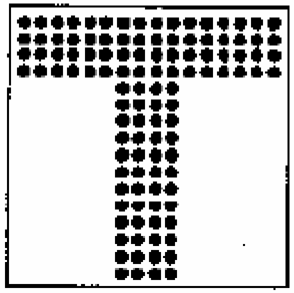
Hybrid Intelligent Systems

Lecture 5. Part 4

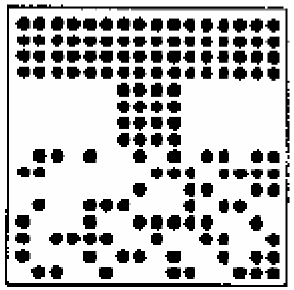
Hopfield associative memory

Tasks solved by associative memory:

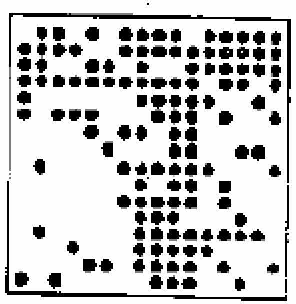
1) restoration of noisy image



Original 'T'



half of image corrupted by noise



20% corrupted by noise (whole image)

2) rememoring of associations

Input image

Image – result of association

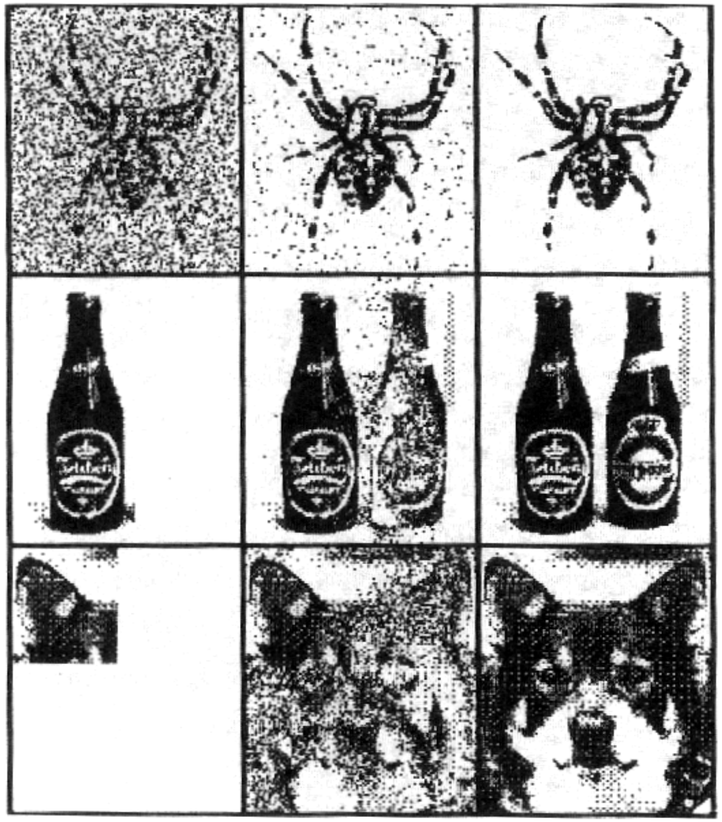
Hopfield model

Sub-type of recurrent neural nets

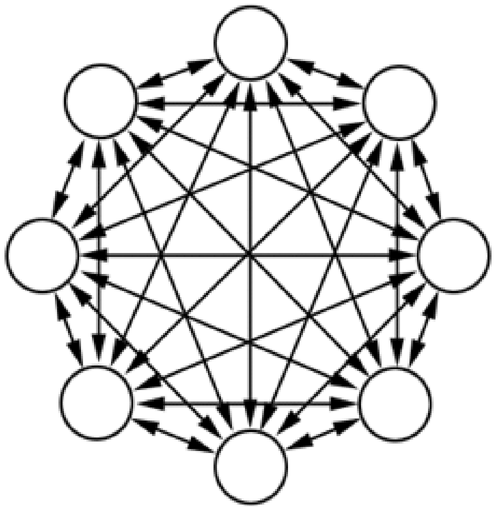
- Fully recurrent
- Weights are symmetric

