## horizontal line

LSTMs

26.05.2025

# What is the problem in RNNs ?

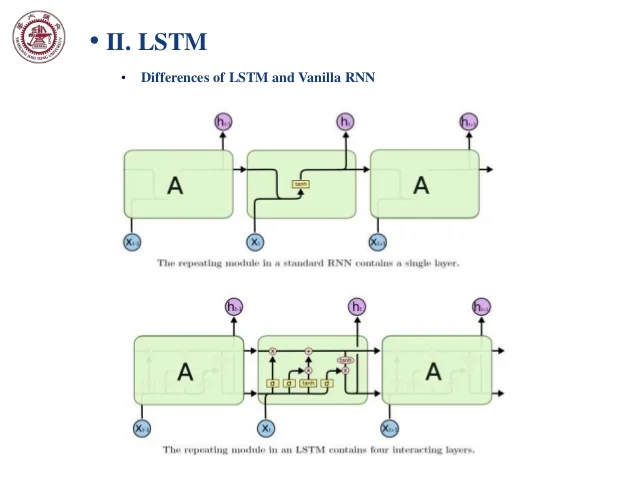
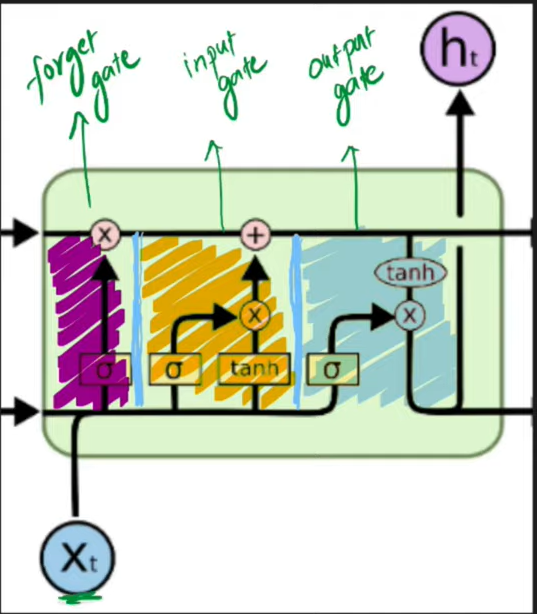
* RNN although capable of working on Text based data or Sequential data lacks on the context of Long term dependency .
* Can’t able to remember long term context in comparison to short term due to Vanishing / Exploding gradient problem.

# LSTM Core Idea

* RNN can only be able to maintain STC , by design can’t remember LTC (Long term Context).
* That’s why it came into mind to also maintain a path i.e. LTM.

