learning long-range dependencies.

Depending on activation and parameters we get explode / vanish. If the gradients are large that's called exploding gradient problem.

⇒ The basic idea of using a gating mechanism to learn long-term dependencie

★ Implementation of Vanilla RNN :-

Let suppose we have two sentences.

1. Vocabulary = ['Mango'(0), 'Banana'(1), 'colour'(2), 'is'(3), 'red'(4), 'yellow'(5)]

(i) Mango is red colour.

(ii) Banana is yellow colour.

(iii) hair has black colour.

Now, how we represent each word?

For instance it represent them with one-hot-encoding:

Maya is red colour.

$$\begin{bmatrix}1\\0\\0\\0\\0\\0\end{bmatrix} \quad \begin{bmatrix}0\\0\\0\\1\\0\\0\end{bmatrix} \quad \begin{bmatrix}0\\0\\0\\0\\1\\0\end{bmatrix} \quad \begin{bmatrix}0\\0\\1\\0\\0\\0\end{bmatrix}.$$

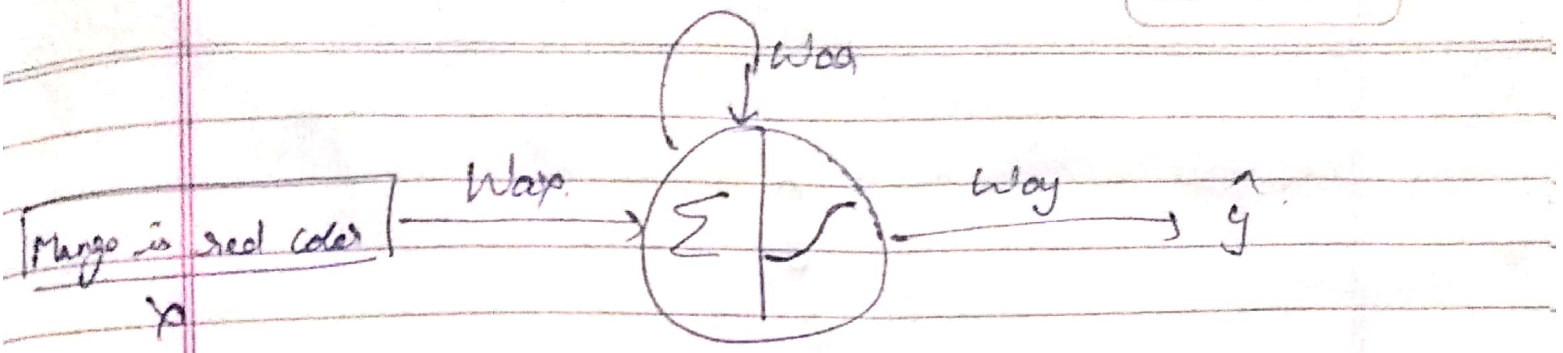
Input_size = [vocabulary_len × 1].

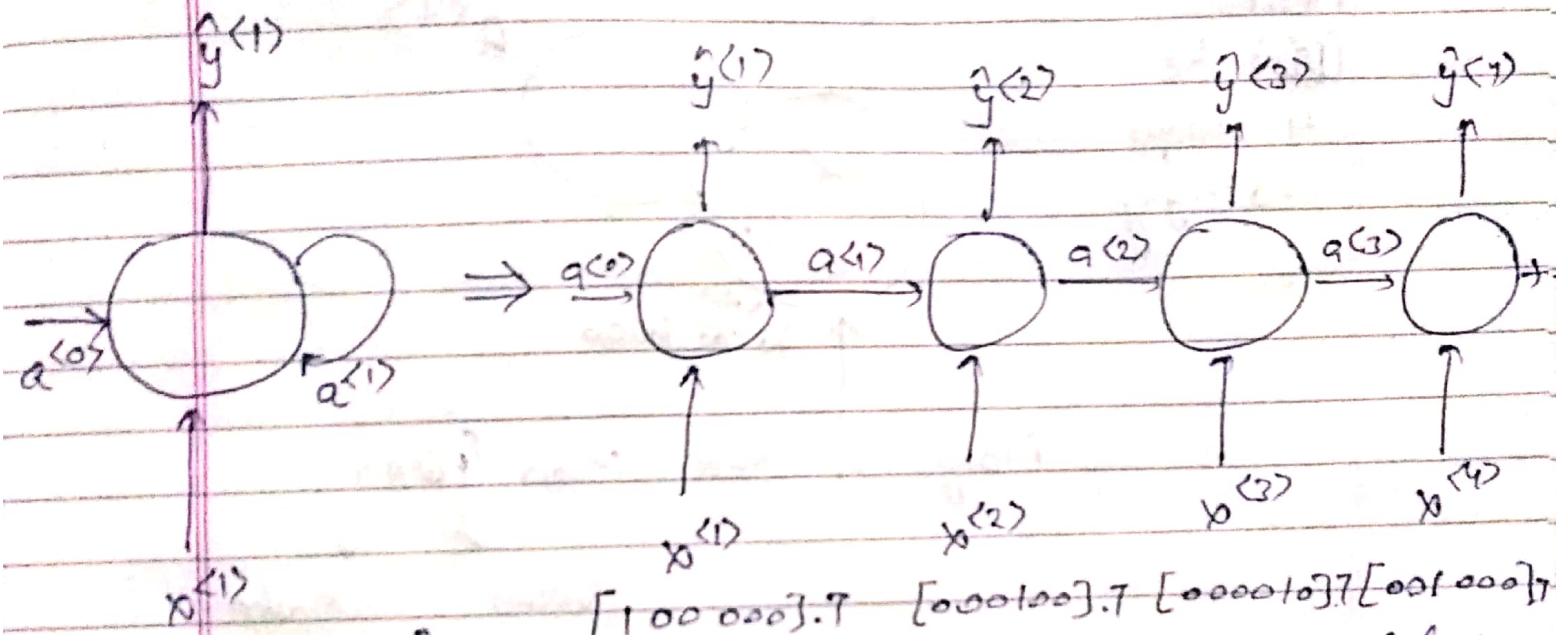
For instance, we don't train these sentence that's why these are OHE but after we use pre-trained model which are trained on 1000 corporus so then this matrix is not represent by OHE. It is content some positive values which represent some dependencies.