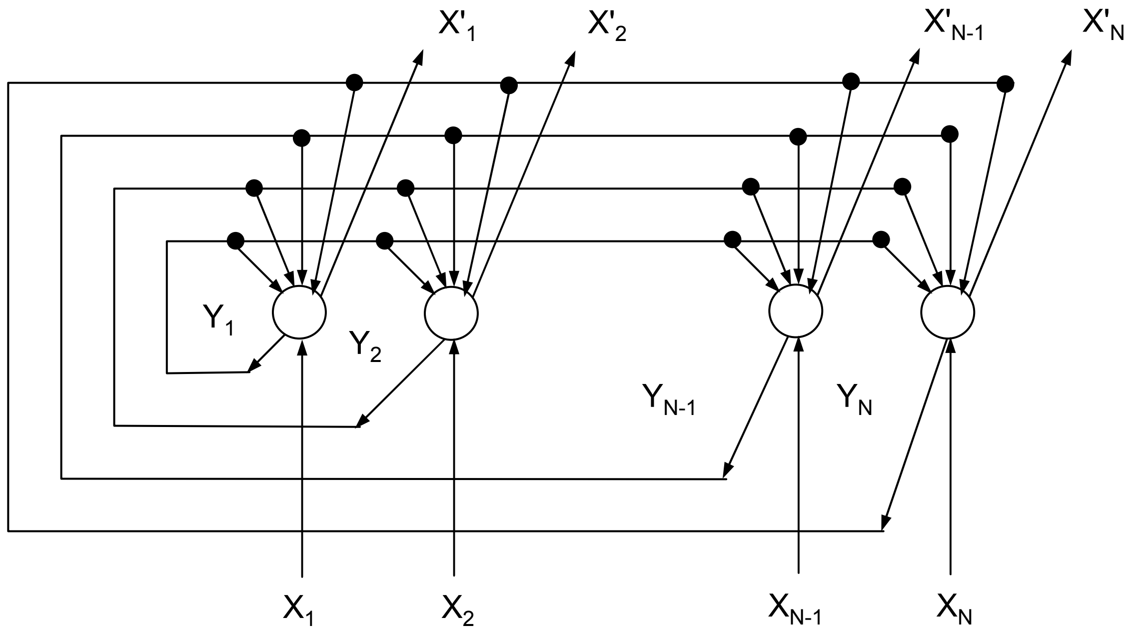
Learning: **Hebb rule** (cells that fire together wire together)
Can recall a memory, if presented with a corrupt or incomplete version

→ **auto-associative** or **content-addressable memory**



Hopfield Model (2)





Features of structure:

- Every neuron is connected with all others
- Connections are symmetric, i.e. for all *i* and *j* $w_{ij} - w_{ji}$
- Every neuron may be Input and output neuron
- Presentation of input is set of state of input neurons

Neurons in Hopfield Network

- The neurons are binary units
  - They are either active (1) or passive
  - Alternatively + or –
  - May be two variants of performance: (-1,1) or (0,1)
- The network contains *N* neurons
- The state of the network is described as a vector from 0 and 1 (or -1 and 1):

$$U = (u_1, u_2, \ldots, u_N) = (0,1,0,1,\ldots,0,0,1)$$

Updating the Hopfield Network (during recall)

- The state of the network changes at each time step. There are four updating modes:
  - Serial – Random:
    - The state of a randomly chosen single neuron will be updated at each time step
  - Serial-Sequential :
    - The state of a single neuron will be updated at each time step, in a fixed sequence
  - Parallel-Synchronous:
    - All the neurons will be updated at each time step synchronously
  - Parallel Asynchronous:
    - The neurons that are not in refractoriness will be updated at the same time

The updating Rule (1):

- Here we assume that updating is serial-Random
- Updating will be continued until a stable state is reached.
  - Each neuron receives a weighted sum of the inputs from other neurons:

$$h_j = \sum_{\substack{i=1 \\ i \neq j}}^{N} u_i . w_{j,i}$$

  - If the input h_j is positive the state of the neuron will be 1, otherwise -1:

$$u_j = \begin{cases} 1 & \text{if } h_j \geq 0 \\ -1 & \text{if } h_j < 0 \end{cases}$$