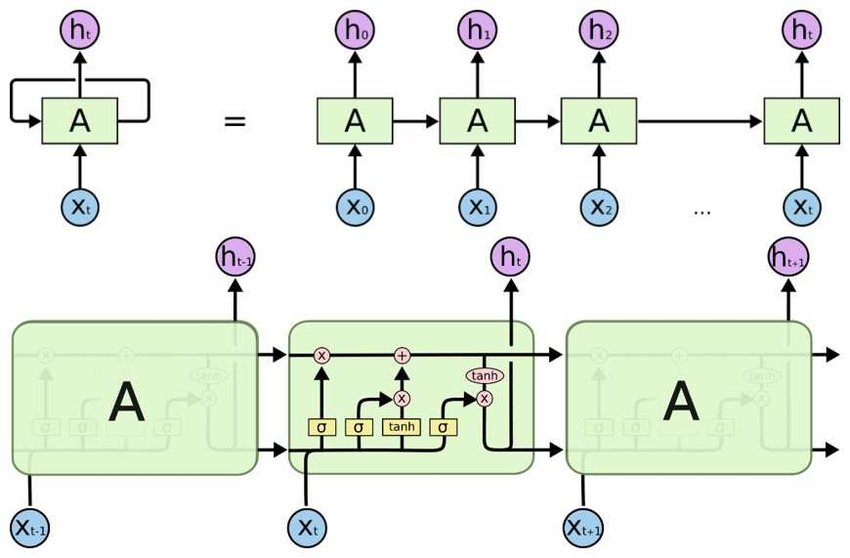
# Point of Difference in RNN and LSTM





## 

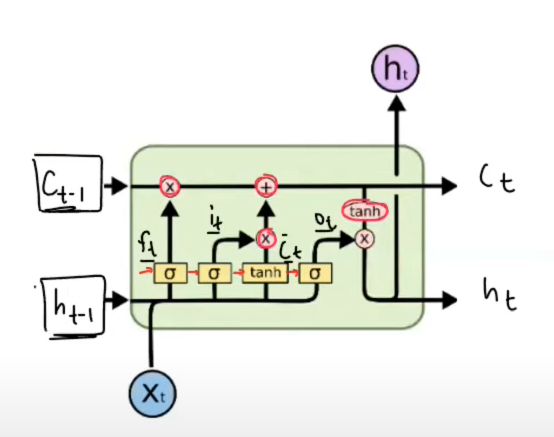
# The Architecture



Steps involved in LSTM :

* Inputs - Xt , Ht-1 , Ct-1
* Process - Update Ct-1 (remove and add) to make Ct
* Output - Ct , Ht

# Vectors , Neural Network layers , Pointwise operations

Vectors Ct , Ht , Ct-1 , Ht-1 , ft , it , Ct\_bar , Ot have same dimensionality. 

Xt can have any dimension.

Sigmoid and tanh are neural network layers that have n no. of nodes same for all . It is a hyper parameter decided by us .

Dimensions of vectors == no. of nodes.

Pointwise multiplication , addition and tanh are done on same dimension vectors element by element.

Xt is given after vectorization OHE in the form of vector

Let’s the Xt be 4 dim vector and Ht-1 be 3 dim vector . Thus no. of nodes also 3.

# The Gates

## Forget Gate : remove something from previous cell state.

Stages :

1. Calculate ft
2. Ct-1 ft



Ft = (Wf [Ht-1, Xt] + Bf)

Ft = (3X7)(7X1) + (3X1)

Ft = (3X1) + (3X1)

Ft = (3X1)

Ft is basically a 3 dim vector that is a sigmoid weighted sum i.e. it contains the probability of each element how much it is important .

If [0 0 0] nothing removed .

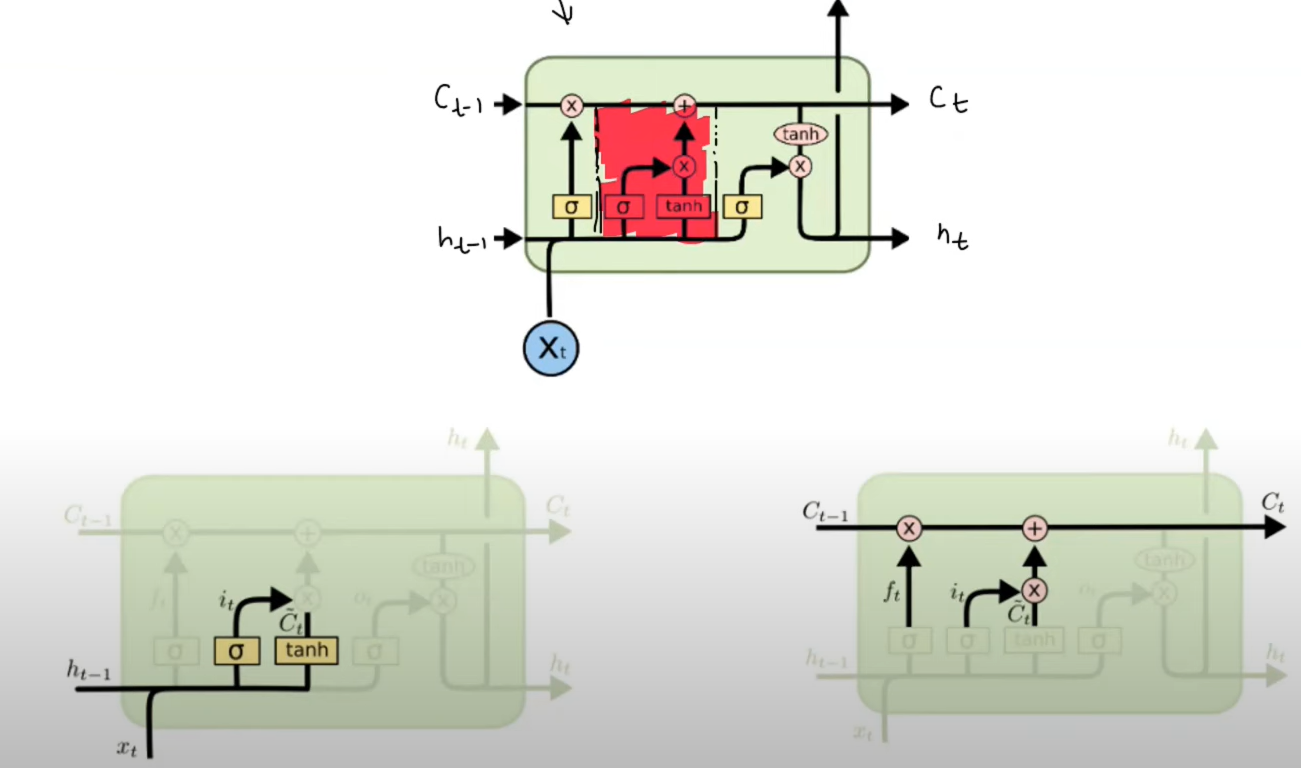
If [1 1 1] fully removed .

