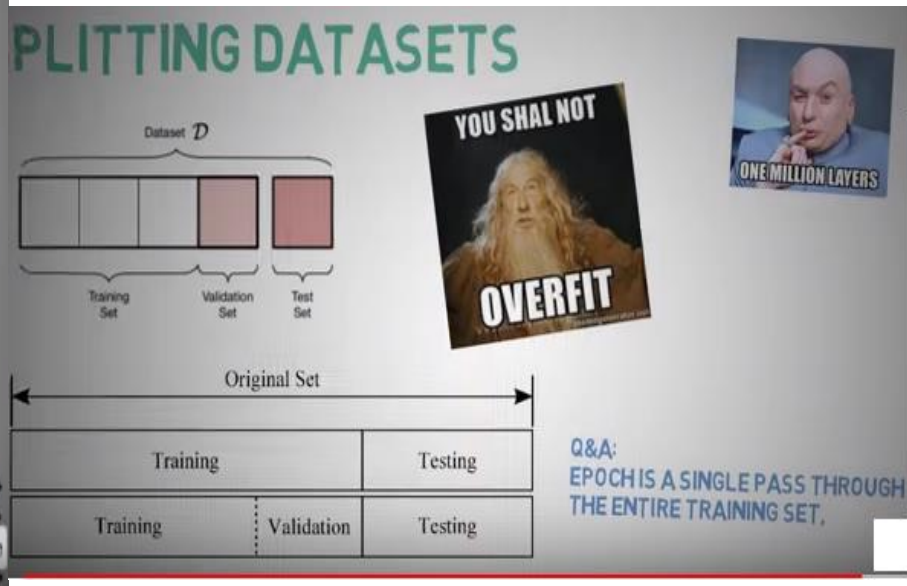
- Input layer—receives all inputs (4 perceptron)
- Output layer—desired o/p
- Hidden layer (2 Hidden layer)
- There can be n-number of hidden layers due to high end resources available
- Number of perceptron's each layer and number of hidden layer ---depending on use cases
- There is mechanics to decide the number of hidden layer

Artificial Neuron—Heart of neural Network



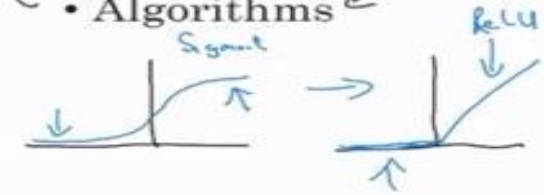
Activation node takes the input from other predecessor node and applies learning parameter to generate weight sum—Then passes to activation function that computes the composite prediction of probabilities. Known as Perceptron (Multiple input and one output)

