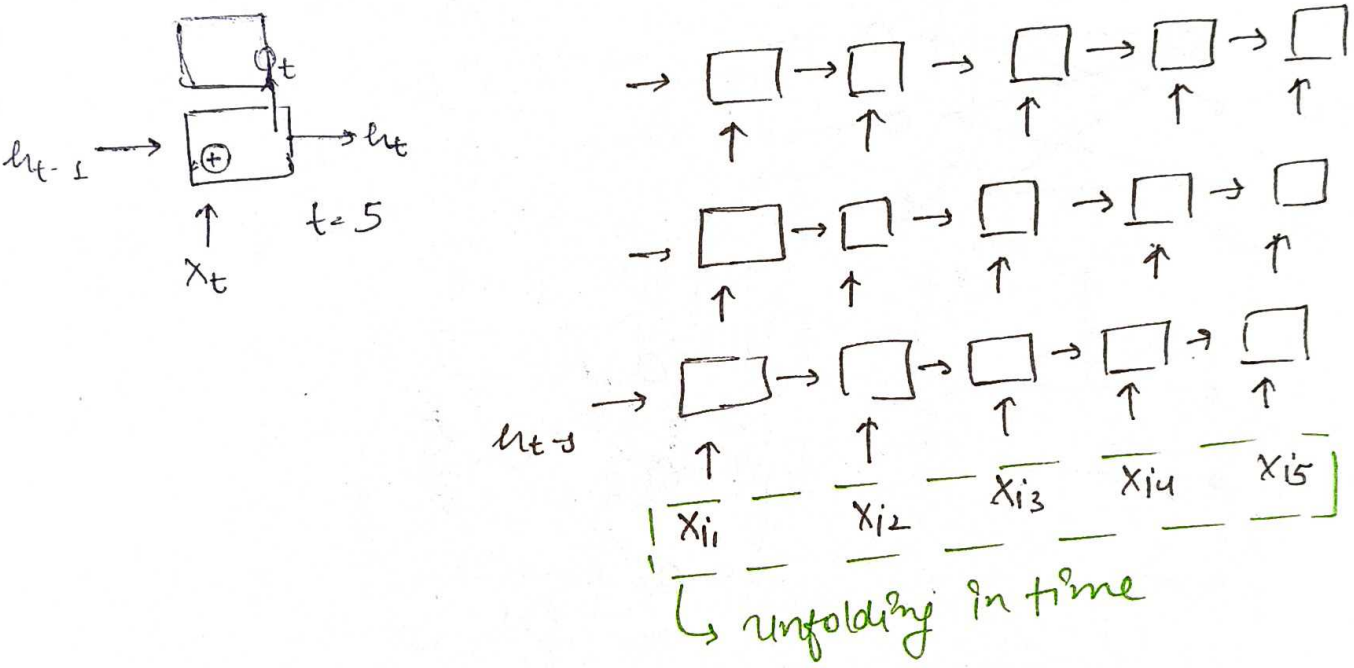
Deep RNN

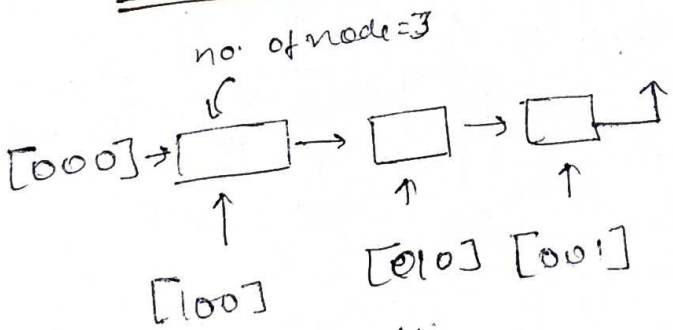
eg:- Sentiment

reviews	Positive / Neg
5 words	



↳ unfolding in time

Architecture

