Course: Artificial Intelligence and Machine Learning Code: 20CS51I

WEEK - 10: DEEP LEARNING

- **Limitations of Machine Learning**
- **What is deep learning?**
- **Deep learning models**
- **Deep Learning Applications**
- **Deep learning frameworks**

Session No. 1

Limitations of Machine Learning :

The following factors serve to limit it:

1. Data Acquisition

Machine Learning requires massive data sets to train on, and these should be inclusive/unbiased, and of good quality. There can also be times where they must wait for new data to be generated.

2. Time and Resources

ML needs enough time to let the algorithms learn and develop enough to fulfill their purpose with a considerable amount of accuracy and relevancy. It also needs massive resources to function. This can mean additional requirements of computer power for you.

3. Interpretation of Results

Another major challenge is the ability to accurately interpret results generated by the algorithms. You must also carefully choose the algorithms for your purpose.

4. High error-susceptibility

ML is autonomous but highly susceptible to errors. Suppose you train an algorithm with data sets small enough to not be inclusive. You end up with biased predictions coming from a biased training set. This leads to irrelevant advertisements being displayed to customers. In the case of ML, such blunders can set off a chain of errors that can go undetected for long periods of time. And when they do get noticed, it takes quite some time to recognize the source of the issue, and even longer to correct it.

5. Data

This is the most obvious limitation. If the model you feed is poor, it can only give you bad results. This can be manifested in two ways: lack of data and lack of**Good**data.

Missing data

Many machine learning algorithms require a lot of data before they begin to provide useful results. A good example is the neural network. A neural network is a data phagocytic machine that requires a lot of training data. The larger the architecture, the more data is needed to produce a viable outcome. Reusing data is a bad idea, and data growth is useful to some extent, but having more data is always the preferred solution.

Lack of good data

Despite its appearance, this is different from the above comments. Let's imagine that you think you can trick your neural network by generating 10,000 fake data points. What happens when you put it in? It will train itself, and then when it's tested on an invisible data set, it won't work. You have data, but the quality of the data does not meet the standard.

6. Interpretability:

Interpretability is one of the main problems of machine learning. Machine learning model fairness and interpretability are critical for data scientists, researchers and developers **to explain their models and understand the value and accuracy of their findings.** Unless these models can be explained, these models can become powerless, and the process of human interpretation follows rules far beyond the technical strength.

What is deep learning?

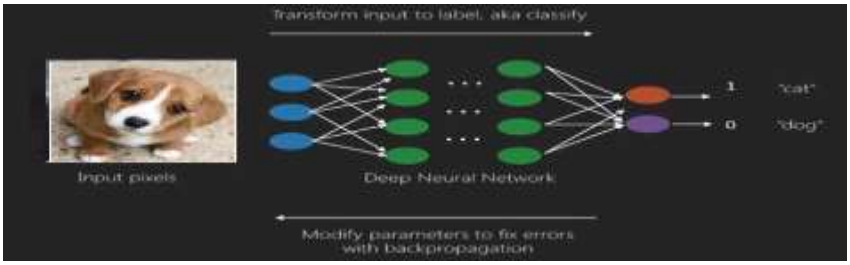
Deep learning is a branch of machine learning which is completely based on artificial neural networks, as neural network is going to mimic the human brain so deep learning is also a kind of mimic of human brain. In deep learning, we don't need to explicitly program everything.

OR

Deep learning is a type of machine learning that works based on the structure and function of the human brain. It uses artificial neural networks to perform sophisticated computations on large amounts of data

Deep learning models:

1. **Deep Neural Network** – It is a neural network with a certain level of complexity (having multiple hidden layers in between input and output layers). They are capable of modeling and processing non-linear relationships.



2. **Deep Belief Network(DBN)** – It is a class of Deep Neural Network. It is multi-layer belief networks.

