Deep Learning in Computer Vision

Sheryl Mathew
School of Electrical Engineering and Computer Science
Washington State University
Pullman, WA, 99164

Abstract

Deep Learning is considered to be the golden era of Machine learning. The different types of Machine learning like Supervised learning, Unsupervised learning and Reinforcement learning are discussed. The most significant deep learning schemes like Artificial Neural Network, Convolutional Neural Network, Deep Boltzmann Machines and Deep Belief Networks are analyzed along with their advantages and disadvantages. An overview of computer vision applications which includes Image Classification, Object Detection and Face recognition is given.

1 Introduction

Abstract

Deep Learning is considered to be the golden era of Machine learning. The different types of Machine learning like Supervised learning, Unsupervised learning and Reinforcement learning are discussed. The most significant deep learning schemes like Artificial Neural Network, Convolutional Neural Network, Deep Boltzmann Machines and Deep Belief Networks are analyzed along with their advantages and disadvantages. An overview of computer vision applications which includes Image Classification, Object Detection and Face recognition is given.

2 Introduction

3 Types of Deep Learning

3.1 Supervised Learning

Supervised learning is the method in which we train a model to learn the mapping from the input x which is a d-dimensional vector called features or attributes to the output y which can be either categorical or real-valued scalar.

If the output is a categorical value, then it is called classification problem and if it is a real-valued scalar, then it is called a regression problem. Examples of binary label classification includes malware classification [**?**], comment abuse classification [**?**] and multi-label classification using deep learning includes classifying brain tumors into normal, glioblastoma, sarcoma and metastatic bronchogenic carcinoma tumors [**?**], classifying plants into 22 different species [**?**] among others. Examples of regression includes monocular global localization in robots [**?**].

The dataset is divided into train and test dataset. The model is trained on the training data which contains the output y which needs to be predicted. This model is then tested on the testing data to determine the ability of the model to generalize on any new data based on its accuracy, precision, recall and F1 score. The process of supervised machine learning is explained in Figure 1 [**?**].

The advantage of supervised learning is its ability to learn a perfect decision boundary that is we can classify the data accurately into its different labels and its disadvantages includes overfitting to the data due to the lack of availability of large amount of labelled data. If new data occurs during testing which was not present in training, then the classifier cannot predict the new data.

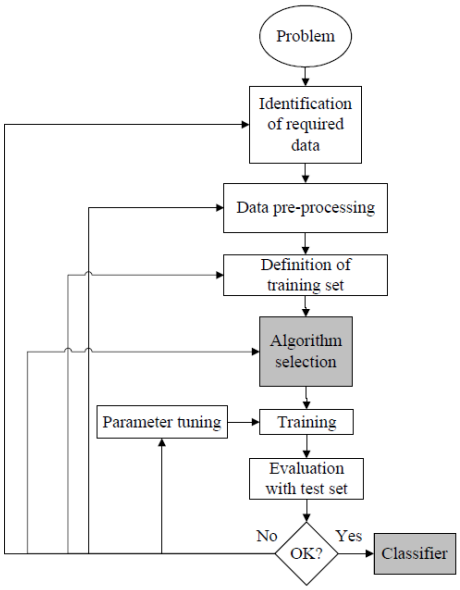

Figure 1. Supervised Machine Learning Workflow

3.2 Unsupervised Learning

Unsupervised learning is a method in which no output labels are provided on which the data can be trained instead we find interesting patterns in the data. Inferences are made on data to better understand the features of the dataset and hence make decisions which are data driven. This is done through clustering, principal component analysis and density estimation.

Clustering is the process of dividing the data into different groups called clusters such that data which are similar to each other are present in the same cluster. This can be visualized in Figure 2 [?]. It is an iterative process. In K-means clustering, each data point is assigned to one of the K clusters and in hierarchical clustering, a hierarchy of cluster is built by initially assigning each data point to its own cluster and then merging the two closest clusters together until only one cluster remains. This is shown in Figure 3 [?]. Example for clustering in deep learning is text clustering [?].