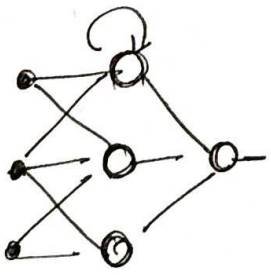


unfolding

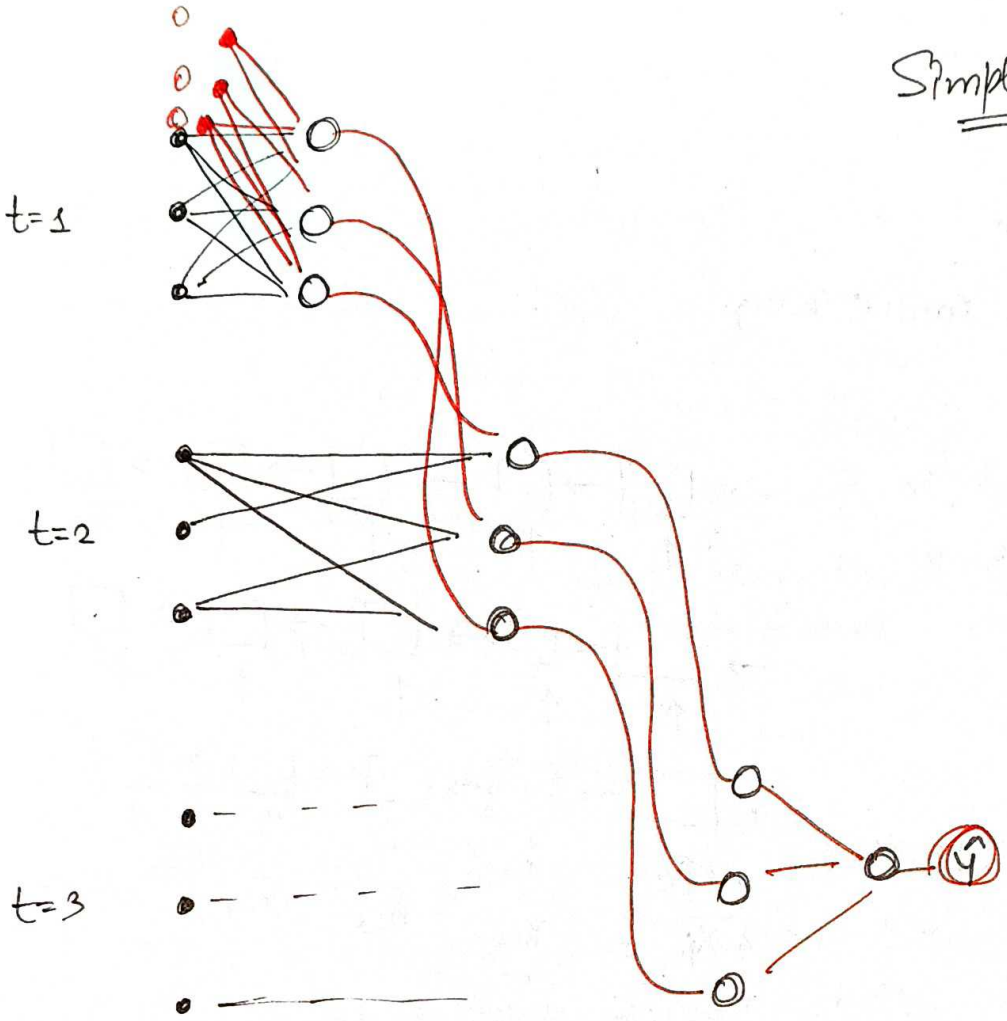
review	Sentiment
cat mat rat	1
rat rat mat	0
mat cat cat	1

[100] [010] [001]