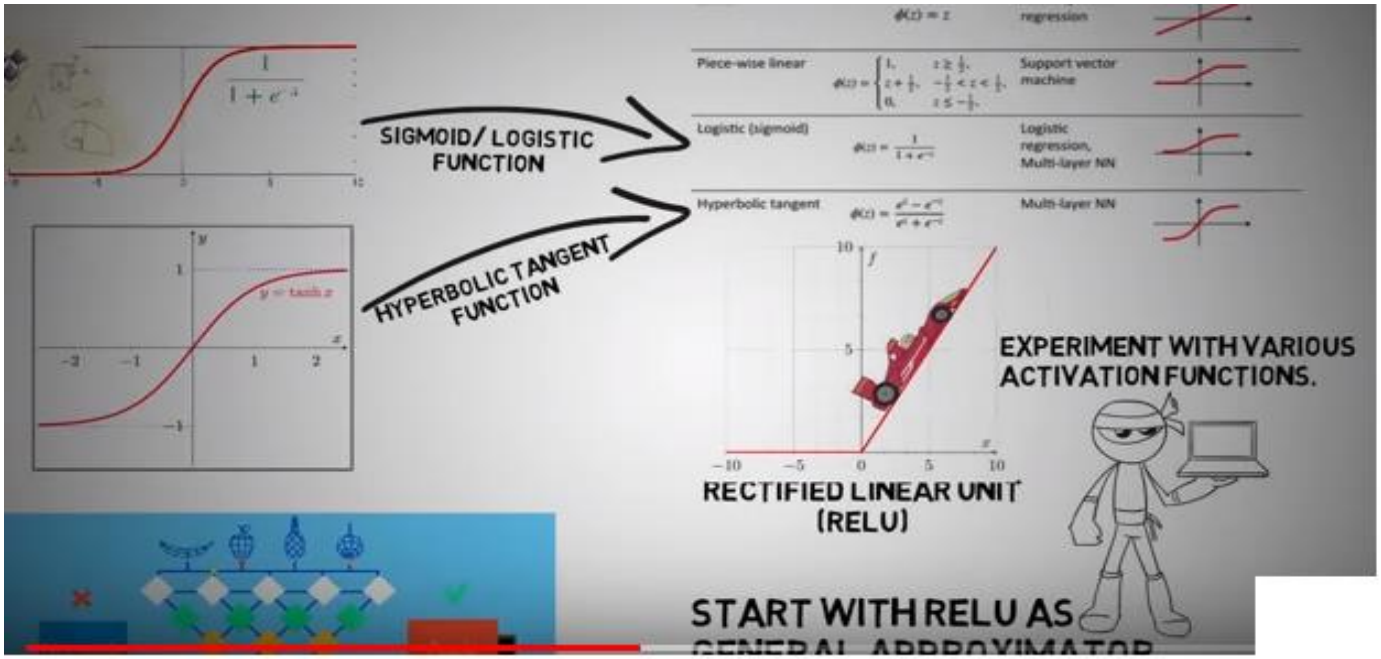




Scale drives deep learning progress

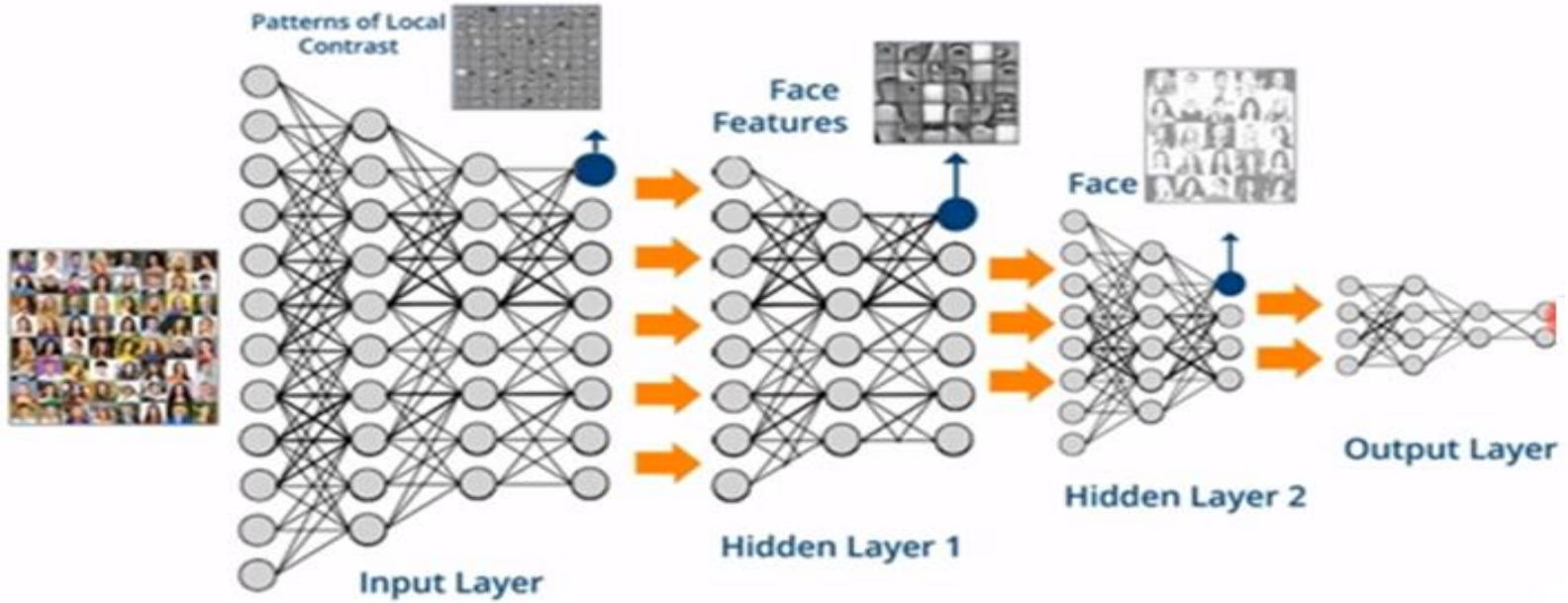
- Data
- Computation
- Algorithms



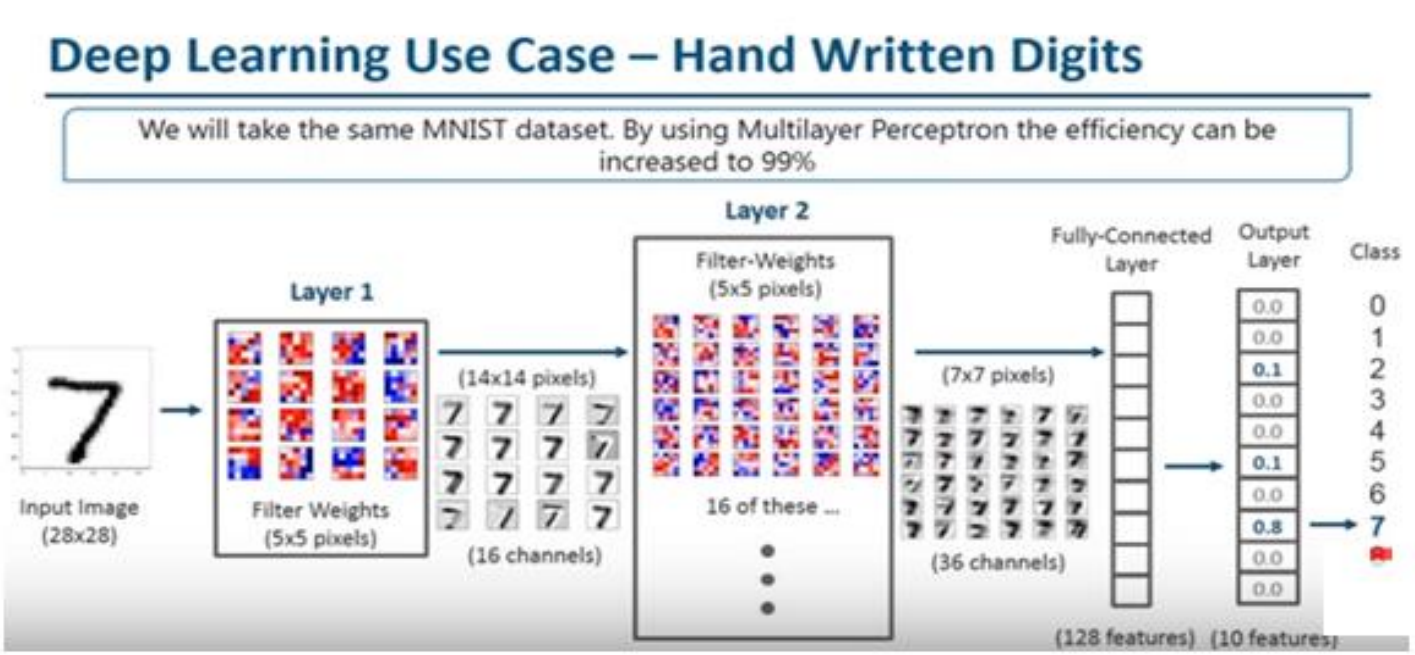


Deep Learning Example

Perform Image recognition using DNN



- Pass the high dimensional data to the input layers.
- In order to match the dimensionality of data, the input layer will have multiple sub-layers of perceptron's to consume the entire input
- Output received from input layer only find edges, patterns of local contrast...etc and fed to Hidden layer 1—where it will be able to-identify features like eyes, nose...etc
- Hidden layer 2--where it will be able to form the entire face and send to output layer and classified and given a name. (example: This image is of this person, name of the person, face of this person...etc)
- If NN is not deep enough, then it will not be able to accurately identify the image.



