Recurrent Neural Network [Lec : 02]

Date : 30 / 04 / 2020

Deep sequence Modeling

- Audio is a kind of sequential data.
- Useful in analyzing medical signals.
-

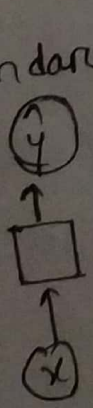
Design criteria:

↳ Handle variable length sequences
↳ Track long term dependencies
↳ Maintain informations about order
↳ Share parameter across the sequence

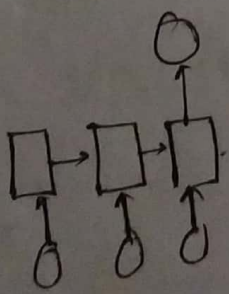
* ↳ RNN as an approach to sequence modeling problems.

RNN

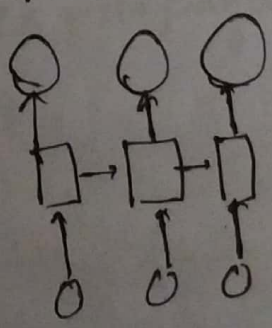
* standard feed forward neural network



one to one

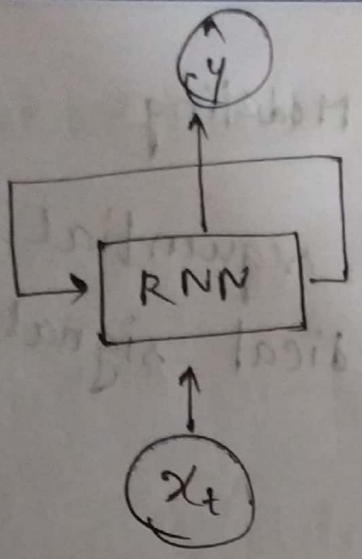


many to one (sentiment)



Many to many

Date : ____/____/____



apply a recurrence relation at every time step to process.

