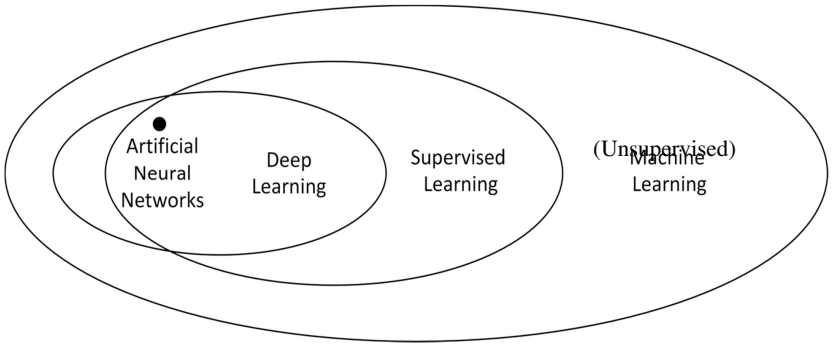
USC EE 542, Lectures 19, 20
Oct.30, Nov.1, 2017

Prof. Kai Hwang,

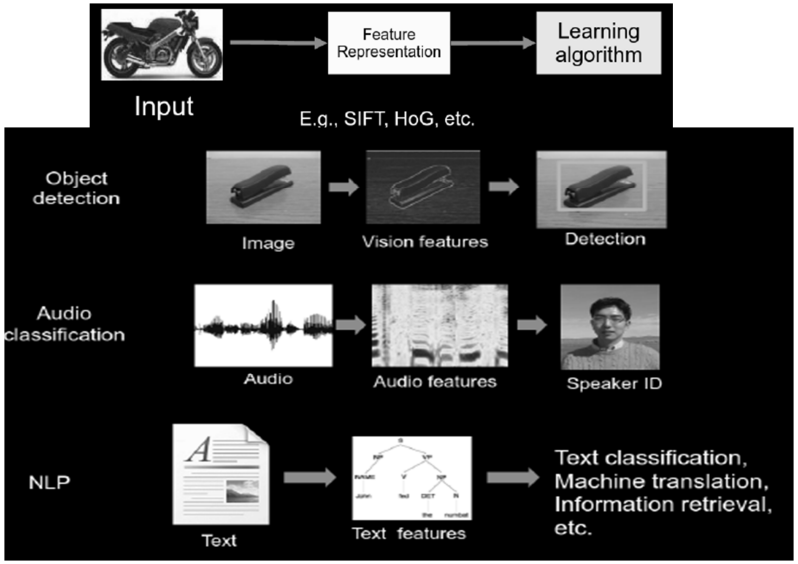
Artificial Intelligence (AI)
and Smart Machines,

Artificial Neural Networks (ANN)
Deep Learning Networks :
Convolutional Neural Networks (CNN)

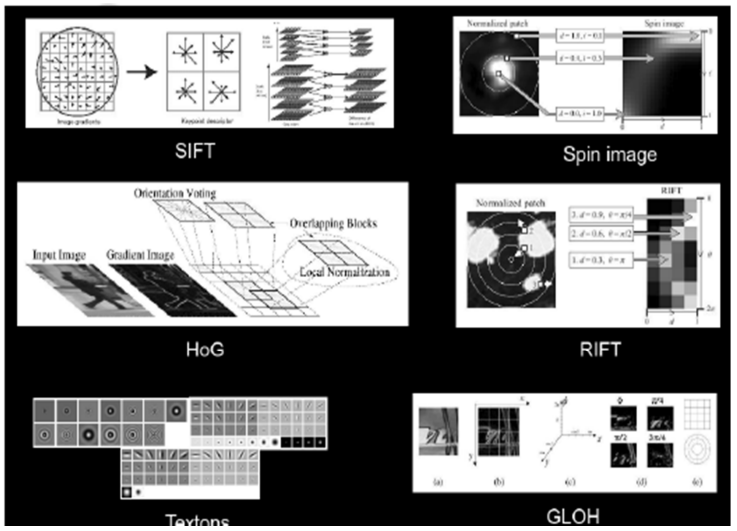
Evolution of Deep Learning from ML and The ANN Approach



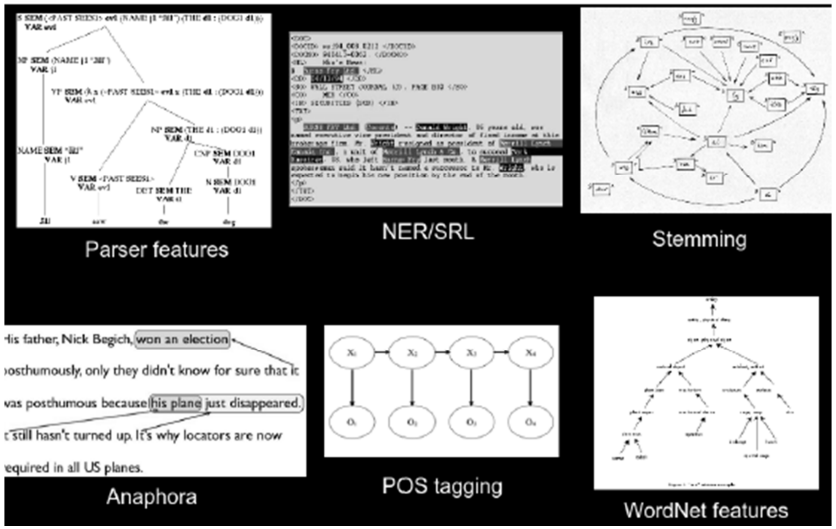
What is Computer Perception ?