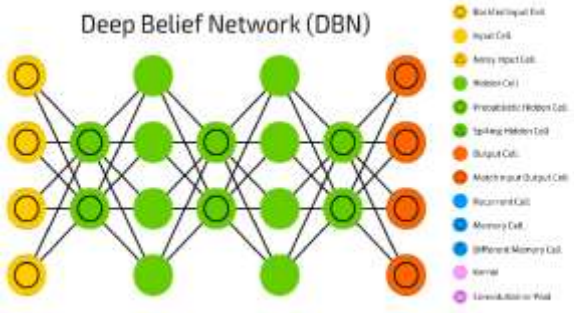
   Steps to performing DBN:

   a. Learn a layer of features from visible units using Contrastive Divergence algorithm.

   b. Treat activations of previously trained features as visible units and then learn features of

features.

c. Finally, the whole DBN is trained when the learning for the final hidden layer is achieved.



3. **Recurrent** (perform same task for every element of a sequence) **Neural Network** – Allows for parallel and sequential computation. Similar to the human brain (large feedback network of connected neurons). They are able to remember important things about the input they received and hence enables them to be more precise.

   RNN works on the principle of saving the output of a particular layer and feeding this back to the input in order to predict the output of the layer.

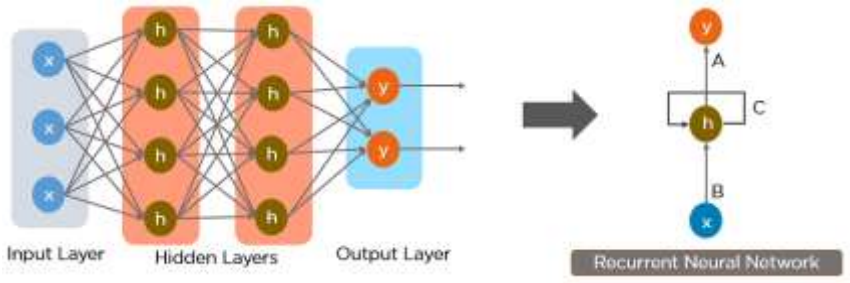


Fig: Simple Recurrent Neural Network

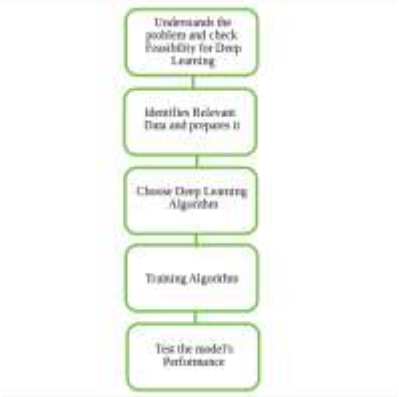
Here is the list of top 10 most popular deep learning algorithms:

1. Convolutional Neural Networks (CNNs)
2. Long Short Term Memory Networks (LSTMs)
3. Recurrent Neural Networks (RNNs)
4. Generative Adversarial Networks (GANs)
5. Radial Basis Function Networks (RBFNs)
6. Multilayer Perceptrons (MLPs)
7. Self Organizing Maps (SOMs)
8. Deep Belief Networks (DBNs)
9. Restricted Boltzmann Machines(RBMs)

10. Autoencoders

Working process :

First we need to identify the actual problem in order to get the right solution and it should be understood, the feasibility of the Deep Learning should also be checked (whether it should fit Deep Learning or not). Second, we need to identify the relevant data which should correspond to the actual problem and should be prepared accordingly. Third, Choose the Deep Learning Algorithm appropriately. Fourth, Algorithm should be used while training the dataset. Fifth, Final testing should be done on the dataset.



Real Life Examples :

1. How to recognize square from other shapes?

a) Check the four lines!

b) Is it a closed figure?

c) Does the sides are perpendicular from each other?

d) Does all sides are equal?

So, Deep Learning is a complex task of identifying the shape and broken down into simpler tasks at a larger side

2.Recognizing an Animal! (Is it a Cat or Dog?)