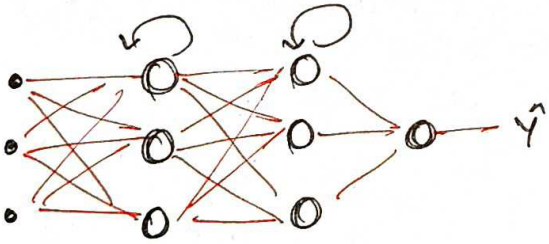
$t=3$

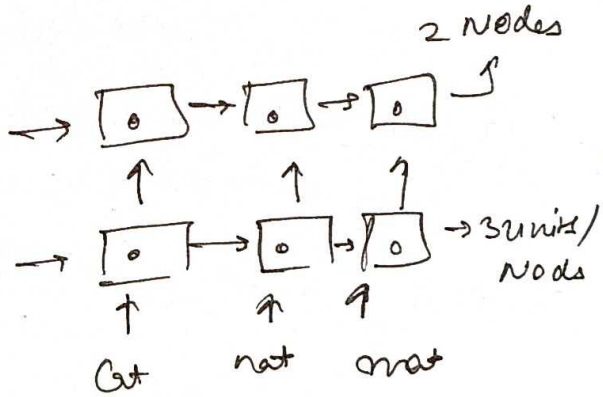


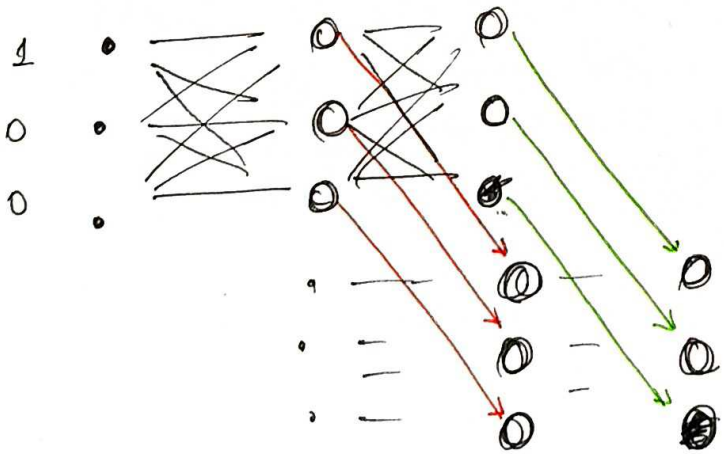
Simple RNN