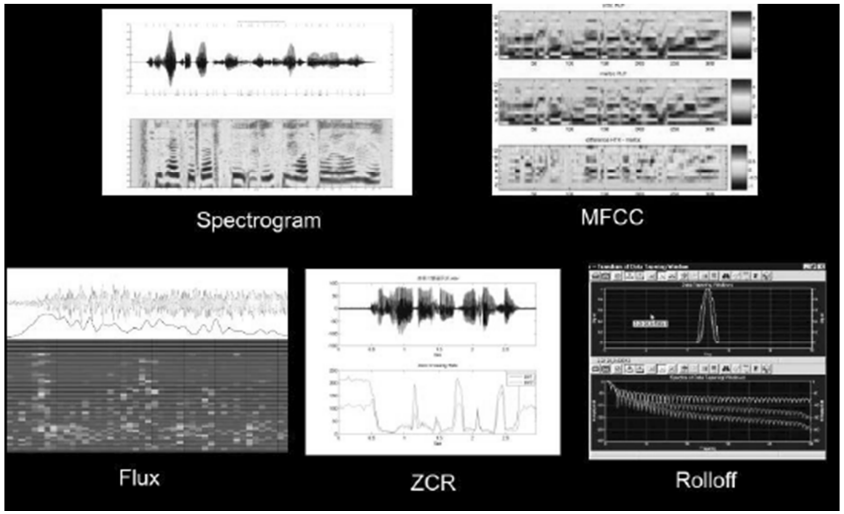
Computer Vision Features



Audio Recognition Features



Nature Language Processing (NLP) Features



Cognitive Computing, AR, VR and MR by IBM, Microsoft, Apple, Facebook, HTC, etc.

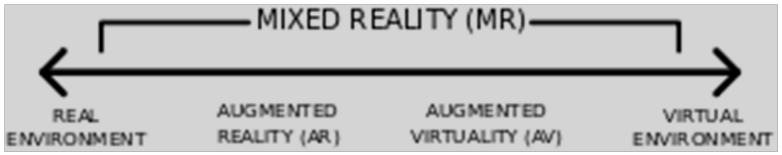
- Cognitive and Neuromorphic Computing
- *Big-Data Analytics* by Hwang/Chen, Wiley 2017
- IBM Watson Project for Health-Care, Cognitive and Smart Clouds
- The AE, VR, MR Spectrum and Commercial Devices by Facebook, Apple, Microsoft, etc.

Reality vs. Virtuality

- The world of events can be described as *reality* or *virtuality* (logical) based on its existence in real world or in the cyber space.
- We characterize them as *real* versus *virtual* events, respectively. These events can be subdivided as *pure*, *augmented, medicated,* or *severly medicated* in ascending degree of virtuality from the extreme end of reality.
- The augmented environments are created by computer images, artificially visual effects, or animated events.
- The mediated events or environments are created out of illusions and special mental conditions.
- The whole space is simply referred to as a spectrum of reality and virtuality as demonstrated in Fig.7.6.

The Spectrum from Real Environment to Virtual Environment

Concepts of AR , AV , VR, and MR

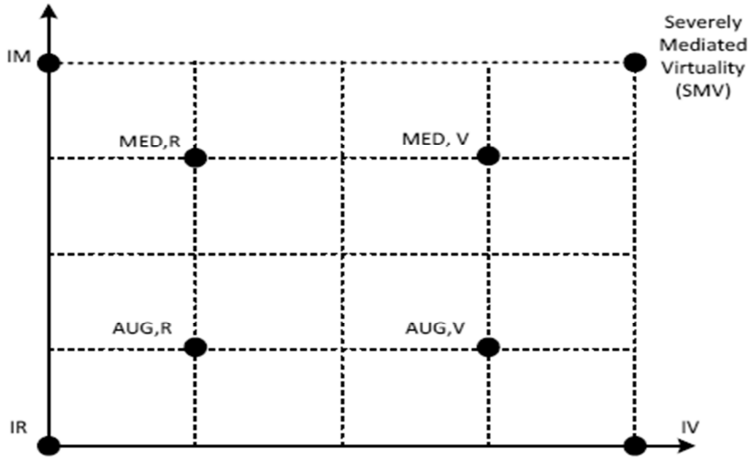


- **VR artificially creates sensory experience, which can include sight, touch, hearing, and smell. The VR can differ significantly from true reality, such as in virtual games.**
- **The immersive environment can be similar to the real world. Good examples include the creation of a lifelike experience, the simulations for pilot or combat training.**

AR, VR and MR at Major IT Companies including Microsoft, Apple, Facebook, HTC, etc

- **The *augmented reality* (AR) and *virural reality* (VR) and *mixed reality* (MR) are all related to each other.**
- **AR is augmented from the real world environment where some real sensor-collected signals or data are involved.**
- **VR is a computer technology that creates a visual environment, some real and some imagined, within which the users can experience a physical presence and interactions.**
- ***Mixed reality* (MR) is more a general term that involves both real and virtual environments in the entire spectrum**

Concepts of Reality, Augmentation, Medication and Virtuality



- **Here we consider a 2-dimensional spectrum of 8 cases of the reality and virtuality, represented by 8 dark dots in the spectrum space.**
- **The x-axis displays from *pure reality* (IR) to *pure virtuality* (IV). The y-axis shows variations from pure to augmented and mediated experience environments.**
- **The 4 interior dots have variable degrees of medication and mediation levels, known also as mixed reality**
- **At the augmentation level, we have the *augmented rea*lity (AR) and *augmented virtuality* (AV).**

AR, VR and MR at Major IT Companies including Sony, Microsoft, Apple, Facebook, HTC, etc

Table 7.5: Some Commercial AR/VR/MR Products

Company	Product	Introduction
Microsoft	HoloLens	A pair of mixed reality head-mounted smart glasses by Microsoft. HoloLens gained popularity for being the first computer running the Windows Holographic platform.
Google	Google Cardboard	This is a VR platform by Google for use with a head mount for a smartphone. Named for its fold-out cardboard viewer, It is a low-cost system to encourage VR applications.
Facebook	Oculus Rift	Oculus Riftis a virtual reality headset developed and manufactured by Oculus VR, released on March 28, 2016.
Samsung	Gear VR	The Samsung Gear VR is a mobile virtual reality headset developed by Samsung Electronics, in collaboration with Oculus, and manufactured by Samsung.
Sony	PlayStation VR	Known by the codename Project Morpheus during development, is a VR gaming head-mounted display developed by Sony Interactive Entertainment and manufactured by Sony.
HTC	HTC VIVE	This is a virtual reality headset developed by HTC and Valve Corporation in 2016. This is designed to utilize "room scale" technology to turn a room into 3D space via sensors
Huawei	Huawei VR	Huawei honor VR is released on May 10, 2016 to match honor V8 smart phone.
Alibaba	Buy+ Plan	Buy+ program uses VR technology to generate interactive three-dimensional shopping environment with computer graphics systems and auxiliary sensors.