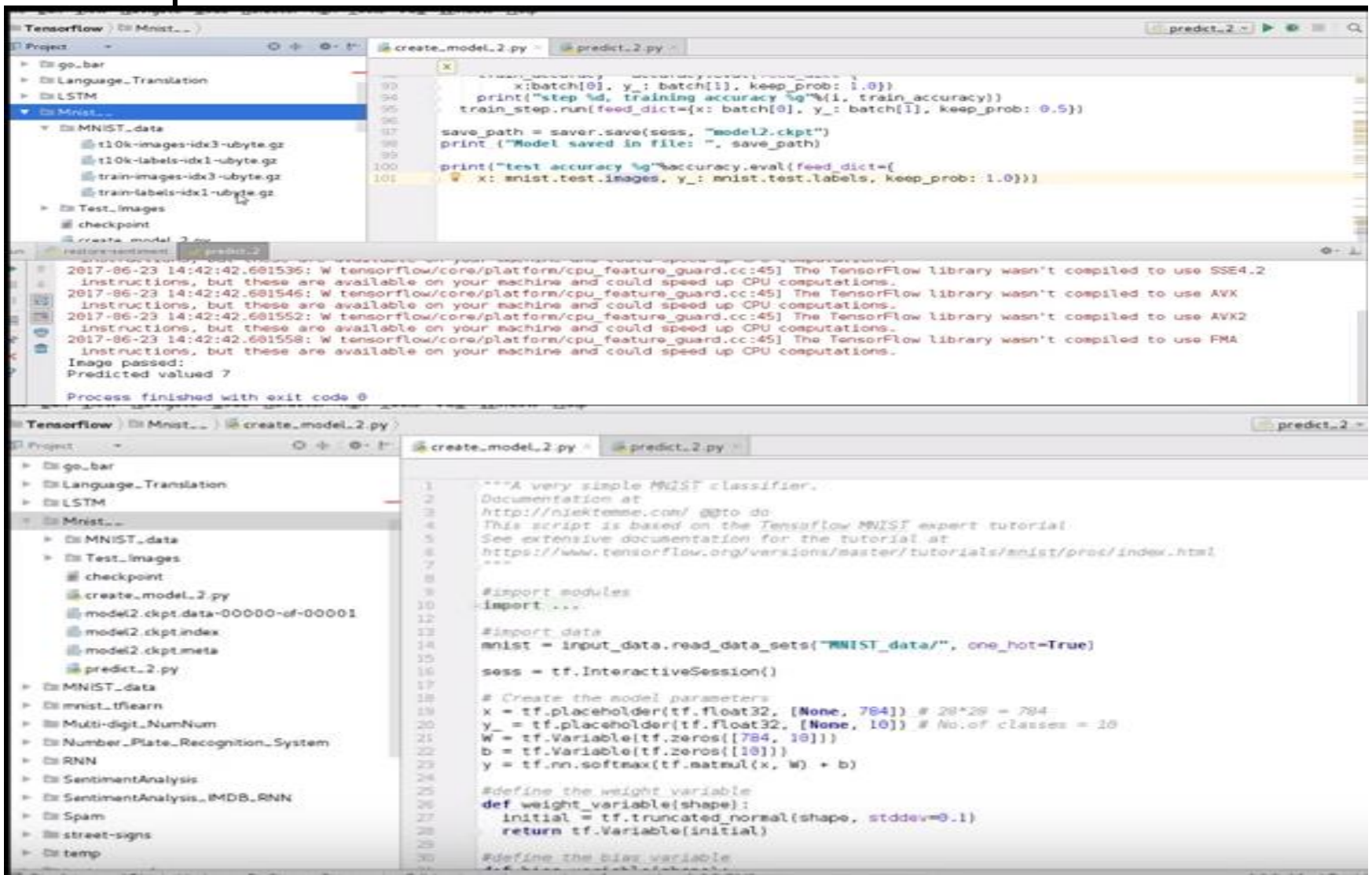
Example Lest consider MNIST dataset

- 60k training sample and 10k testing samples of Hand written digit images.
- The Task to train the model which accurately identifies the digit present on the image
- To solve this use case—create Deep NN with multiple Hidden layer to process 60k image pixel by pixel and final output or the m/c will identify the number that was there on the particular image

- Output layer will be an Array of the index 0-9, where each index correspond for respective digit
- index 0—probability of zero being the digit present on the input image
- index 2---value is 0.1-probability of 2 of being the digit
- The highest probability in this array is 0.8 which is present at the index 7 of the array, hence the number present on the image is 7