Convergence of the Hopfield Network

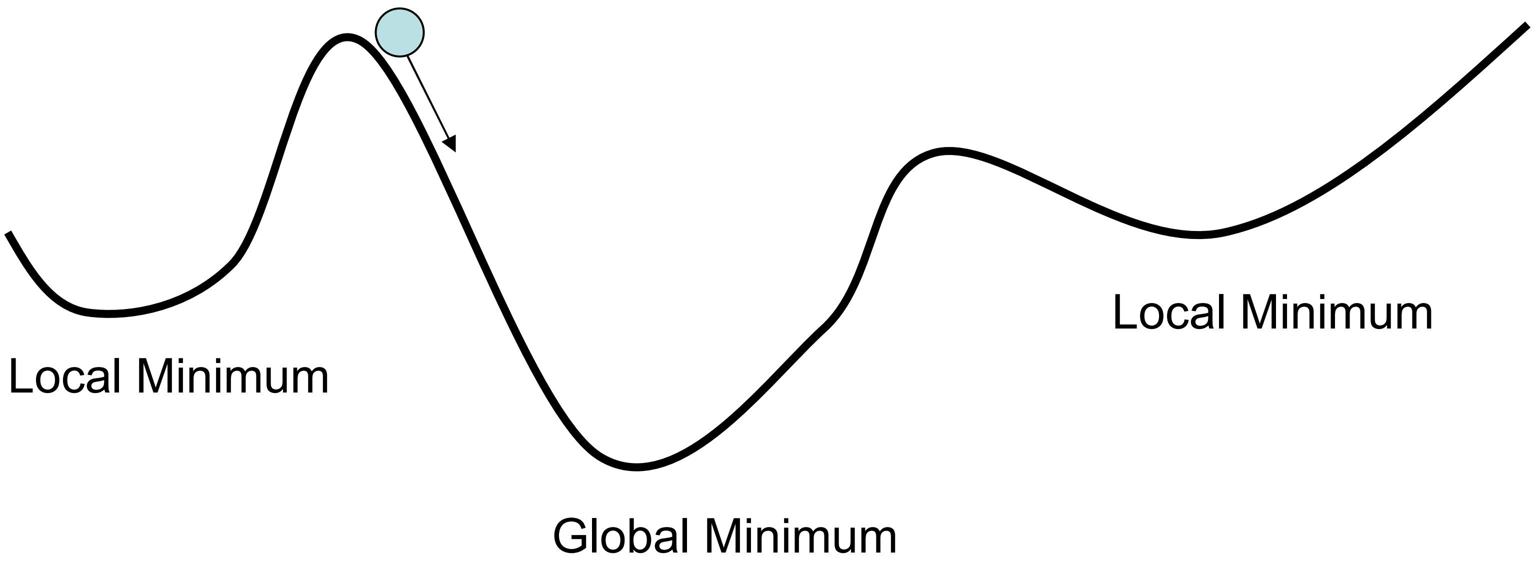
- Does the network eventually reach a stable state (convergence)?
- To evaluate this a ‘energy’ value will be associated to the network:

$$E = -\frac{1}{2}\sum_{j}\sum_{\substack{i=1 \\ i\neq j}}^{N} w_{j,i} u_i \, u_j$$

- The system will be converged if the energy is minimized

The Energy Function:

- The energy function is similar to a multidimensional (N) terrain



Associative memory based on Hopfield model

- Two processes
  - Learning
  - Testing (using, recalling)

Learning

- Each pattern can be denoted by a vector from -1 and 1:

$$S_p = (-1,1,-1,1,\ldots,-1,-1,1) = (s_1^p, s_2^p, s_3^p, \ldots s_N^p)$$

- If the number of patterns is m then:

$$w_{i,j} = \sum_{p=1}^{m} s_i^{\ p} s^{p}{}_{j}$$

  - May be calculated without presentation of examples
- Hebbian Learning:
  - The neurons that fire together , wire together
  - For Hopfield model: Weight of link increases for neurons which fire together (with same states) and decreases if otherwise

Recalling

- Iteration process of calculation of states of neurons until convergence will be achieved
- Input neurons may be freeze (can not change its state), if input pattern has not noise and may be changed otherwise
- To obtain right pattern (one from stored during learning) it is needed to present on inputs enough large vector and model must have enough large information capacity

Example of preparing of data for learning working (task – estimation of prize of flat). Length of vector (N) - 29

Destrict:

Name 1	000
Name 2	001
Name 3	010
Name 4	011
Name 5	100
Name 6	101

Type of flat

no	00
Panel	01
Large size	10
Monolith	11

Floor

1	0000
2	0001
3	0010
4	0011
5	0100
6	0101
7	0111
8	1000
9	1001
10	1010
11	1011
12	1100
13	1101
14	1110

Number of storeys:

1	0000
2	0001
3	0010
4	0011
5	0100
6	0101
7	0111
8	1000
9	1001
10	1010
11	1011
12	1100
13	1101
14	1110

Material:

panels	00
bricks	01
concrete	10

Square all

20-30	00
31-40	01
41-50	10
51-63	11

Square of rooms

10-15	000
16-20	001
21-25	010
26-30	011
31-35	100
36-40	101

Square of kitchen

4-6	00
7-8	01
9-10	10
11-12	11

Balcony

no	00
balcony	01
loggia	10
Balcony + loggia	11

Phone

yes	0
no	1

Prize

71-90	0000
91-110	0001
111-130	0010
131-150	0011
151-170	0100
171-190	0101
191-210	0110
211-230	0111
231-250	1000
251-270	1001
271-290	1010
291-310	1011
311-330	1100
331-350	1101
351-370	1110
371-390	1111

Limitations of Hopfield associative memory

- The evoked pattern is sometimes not necessarily the most similar pattern to the input because local minimums
- Some patterns will be recall more than others
- Spurious states: non-original patterns because symmetry of weight matrix
- Information capacity: $\leq 0.15\ N$

- One of method of fighting with local minimums of E – to introduce in model of random process of updating of weights, i.e. to append to Hopfield model of Boltzmann machine