Remember that weights are shared with all time steps.

Parameter representation:

1. Wea : Weight of activation × activation
2. ~~Wax : Wya~~
2. Wax : Weight of input × activation
3. Way : Weight of activation × output.
4. ba : Activation bias.
5. ~~Output~~ by : Output bias





[100000].T [000100].T [000010].T [001000].T

Mango is red colour.

Mango [100000]
is [000100]
red [000010]
colour [001000]

~~Sentence_len * Vocab_size.~~

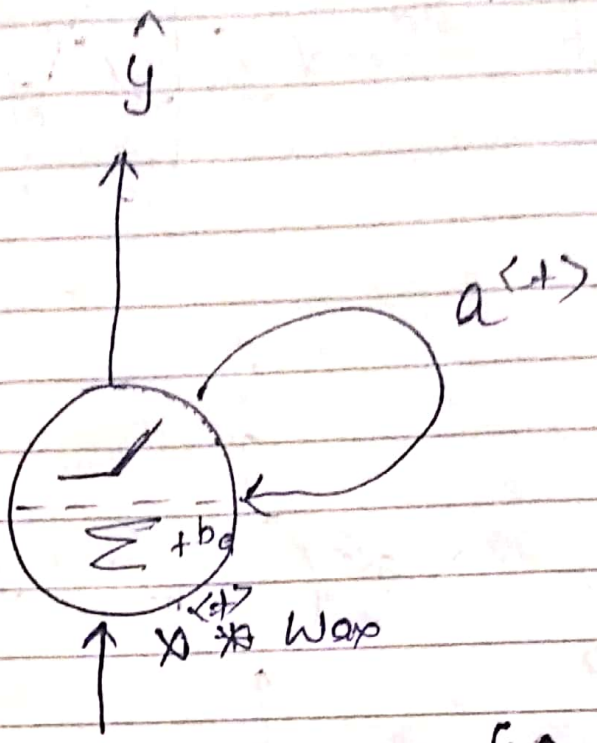
How we represent Input.

Mango is red colour.

1	0	0	0
0	0	0	0
0	0	0	1
0	1	0	0
0	0	1	0
0	0	0	0

10 × 4 → Sentence_len

Vocabulary_len



Here (10) Ke 4 loops chalengy

Mango is red colour. [10 × 4].

vocab dimstep.

1. X_input = [batch_size, vocab_size, sequence/dimestep]

2. W_aa = [hidden_unit, hidden_unit]

3. W_ax = [batch_unit, hidden_unit]

4. W_gy = [hidden unit, output_unit]

5. b_a = [hidden_unit, 1]

6. by = [output_unit, 1]

You can say that number of vocab = no. of hidden units.

7. yt_pred = [output_unit, vocab_size, timestep]

8. a_prev = [hidden_unit, vocab_size]

In our example we have 3 sentence.

1. Mango is red color.
2. Banana is yellow color.
3. Cat is black pool.

Vocab_size = [10] # Unique words.

b_a = (2 x 1) b_y = (1 x 1)

x_input = (1 x 10 x 4).

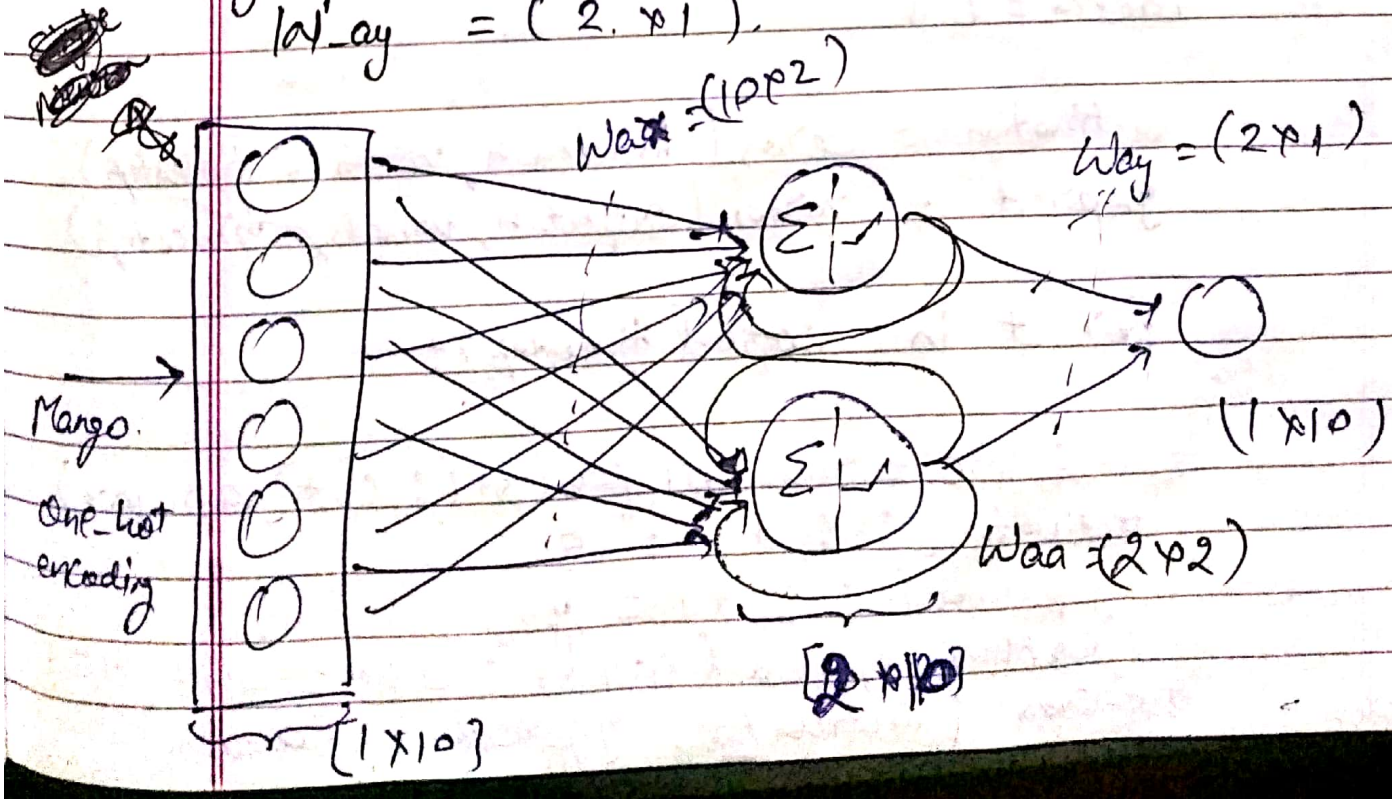
W_ax = (10 x 2)