AR/VR in Video Games:

- The use of graphics, sound and input technology in video games can be incorporated into VR.
- Several Virtual Reality *head mounted displays* (HMD) were released for gaming, including the Virtual Boy developed by Nintendo and iGlasses developed by Virtual I-O.
- Several companies are working on a new generation of VR headsets: Oculus Rift is a head-mounted display for gaming purposes, which was acquired by Facebook in 2014.
- Sony has PlayStation VR (codenamed Morpheus). Valve Corporation announced their partnership with HTC VIVE to use a VR headset to track the exact position of its user

VR in Education and Training:

- Strides are being made in the realm of education, although much needs to be done. The possibilities of VR and education are endless and bring many advantages to pupils of all ages.
- Few are creating content that have be used for educational purposes, with most advances being made in the entertainment industry, but many understand and realize the importance of education and VR.
- Navy personnel using a VR parachute training simulator. The usage of VR in a training perspective is to allow professionals to conduct training in a virtual environment.

Reading Assignments in Chapter 7: AI Machines and Deep Learning

Sec. 7.1 ~ 7.1.2: AI and Smart Machines

Sec.7.1.3 : Neural Computing Chips (lecture 9)

Sec. 7.2.1 ~ 7.2.2 : AR and VR (Lecture 9)

Sec. 7.3.1 ~ 7.3.4 : Artificial Neural Networks

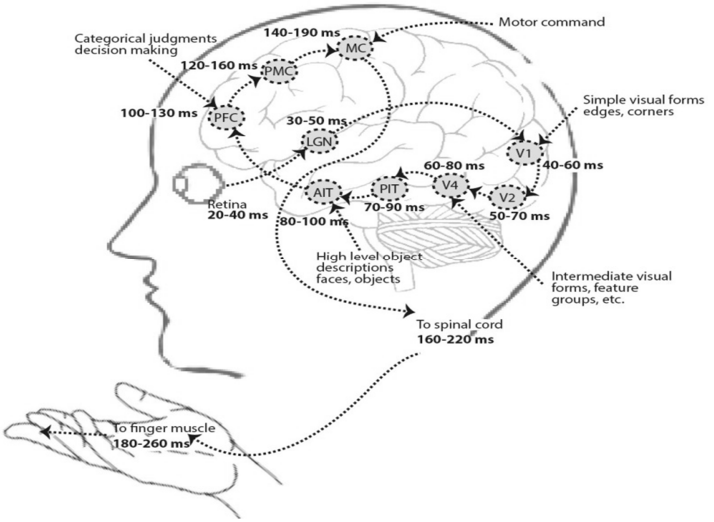
Sec. 7.4.1 ~ 7.4.3 : Convolutional Neural Nets

Sec. 7.4.4 : RNNs and DLNs (Lecture 9)

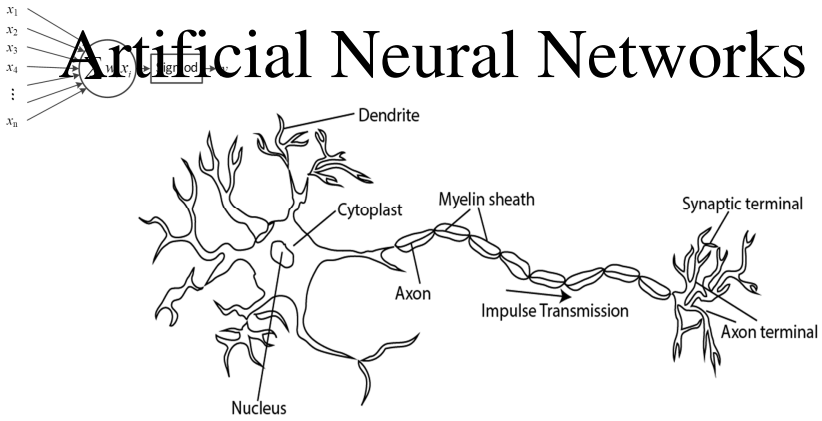
Skip Sec. 7.5 completely

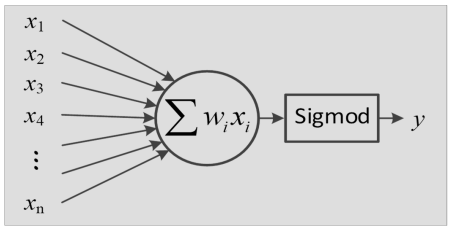
HW #2 : Problems 5.1, 7. 3, 7.8, and 7.9

Biological Neural Networks



Artificial Neural Networks





The Perceptron Model

Artificial Neural Networks (ANN) (1)

NEURAL NETWORKS

A first perspective on the origin of neural networks states that they are mathematical representations inspired by the functioning of the human brain. Another more realistic perspective sees neural networks as generalizations of existing statistical models. Let's take logistic regression as an example:

$$P(Y=1|X_1,\ldots,X_N)=\frac{1}{1+e^{-(\beta_0+\beta_1X_1+\quad+\beta_NX_N)}},$$

This model can be seen in Figure 3.11.

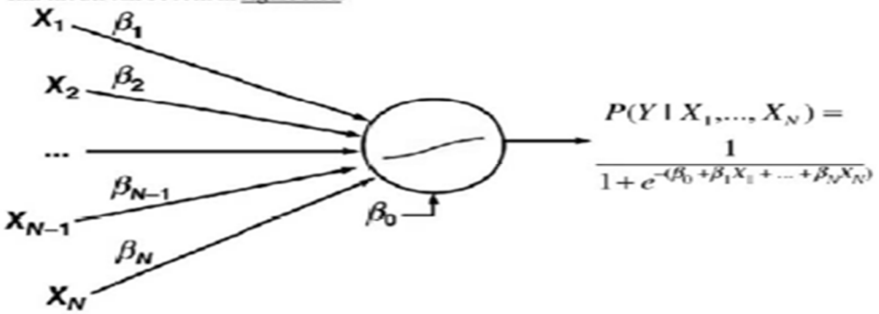


Figure 3.11 Neural Network Representation of Logistic Regression

Artificial Neural Networks (ANN) (2)

