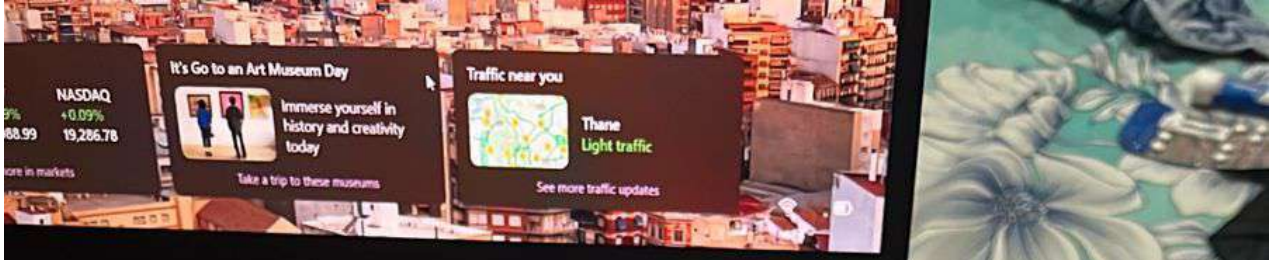
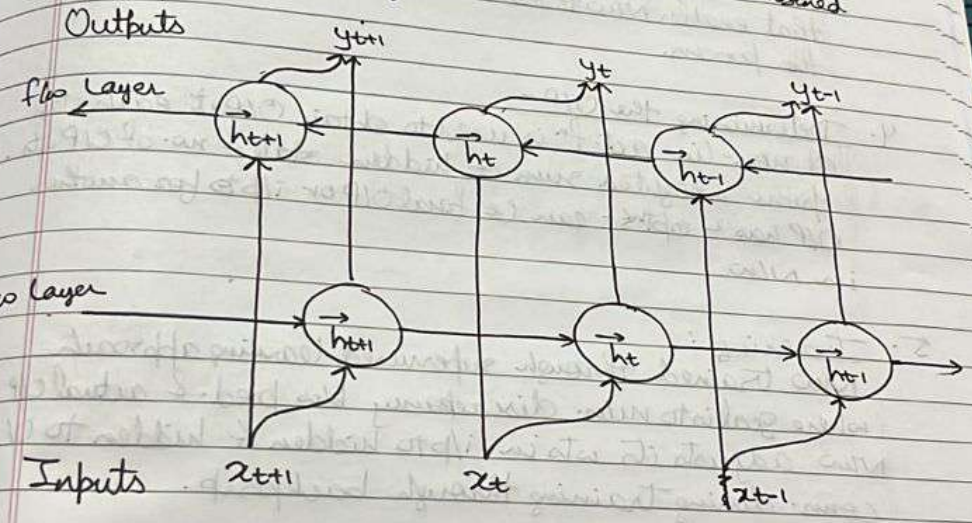


Backprop. in RNN -
Bi-dir. RNN -

- Made to process seq. data.
- In order for NWs to process use info. from both past & future context in its predictns, BRNNs process i/p seq. in both Flw & Bkw directn.
- Has 2 distinct hidden layers One of which proc. i/p seq. Flw & other proc. it Bkw. After that results from these hidden layers are collected & i/p to pred. making final layer.
- Any RNN cell such as LSTM & GRU can be used to create the recurrent hidden layers.
- Functn BRNN fns similarly to conventional RNN in Flw direct updating the hidden state depending upon curr. i/p & prior hidden state at each time step.
- Bkw hidden layer, on other hand analyses i/p seq in opp. manner, updating hidden state based on current i/p & hidden state of next time step.
- Compared to conventional unidir RNNs, accuracy of BRNN improved since it can process info. in both direct's & acc. for both future & past contexts.
- Bcoz the 2 hidden layers can complement one another & give final pred. layer more data, using 2 distinct hidden layers also offer type of model regularizatn.

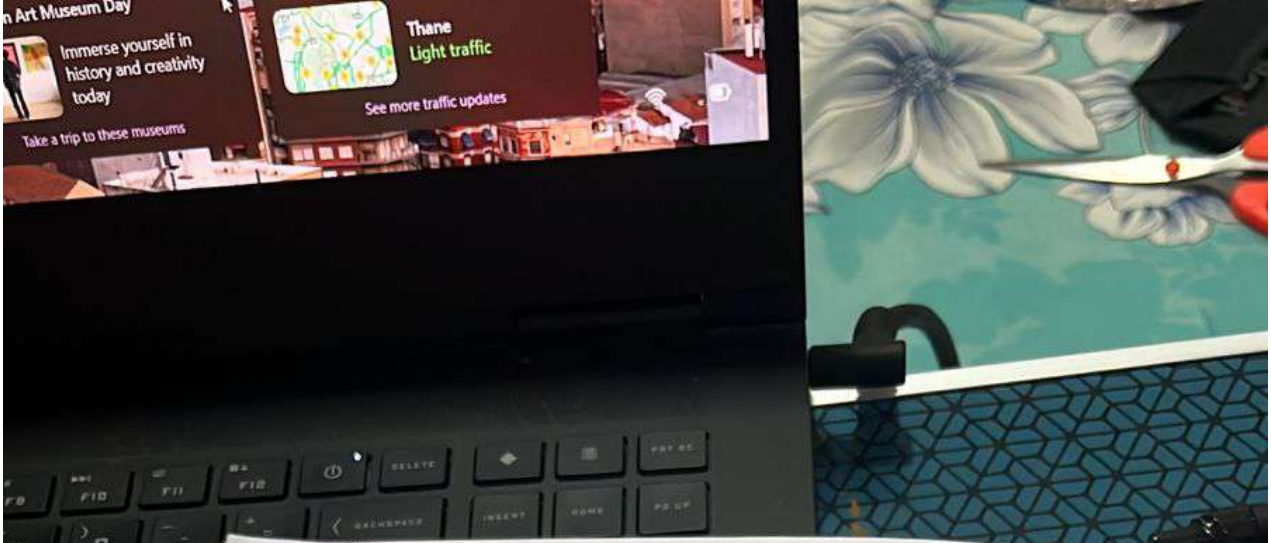
- To update model params., grads are computed for both fw & bw passes of back prop. through time tech. & is typically used to train the ~~model~~ BRNN.
- I/P seq. is processed by BRNN in single fw pass at inference time & preds are made based on combined O/Ps of hidden layers.



Working -

① Inputting a seq:
A seq. of data pts., each rep. as a vector with the same dimensionality are fed into a BRNN. Seq. might have diff. lengths.

② Dual Processing:
Both fw & bw direct^ns are used for processing the data. On the basis of i/p at that step & hidden state at step t-1, hidden state at timestep t is determined in fw direct^n. I/P at step t & hidden state at step t+1 are used to



compute hidden state at step t in reverse way.

3. Computing hidden state:
A non-lin. act. fn at on weighted ~~sum~~ sum of i/p & prev. hidden state is used to calc. hidden state at each ~~time~~ step. This creates a memory mechanism that enables NNw to remember data from earlier steps in the process.

4. Determining the O/P -
A non-lin act. fn is used to obtain O/P at each step from weighted sum of hidden state & no. of O/P nds. O/P has 2 opts - can be final O/P or i/p to for another in NNw

5. Training :
NNw trained through supervised learning approach where goal is to min. discrepancy btw pred. & actual op. NNw adjusts its wts in i/p to hidden & hidden to O/P conn. during training through backprop.