a_prev ⇒ (2 x 10) ⇐ anext

W_aa = (2 x 2)

yt_pred = (1, 10 x 4)

W_ay = (2 x 1).



t - timesteps

$$a^{next} = \tanh(\text{dot}(W_{aa}.T, a_{prev}) + \text{dot}(W_{ax}.T, x_t) + b_a)$$

$$y\_cap = \text{softmax}(\text{dot}(W_{ay}.T, a^{next}) + b_y)$$

```
def rnn_cell_forward(xt, aprev, parameter):

    anext = tanh(xt, aprev, parameter)
    y_cap = softmax(anext)

    cache = (aprev, xt, anext, parameters)

    return anext, y_cap, cache

def rnn_forward(xt, a_prev, parameter):

    caches = []

    activation = zeros(hidden_n, vocab, timestep)
    y_pred = zeros(output_n, vocab, timestep)

    for t in range(timestep):

        a, y, c = rnn_cell_fow(xt[:, :, t], ap, par)
        activation[:, :, t] = a
        y_pred[:, :, t] = y
        caches.append(cache); caches = (caches, xt)
    return (activation, y_pred, caches)
```

Steps In forward propagation we dont have, x, and a_{prev}.

$$a_{next} = \tanh(dot(W_{aa}, a_{prev}) + dot(W_{ax}, x) + b_a)$$

$$y\_cap = \text{Softmax}(dot(W_{ay}, a_{next}) + b_y).$$

Back propagation through time

Weight are same for all timesteps in forward propagation.

Cost function $J^{<t>}(\theta) = -\sum_{j=1}^{|M|} y_{t,j} \log \hat{y}_{t,j}$

$$J(\theta) = -\frac{1}{T} \sum_{t=1}^{T} \sum_{j=1}^{|M|} y_{t,j} \log \hat{y}_{t,j}$$

where M = Vocabulary $J(\theta)$ = cost function
In, our example we have 4 time steps.

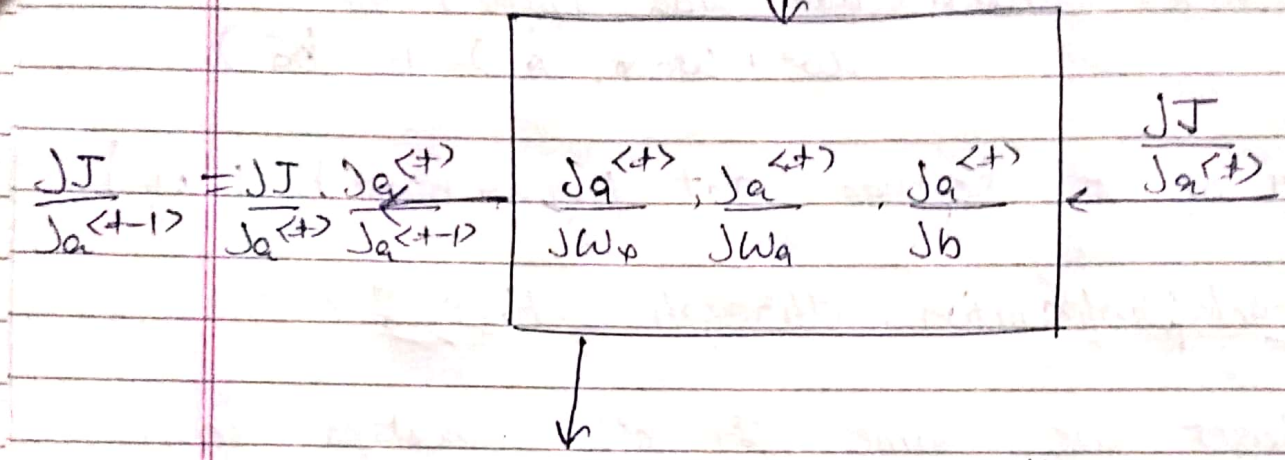
$$J(\theta) = -\frac{1}{4}\left[\sum_{j=1}^{10} y_{0,j} \log \hat{y}_{0,j} + \sum_{j=1}^{10} y_{1,j} \log \hat{y}_{1,j} + \sum_{j=1}^{10} y_{2,j} \log \hat{y}_{2,j} + \sum_{j=1}^{10} y_{3,j} \log \hat{y}_{3,j}\right]$$

Actually, we are considering Average cost in all time steps.