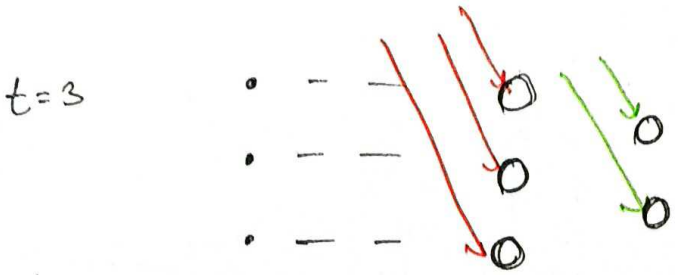


deep RNN

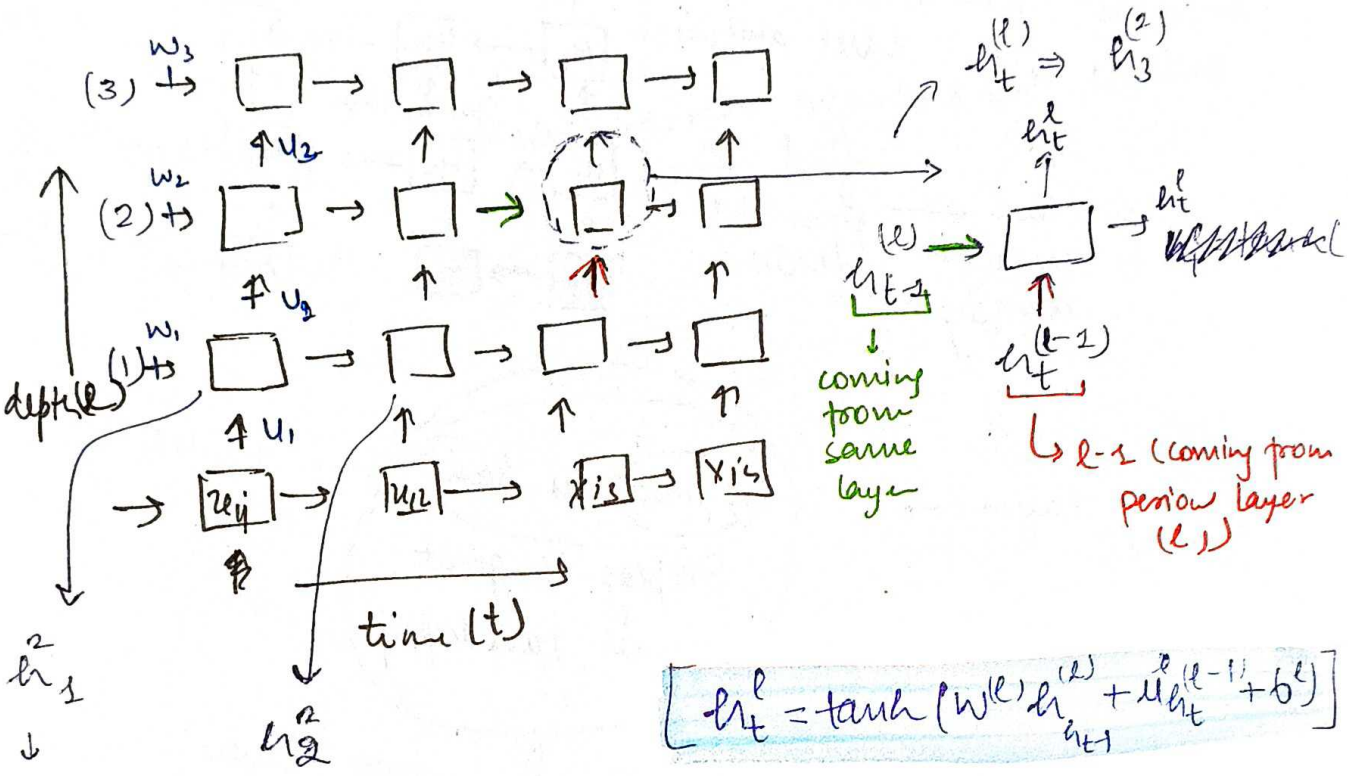






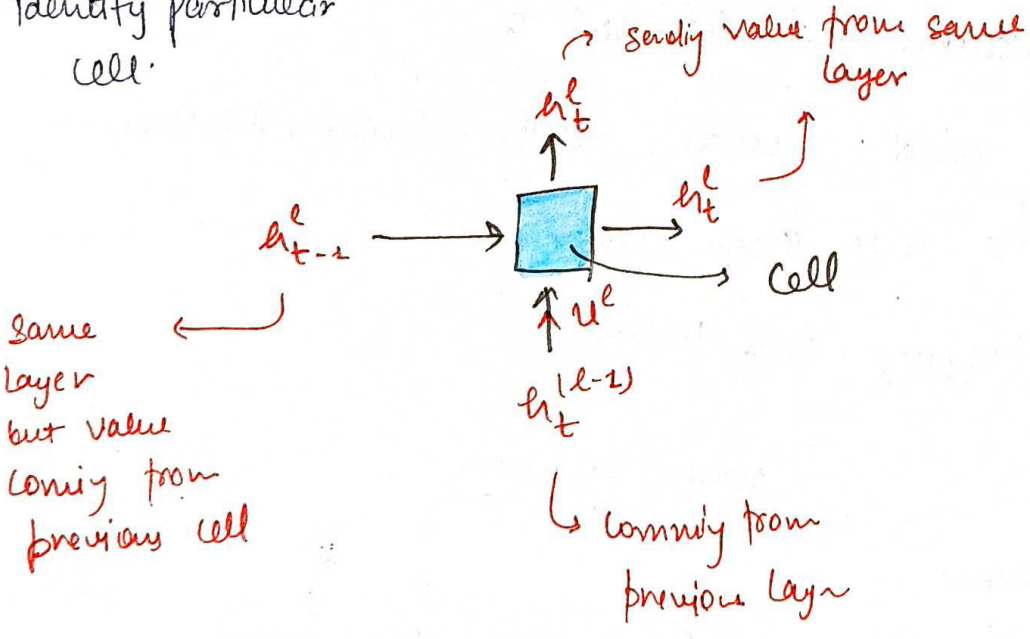