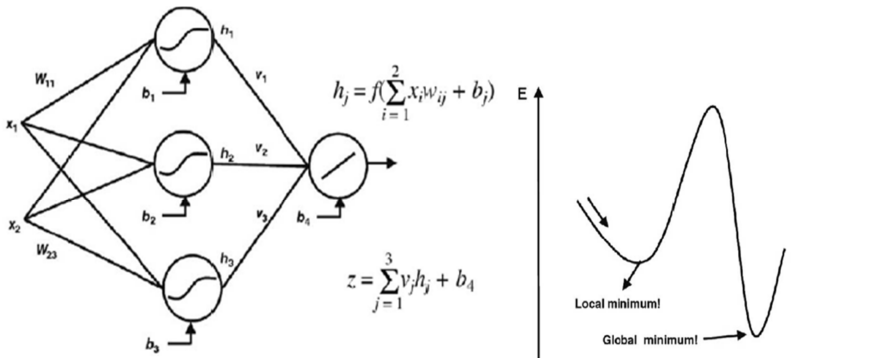


Figure 3.12 A Multilayer Perceptron (MLP) Neural Network

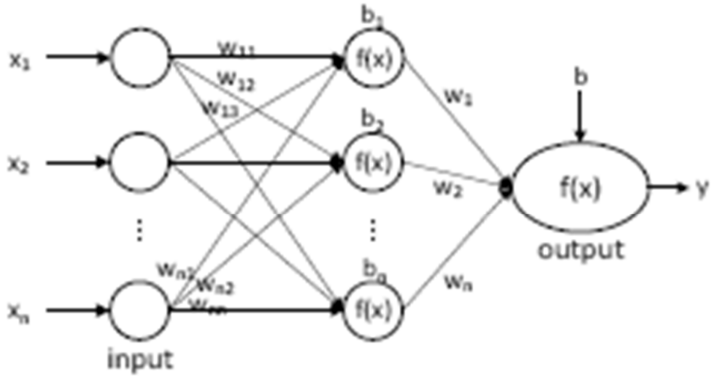
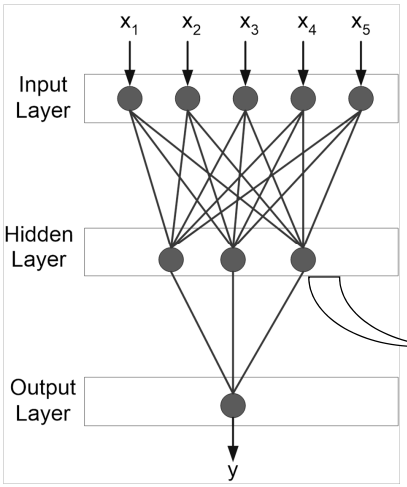
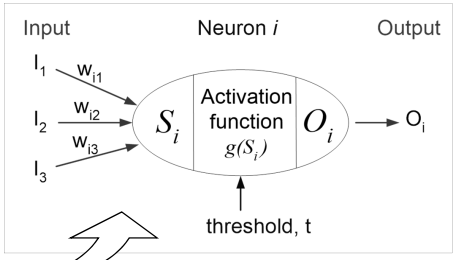


Figure 7.20 : Structure of a two-layer artificial neural network

General Structure of ANN (5)





Training ANN means learning the weights of the neurons

Algorithm for learning ANN (6)

- **Initialize the weights (w_0, w_1, …, w_k)**
- **Adjust the weights in such a way that the output of ANN is consistent with class labels of training examples**
  - **Objective function:**

$$E = \sum_i \left[Y_i - f(w_i, X_i) \right]^2$$

  - **Find the weights w_i's that minimize the above objective function**
    - **e.g., back-propagation algorithm**

7.8: Given in Figure 7.30, an ANN with linearly activated neurons, i.e., the output y, is a weighted summation of its input signals $y = \sum_{i=1}^{n} w_i x_i$, where w_i is the weight of input signal x_i to that neuron. The numbers at the input arrows at neurons n1 and n2 are the input signals. The numbers at the edges are weights applied on that edge. For example, the edge between n1 and n3 has a weight $w_{13} = -1$. Assume the weights on two input edges are equal to 1.

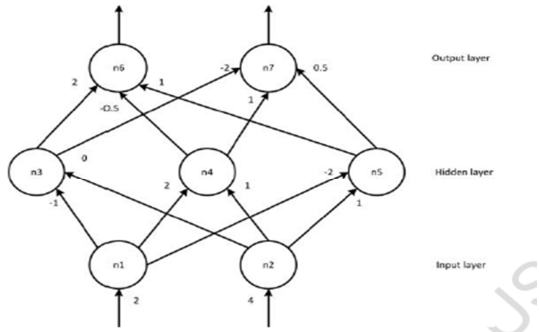


Figure 7.30
An ANN for Problem 7.8.

ANN Types and Functional Classes

- **We classify ANNs into two major types: static versus dynamic ANNs. Some ANNs are static in nature which do not change much with environment. For example, the perceptron and neocognitron are statically designed for fixed purpose.**

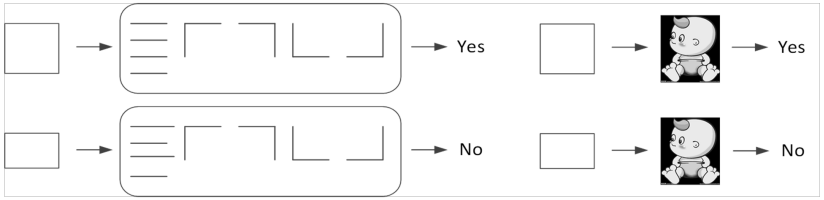
- **Some ANNs are dynamically structured or known as adaptive systems, For example, ANNs used for modeling populations and environments are changing constantly. Dynamic neural networks deal with not only nonlinear multivariate behavior, but also time-dependent behavior.**

ANN Types and Functional Classes

ANN Types	Reported ANNs (Check Wikipedia to dig out details of each Type)
Static ANNs	Neocognitron, McCulloch-Pitts cell, Radial basis function network RBF, Learning vector quantization, Perceptron (Adaline model, convolutional neural networks), Modular neural networks, Associative neural network
Dynamic ANNs,	Feedforward neural network FFN, Recurrent Neural Networks RNNs (Hopfield network, Boltzmann machine, Simple recurrent networks, Echo state network, Long short term memory network, Bi-directional RNN, Hierarchical RNN, Stochastic neural networks), Kohonen Self-Organizing Maps, Autoencoder, probabilistic neural network PNN, Time delay neural network TDNN, Regulatory feedback network RFNN

What is Deep Learning and What are the available Deep Learning Neural Networks ?