Defining facial features which are important for classification and system will then identify this automatically.(Whereas Machine Learning will manually give out those features for classification)

Advantages :

1. **Best in-class performance on problems.**
2. **Reduces need for feature engineering.**
3. **Eliminates unnecessary costs.**
4. **Identifies defects easily that are difficult to detect.**

Disadvantages :

1. **Large amount of data required.**
2. **Computationally expensive to train.**
3. **No strong theoretical foundation.**

Applications :

1. **Automatic Text Generation** – Corpus of text is learned and from this model new text is generated, word-by-word or character-by-character. Then this model is capable of learning how to spell, punctuate, form sentences, or it may even capture the style.
2. **Healthcare** – Helps in diagnosing various diseases and treating it.
3. **Automatic Machine Translation** – Certain words, sentences or phrases in one language is transformed into another language (Deep Learning is achieving top results in the areas of text, images).
4. **Image Recognition** – Recognizes and identifies peoples and objects in images as well as to understand content and context. This area is already being used in Gaming, Retail, Tourism, etc.
5. **Predicting Earthquakes** – Teaches a computer to perform viscoelastic computations which are used in predicting earthquakes.

Deep Learning Frameworks :

Deep learning (DL) frameworks offer building blocks for designing, training, and validating deep neural networks through a high-level programming interface. Widely-used DL frameworks, such as PyTorch, TensorFlow, PyTorch Geometric, DGL, and others, rely on GPU-accelerated libraries, such as cuDNN, NCCL, and DALI to deliver high-performance, multi-GPU-accelerated training.

1. **PyTorch**

PyTorch is a Python package that provides two high-level features:

- Tensor computation (like numpy) with strong GPU acceleration.
- Deep Neural Networks (DNNs) built on a tape-based autograd system.

Reuse your favorite Python packages, such as numpy, scipy and Cython, to extend PyTorch when needed.

2. TensorFlow

TensorFlow is an open source software library for numerical computation using data flow graphs. Nodes in the graph represent mathematical operations, while the graph edges represent the multidimensional data arrays (tensors) that flow between them. This flexible architecture allows you to deploy computation to one or more CPUs or GPUs in a desktop, server, or mobile device without

rewriting code. For visualizing TensorFlow results, TensorFlow offers TensorBoard, a suite of visualization tools.

3. JAX

JAX is a Python library designed for high-performance numerical computing and machine learning research. JAX can automatically differentiate native Python and implement the NumPy API. With just a few lines of code change, JAX enables distributed training across multi-node, multi-GPU systems, with accelerated performance through an XLA-optimized kernel on NVIDIA GPUs. Both Python and NumPy are widely used and familiar, making JAX simple, flexible, and easy to adopt.

4. PaddlePaddle

PaddlePaddle provides an intuitive and flexible interface for loading data and specifying model structures. It supports CNN, RNN, and multiple variants, and easily configures complicated deep models. PaddlePaddle also provides extremely optimized operations, memory recycling, and network communication, and makes it easy to scale heterogeneous computing resources and storage to accelerate the training process.

5.MXNet

MXNet is a DL framework designed for both efficiency and flexibility. It allows you to mix the flavors of symbolic programming and imperative programming to maximize efficiency and productivity.At its core is a dynamic dependency scheduler that automatically parallelizes both symbolic and imperative operations on-the-fly. A graph optimization layer on top of that makes symbolic execution fast and memory efficient. The library is portable and lightweight, and it scales to multiple GPUs and machines.

Difference between Machine Learning and Deep Learning:

Sl.no	Machine Learning	Deep Learning
1	Works on small amount of Dataset for accuracy	Works on Large amount of Dataset.
2	Dependent on Low-end Machine.	Heavily dependent on High-end Machine
3	Divides the tasks into sub-tasks, solves them individually and finally combine the results.	Solves problem end to end.
4	Takes less time to train.	Takes longer time to train.
5	Testing time may increase.	Less time to test the data.