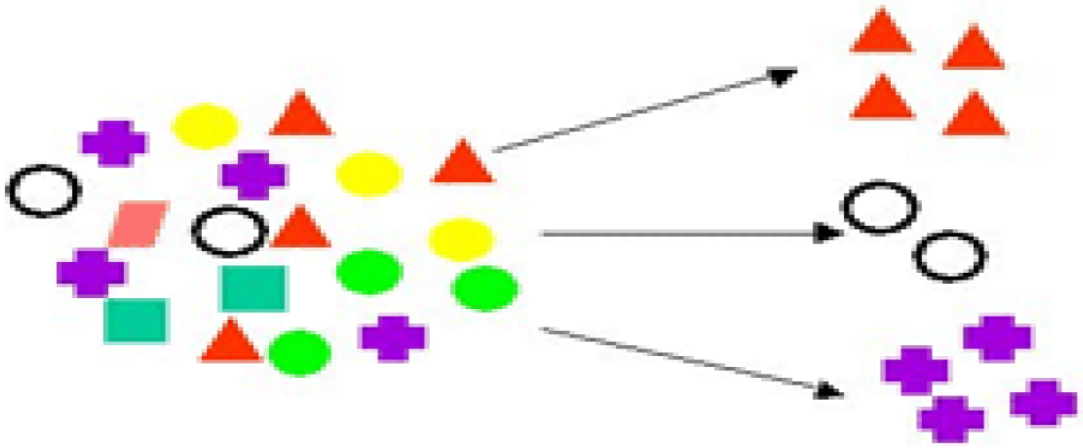

Figure 2. Clustering

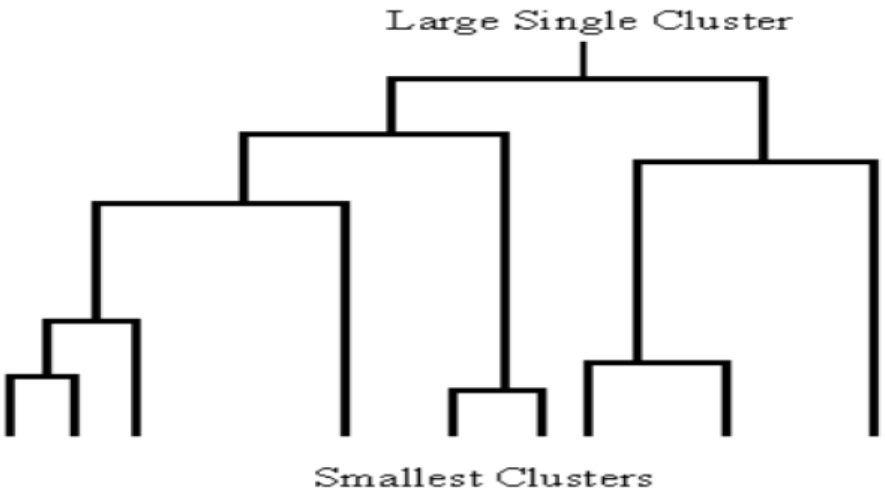

Figure 3. Hierarchical clustering

Principal Component Analysis is a dimensionality reduction method which is used for feature extraction. This is done by combining the input variables in a specific manner and finding the most important variables so that we can drop the unimportant ones. All the variables after PCA are independent of each other. In Figure 4 [**?**], the biplot for the correlation matrix of the PCA of fossil teeth data is shown. The different variables are represented as arrows and the tooth markers as numbers. The high proportion of variability means that there is a group of highly correlated variables [**?**]. Example for PCA in deep learning is document image analysis [**?**].

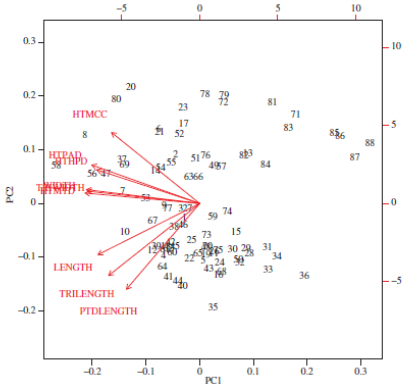

Figure 4. Principal Component Analysis

Density estimation finds underlying probability distribution of the input variables using different statistical models. It is the process of using discrete data points to estimate a continuous density field. Figure 5 [**?**], shows the kernel density estimate and contributions of different data points (dashed curve) along the true underlying density (solid curve). Example for Density estimation in deep learning is traffic density estimation [**?**].

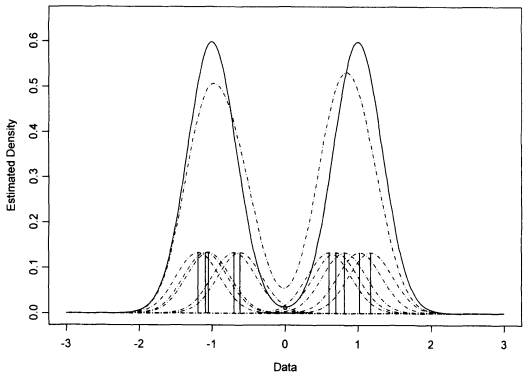

Figure 5. Kernel Density Estimate

The advantages of unsupervised learning are that it computation time and speed is less, there is no need for domain knowledge about the data and there is a huge reservoir of unlabeled data. The disadvantage is that after classification the data scientist must interpret and label the classes.

3.3 Reinforcement Learning

Reinforcement learning is a method that contains an agent to learn about its environment through trial and error. This is done by using feedback that it receives when it performs an action in the environment. The feedback is in the form of rewards or penalties. The goal is the maximize the reward. The process is explained in Figure 6 [?].

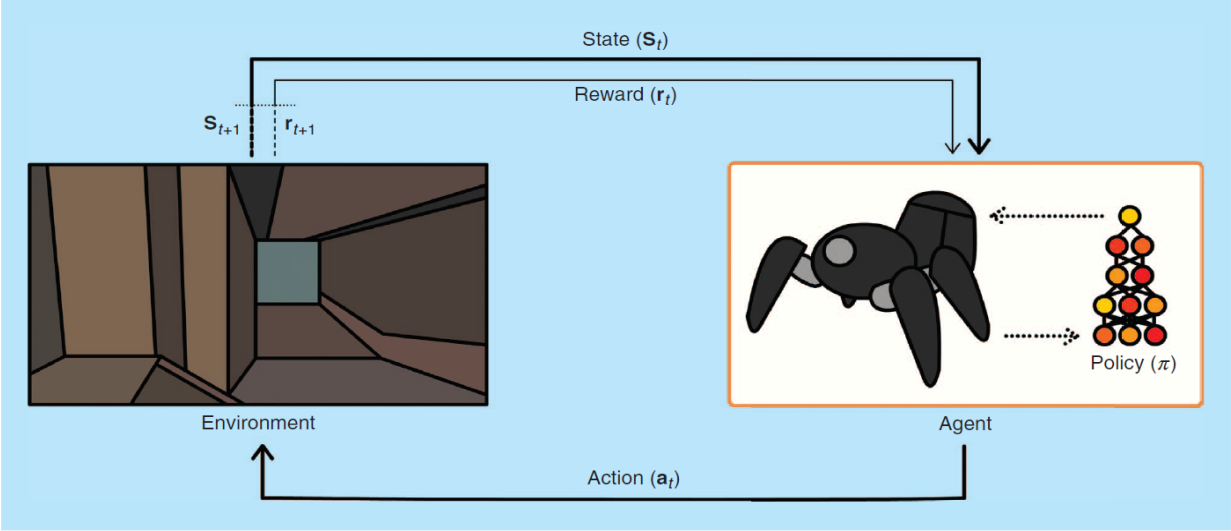

Figure 6. Perception-Action-Learning Loop

It involves making decisions sequentially therefore labels are given to the sequences of dependent decisions. The main approaches in Reinforcement learning are Policy Learning and Q-learning. Examples of Reinforcement learning in Deep learning includes Playing Atari 2600 games [?] and human level control [?].

Policy Learning is the method in which only the state is known. Policy is detailed instructions on what to do at every state based on which the agent will select the action to perform. It is very difficult to know the policy beforehand and required a lot of knowledge of the function that will map the different states to a goal. Q-learning is the method in which the state and action is known. The agent is not told what to do at each state instead it is given a framework in which it can make its own decisions.

The main thing to be aware of is the exploration versus exploitation trade-off policy in reinforcement learning. There should be an optimal method that is used to decide whether the agent has to explore i.e. agent takes a random action or exploit i.e. agent takes an action in such a way that it will maximize the reward.