How can deep learning complete with human on self-learning through education? If we want to judge whether a quadrangle is square or not, the rationale approach is to seek features of square, such as same length for four sides and four 90 degree corner angles. This requires comprehension of the concept of right angle and the length of the side view as demonstrated below:



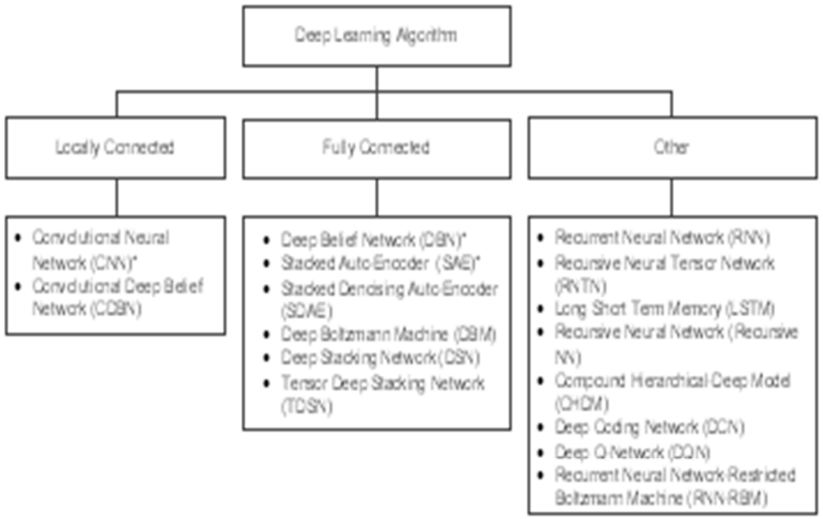


Figure 7.17 : A Taxonomy of various deep learning neural network models. (Reprint with permission from K. Hwang and M. Che, Big Data Analytics, Wiley, 2017)

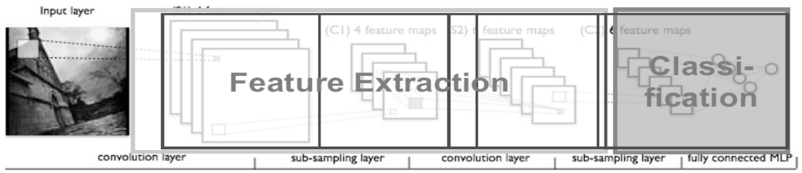
- *Fully connected networks:* In the traditional neutral network, the connections between layers from input layer, hidden layer to output layer are fully connected. One neutron in the previous layer connects with every neurons in the next layer. Deep learning architecture like Deep belief networks (DBN), Deep Boltzmann Machines (DBM), *Stacked Auto-Encoders* (SAE), *Stacked Denoising Auto-Encoders* (SDAE), *Deep Stacking Networks* (DSN), *Tensor Deep Stacking Networks* (TDSN), are fully connected.
- *Locally connected Networks:* Locally connected deep learning architecture means the connection mode between input layer and output layer is locally connected. This kind of deep learning architecture takes *convolutional neural network* (CNN) as the representative class. It use the concept of partial connection and weight sharing of convolutional operation to describe the overall by local features. Thus, it reduces the number of weight greatly. *Convolutional Deep Belief Networks* (CDBN) are also locally connected.
- *Other Neural Networks* : This class includes recurrent neural network (RNN), Recursive neural tensor network (RNTN), Long short term memory (LSTM), etc. Other related networks include the *Recurrent Neural network-Restricted Boltzmann Machine* (RNN-RBM), *Deep Q-Network* (DQN), *Compound Hierarchical-Deep Models* (CHDM), *Deep coding network* (DPCN), etc. Conventional ANNs, with locally connected or fully connected, may have limited applicability, because they perform badly or become powerless when dealing with data streams.

What is Convolutional NN ?

CNN - multi-layer NN architecture

- Convolutional + Non-Linear Layer
- Sub-sampling Layer
- Convolutional +Non-L inear Layer
- Fully connected layers

- Supervised



Essentially neural networks that use convolution in place of general matrix multiplication in at least one of their layers.

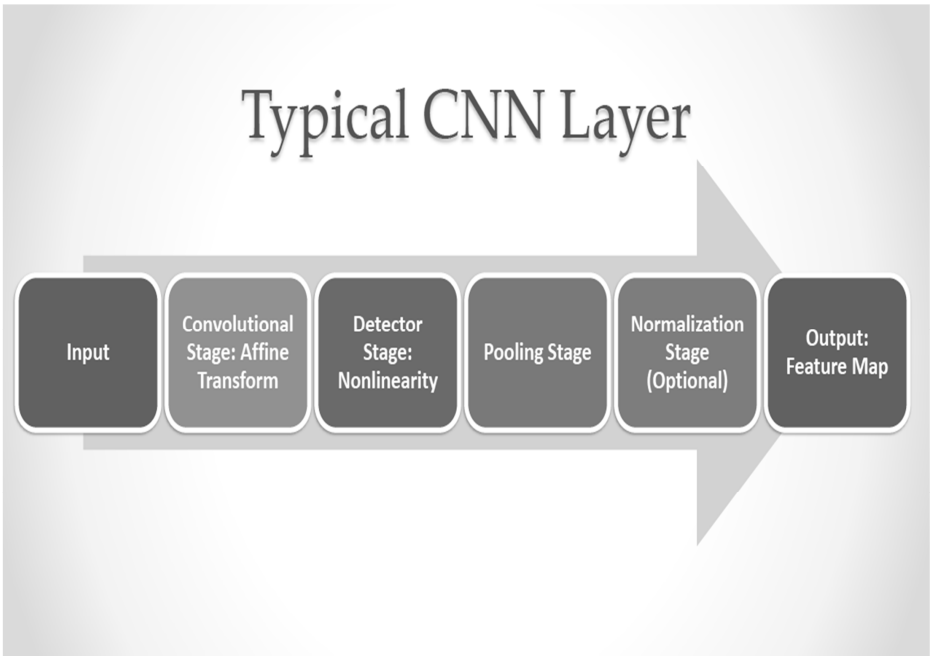
Basic Concept of Convolution and Polling in CNN Operations

- **Consider an image of 500x500 pixels, the no. of neurons in input layer is set to 500x500. With 10^8 neurons in hidden layer. Each connection between one input neuron and one neuron in hidden layer is set with a weight parameter.**
- **The no.of weight parameters between input layer and hidden layer is $500x500x10^8 = 25x10^{12}$. This may demand enormous amount of computations.**
- **Design a 10x10 filter to extract the local features of an image. This operation imitate human eyes to feel local image region. Then a hidden layer neuron is connected to a 10x10 area of the image through the filter.**

- **Hidden layer has 10^8 hidden neurons so there are 10^8 filters used between hidden layer and input layer. Thus the no. connection weights between them becomes $10x10x10^8$,= 10^{10} . Thus the feature learned from a filter can be used by many hidden neurons.**

- **By sharing weight, we reduce from $25x10^{12}$ computations to 100. This concept of local weight sharing make the convolution operation simpler and faster.**

- **The filter of 10x10 is called a convolution kernel. When we need to represent more local features of an image, we can use multiple convolution kernels.**



Convolutional neural network (CNN)

is a kind of feed forward neural network, which uses convolution to reduces the weights and thus reduce the complexity of calculation. CNN belongs to supervised learning method. And it is widely applied in voice recognition and image understanding field. CNN is composed of many layers.