Let first write all parameters

$\Rightarrow$ b_a, b_y, W_{ap}, W_{aa}, W_{ay}

cache = ($a^{\langle t \rangle}$, $a^{\langle t-1 \rangle}$, $x^{\langle t \rangle}$, parameters)



$$\frac{\partial J}{\partial x^{\langle t \rangle}} = \frac{\partial J}{\partial a^{\langle t \rangle}} \cdot \frac{\partial a^{\langle t \rangle}}{\partial x^{\langle t \rangle}}$$

We have x.

tanh(—)

tanh(—)

$$a^{\langle t \rangle} = a_{next} = \tanh(W_p \cdot x + W_a \cdot a_{prev} + b_a)$$

$$y\_cap = \text{softmax}(W_y \cdot a_{next} + b_y),$$

$$J = -\frac{1}{T} \sum_{i=1}^{\|m\|} \sum_{j=1}^{T} Y_{j,i} \cdot \log(y\_cap_{j,i})$$

and We have to find

$$\frac{\partial J}{\partial \theta} = \frac{\partial J}{\partial J(W_y, b_y, W_p, W_a, b_a)} = \frac{\partial J}{\partial y\_cap} \cdot \frac{\partial y\_cap}{\partial W_y} \quad \text{III}$$

$$\frac{\partial J}{\partial y\_cap} \cdot \frac{\partial y\_cap}{\partial b_y} \quad \text{II}$$

$$\times \frac{\partial J}{\partial y\_cap} \cdot \frac{\partial y\_cap}{\partial a_{next}} \cdot \frac{\partial a_{next}}{\partial W_p, W_a, b_a} \quad \text{I}$$

$$a^{\langle t \rangle} = \tanh\left(W_{ax} x^{\langle t \rangle} + W_{aa} a^{\langle t-1 \rangle} + b\right)$$

$$\frac{\partial \tanh(u)}{\partial u} = \frac{\partial}{\partial u}\left[\frac{e^{u} - e^{-u}}{e^{u} + e^{-u}}\right]$$

$$= \left[\frac{(e^{u} + e^{-u})^2 - (e^{u} - e^{-u})^2}{(e^{u} + e^{-u})^2}\right]$$

$$= 1 - \left(\frac{e^{u} - e^{-u}}{e^{u} + e^{-u}}\right)^2$$

$$\frac{\partial \tanh(u)}{\partial u} = 1 - \tanh(u)^2$$

1. $$\frac{\partial a^{\langle t \rangle}}{\partial W_{ax}} = \left[1 - \tanh\left(W_{ax} x^{\langle t \rangle} + W_{a} a^{\langle t-1 \rangle} + b\right)^2\right] x^{\langle t \rangle T}$$

2. $$\frac{\partial a^{\langle t \rangle}}{\partial W_{aa}} = \left[1 - \tanh\left(W_{ax} x^{\langle t \rangle} + W_{aa} x^{\langle t-1 \rangle} + b\right)^2\right] x^{\langle t-1 \rangle T}$$

3. $$\frac{\partial a^{\langle t \rangle}}{\partial b_a} = \sum_{\text{Batch}}\left(1 - \tanh\left(W_{ax} x^{\langle t \rangle} + W_{aa} x^{\langle t-1 \rangle} + b\right)^2\right)$$

Is it over? no! Why because a_{prev} is also contain some wights from previous activations.

4. $$\frac{\partial a^{\langle t \rangle}}{\partial x^{\langle t \rangle}} = W_{ax}^{T}\left(1 - \tanh\left(W_{ax} x^{\langle t \rangle} + W_{aa} x^{\langle t-1 \rangle} + b\right)^2\right)$$

5. $\frac{\partial a^{\langle t \rangle}}{\partial a^{\langle t-1 \rangle}} = W_{aa}^{T} \cdot \left(1 - \tanh\left(W_{ax} x^{\langle t \rangle} + W_{aa} a^{\langle t-1 \rangle} + b\right)^2\right)$

dtanh = [hidden_unit, vocab_size].

dWax = [batch_unit, Haiden_unit]

dWaa = [hidden_unit, Haiden_unit]

dba = [hiddenunit, 1]

dby = [output_unit, 1]