$$h_t = f_W(h_{t-1}, x_t)$$

h_t → cell state

h_{t-1} → old state

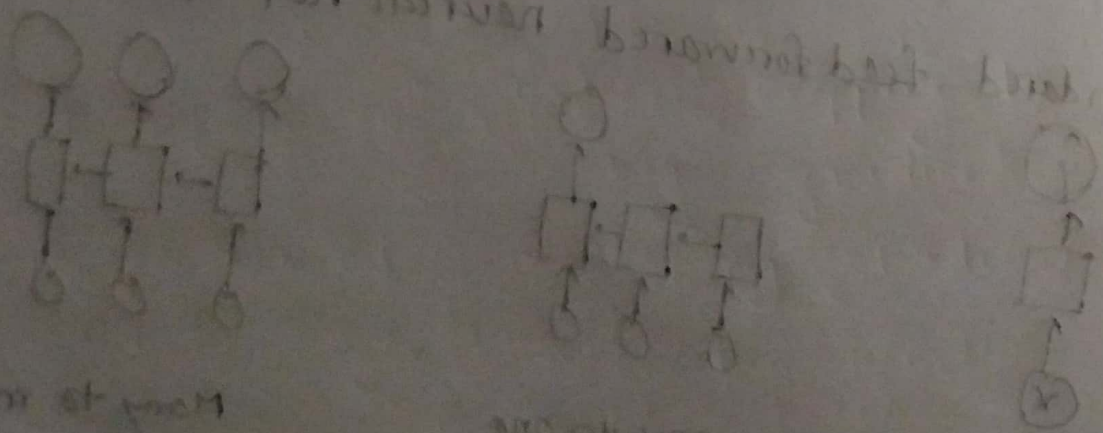
x_t → input vector at t

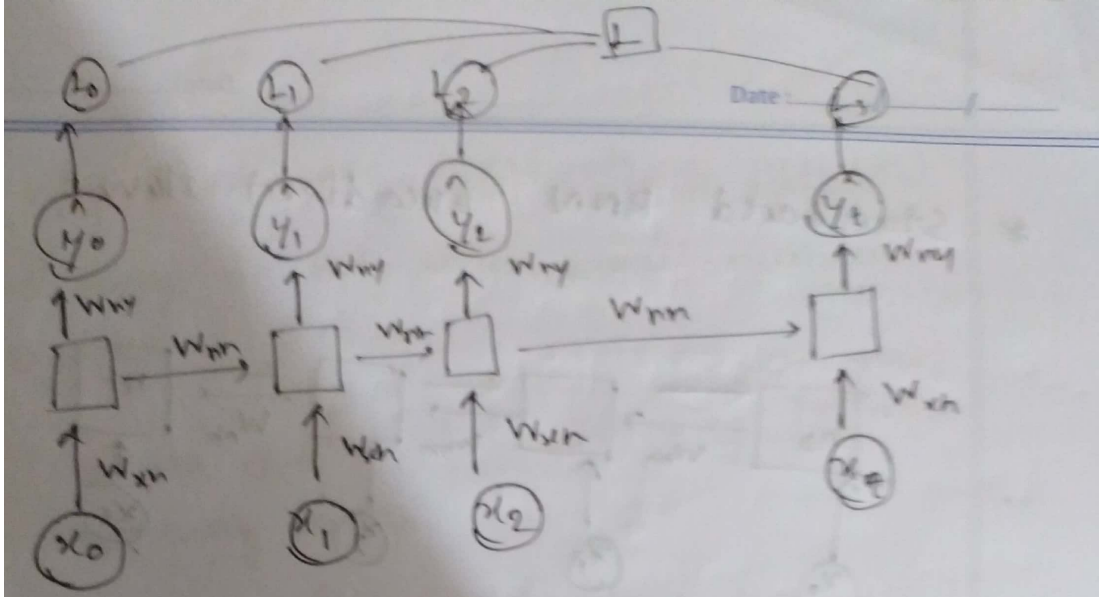
f_W → function parameterized by W

output vector → $\hat{y}_t = W_{hy}^T h_t$

update hidden state → $h_t = \tanh(W_{hh}^T h_{t-1} + W_{xh}^T x_t)$

input vector → x_t





Back propagation Through Time (BPTT)

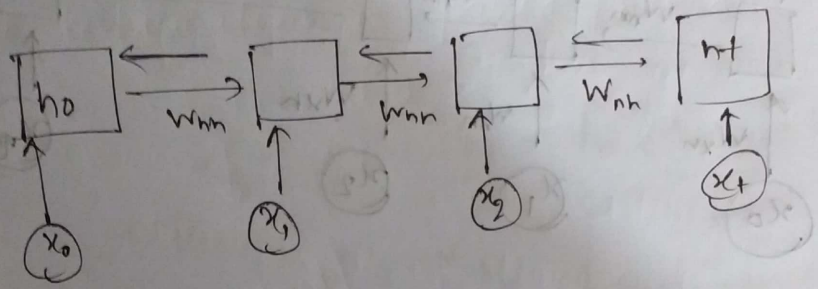
→ Take the derivative of the loss w.r.t to each parameters

→ Shift parameters to minimize the loss

In RNN, errors are back propagated in each individual time step

Date : ..../..../....

* Standard RNN Gradient flow :



Computing h0 involves many factors of W_{hh} + repeated gradient computation

- repeatative use of gradient can be problematic

Many values > 1 : exploding gradient problem

→ gradient clipping to scale big gradients

Many values < 1 : vanishing gradient problem

→ ① Activation function
② Weight Initialization
③ Network architecture

Prottvas

Date : ____/____/____

Why vanishing gradients a problem?