Generally, it includes such basic components: input layer, convolution layer, pooling layer, fully connected layer, output layer. We need decide how many convolution layers and pooling layers should be utilized and what kind of classifier should be selected according to each specific application problem.

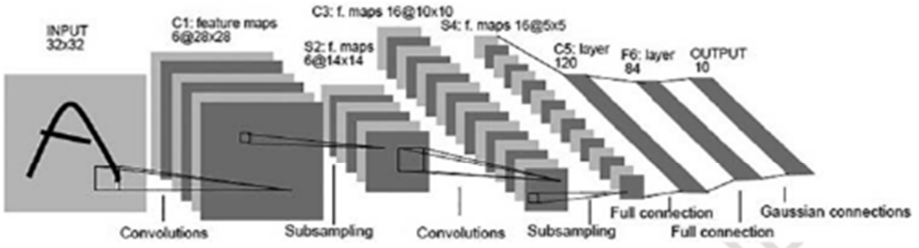
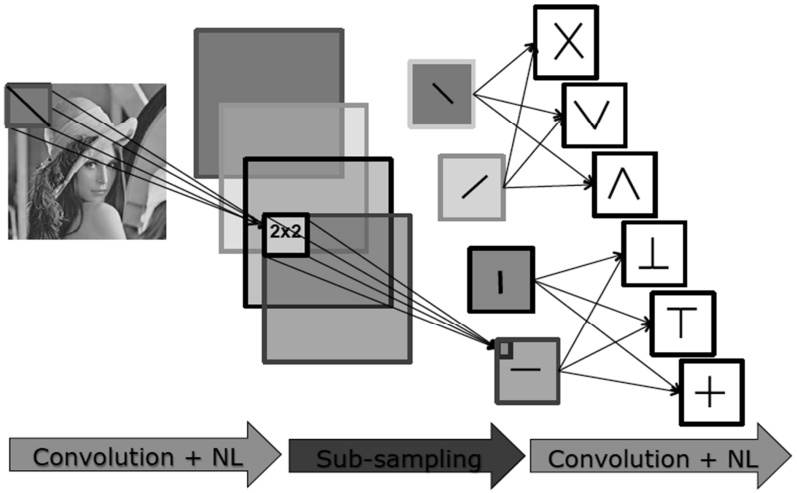


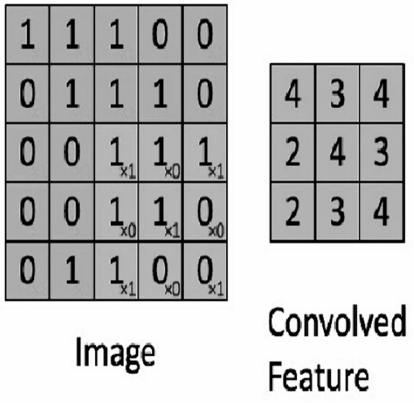
Figure 7.17
Convolutional neural network utilized by LeNet-5. Courtesy of Yann LeCun et al., "Gradient-based Learning Applied to Document Recognition, *Proceedings of the IEEE* 86, no. 11 (1998).

What is Convolutional NN ?



Kai Hwang, July 26, 2017

2-D Convolution in Action!



Pooling reduces computational complexity and thus shortens image recognition time: One can calculate the mean value (or maximum value) in an area of image. The statistical features obtained by such aggregation can reduce the dimensionality and thus improve the results. The operation of such a data merging is called pooling.

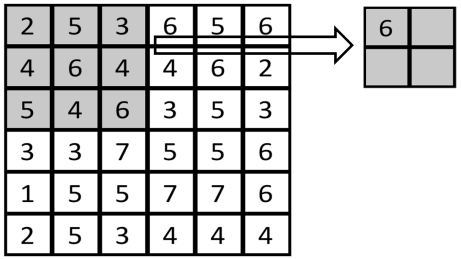


Figure 7.13: The concept of pooling from the 6 x 6 grid to a 2 x 2 grid

Example 7.2 : Convolution and pooling for Convolutional Neural Networks

CNN has been widely used in digital image processing with the rapid development of CNN. For example, using this DeepID convolutional neural network, the recognition rate of the human face can reach a maximum of 99.15% of the correct rate. This technique can play an important role in the search for missing people and the prevention of terrorist crime. Fig. 7.14 shows CNN used. If the given input image as shows in Fig. 7.15(a), the image size is 8x6. We adopt the size of convolution kernels is 3x3 and the size of one feature graph in convolutional layer 1 is ((8-3)+1)x((6-3)+1)=6x4. Assuming we use 3 filters, the corresponding weight matrices are:

$$w_1 = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 1 \end{bmatrix}, w_2 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, w_3 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

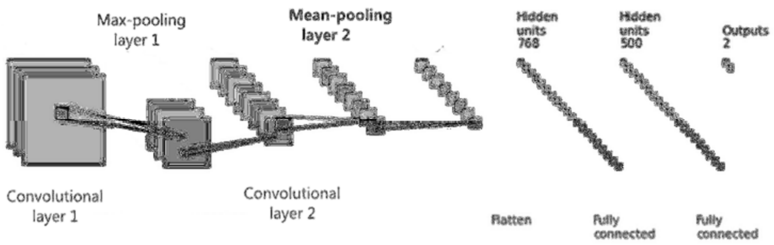


Figure 7.14 Schematic diagrams of Convolutional Neural Network

If the given input image as shows in Fig. 7.15(a), the image size is 8x6. We adopt the size of convolution kernels is 3x3 and the size of one feature graph in convolutional layer 1 is ((8-3)+1)x((6-3)+1)=6x4. Assuming we use 3 filters, the corresponding weight matrices are:

$$w_1 = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 1 \end{bmatrix}, w_2 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, w_3 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

