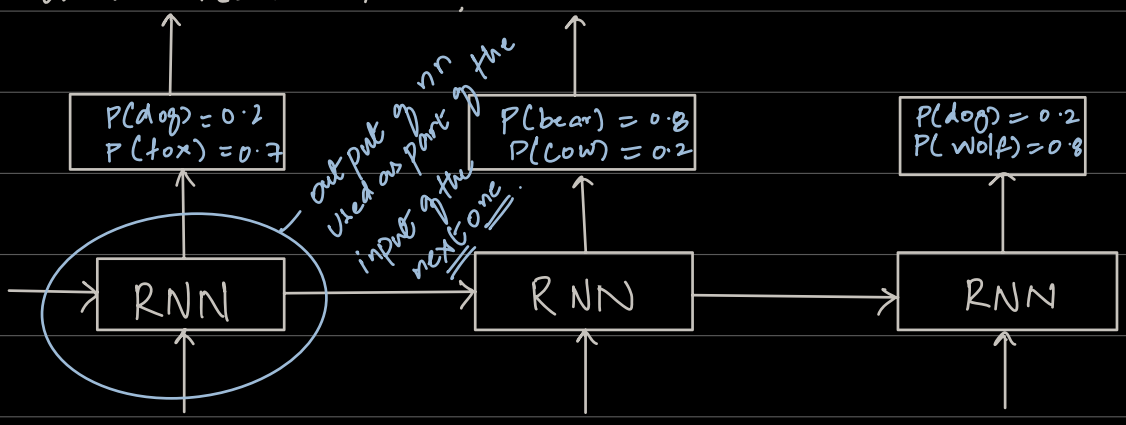
- LSTM (Long Short term memory networks) - widely used modification of RNNs that incorporates modification to the memory structure. First modification - inclusion of short term memory and second modification - inclusion of long term memory. Selecting relevant patterns for our network to learn.

σ - Sigmoid function } Activation
tanh - hyperbolic tangent } functions

⇒ Neural Network:

image ⟶ neural network ⟶ P(dog) = 0.8
⟶ P(wolf) = 0.15
⟶ P(fish) = 0.05

What if the image is actually wolf how does neural network learn this?

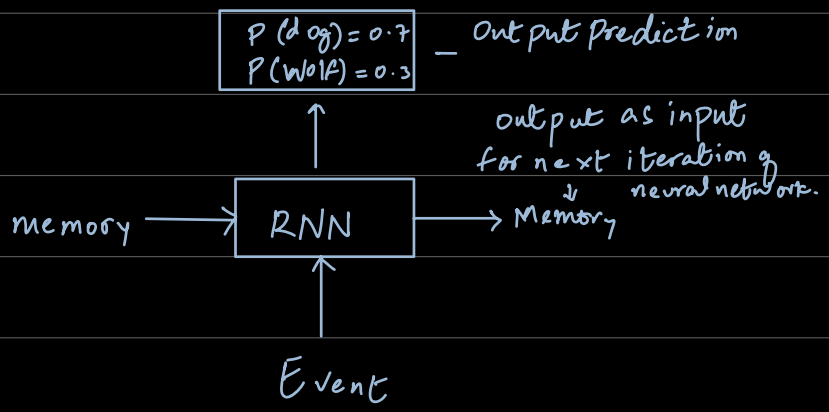


⇒ We want to use previous images output to hint our neural network that the image currently

processing is that of an Wolf.

⇒ Combining vectors which would then be squished through an activation fuction — which could be sigmoid or tanh.

⇒ Drawbacks: If bear appeared a while ago and recent images were of an tree, squirrel — but those images don't give much information. And information coming in gets repeatedly squished through activation and training a network using back propogation all the way leads to vanishing gradient problem. ∴ Bear information lost. RNN memory — short term memory.