Notation



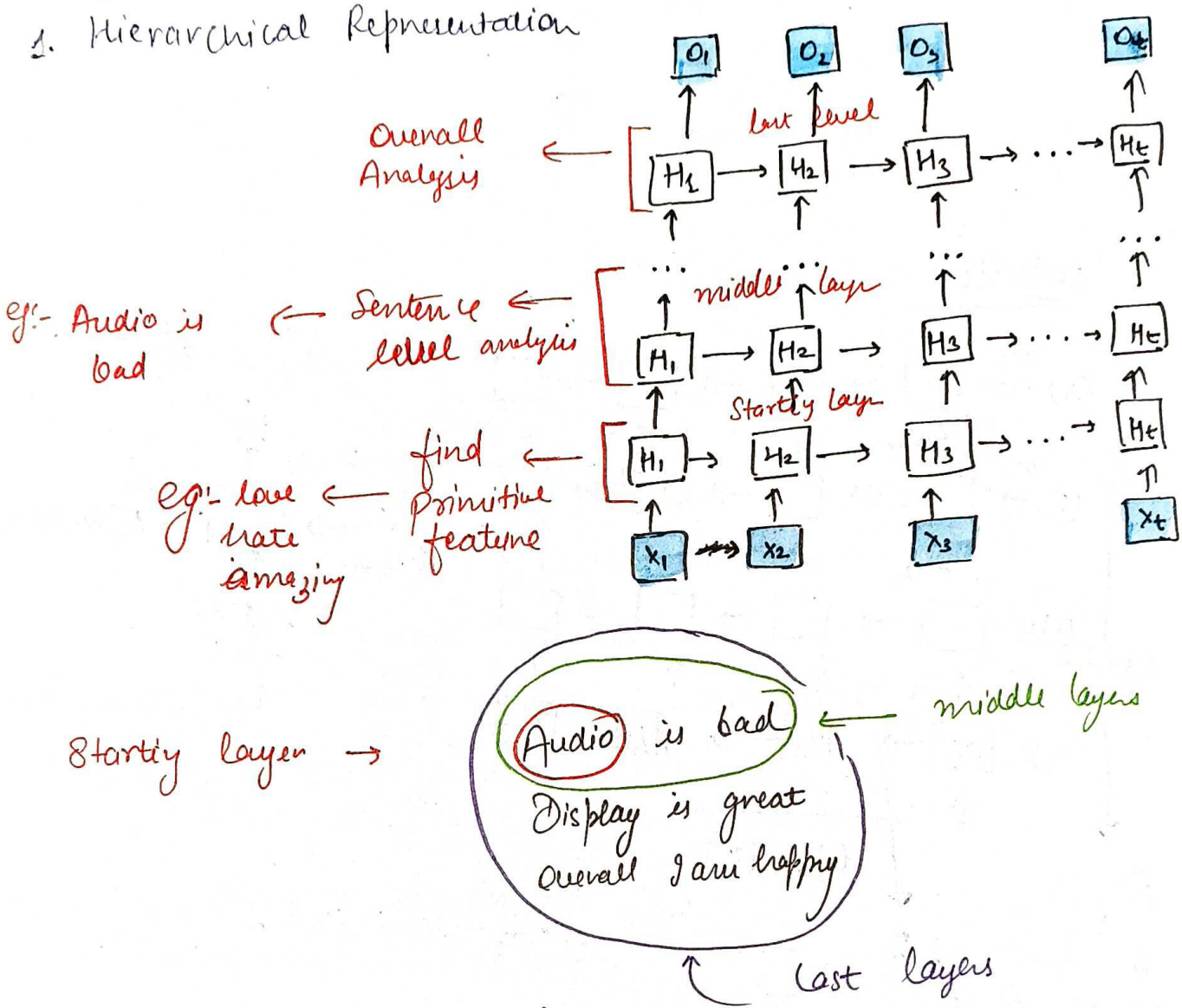
$$h_t^l = \tanh\left(W^{(l)} h_{t-1}^{(l)} + U^l h_t^{(l-1)} + b^l\right)$$