The advantage is that it uses the deeper knowledge about the domain and the disadvantage are that it requires a lot of data to train on, the model that is generated is not very generalizable and instead it is very domain specific and the model is also created based on the rewards set-up by the developers therefore it is based on the bias of the developers.

4 Deep Learning Schemes

4.1 Artificial Neural Network (ANN)

Artificial Neural Networks are computers whose architecture is modelled after the brain structure which contains several neurons which all are interconnected to each other. The output of one neuron may be the input to several other neurons. The output state is changed based on the strength of the electrical or chemical impulse. The learning process in our brain involves activating certain neural connections which reinforces these connections and hence increases the chances of providing the required response given an input. The learning process includes feedback i.e. when the required response is produced the corresponding neural connections get strengthened.

ANN's are the simple clustering of primitive artificial neurons. This is done by creating various layers which are connected to one another. The general structure of ANN is shown in Figure 7 [**?**]. Neurons are called nodes. Some nodes interface with the real world to get the inputs, some nodes interface with the real world to provide the output and the other nodes are hidden. These nodes are grouped together into layers. The 3 different layers are input, hidden and output [**?**]. Most neural networks are fully connected i.e. every hidden node and every output node is connected to every node on either side.

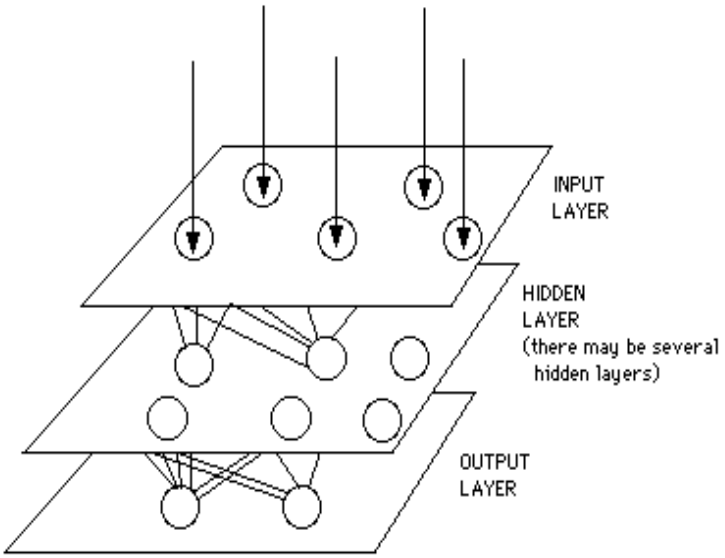

Figure 7. Simple Neural Network

The two different types of connection include Competition where lateral inhibition takes place i.e. the network will choose the output with highest probability and considers the neurons which were responsible for that output and inhibit all the others and Feedback is when the output of one-layer routes back to the previous layer.

The connections between one node and another is called weight. Higher the weight the more influence one node has over another. If the weight is positive, it implies that the one node excites another and if the weight is negative, it implies that one node inhibits another.

Supervised learning implies that both the inputs are outputs are given during training. The outputs are given either along with the input or by manually grading the network's performance. When a neural network is being trained, input is given to the input nodes which is then used to trigger the nodes in the hidden layer and they in turn reach the output nodes. This is called feedforward network Figure 8a. [**?**]. Each node receives input from the nodes on its left, then these inputs are multiplied by the weights of the connections that they move along. Every node adds up all its input values and if the sum is greater than the given threshold value then that node is activated, and this triggers all the nodes which are connected to it on it's right. Backpropagation is used to provide feedback to the network based on comparing the output from the network and the correct output and using the difference between them to modify the weights of the connections between the nodes in the network. This is called feedback network Figure 8b. [**?**]. This causes the network to learn and with time the difference between the actual and the network output reduces.

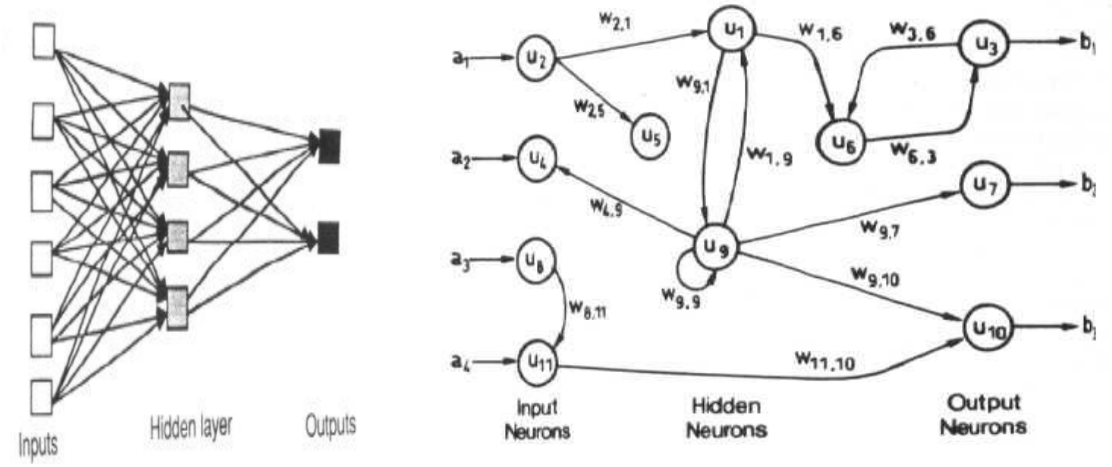

Figure 8. a) Feed Forward Network b) Feed Back Network

In unsupervised learning the network tries to understand the network without any outside help. The network decides which features of the input are to be grouped together. This is referred to as self-organization or adaptation.

Advantages includes the ability to learn how to do tasks based on the training data, ability to represent the training information, parallel computations, recognize patterns in the training data, quick adaptation to changing data and higher performance when compared to other conventional approaches.

Disadvantages includes the necessity of requiring processors with the ability for parallel processing, it is difficult to explain the behavior of the network, structure of the network is determined based on trial and error and training time is very high.