→ ~~many~~ multiply many small numbers together

↓

as the number will shrinking, it will eventually vanish

↓

errors due to further back time steps have smaller gradients

↓

biasing the network to capture short term dependencies

Solution

1. Activation function:

Using ReLU prevents f' from shrinking the gradients when $x > 0$

2. Parameter initialization:

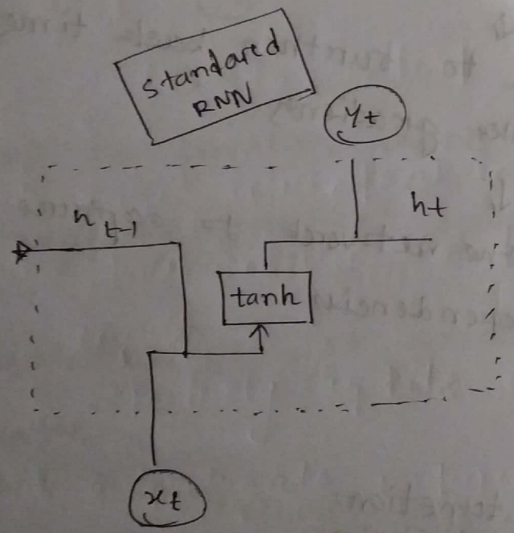
Initialize weights to the identity matrix helps to prevent them to shrinking to zero

Protivas

Date :...../...../.....

3. Gated cells: use more complex recurrent unit with gates to control what information is passed through

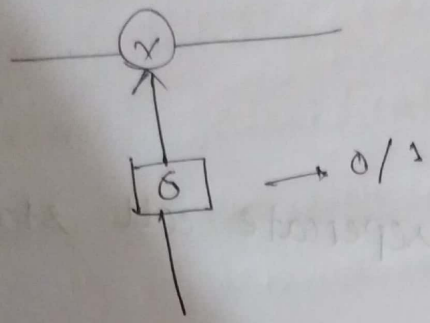
ex – LSTM, GRU



LSTM:

– contains computational blocks that control information flow.

– LSTM cells are able to track information throughout many time steps.

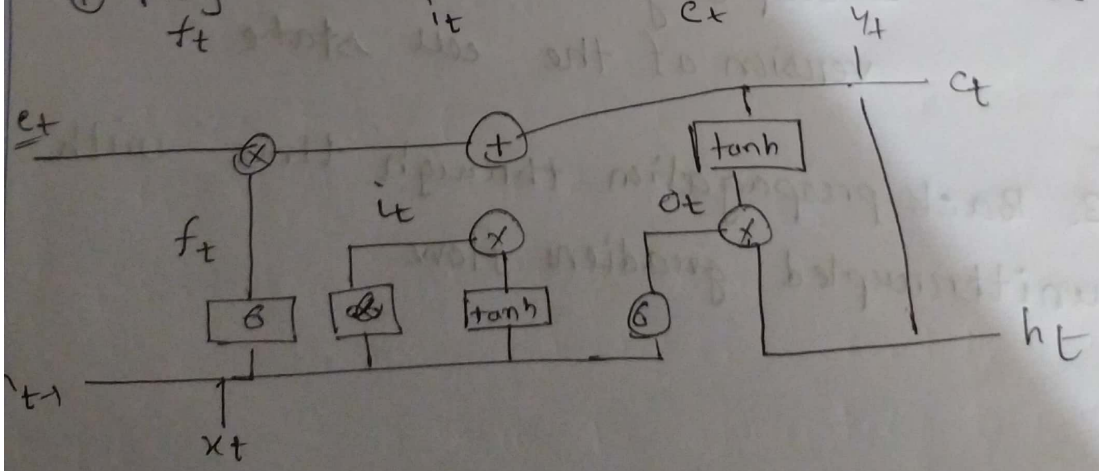


- Information is added or removed through structures called gates.

gates → consists of a neural net layer i.e. sigmoid

How LSTM works:

① Forget f_t ② Store i_t ③ Update c_t ④ Output o_t



Date :...../...../.....

* ~~Urde~~ Uninterrupted gradient flow.

Key concepts:

1. Maintain a seperate cell states from what is outputted
2. Use gates to control information flow
   - Forget irrevalent information
   - Store information from currecent input
   - selectively update cells
   - output gate returns a filtered version of the cell state
3. Back propagation through time with uniterrupted gradient flow.

RNN applications

1. Music generation
2. Sentiment classification
3. Machine Translation
   - attention mechanism
4. Environmental modeling