dx = [batch_unit, vocab_size

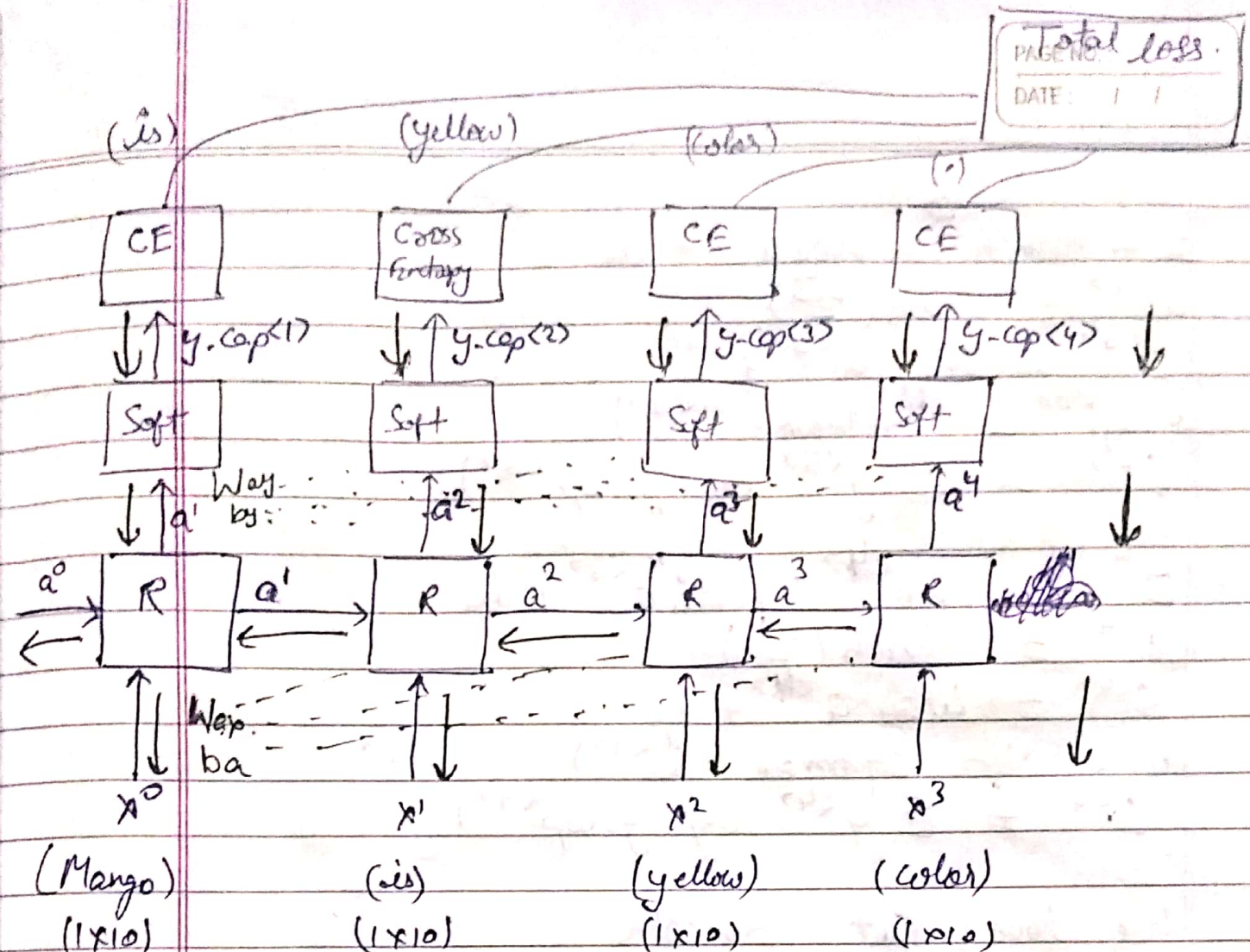
daprev = [hiddenunit, vocab_size]

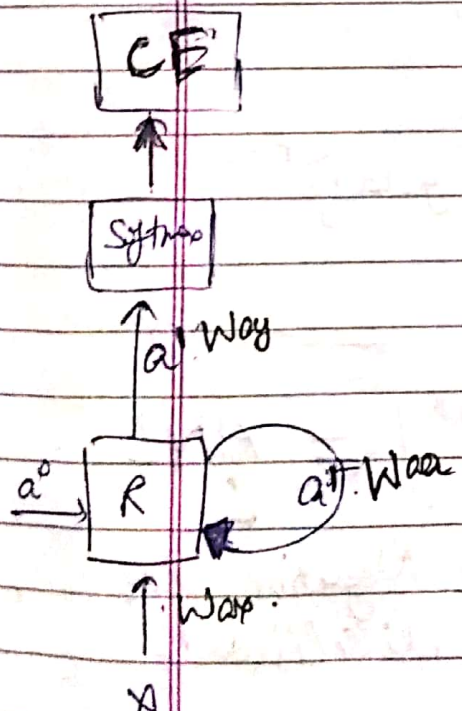
loss for single time step. [for BiClassification]

$L^{\langle T \rangle}(\hat{y}^{\langle t \rangle}, y^{\langle t \rangle}) = -y^{\langle t \rangle} \log \hat{y}^{\langle t \rangle} - (1 - y^{\langle t \rangle}) \log (1 - y^{\langle t \rangle})$

Loss for all time step. [for Multiclassification]

$$L(\hat{y}, y) = \sum_{t=1}^{T} L^{\langle t \rangle}(\hat{y}^{\langle t \rangle}, y^{\langle t \rangle})$$





$$z^{<1>} = W_{ax} x^{<1>} + W_{aa} a^{<0>} + b_a$$
$$a^{<1>} = \tanh(z^{<1>})$$
$$s^{<1>} = W_{ay} a^{<1>} + b_y$$
$$y\_cap^{<1>} = \text{Softmax}(s^{<1>})$$
$$L^{<1>} = \sum y^{<1>} \log(y\_cap^{<1>})$$

$$z^{<2>} = W_{ax} x^{<2>} + W_{aa} a^{<1>} + b_a$$
$$a^{<2>} = \tanh(z^{<2>})$$
$$s^{<2>} = W_{ay} \cdot a^{<1>} + b_y$$
$$y\_cap^{<2>} = \text{Softmax}(s^{<2>})$$
$$L^{<2>} = \sum y^{<2>} \log(y\_cap^{<2>})$$

$$z^3 = W_{ax} x^{<3>} + W_{aa} a^{<2>} + b_a$$
$$a^{<3>} = \tanh(z^3)$$
$$s^{<3>} = W_{ay} a^{<3>} + b_y$$
$$y\_cap^{<3>} = \text{Softmax}(s^{<3>})$$
$$L^{<3>} = \sum y^{<3>} \cdot \log(y\_cap^{<3>})$$

$$z^4 = W_{ax} x^{<4>} + W_{aa} a^{<3>} + b_a$$
$$a^{<4>} = \tanh(z^4)$$
$$s^{<4>} = W_{ay} a^{<4>} + b_y$$
$$y\_cap^{<4>} = \text{Softmax}(s^{<4>})$$
$$L^{<4>} = \sum y^{<4>} \cdot \log(y\_cap^{<4>})$$

We have cost function

$$J^t(\theta) = -\sum_{j=1}^{\|m\|} y_j \cdot \log(y\_cap_j)$$

$$J(\theta) = -\frac{1}{T} \sum_{t=1}^{T} \sum_{j=1}^{|m|} y_j^{<T>} \cdot \log(y\_cap_j^{<T>})$$

We have,

$$-\frac{1}{4}\left[L^{<1>} + L^{<2>} + L^{<3>} + L^{<4>} \right] \Leftarrow ANL$$

↓ Average Negative log likelihood.