⇒ Basics of LSTM :

Gates:

- Forget gate
- Learn gate
- Remember gate
- Use gate

Outputs

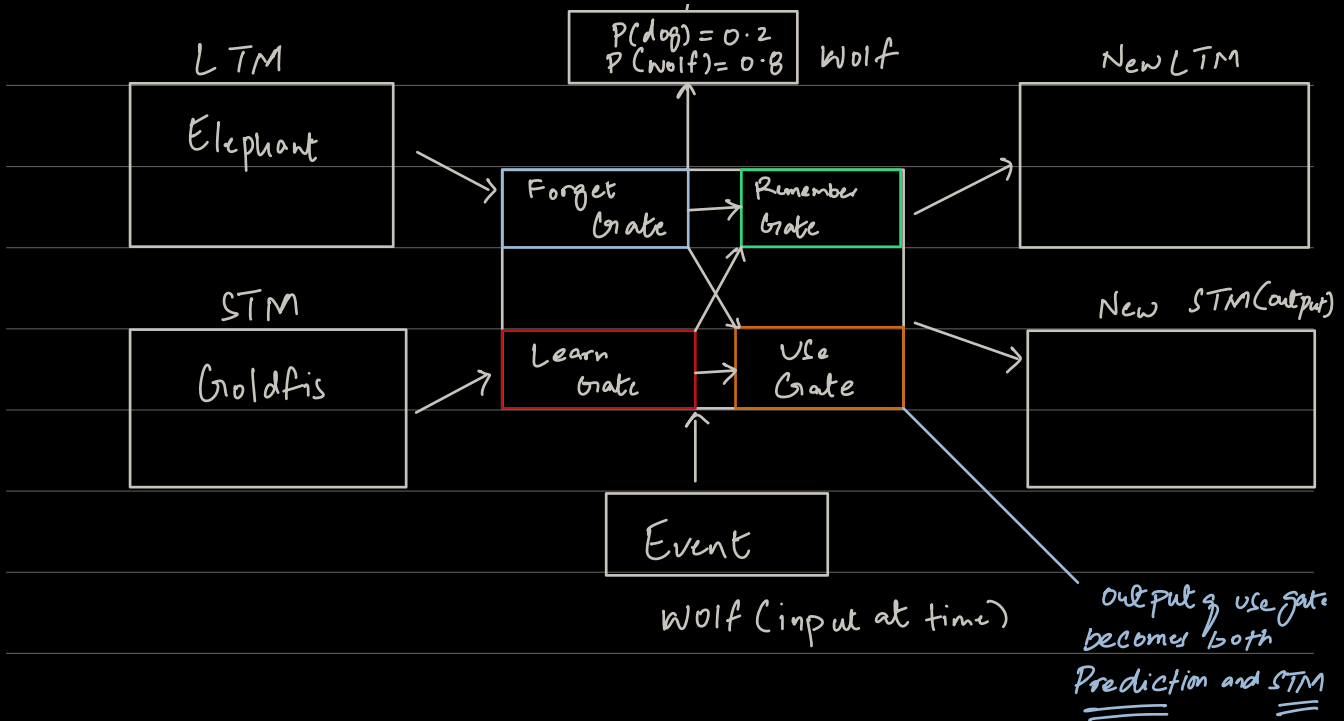
- New Long-term memory
- New Short-term memory

Inputs:

- Long term Memory
- Short term Memory
- Input Vector (Event)

- In updating Long term memory we add a bit a remove a bit to the long term memory.
- In updating Short term memory we remember some bit and forget some bit.

output



1) The Long term memory goes to the forget gate, where it forgets every thing that it doesn't consider useful.