Applications include estimating participation in elections [**?**], designing generic distance relays in power systems [**?**] and prediction of PVT properties of Iran crude oil [**?**]

4.2 Deep Boltzmann Machines (DBM) And Deep Belief Network (DBN)

Deep Boltzmann Machine is a network of symmetrically coupled stochastic binary units. It contains a set of visible units $v \in \{0,1\}^v$ and set of hidden units $h \in \{0,1\}^u$ that are used to learn how to model higher-order correlations between the visible units. In Figure 9a. [?] the General Boltzmann machine consists of a top layer which represents a vector of stochastic binary hidden variables and bottom layer which represents a vector of stochastic binary visible variables. In Figure 9b. [?] the Restricted Boltzmann machine consists of no hidden-to-hidden or visible-to-visible connections.

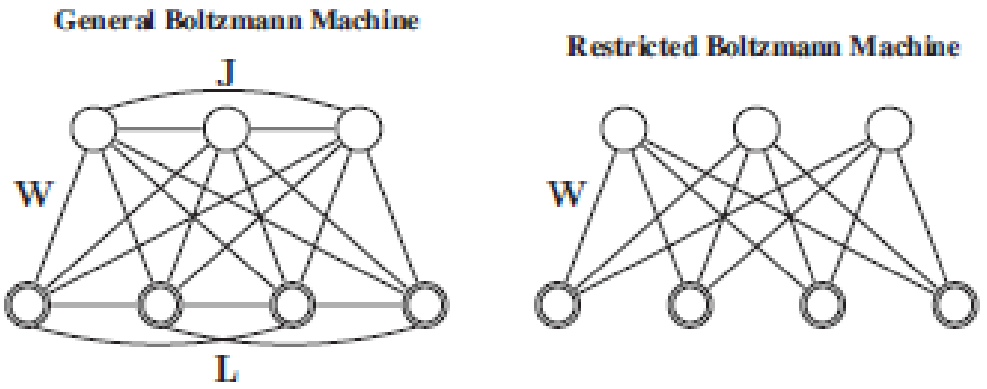

Figure 9. a) General Boltzmann machine b) Restricted Boltzmann machine

Deep Belief Networks consists of a stack of Restricted Boltzmann machines (RBM) wherein each RBM layer communicates with both the previous and subsequent layers. The output of the upper RBM's hidden layer is used as the input of the lower RBM's visible layer. This is shown in Figure 10 [?]. The learning process of RBM is an unsupervised learning approach so a DBN works without supervision but for classification a new supervised learning network has to be added which is used to classify the training data based on the features that have been extracted from DBN. They are used to recognize and generate images, video sequences, and motion capture data.

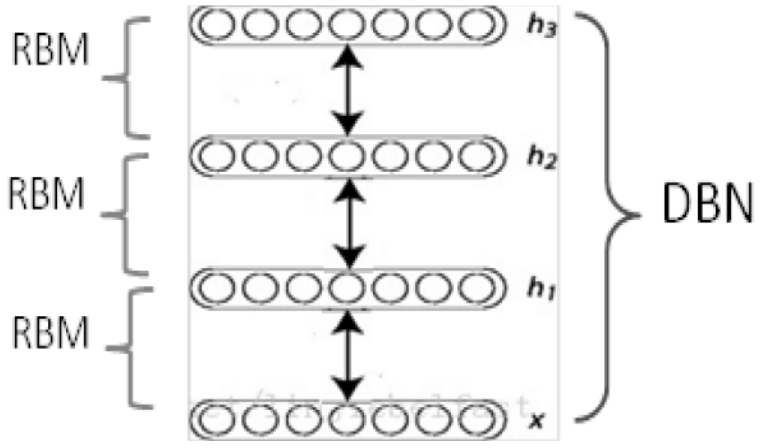

Figure 10. Deep Belief Network

Both Deep Belief Networks and Deep Boltzmann machines are capable of learning internal representations of the data that is used to capture very complex structure in the higher layers. The fastest way to initialize the states of the units in all the layers is by performing a single bottom-up pass using twice the weights to compensate for the initial lack of top-down feedback and the approximate inference procedure after the initial bottom-up pass can include top-down feedback which allows the use of higher-level knowledge to resolve uncertainty about the intermediate-level features. This creates better data-dependent representations and statistics for learning [?].

In Figure 11a. [?], the top two layers form an undirected graph and the remaining layers form a DBN with directed top-down connections and Figure 11b. [?], the DBM consists of both hidden-to-hidden and visible-to-hidden connections and no within-layer connections with all the connections being undirected.

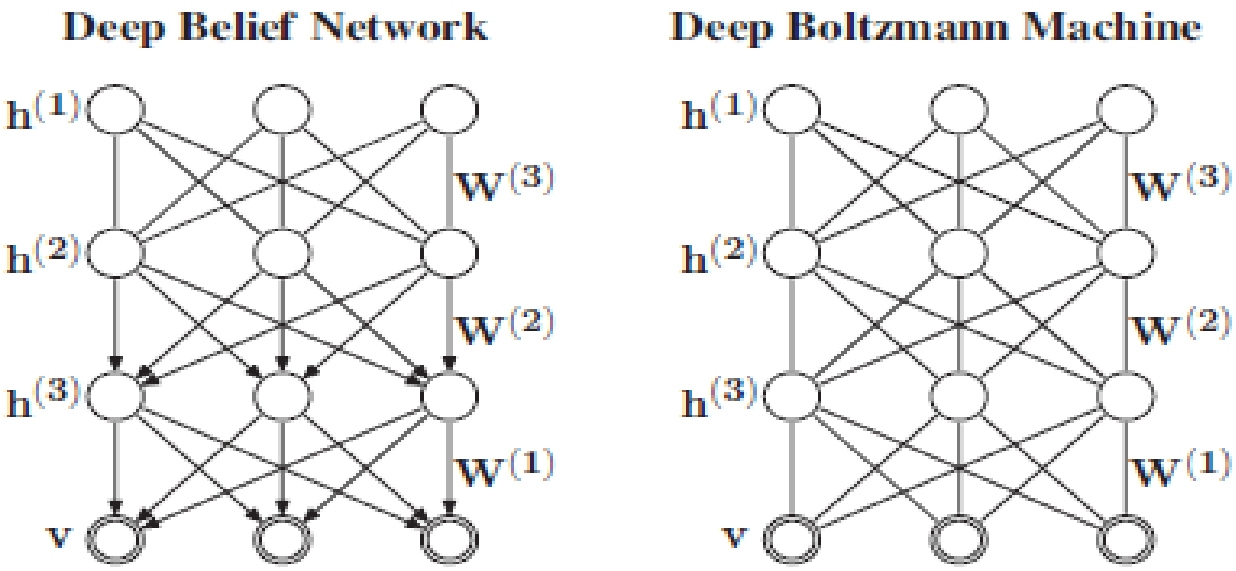

Figure 11. a) Deep Belief Network b) Deep Boltzmann machine

Deep Boltzmann machine is used in Image Recognition [?], 3D Object Detection [?] and for Learning Spam Features [?]. Deep Belief Network is used in Natural Language Understanding [?], Precision Mechanism Quality Inspection [?] and 3D Image Reconstruction [?].