We know that,

$$\frac{\partial L}{\partial A} = A - y, \quad \frac{\partial L}{\partial \theta} = (A - y) A'$$

from NN Notes.

Here,

(i) $\frac{\partial L^{\langle 4 \rangle}}{\partial W_{ay}} = [y\text{-}cap^{\langle 4 \rangle} - y^{\langle 4 \rangle}] \cdot a^{\langle 4 \rangle}$

from here we can clearly see that $\frac{\partial L}{\partial W_{ay}}$, only depends on the values at the current time steps. $y\text{-}cap^{\langle 4 \rangle}$, $y^{\langle 4 \rangle}$, $a^{\langle 4 \rangle}$. If we have these calculate the gradient of Way a simple matrix multiplication

$$dWay = \frac{\partial L^{\langle 4 \rangle}}{\partial W_{ay}} + \frac{\partial L^{\langle 3 \rangle}}{\partial W_{ay}} + \frac{\partial L^{\langle 2 \rangle}}{\partial W_{ay}} + \frac{\partial L^{\langle 1 \rangle}}{\partial W_{ay}}.$$

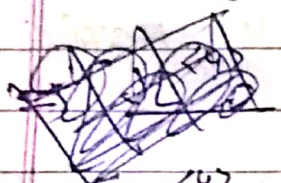
This is change in $L^{\langle t \rangle}$ w.r.t Way at each time steps.

(ii) $\frac{\partial L^{\langle 4 \rangle}}{\partial W_{aa}} = \frac{\partial L^{\langle 4 \rangle}}{\partial s^{\langle 4 \rangle}} \cdot \frac{\partial s^{\langle 4 \rangle}}{\partial a^{\langle 4 \rangle}} \cdot \frac{\partial a^{\langle 4 \rangle}}{\partial z^{\langle 4 \rangle}} \cdot \frac{\partial z^{\langle 4 \rangle}}{\partial W_{aa}}$.

Now, note that we can't treat $a^{\langle 3 \rangle}$ as an constant because it depends on $a^{\langle 2 \rangle}$ which also depends on, $a^{\langle 1 \rangle}$, $a^{\langle 0 \rangle}$. then what we have,

$$\frac{\partial L^{\langle 4 \rangle}}{\partial W_{aa}} = \sum_{K=1}^{4} \frac{\partial L^{\langle 4 \rangle}}{\partial a^{\langle 4 \rangle}} \frac{\partial a^{\langle 4 \rangle}}{\partial a^{\langle K \rangle}} \frac{\partial a^{\langle K \rangle}}{\partial W_{aa}}$$

At $t = 4$.



$$= \left[\frac{\partial L^{\langle 4 \rangle}}{\partial a^{\langle 4 \rangle}} \cdot \frac{\partial a^{\langle 4 \rangle}}{\partial a^{\langle 1 \rangle}} \cdot \frac{\partial a^{\langle 1 \rangle}}{\partial W_{aa}} + \frac{\partial L^{\langle 4 \rangle}}{\partial a^{\langle 4 \rangle}} \cdot \frac{\partial a^{\langle 4 \rangle}}{\partial a^{\langle 2 \rangle}} \cdot \frac{\partial a^{\langle 2 \rangle}}{\partial W_{aa}} \right.$$

$$\left. + \frac{\partial L^{\langle 4 \rangle}}{\partial a^{\langle 4 \rangle}} \cdot \frac{\partial a^{\langle 4 \rangle}}{\partial a^{\langle 3 \rangle}} \cdot \frac{\partial a^{\langle 3 \rangle}}{\partial W_{aa}} + \frac{\partial L^{\langle 4 \rangle}}{\partial a^{\langle 4 \rangle}} \cdot \frac{\partial a^{\langle 4 \rangle}}{W_{aa}} \cdot \right]$$

$K=2$ $$= \left[\frac{\partial L^{\langle 4 \rangle}}{\partial a^{\langle 4 \rangle}} \cdot \frac{\partial a^{\langle 4 \rangle}}{\partial a^{\langle 3 \rangle}} \cdot \frac{\partial a^{\langle 3 \rangle}}{\partial a^{\langle 2 \rangle}} \cdot \frac{\partial a^{\langle 2 \rangle}}{\partial W_{aa}} + \right.$$

$K=1$ $$\frac{\partial L^{\langle 4 \rangle}}{\partial a^{\langle 4 \rangle}} \cdot \frac{\partial a^{\langle 4 \rangle}}{\partial a^{\langle 3 \rangle}} \cdot \frac{\partial a^{\langle 3 \rangle}}{\partial a^{\langle 2 \rangle}} \cdot \frac{\partial a^{\langle 2 \rangle}}{\partial a^{\langle 1 \rangle}} \cdot \frac{\partial a^{\langle 1 \rangle}}{\partial W_{aa}} +$$

$K=3$ $$\frac{\partial L^{\langle 4 \rangle}}{\partial a^{\langle 4 \rangle}} \cdot \frac{\partial a^{\langle 4 \rangle}}{\partial a^{\langle 3 \rangle}} \cdot \frac{\partial a^{\langle 3 \rangle}}{\partial W_{aa}} +$$