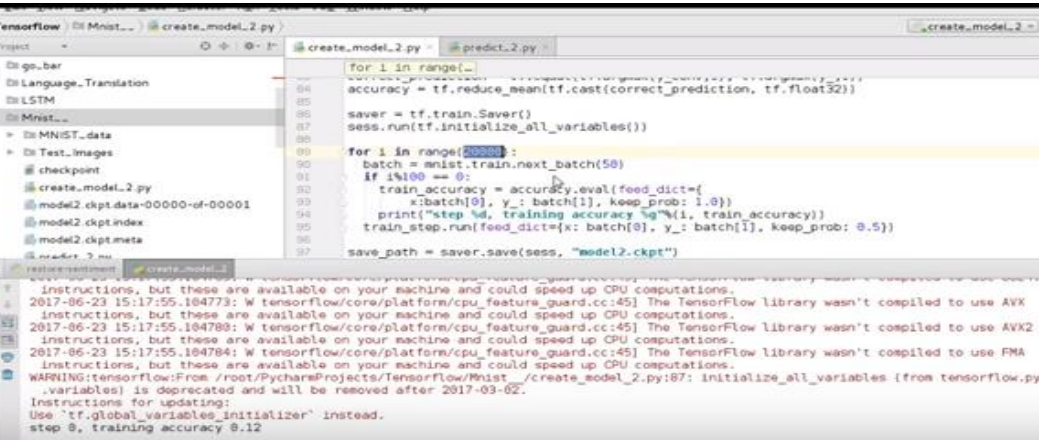
Framework or technique used isTensorflow—open source Google library for DL