The advantages of DBM are that samples can be created that are similar to the distribution of the training data and includes pattern completion. It comes along with an efficient layer-by-layer pre-training procedure which can then be used to train on unlabeled data and later fine-tuned for a specific task using labeled data. It incorporates uncertainty about ambiguous inputs. It has better generative models. The disadvantage is that the mean-field inference must be calculates for every new test input [?].

The advantages of DBN are that it has similar criteria for the standard deviation in all the sub-voxels and is its capability of capturing more similar images with similar standard deviations in one scene. The disadvantage is that the high complexity of running time [?].

4.3 Convolutional Neural Network (CNN)

Convolutional Neural Networks are similar to Artificial Neural Networks. It just assumes that the input will always be images. The layers are arranged in three dimensions: width, height and depth since the CNN sees images as volumes. The neurons in one layer will not connect to all the neurons in the next layer instead only to a small part of it. The output of the network will be along the depth direction and represented as a single vector containing the probability scores. There are three different types of layers which include convolutional layers, pooling layers and fully connected layers. The stacked form of these layers forms CNN. This is shown in Figure 12 [?].

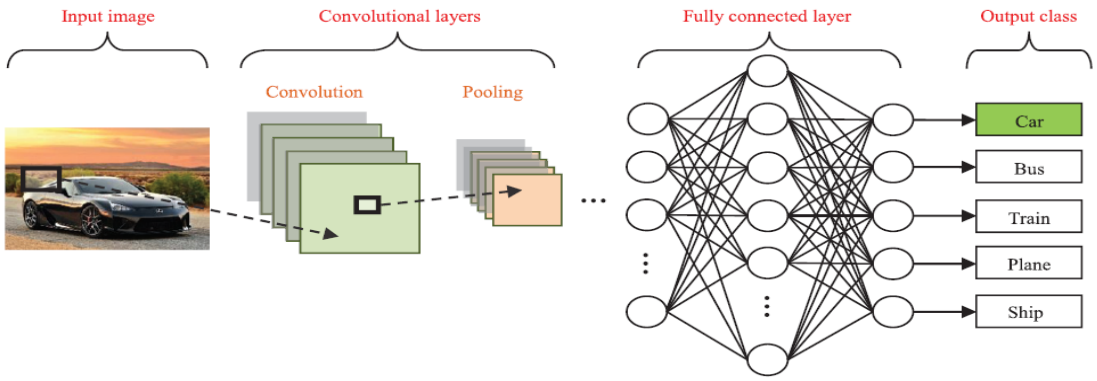

Figure 12. Convolutional Neural Network

Convolutional layers are used to learn the feature representation of the given input image and act as feature extractors. A filter is passed over the input image and as it is being passed over the image, the values present in the filter are multiplied with the image pixels. After multiplying the values, they are summed to get a single number which is stored in the feature map. Then the filter is moved by the number of steps in stride and the process is repeated and the values stored in the feature/activation map. This process is called convolving. They are used to reduce the model complexity and can be optimized by varying the depth, stride and adding zero-padding.

Pooling layers are added to reduce the spatial dimensions of the input. This is done to reduce the number of parameters and hence the computational complexity. The pooling layer is applied over each feature/activation map independently.

Fully connected layers are used to interpret the feature representations that have been generated by the convolutional and pooling layers and performs high-level reasoning. They consist of neurons which are connected to all the neurons before and after it without any layers in between them. This is similar to Artificial Neural Network.

Applications include detection and classification of lung abnormalities [?], human object identification for human robot interaction [?] and Chinese character recognition [?].

Advantages includes the ability of automatic feature extraction and the predications are very fast after training. Disadvantages include its inability to explain its decisions, training time is high and requires a large amount of training data.

Figure 13 [?] shows the training sample of digit images from MNIST dataset. After training the CNN over the data, the activations which are obtained are shown in Figure 14 [?]. It can be observed that CNN has picked up the unique characteristics of each digit. Figure 15 [?] explains the convolutional operation with a filter of 3x3 and stride 1 to produce a feature map of 5X5. While Figure 16 [?] shows the visual representation of the convolutional layer. Figure 17 [?] represents the max pooling operation with filter of size 2x2 and Figure 18 [?] is the example when max pooling is done over the images.

Figure 13. 100-digit images from MNIST dataset

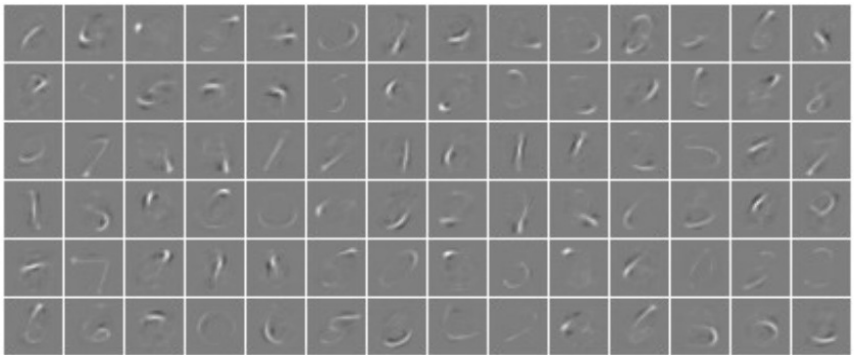

Figure 14. Activations taken from first convolutional layer after training on MNIST data