$K=4$ $$\left. \frac{\partial L^{\langle 4 \rangle}}{\partial a^{\langle 4 \rangle}} \cdot \frac{\partial a^{\langle 4 \rangle}}{\partial W_{aa}} \right]$$

We sum up the contributions of each time step to the gradient. because W_{aa} is used in every step up to the output we care about, we need to backpropagate gradients from $t=4$ through the network all the way to $t=0$.

$$dW_{aa} = \frac{\partial L^{\langle 4 \rangle}}{\partial W_{aa}} + \frac{\partial L^{\langle 3 \rangle}}{\partial W_{aa}} + \frac{\partial L^{\langle 2 \rangle}}{\partial W_{aa}} + \frac{\partial L^{\langle 1 \rangle}}{\partial W_{aa}}$$

$$\star\star \Rightarrow \frac{\partial L^{\langle T \rangle}}{\partial W_{aa}} = \sum_{k=1}^{T} \frac{\partial L^{\langle T \rangle}}{\partial a^{\langle T \rangle}} \left(\prod_{j=k+1}^{T} \frac{\partial a^{\langle j \rangle}}{\partial a^{\langle j-1 \rangle}} \right) \cdot \frac{\partial a^{\langle k \rangle}}{\partial W_{aa}}$$

This is the reason of exploding gradient problem.

$\prod_{j=k+1}^{T} \frac{\partial a^{\langle j \rangle}}{\partial a^{\langle j-1 \rangle}}$ the derivative of a vector w.r.t vector, the result is a Jacobian Matrix.

This product can shrink to zero or explode to infinite

When it goes explode or Vanish?

let, λ_1 = largest singular value of W_{aa}.

If you use 'tanh'

If $\lambda < 1$: Vanishing gradient.

If $\lambda > 1$: Exploding gradient.

For sigmoid

If $\lambda < 0.25$: Vanishing gradient.

If $\lambda > 0.25$: Explode gradient.

One solution of this is Truncate backpropagate. Skip some timesteps.

1. ⇒ Using an L1 and L2 form penalty on recurrent weights can help with exploding gradient. The L1 and L2 term can ensure that during training λ_1 stay smaller than 1. and in this regime gradient can not explode. also, it help in vanishing problem. it ensure that output of sigmoid or any function lies in linear region i.e. its derivative also not too high or too low.

2. ★ Scaling down the gradient [clipping]

It reduce Exploding problem.

$$\hat{g} = \frac{\partial L}{\partial W_{ab}}$$

If $\|\hat{g}\| \geq$ threshold then

$$\hat{g} = \frac{\text{threshold}}{\|\hat{g}\|} \cdot \hat{g}$$

end if

What we do here, we Clipping the norm of the exploded gradient.

At $t = 2$.

$$\frac{\partial L^{<2>}}{\partial W_{aa}} = \left[\frac{\partial L^{<2>}}{\partial a^{<2>}} \cdot \frac{\partial a^{<2>}}{\partial a^{<1>}} \cdot \frac{\partial a^{<1>}}{\partial W_{aa}} + \frac{\partial L^{<2>}}{\partial a^{<2>}} \cdot \frac{\partial a^{<2>}}{\partial W_{aa}} \right].$$

(i) $$\frac{\partial L^{<t>}}{\partial a^{<t>}} = \frac{\partial L^{<t>}}{\partial S^{<t>}} \cdot \frac{\partial S^{<t>}}{\partial a^{<t>}}$$

$$= [y\text{-}cap^{<t>} - y^{<t>}] \cdot W_{ay}.$$

(ii) $$\frac{\partial a^{<t>}}{\partial a^{<t-1>}} = \left[1 - \tanh\left(W_{ax} x^{<t>} + W_{aa} a^{<t-1>} + b_a\right)^2\right] W_{aa}.$$

(iii) $$\frac{\partial a^{<t>}}{\partial W_{aa}} = \left[1 - \tanh\left(W_{ax} x^{<t>} + W_{aa} a^{<t-1>} + b_a\right)^2\right] a^{<t-1>}.$$

(iii) $$\frac{\partial L^{<4>}}{\partial W_{ax}} = \frac{\partial L^{<4>}}{\partial a^{<4>}} \cdot \frac{\partial a^{<4>}}{\partial W_{ax}}$$

$$= (y\text{-}cap^{<4>} - y^{<4>}) \cdot W_{ay} \cdot x^{<4>}.$$

Here, also we cannot treat $x^{<t>}$ as an constant because it varies at each time step.

$$\frac{\partial L^{\langle 4\rangle}}{\partial W_{ax}} = \sum_{k=1}^{4} \frac{\partial L^{\langle 4\rangle}}{\partial a^{\langle 4\rangle}} \cdot \frac{\partial a^{\langle 4\rangle}}{\partial a^{k}} \cdot \frac{\partial a^{k}}{\partial W_{ap}}$$

At $t = 2$

$$\frac{\partial L^{\langle 2\rangle}}{\partial W_{ax}} = \sum_{k=1}^{2} \frac{\partial L^{\langle 2\rangle}}{\partial a^{\langle 2\rangle}} \cdot \frac{\partial a^{\langle 2\rangle}}{\partial a^{\langle k\rangle}} \cdot \frac{\partial a^{k}}{\partial W_{aa}}$$

$$= \left[\frac{\partial L^{\langle 2\rangle}}{\partial a^{\langle 2\rangle}} \cdot \frac{\partial a^{\langle 2\rangle}}{\partial a^{\langle 1\rangle}} \cdot \frac{\partial a^{\langle 1\rangle}}{\partial W_{ax}} + \frac{\partial L^{\langle 2\rangle}}{\partial a^{\langle 2\rangle}} \cdot \frac{\partial a^{\langle 2\rangle}}{\partial W_{aa}} \right]$$