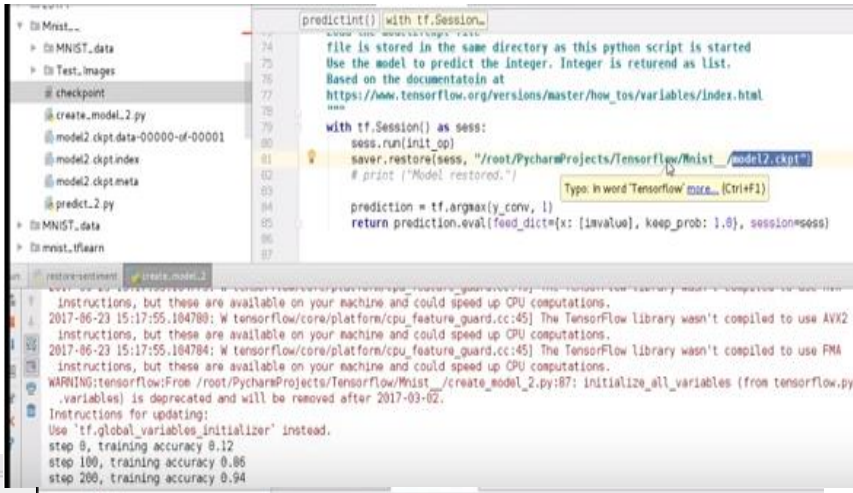
Example Lest consider MNIST dataset

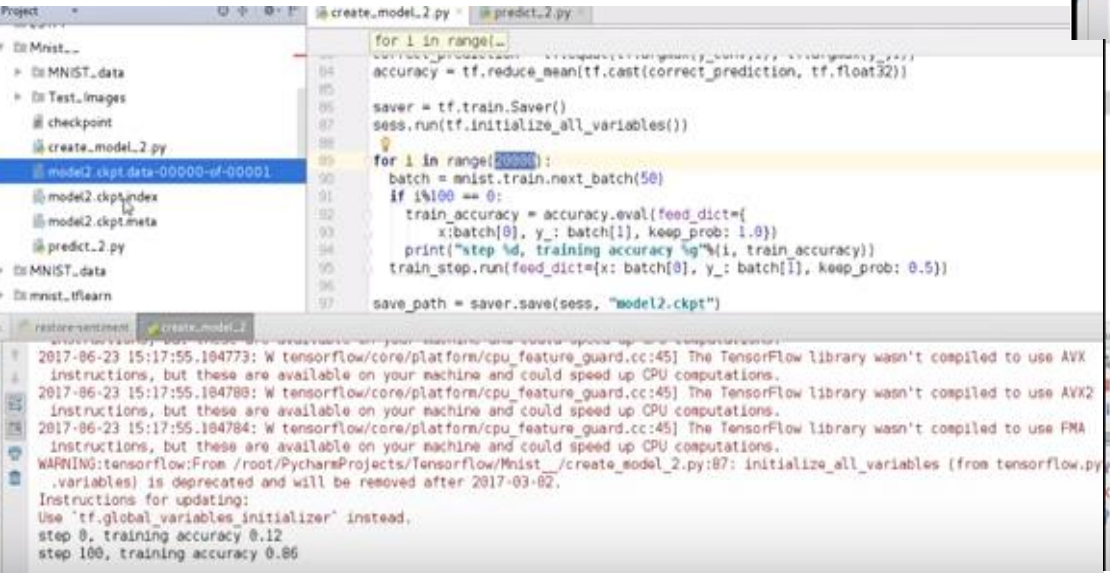


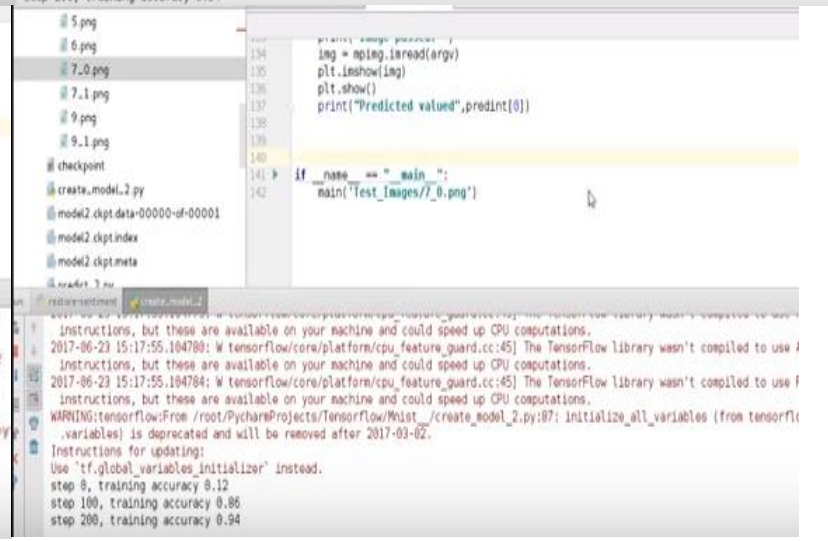
Framework or technique used is Tensorflow—open source Google library for DL