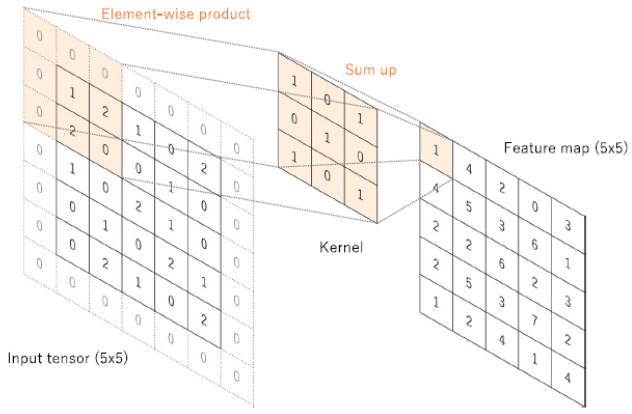

Figure 15. Convolutional operation with zero padding

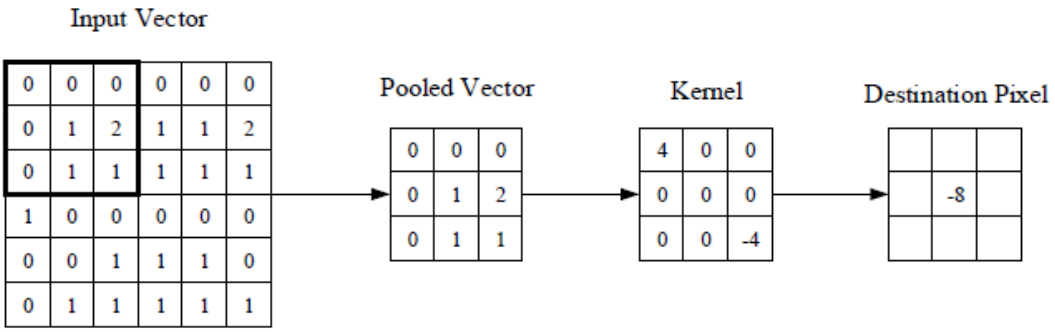

Figure 16. Visual representation of convolutional layer

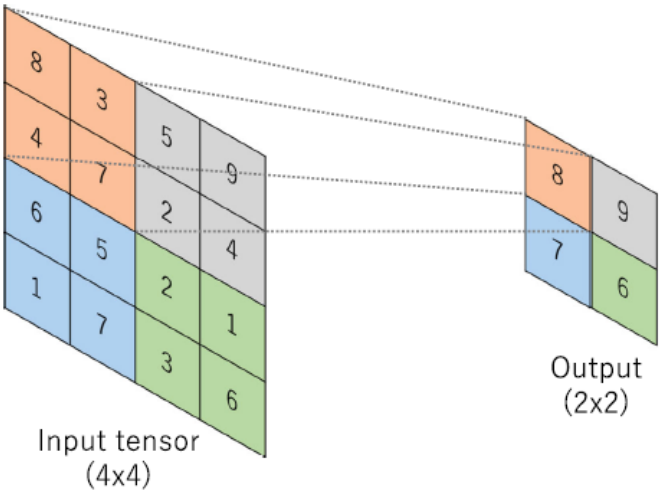

Figure 17. Max Pooling Operation

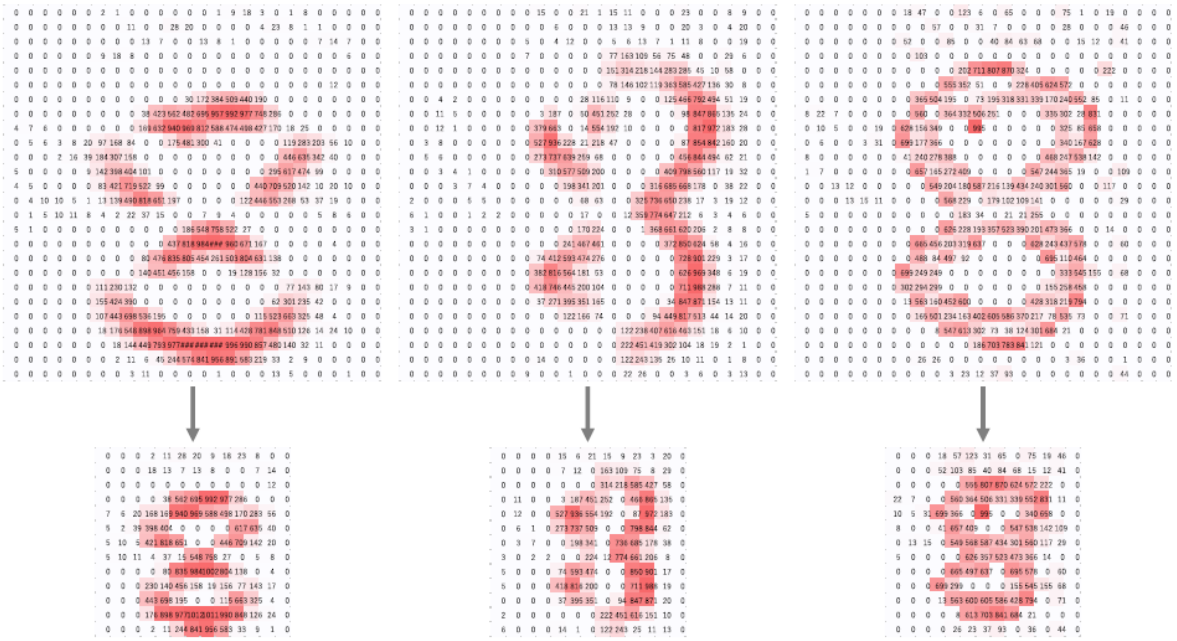

Figure 18. Max Pooling over the digit 8

5 Applications of Deep Learning

5.1 Image Classification

Image classification is the process of predicting the labels for any input image using deep learning especially Convolutional Neural Network. The different methods which are used in image classification are discussed below.

AlexNet [?] uses a deep convolutional neural network to classify images in the ILSVRC-2010 dataset. This brought deep learning into focus and is considered as one of the milestones in deep learning. It produced a top-5 test error rate of 15.4%. It contains 5 convolutional layers, 3 fully connected layers, dropout is applied before the first and second fully connected layer and Relu is applied after every convolutional and fully connected layer.

Figure 19 [?] left shows 8 different ILSVRC-2010 test images along with the correct label written at the bottom of each image. It also contains the 5 most probable labels assigned to them by the model and their corresponding probability. Figure 19 [?] right shows 5 ILSVRC-2010 test images in the first column and the remaining columns contains the 6 training images that produce feature vectors in the last hidden layer with the smallest Euclidean distance from the feature vector for the test image.