4	3	1	6	8	0	0	0
0	0	0	5	3	2	1	0
0	0	0	4	5	7	9	1
0	0	0	0	0	0	0	0
1	2	6	2	1	6	3	1
0	0	0	0	0	0	1	1

(a) Input image

5	13	14	17	22	8
0	5	3	7	4	2
7	8	12	19	18	15
0	0	0	0	1	1

(b) After first convolution

-5	3	4	7	12	8
-10	-5	-7	-3	-6	-8
-3	-2	2	9	8	5
-10	-10	-10	-10	-9	-9

(c) After deviation

0	3	4	7	12	8
0	0	0	0	0	0
0	0	2	9	8	5
0	0	0	0	0	0

(d) Feature map at layer 1

0	0	3	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0

(e) After second convolution

0	0	3	0	0	0
0	0	0	0	0	0
0	0	1	0	0	0
0	0	0	0	0	0

(f) After another convolution

1			2			3		
3	7	12	0	3	0	0	3	0
0	9	8	0	0	0	0	1	0

(g) Feature may after maximum polling

Figure 7.15: Successive Convolutional and Pooling steps in building a CNN

Example 7.8 How a CNN Works with Pooling and Overlapped Processing

We can understand how to realize convolution through the convolution operation for an image of 8×8, as shown in Figure 7.21. The dimension of the convolution kernel is 4×4; and the feature matrix is:

$$w = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 \end{pmatrix}.$$

We extract an image x_1 of 4×4 from the image of 8×8 for the convolution operation with the feature matrix. Here, we obtain the value y_1 for the first neuron in the hidden layer by utilizing the equation $y_i = w \times x_i$. The step size of the convolution is set as 1. We continue to extract image x_2 of 4×4, and obtain the value y_2 for the second neuron through the convolution operation. We repeat the aforementioned steps until the traverse of the whole image is completed.

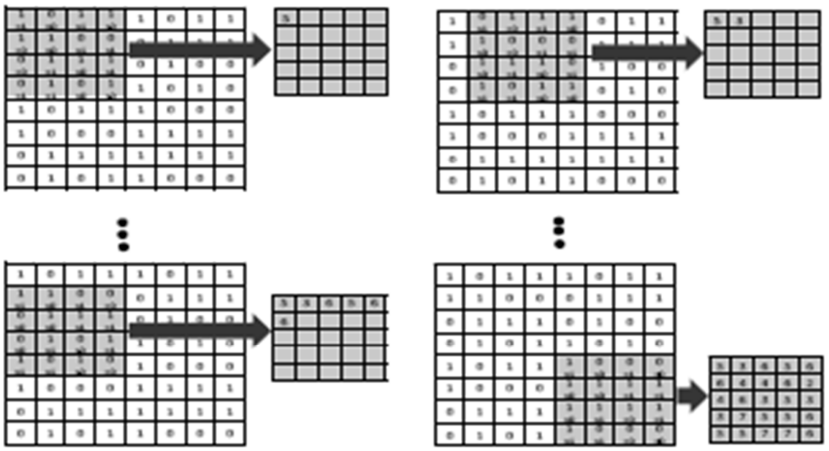


Figure 7.23: Schematic diagram for a convolutional ANN (CNN) (Source: http://ufldl.stanford.edu/wiki/index.php/UFLDL_Tutorial)

Variants

Full
- Add zero-padding to the image enough for every pixel to be visited k times in each direction, with output size: $(m + k - 1) \times (m + k - 1)$

Valid
- With no zero-padding, kernel is restricted to traverse only within the image, with output size: $(m - k + 1) \times (m - k + 1)$

Same
- Add zero-padding to the image to have the output of the same size as the image, i.e., m x m

Stride *s*
- Down-sampling the output of convolution by sampling only every s pixels in each direction.
- For instance, the output of 'valid' convolution with stride s results in an output of size $\frac{m-k+s}{s} \times \frac{m-k+s}{s}$

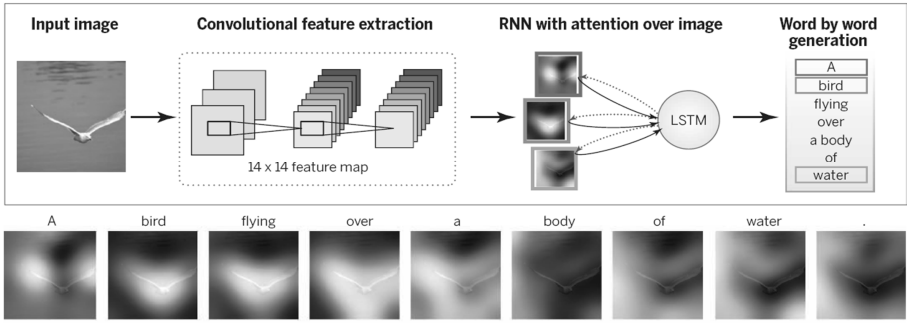
http://goodfeli.github.io/dlbook/contents/convnets.html

Some Reported Performance Results on the ImageNet Project

ImageNet for large-scale visual recognition is a benchmark in object classification and detection, with millions of images and hundreds of object classes. In the ILSVRC 2014, the winner was GoogLeNet from the DeepDream project. This team achieved a mean average precision of object detection of 0.439329 and a reduced classification error of 0.06656. Their CNNs applied more than 30 layers. Performance of CNN on the ImageNet tests is now close to that of humans. In 2015 a many layered CNN demonstrated the ability to spot faces from a wide range of angles, including upside down, even when partially occluded, with competitive performance.

Deep Learning Example

from an Image to a Verbal Description



Automatic generation of text captions for images with deep learning. A convolutional neural network is trained to interpret images, and its output is used by a recurrent neural network (RNN) trained to generate a text caption (top). The sequence at the bottom shows the word-by-word focus of the network on different parts of input image while it generates the caption word-by-word.

7.9: This problem requires you to use convolutional neural networks (CNNs) to perform image classification tasks. The image has an input layer with 5×5 resolution, and the network includes a convolution layer (C1), a max-pooling layer (P1), and a fully-connected layer. Input binary image as seen here:

$$input\_matrix = \begin{Bmatrix} 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \end{Bmatrix}.$$

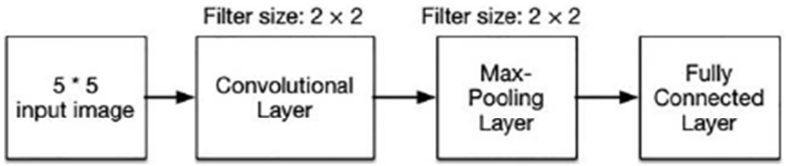


Figure 7.31
The convolutional neural network image for Problem 7.9.

