LSTM(Long Short Term Memory)

- What
  - LSTMs are a special kind of RNN — capable of learning long-term dependencies by remembering information for long periods is the default behavior.
    - All RNN are in the form of a chain of repeating modules of a neural network. In standard RNNs, this repeating module will have a very simple structure, such as a single tanh layer.
    - LSTMs also have a chain-like structure, but the repeating module is a bit different ... structure. Instead of having a single neural network layer, four interacting layers are communicating extraordinarily.
- Working
  - Workings of LSTMs in RNN - The LSTM cell also has a memory cell that stores information from previous time steps and uses it to influence the output of the cell at the current time step. The output of each LSTM cell is passed to the next cell in the network, allowing the LSTM to process and analyze sequential data over multiple time steps.
    - Decide How Much Past Data It Should Remember
      - $f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$
        - f_t : Forget gate
        - Decide what information to throw away or remember. Looks at h_{t-1} (from previous time step) and x_t (current input).
    - Decide How Much This Unit Adds to the Current State
      - $i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$
      - $\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$
        - i_t : Input gate
        - Determines which of the information to let through (or update) the cell state.
    - Decide What Part of the Current Cell State Makes It to the Output
      - $o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$
      - $h_t = o_t * \tanh(C_t)$
        - o_t : Output gate
        - Decides what information to output (the output is the current time step).
  - Now Let's Get Deeper
    - Just Imagine Instead Of Using The Simple Tanh Layer We Have Four Interacting Layers
    - Long Ago: We Rarely Put Sequential Thing Ago
      - Long Ago: Long Ago We Rarely Put Sequential Learning ... Yesterday
      - Long Ago: Long Ago We Rarely Put Sequential Learning ... Today
      - Long Ago: Long Ago We Rarely Put Sequential Learning ... Yesterday, Today, Tomorrow
      - LSTM Uses Two Path
      - Long Ago: Long Ago We Rarely Put Sequential Learning ... Yesterday, Today, Tomorrow
        - But Hold On , It's Not That Easy
          - In Normal Vanilla RNN We Have Simpl Unit
          - LSTM Have Complex One
    - It Uses Two Activation function
      - We Already know
    - Blue Indicating The Sigmoid And Orange The Tanh
    - So How?
      - Secondly
        - Although We Can Not Add Or Modify Its Elements
        - Gate Is Just A Way To Optionally Let Information Through
        - The Gate Is Parameterized By Weights Which Are Learned During The Learning Process
        - Gate Is Just A Way To Optionally Let Information Through
        - The Gate Is Parameterized By Weights Which Are Learned During The Learning Process
        - Let's Add Some Math
          - The First
            - Step 1: Decide What Information We Are Going To Throw Away From The Cell State
            - Step 1: Decide What Information We Are Going To Throw Away From The Cell State
            - Forget Gate Layer Is A Sigmoid Layer
            - Forget Gate Layer Looks At h_{t-1} And x_t, And Outputs A Number Between 1 And 0 For Each Number In The Cell State C_{t-1}
            - Now That We Have Analyzed Some Basic Concepts Let's Take Input As 1
              - We Start With We Know Every Input Is Given A Weight Of 0.5
              - Let's Take Input As 1 Multiply Weight 0.5 =