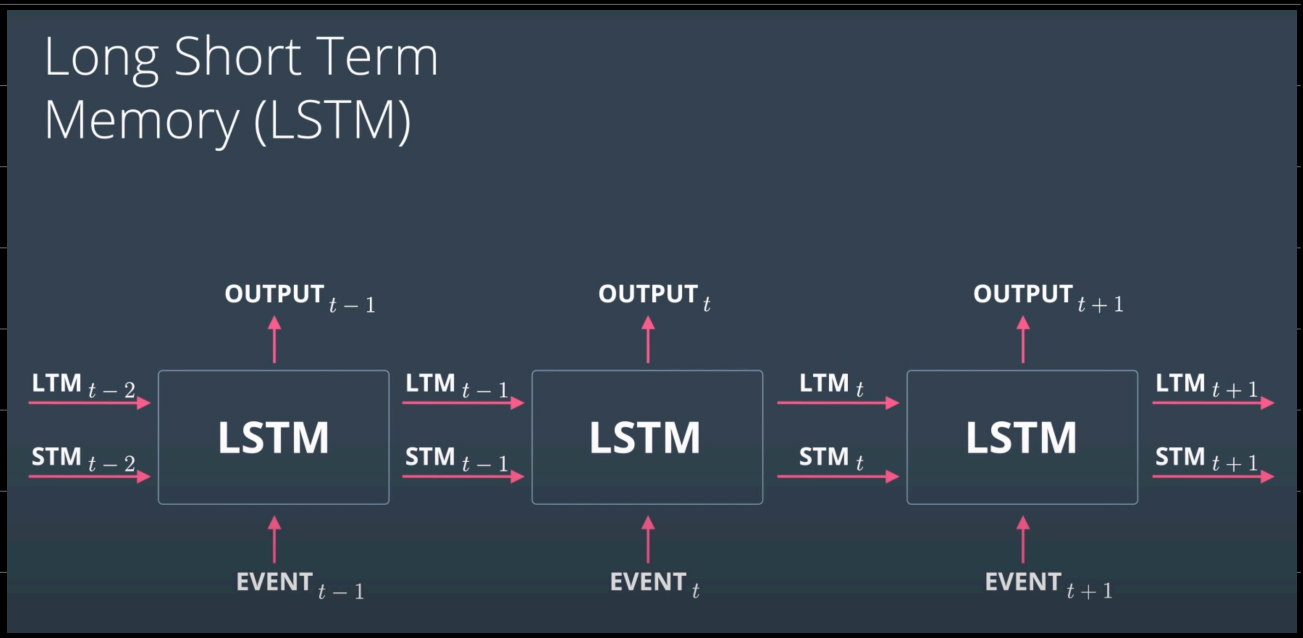
2) Short term memory and the event are joined together in the learn gate. (removes unnessecary info)

3) The LTM that we haven't forgotten yet plus the new learned information gets joined in the remember gate — outputs an updated LTM

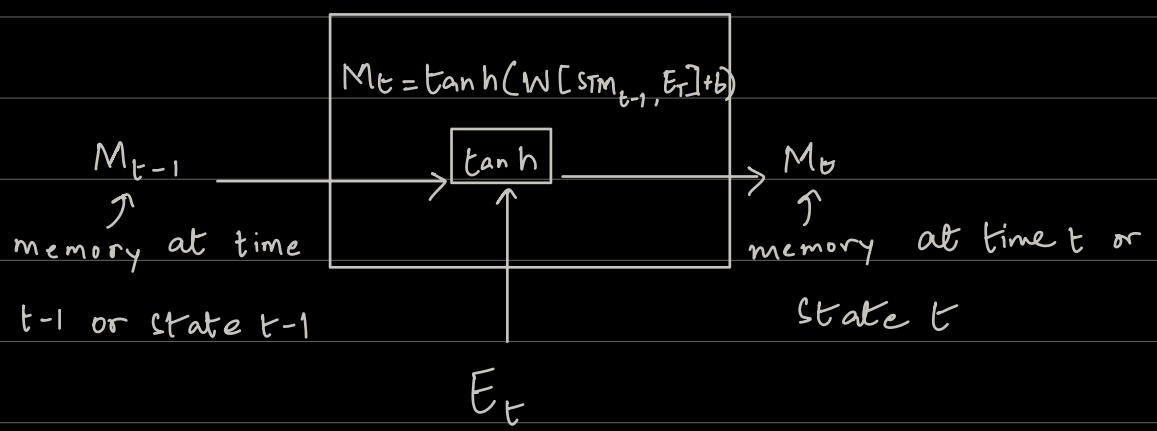
4) What information we use from what we previously know plus what we just learned to make a prediction.

what we know → Use gate → To make prediction and what is our new STM (output)

