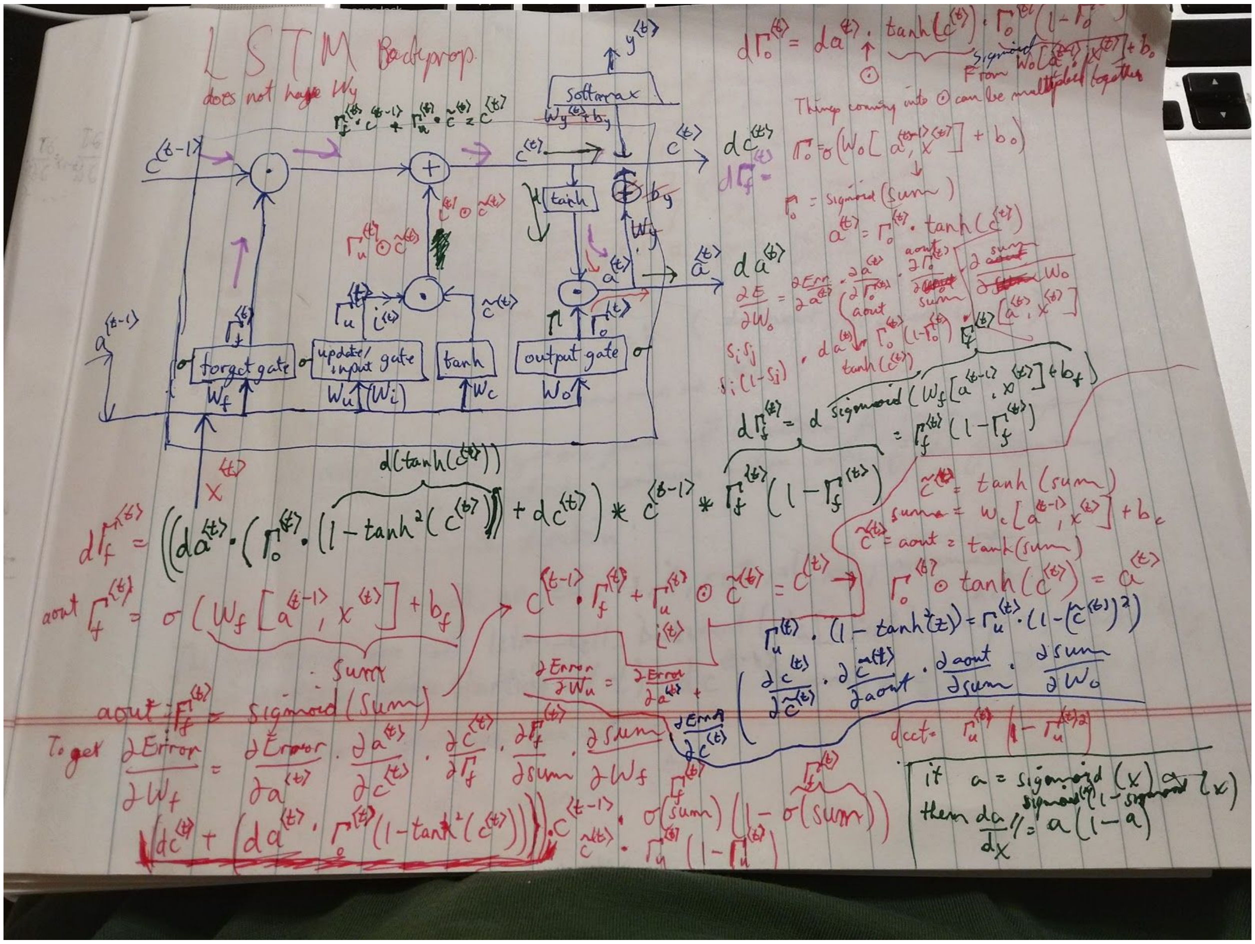
Understanding Backpropagation as Applied to LSTM

By Abraham Kang and Jae Duk Seo with special thanks to Kunal Patel and Mohammad-Mahdi Moazzami for reviewing the this paper and providing feedback.