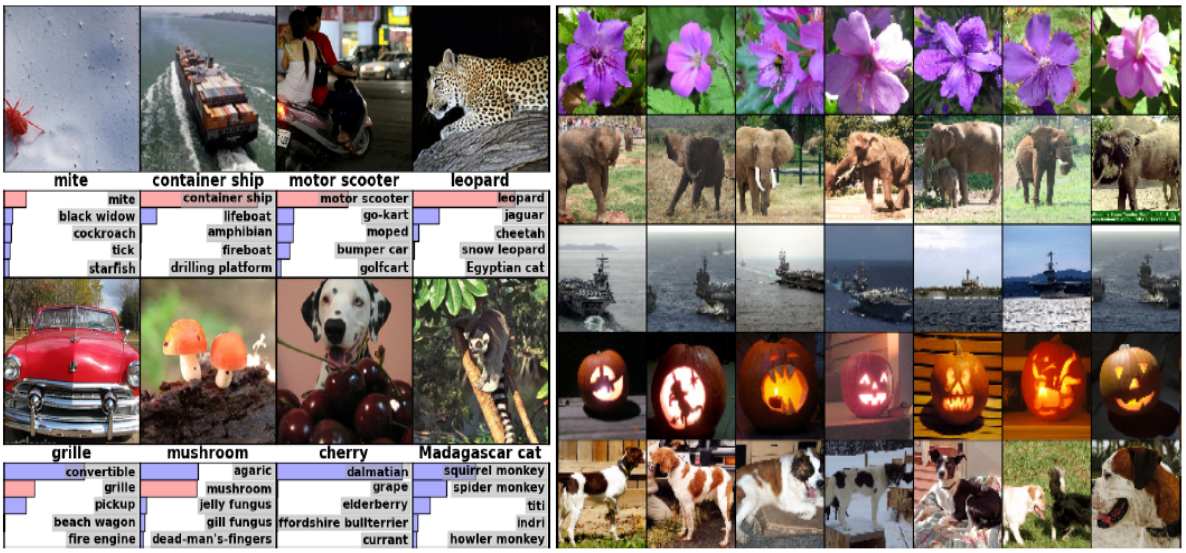

Figure 19. AlexNet

ZFNet [?] uses to classify images in the ImageNet 2012 dataset. It produced a top-5 test error rate of 14.8%. This was done by optimizing the hyper-parameters of AlexNet. Deconvolutional network was a visualization technique that was developed to examine the different feature activations and their relationships to the input.

Figure 20 [?] shows the visualization of features using the Deconvolutional network of ZFNet. Layers 2-5 shows the top 9 activations in the random subset of feature maps and for each feature map the corresponding image patches are shown.

Figure 20. ZFNet

GoogLeNet [**?**] uses a 22-layer deep CNN to classify images in the ILSVRC-2010 dataset. The network used batch normalization, RMSprop and image proportions to produce a top-5 test error rate of 6.67%. It consists of more than 100 layers and 20 parameter layers depth. It also used scale invariance and Hebbian principles. This network reduced the number of parameters from 60 million that was used in AlexNet to 4 million.

5.2 Object Detection

Object Detection is the process of identifying and labelling all the objects in an image. The different methods which are used in object detection are discussed below.

OverFeat [?] used a multiscale and sliding window approach in its convolutional neural network. This model was trained on ILSVRC-2013 dataset to perform classification, localization and detection. It produced a 24.3% mean average precision in object detection. Figure 21 [?] left contains the predictions and Figure 21 [?] right contains the ground truth labels. This shows the difficulty in detection since the image can contains large number of small objects.

Figure 21. OverFeat

R-CNN [?] was trained on ILSVRC-2013 dataset and produced a 29.7% mean average precision. All possible objects were extracted using Selective Search which is a region proposal method. After which CNN is used to extract the features from each region. Lastly SVM is used to classify each region. Figure 22 [?] shows the architecture of R-CNN.

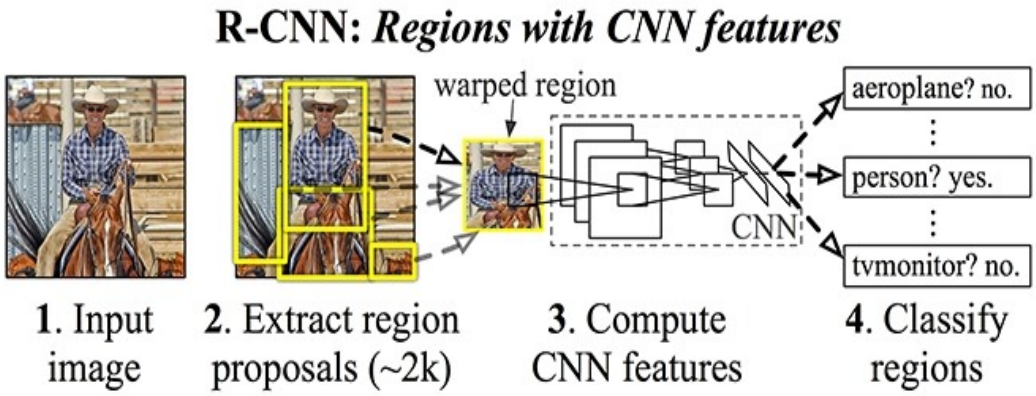

Figure 22. R-CNN

YOLO [?] uses a single CNN which simultaneously predicts multiple bounding boxes and the class probabilities of each box. The detection problem is considered as a regression problem which increases the speed of detection. In Figure 23 [?], input image is resized to size 448x448, single CNN is run and thresholding the resulting detection by the model's confidence is performed. When it was trained on PASCAL VOC 2007 dataset, produced a 66.4% mean average precision.

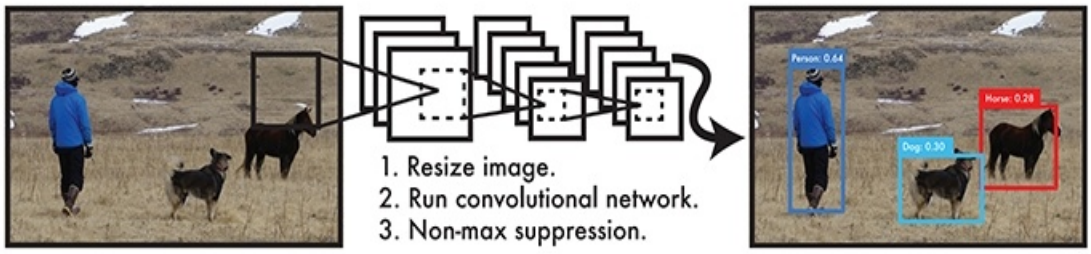

Figure 23. YOLO

5.3 Face Recognition

Face recognition is the process of identifying an individual from an image of their face. The different methods which are used in face recognition are discussed below.

