large.

Sub :

Day 18/2/2046

Time : Date : / /

## Ans. to the Ques. No. 4

**Vanishing gradient problem:** In RNN to train the network one back-propagate though time and calculate the gradient at each step. Also, the weights beign added at each step. As, here one dealt with a time series, the network tends to forget the old information when the gradient value is small. As smaller gradient do not affect the weight updation, the system do not learn any thing. This is called vanishing gradient problem.

**How does LSTM solve the vanishing gradient problem:** From the cell structure we can see, LSTM has, input vector at time step $t$. And the network cell is denoted by $c_t$. the output here passed through time step $t$, $t+1$ is $h_t$. LSTM has 3 gates and forget gate controls what information in cell state to forget and give new information.

Sub :

Day

Time :  Date : / /

$$f_t = \sigma(W_f [x_t, h_{t-1}])$$

input gate control information into the cell state.

$$i_t = \sigma(W_i [x_t, h_{t-1}])$$

output gate,

$$o_t = \sigma(W_o [h_{t-1}, x_t])$$

$$h_t = o_t * \tanh(c_t)$$

cell state,

$$c_t = f_t * c_{t-1}$$

$$\tilde{c}_t = \tanh(W_c [x_t, h_{t-1}])$$

Here, the system back propagate with time, to preserve the memory and avoid vanishing gradient problem.