(i) $$\frac{\partial a^{\langle t\rangle}}{\partial W_{ax}} = \left[1 - \tanh\left(W_{ap} x^{\langle t\rangle} + W_{aa} a^{\langle t-1\rangle} + b_a\right)^2\right] x^{\langle t\rangle}$$

$dW_{ap} \Rightarrow$ $$\frac{\partial L}{\partial W_{ap}} = \frac{\partial L^{\langle 1\rangle}}{\partial W_{ap}} + \frac{\partial L^{\langle 2\rangle}}{\partial W_{ap}} + \frac{\partial L^{\langle 3\rangle}}{\partial W_{ap}} + \frac{\partial L^{\langle 4\rangle}}{\partial W_{ap}}$$

$$\frac{\partial L^{\langle T\rangle}}{\partial W_{ap}} = \sum_{k=1}^{T} \frac{\partial L^{\langle T\rangle}}{\partial a^{\langle T\rangle}} \left(\prod_{j=k+1}^{T} \frac{\partial a^{\langle j\rangle}}{\partial a^{\langle j-1\rangle}} \right) \cdot \frac{\partial a^{\langle k\rangle}}{\partial W_{aa}}$$

(iv) $$\frac{\partial L^{\langle 4\rangle}}{\partial b_a} = \frac{\partial L^{\langle 4\rangle}}{\partial a^{\langle 4\rangle}} \cdot \frac{\partial a^{\langle 4\rangle}}{\partial b_a}$$

$$= \left[\hat{y}^{\langle 4\rangle} - y^{\langle 4\rangle}\right] \cdot W_{ay} \cdot \frac{\partial a^{\langle 4\rangle}}{\partial b_a}$$

$$\frac{\partial a^{\langle 4 \rangle}}{\partial ba} = \mathcal{E}\left[1 - \tanh\left(W_{aa} x^{\langle 4 \rangle} + W_{aa} a^{\langle 3 \rangle} + ba\right)^2\right].$$

$$dba = \frac{\partial L^{\langle 1 \rangle}}{\partial ba} + \frac{\partial L^{\langle 2 \rangle}}{\partial ba} + \frac{\partial L^{\langle 3 \rangle}}{\partial ba} + \frac{\partial L^{\langle 4 \rangle}}{\partial ba}.$$

(V) $$\frac{\partial L^{\langle 4 \rangle}}{\partial by} = \frac{\partial L^{\langle 4 \rangle}}{\partial S^{\langle 4 \rangle}} \cdot \frac{\partial S^{\langle 4 \rangle}}{\partial by}$$

$$= \left[y\_cap^{\langle 4 \rangle} - y^{\langle 4 \rangle}\right].$$

$$dby = \sum_{i=0}^{T} \frac{\partial L^{\langle i \rangle}}{\partial ba}.$$

(VI) $$\frac{\partial L^{\langle 4 \rangle}}{\partial x^{\langle 4 \rangle}} = \frac{\partial L^{\langle 4 \rangle}}{\partial a^{\langle 4 \rangle}} \cdot \frac{\partial a^{\langle 4 \rangle}}{\partial x^{\langle 4 \rangle}}.$$

When we dealing with one to many structure then input is dependent on previous timestep output's.

dxt ⇒ $$\frac{\partial L^{\langle T \rangle}}{\partial x} = \frac{\partial L^{\langle 1 \rangle}}{\partial x^{\langle 1 \rangle}} + \frac{\partial L^{\langle 2 \rangle}}{\partial x^{\langle 2 \rangle}} + \frac{\partial L^{\langle 3 \rangle}}{\partial x^{\langle 3 \rangle}} + \frac{\partial L^{\langle 4 \rangle}}{\partial x^{\langle 4 \rangle}}$$

$$\frac{\partial L^{\langle T \rangle}}{\partial x} = \sum_{K=1}^{T} \frac{\partial L^{\langle T \rangle}}{\partial a^{\langle T \rangle}} \left(\prod_{j=k+1}^{T} \frac{\partial a^{\langle j \rangle}}{\partial a^{\langle j-1 \rangle}}\right) \cdot \frac{\partial a^{\langle k \rangle}}{\partial x^{\langle k \rangle}}$$

(vii) $$\frac{\partial L^{<4>}}{\partial a\_o} = \frac{\partial L^{<4>}}{\partial a^{<4>}} \cdot \frac{\partial a^{<4>}}{\partial a^{<3>}} \cdot \frac{\partial a^{<3>}}{\partial a^{<2>}} \cdot \frac{\partial a^{<2>}}{\partial a^{<1>}}$$

We also need to get optimal value of a_prev or a_0 because we initialize it as random.

$$da0 = \frac{\partial L^{<1>}}{\partial a\_o} + \frac{\partial L^{<2>}}{\partial a\_o} + \frac{\partial L^{<3>}}{\partial o\_o} + \frac{\partial L^{<4>}}{\partial a\_o}$$

$$\frac{\partial L^{<1>}}{\partial a\_o} = \frac{\partial L^{<1>}}{\partial a^{<1>}} \cdot \frac{\partial a^{<1>}}{\partial a^{<0>}}$$

$$\frac{\partial L^{<1>}}{\partial a^{<1>}} = [y\_cap^{<1>} - y^{<1>}] \cdot W_{ay}$$

$$\frac{\partial a^{<1>}}{\partial a^{<0>}} = [1 - \tanh(W_{aax} x^{<1>} + W_{aa} \cdot a^{<0>} + b_a)]^2 W_{aa}$$