If [0.5 0.5 0.5] half context removed .

Now step 2 is :

Ct-1 ft i.e. removed part from previous LTM.

## Input Gate : add something to previous LTM.

Stages : 

1. Calculate Candidate cell state Ct\_bar.
2. Calculate it for filtering it.
3. Calculate Ct



Ct\_bar = (Wc [Ht-1, Xt] + Bc)

Ct\_bar = (3X7)(7X1) + (3X1)

Ct\_bar = (3X1) + (3X1)

Ct\_bar = (3X1)

Similiarly Ct\_bar is a 3 dim vector , used find out what is useful to add.

it is also a 3 dim vector for calculating how much percent is useful to add.



it = (Wf [Ht-1, Xt] + Bf)

it = (3X7)(7X1) + (3X1)

it = (3X1) + (3X1)

it = (3X1)

It is used to filter on Ct\_bar to add most important information.

For that we do :

Ct\_bar it

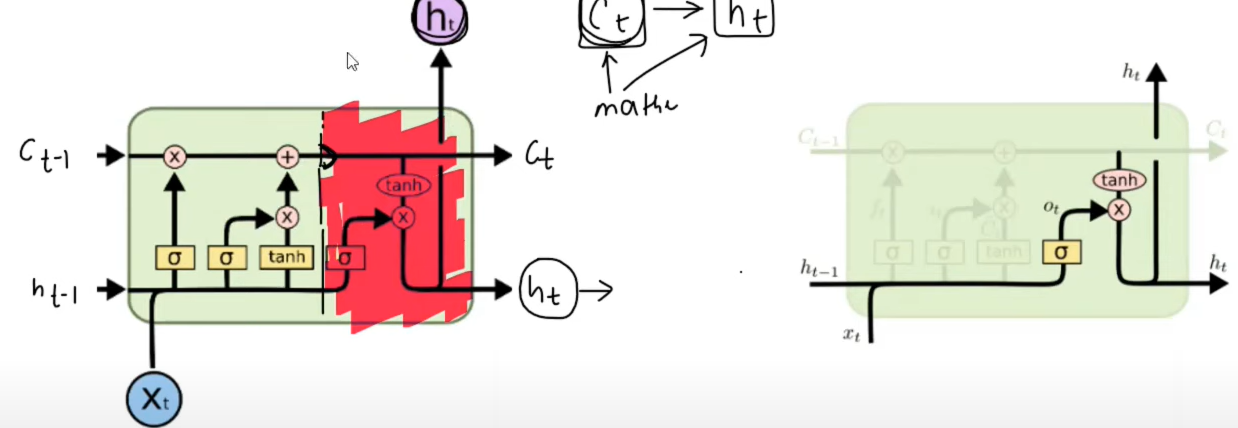
3rd step is to :

Ct = (Ct-1 ft) (Ct\_bar it)

Note : To find out part of important thing in Ct-1 we use “”

To add something useful we use “”

## Output Gate : Create current STM



Stages :

1. Calculate tanh(Ct).
2. Calculate Ot
3. Ot tanh(Ct)



Ot = (Wf [Ht-1, Xt] + Bf)

Ot = (3X7)(7X1) + (3X1)

Ot = (3X1) + (3X1)

Ot = (3X1)

Ot is also a sigmoid activated weighted sum giving probability means it find out parts of Ct which is more important to be in STM.

That’s why we use tanh(Ct).

Ht = Ot tanh(Ct)

* [Next Word Predictor Application using LSTM](https://colab.research.google.com/drive/1KWghhBvM7eARxXLnbcxwALtZFcWIs_J3)