Example Lest consider MNIST dataset

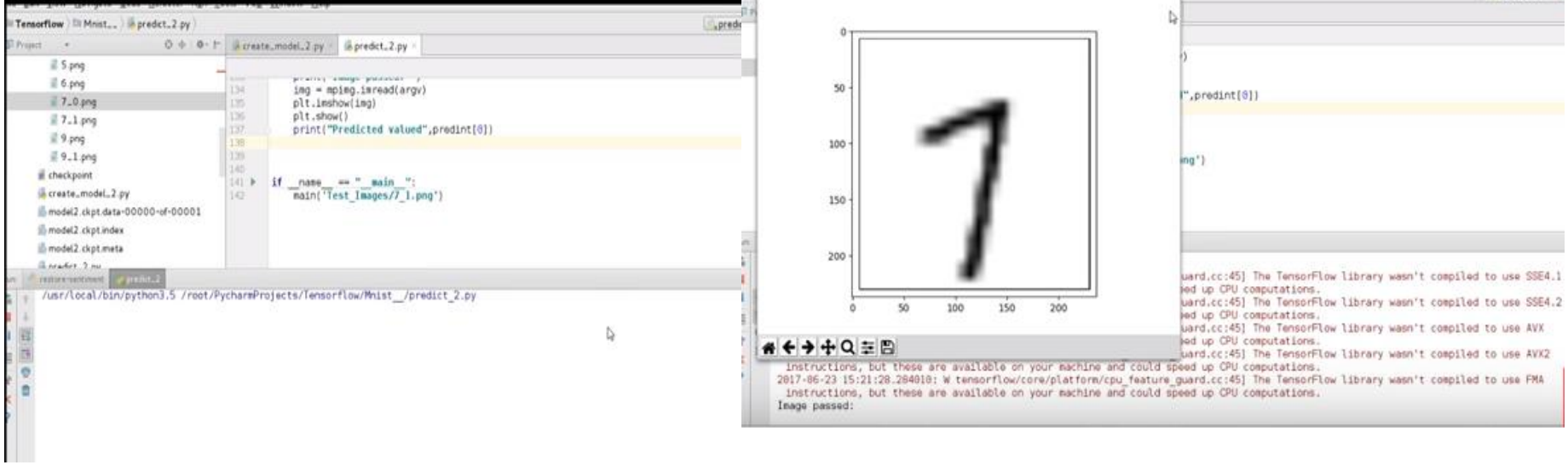


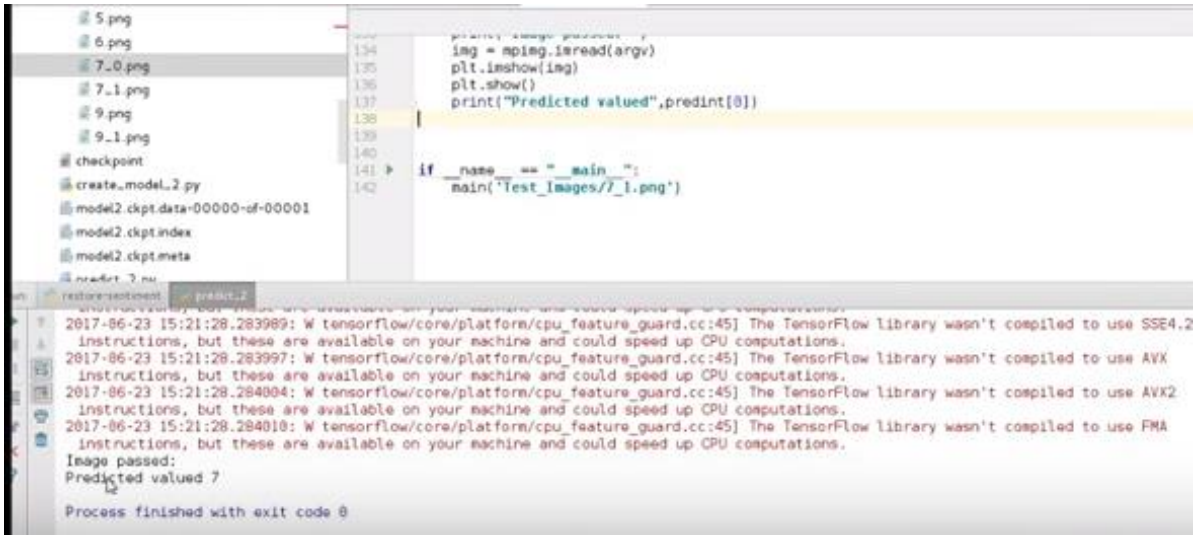






- MNIST dataset download—which get extracted when program get executed
- Deep NN—able to create a model and to process all images to train the m/c
- Crete DNN—Hidden layer to process the image
- Complete the training steps—(which is set to 20 in this example to achieve particular accuracy)
- A model is created with 92% accuracy—92 predictions will be correct meaning out of 100 images 92 predictions are correct


Tensorflow
Mnist_
predct_2.py
create_model_2.py
predct_2.py
5.png
6.png
7_0.png
7_1.png
9.png
9_1.png
checkpoint
create_model_2.py
model2.ckpt.data-00000-of-00001
model2.ckpt.index
model2.ckpt.meta
img = mpimg.imread(argv)
plt.imshow(img)
plt.show()
print("Predicted valued",predint[0])
if __name__ == "__main__":
main('Test_Images/7_1.png')
/usr/local/bin/python3.5 /root/PycharmProjects/Tensorflow/Mnist_/predict_2.py
2017-06-23 15:21:28.284010: W tensorflow/core/platform/cpu_feature_guard.cc:45] The TensorFlow library wasn't compiled to use FMA
instructions, but these are available on your machine and could speed up CPU computations.
Image passed:


5.png
6.png
7_0.png
7_1.png
9.png
9_1.png
checkpoint
create_model_2.py
model2.ckpt.data-00000-of-00001
model2.ckpt.index
model2.ckpt.meta
img = mpimg.imread(argv)
plt.imshow(img)
plt.show()
print("Predicted valued",predint[0])
if __name__ == "__main__":
main('Test_Images/7_1.png')
2017-06-23 15:21:28.283989: W tensorflow/core/platform/cpu_feature_guard.cc:45] The TensorFlow library wasn't compiled to use SSE4.2
instructions, but these are available on your machine and could speed up CPU computations.
2017-06-23 15:21:28.283997: W tensorflow/core/platform/cpu_feature_guard.cc:45] The TensorFlow library wasn't compiled to use AVX
instructions, but these are available on your machine and could speed up CPU computations.
2017-06-23 15:21:28.284004: W tensorflow/core/platform/cpu_feature_guard.cc:45] The TensorFlow library wasn't compiled to use AVX2
instructions, but these are available on your machine and could speed up CPU computations.
2017-06-23 15:21:28.284010: W tensorflow/core/platform/cpu_feature_guard.cc:45] The TensorFlow library wasn't compiled to use FMA
instructions, but these are available on your machine and could speed up CPU computations.
Image passed:
Predicted valued 7
Process finished with exit code 0

Applications Of Deep Learning

Some amazing and recent applications of Deep Learning are:

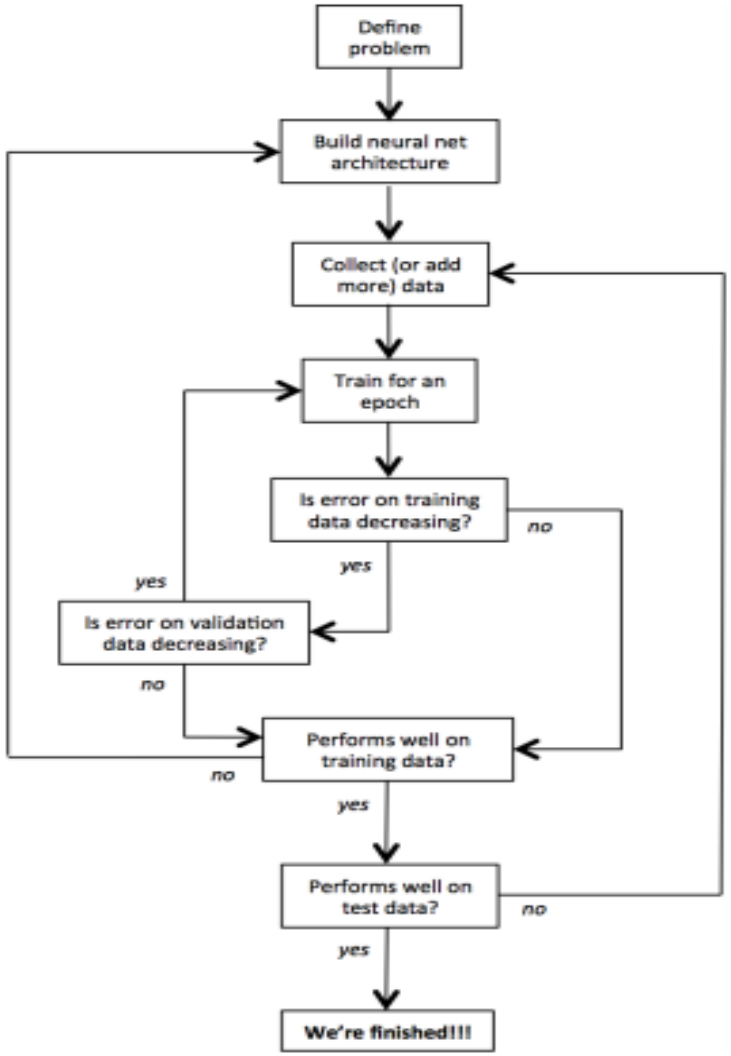
- Automatic Machine Translation.
- Object Classification in Photographs.
- Automatic Handwriting Generation.
- Character Text Generation.
- Image Caption Generation.
- Colorization of Black and White Images.
- Automatic Game Playing.

Face recognition

Applications Of Deep Learning - Google Lens

- Google Lens is a set of vision-based computing capabilities that allows your smartphone to understand what's going on in a photo, video or a live feed.
- For instance, point your phone at a flower and Google Lens will tell you on the screen which type of flower it is.
- You can aim the camera at a restaurant sign to see reviews and other information.

Subscribe



- Gathering Data
- Preparing that Data
- Choosing a Model
- Training
- Evaluation
- Hyperparameter Tuning

Detailed workflow for training and evaluating a deep learning model

Thank you
for your Attention!