what we learned ↗



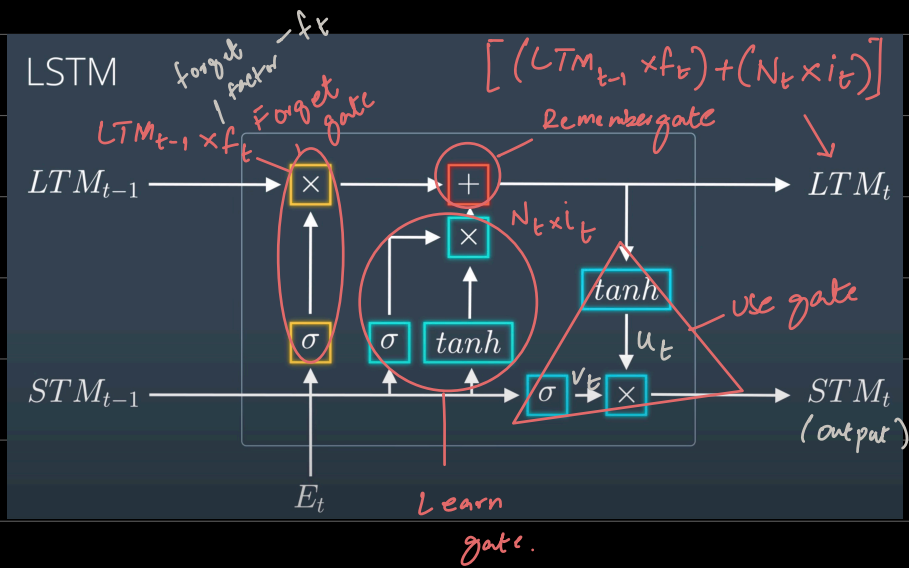
RNN:



In RNN, M_{t-1} or STM_{t-1} vector is multiplied to weight matrix and E_t vector is multiplied to weight matrix and both vectors are then added together plus a b (bias). Their result squished

through tanh function, which gives us output M_t or S_t. This further can be multiplied to a weight matrix to give output Y_t or can be fed to another iteration of neural network.

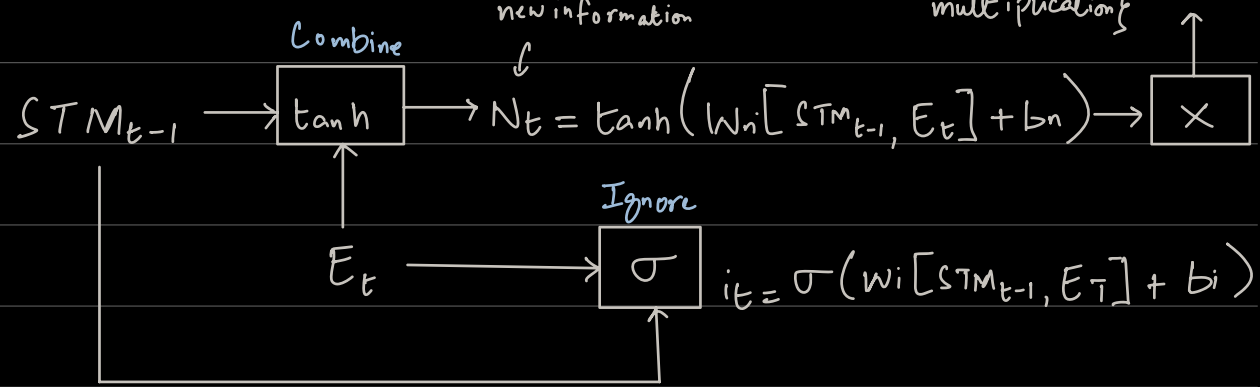
=> The output at this neuron acts as prediction and also memory that we carry to the next node.



(vector) Learned Information

{element wise multiplication} $N_t \cdot i_t$

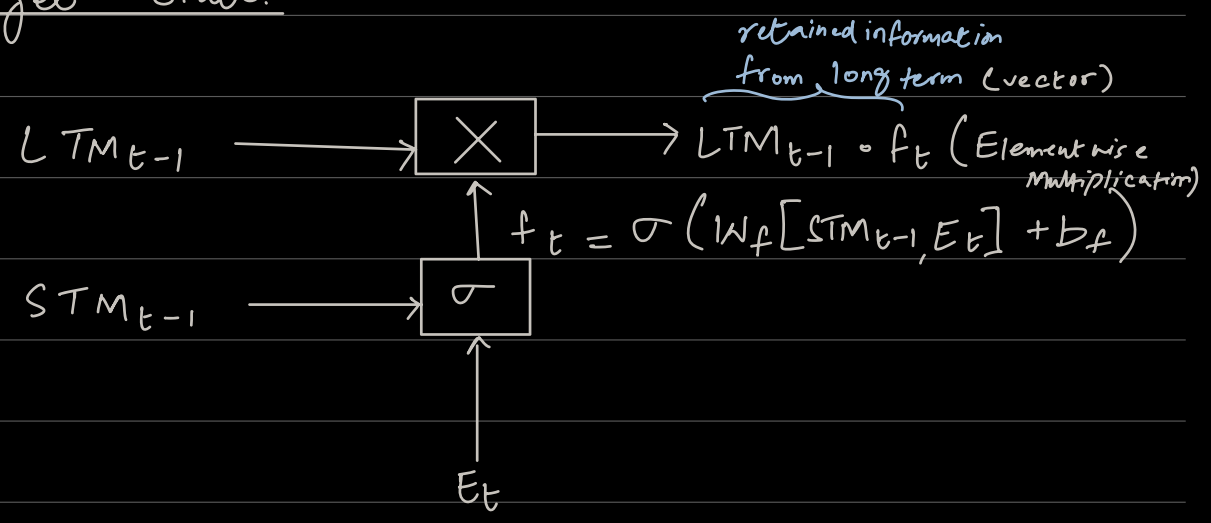
Learn Gate:



*) First, we combine STM_{t-1} and E_T by putting them through a linear function, which consists of joining the vectors (by addition) and multiplying joined vector by weight matrix - W_n and adding bias - b_n to it and squishing the result with tanh activation function.

*) Second, we calculate ignore factor i_t, which gets (multiplied element-wise to N_t (new info)). The i_t is calculated by joining vectors STM_{t-1} and E_t and multiplying new weight matoix - W_i and then adding a bias — b_i and then squishing the result through an activation function.

Forget Gate:



Remember Gate:

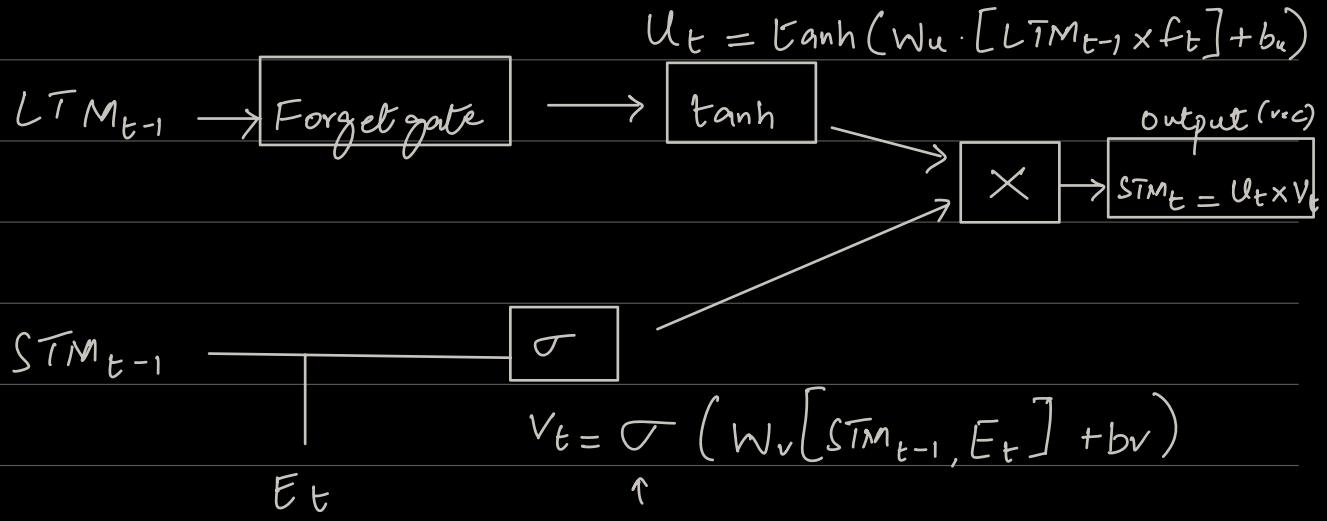
LTM (info) $t-1$ ↓ — (STM_{t-1} and E_t) ↓

Add output of **forget gate** and **learn gate**.

returns output for new LTM — LTM_t

Use Gate (output gate):

Uses Longterm memory just came out of forget gate and short term memory just came out of learn gate.



Learn gate + forget gate (retained info) ⟶ output Prediction

Training neural network on series of pairs of (movie review (text), rating (1 – 5)) ⇒ and after training for evaluation give a review and let neural network predict rating. (words play an important role)