DeepFace [?] takes the image of a face as input and performs Frontalization i.e. the variations in the images are removed so that every face would like they are looking straight at the camera. 2D alignment is used to normalize variations which are not out-of-plane and 3D alignment is used to normalize variations which are out-of-plane. This is then given to the CNN as input. This model was trained on Labeled Faces in the Wild dataset and produced a 97.3% accuracy in recognizing facial images. Figure 24 [?] shows the architecture of DeepFace.

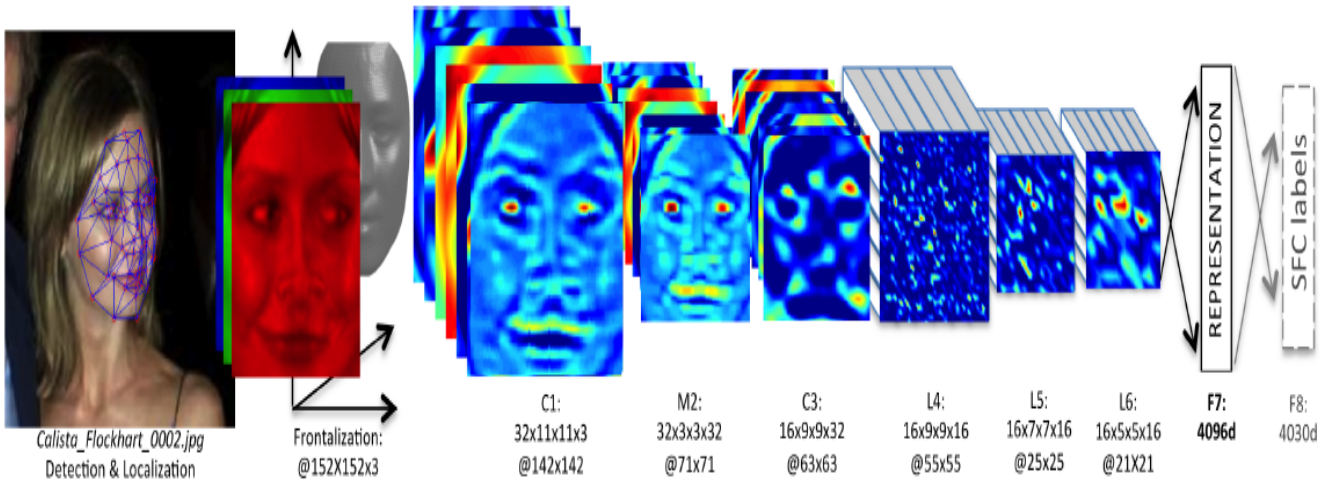

Figure 24. DeepFace

FaceNet [?] uses triplet loss-based function along with ZFNet and GoogLeNet to minimize the distance between two images of the same person and maximize the distance between the two images of different persons. This model was trained on Labeled Faces in the Wild dataset and produced a 99.63% accuracy in recognizing facial images. Figure 25 [?] shows the output distances of FaceNet between pairs of faces of the same person and different people.

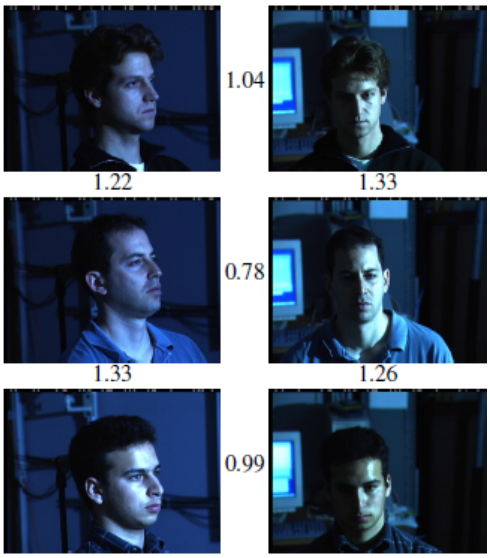

Figure 25. FaceNet

OpenFace [?] uses a combination of Histogram of Oriented Gradient) and Support Vector Machine or OpenCV's Haar cascade classifier to detect faces, then the obtained faces are preprocessed before giving it to the neural network for classification. This model was trained on Labeled Faces in the Wild dataset and produced a 92.9% accuracy in recognizing facial images. Figure 26 [?] shows the affine transformation in OpenFace where the transformation is based on the large blue landmarks and the final image is cropped at the boundaries and then resized to size 96X96 pixels.

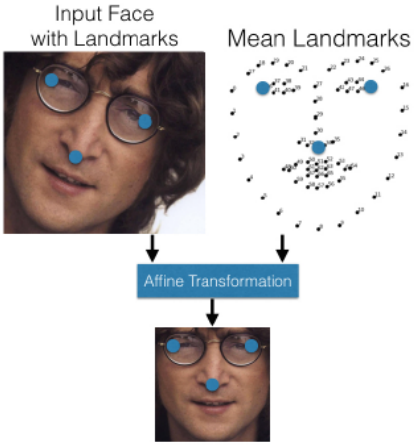

Figure 26. OpenFace

6 Conclusion

The different models in Deep learning have been explored in the paper, their advantages and disadvantages analyzed and their various applications in different fields discussed. We have found that CNN is the most commonly used neural network in the field of computer vision.

The future scope of Deep Learning is discussed below:

- Unsupervised deep learning since both animals and humans both learn by observation.
- Systems which are trained end-to-end and which combines CNN and RNN's that learn based on reinforcement learning.
- Systems which combine deep learning and reinforcement learning.
- Defining optimal method to select the model type and structure of the neural network for any given task.
- Generalizing neural network models and using a single model to work for different applications.
- Systems which consists of both representational learning and complex reasoning.
- Combining different neural networks which perform separate tasks to communicate with each other, so the information learnt from one model can be used as input to another model.