identity particular cell.



Why and when to use?

1. Hierarchical Representation



2. Customization for Advanced Tasks

encoder and decoder with Attention Mechanism

↳ use Deep RNN

for eg:- Language translation

when to use

- complex tasks
  - Speech Recog
  - Machine translation
- Large datas
- computational low PC take large time
- Simple RNN

Variants

Simple RNN

- Deep LSTM
- Deep GRU

Disadvantages

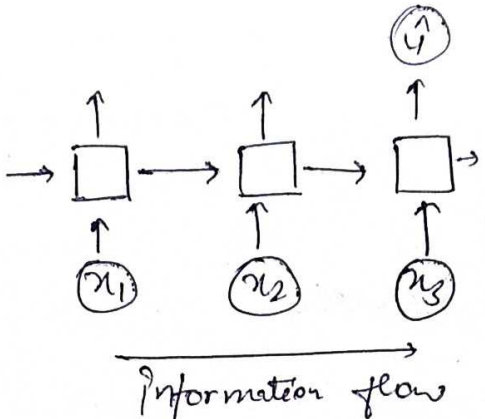
(i) Overfitting → careful during hyperparameter selection

(ii) Training data

Bidirectional RNN

The why?

Simple RNN →



y_i depend on the past input

we are in this scenario where future input affect past output → this fail Simple RNN.

eg: [NER] → Named entity recognation } breaks sentence word by word and extract entities.

(i) I live in gurgaon. → place

(ii) I love amazon, its a great website

I love amazon its a beautiful river

When we go left to right, we don't know amazon is a organisation or LOC (Place). At this point future input affect past output. And we cannot use RNN, LSTM and GRU because of left to Right information. That why use bidirectional