Assume the convolution kernel size (also called the receptive field) of C1 is 2×2, the stride is 1, and the number of filters is 1 (see Figure 7.31). You are expected to answer the following problems:

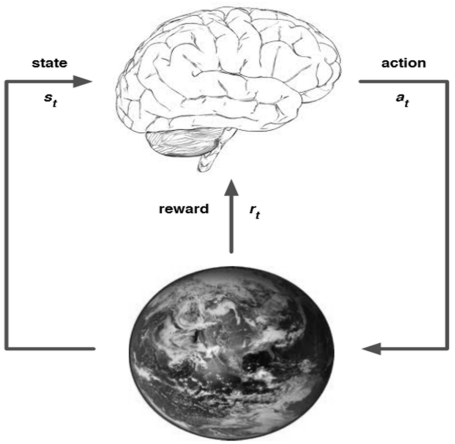
(a) Calculate the number of neurons in the convolutional layer.

(b) Compute the output feature map of this layer, where:

The weight matrix of the filter is $w = \begin{Bmatrix} 1 & 0 \\ 1 & 1 \end{Bmatrix}$.

(c) Define a max-pooling layer with a receptive field with a size of 2×2 matrix. Use a stride of 2 to ensure that there is no overlapping. Compute the output feature map of this layer.

The Concept of Reinforcement Learning



- At each step t the agent:
  - Receives state s_t
  - Receives scalar reward r_t
  - Executes action a_t
- The environment:
  - Receives action a_t
  - Emits state s_t
  - Emits scalar reward r_t

DeepMind AlphaGo Program

- AlphaGo program defeated the world Go Champaign in March 2016. The Go is the most complicated chase-board game played on a 19×19 grid by two players placing black and white stones on the board alternatively.
- There are 361 board points that can be placed. The whole game may end up with 10^{170} possible choices for the player to consider.
- The AlphaGo AI program was developed by DeepMind, an AI subsidiary under the Alphabet Company. This milestone achievement marked the era that machine intelligence can beat human players in some selected areas.

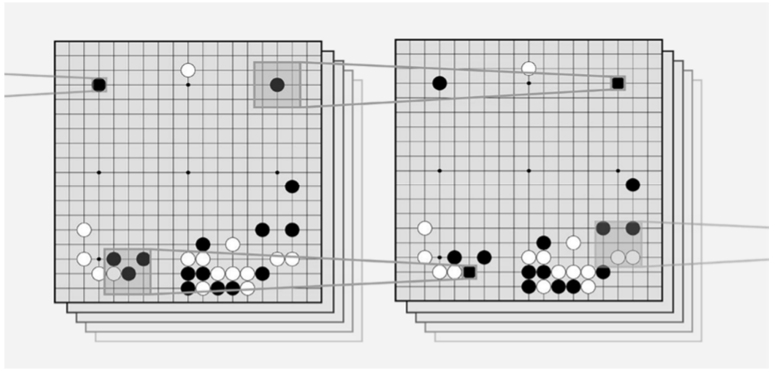
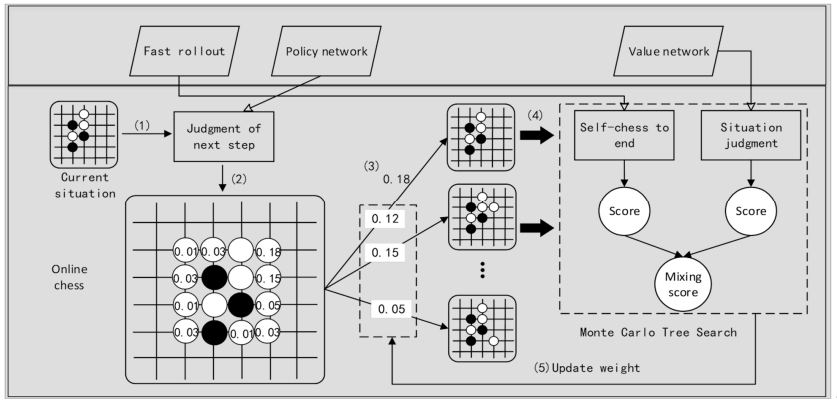


Figure 9.13: Convolutional neural network for processing the Go playing board.

DeepMind and Brain Projects at Google

- In 2016, Google AlphaGo program defeated the top Go player. This opened up the debate between human intelligence vs. machine intelligence.
- The Google Brain Team has used large CPU/GPU/TPU clusters to recognize 2000 classes of photo images trained from billions of YouTube images.



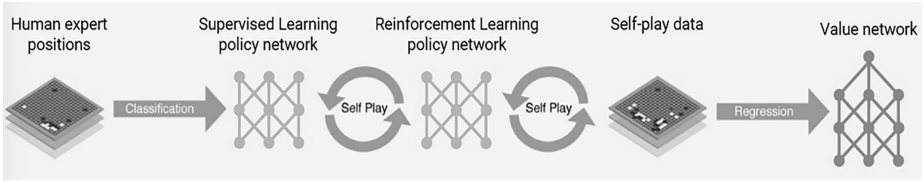


Figure 9.15 The self-play training pipeline between policy network and value networks by human experts.

Program	Accuracy
Human 6-dan	~ 52%
12-Layer ConvNet	55%
8-Layer ConvNet*	44%
Prior state-of-the-art	31-39%

Program	Winning rate
GnuGo	97%
MoGo (100k)	46%
Pachi (10k)	47%
Pachi (100k)	11%

Figure 9.17 Performance of different AlphaGo programs.

EE 542 Home Work #4 (6%)

for Chapters 9 and 10 Due date: Nov.15, 2017

Chapter 9: Prob. 9.3, Prob.9.4, Prob.9.5,
(Lectures 21 ~ 24)

Chapter 10: Prob.10.2, Prob.10.6, Prob.10.8,
(Lectures 14, 15, 23, 24)

Note: Some problems require Hadoop/Spark/Tensor programming effort on the AWS cloud. Some require investigated research reports. Do not wait until November, you cannot finish solving all 6 problems, which counts 6% of the course grade.