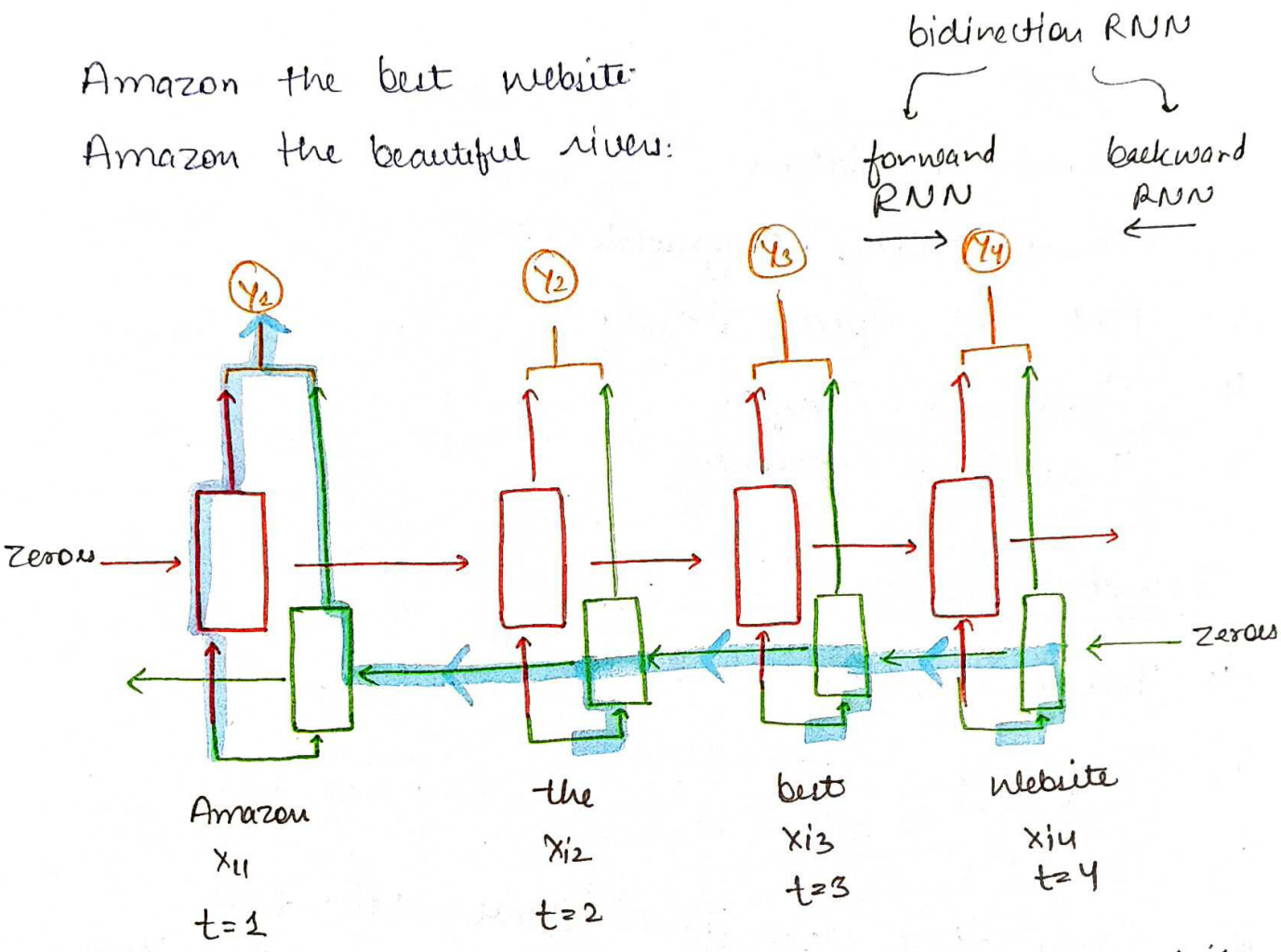
← I love amazon, its a great website →

bidirection takes input from both side.

Bidirectional RNN Architecture

Amazon the best website.

Amazon the beautiful rivers.



for Y_1 calculation → Amazon, the, best and website contribute.

Y_2 ← calculate karne time website word bhi aaya to matlb mean we are taking about amazon website.

$$\vec{h}_t \text{ (for red box)} = \tan(\vec{W}\vec{h}_{t-1} + \vec{U}x_t + \vec{b})$$

$$\overleftarrow{h}_t \text{ (for green)} = \tan(\overleftarrow{W}\overleftarrow{h}_{t+1} + \overleftarrow{U}x_t + \overleftarrow{b})$$

↳ why plus? because $t=3$ ko $t=4$ affect kar rha (next t)

$$y_t = \sigma(\vec{V}[\vec{h}_t, \overleftarrow{h}_t] + b)$$

some weight

Applications and Drawbacks

Application

1. Machine translation
2. Name Entity Recognition
3. Part of Speech tagging
4. Sentiment Analysis
5. Time Series forecasting

Drawback

(i) Training time ↑↑

(ii) Complexity

(iii) Overfitting

(iv) Some time we don't have all data like real time Speech recognition. Bidirection RNN cause delay.