Backpropagation is one of those topics that seem to confuse many (except for in straightforward cases such as feed-forward neural networks). If you try to Google the topic you get a number of articles that make a valiant effort to explain the area but sometimes the notation and pictures are confusing because of the way they orient the inputs and outputs or variable names being represented differently. Even the representation of things like forking within LSTM is assumed and omitted from discussion. (https://towardsdatascience.com/back-to-basics-deriving-back-propagation-on-simple-rnn-lstm-feat-aidan-gomez-c7f286ba973d, https://medium.com/@aidangomez/let-s-do-this-f9b699de31d9, http://www.wildml.com/2015/10/recurrent-neural-networks-tutorial-part-3-backpropagation-through-time-and-vanishing-gradients/, https://arxiv.org/abs/1610.02583, https://machinelearningmastery.com/gentle-introduction-backpropagation-time/, and https://www.coursera.org/lecture/nlp-sequence-models/backpropagation-through-time-bc7ED). We are going to explain backpropagation through an LSTM RNN in a different way. We want to give you general principles for deciphering backpropagation through any neural network. Then apply those principles to LSTM (Long-Short Term Memory) RNNs (Recurrent Neural Networks). Let’s start with the notes we took to figure this stuff out.



Just kidding. This is just our thought process. We will make it easier.

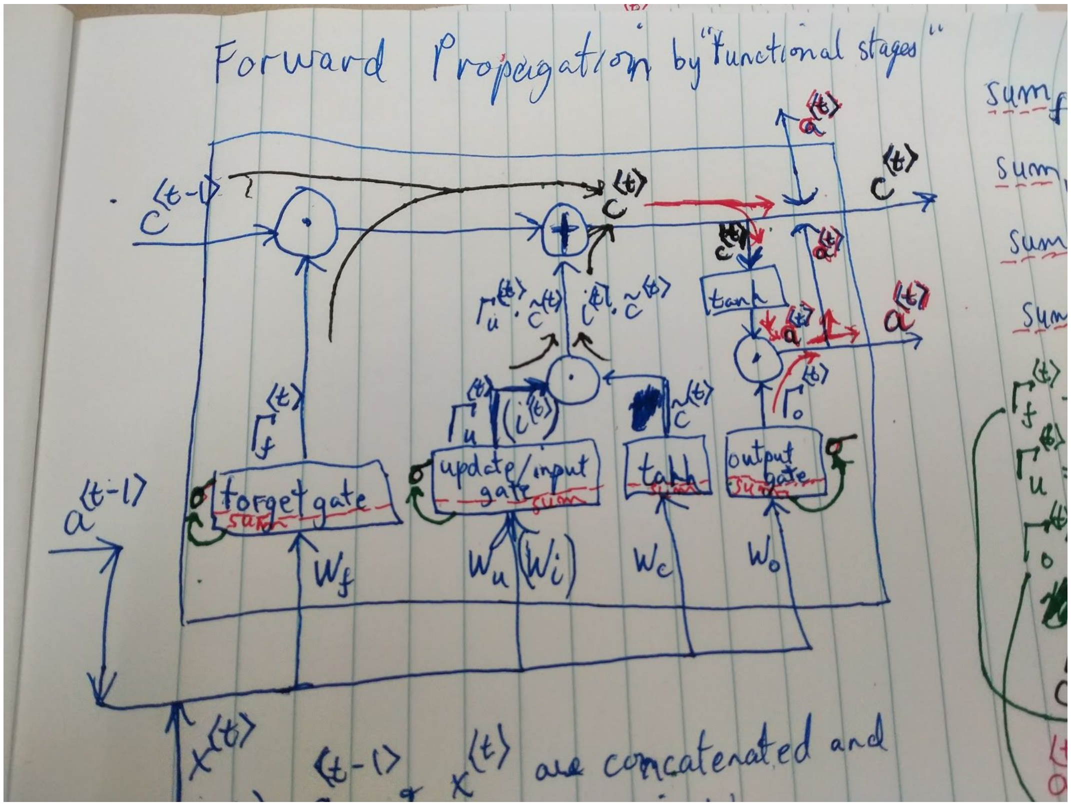
If you haven’t read Matt Mazur’s excellent A Step by Step Backpropagation Example please do so before continuing. It is still one of the best explanations of backpropagation out there and it will make everything we talk about seem more familiar.

There are 3 principles that you have to understand in order to comprehend back propagation in most neural networks: forward propagation in functional stages, the chain rule, and working in reverse through the functional stages and forks using partial derivatives.

Forward Propagation in Functional Stages

When you look at a neural network, the inputs are passed through functional stages to become outputs. Those outputs become inputs to the next functional stage and turn into outputs. This continues until the final output is the result at the end of the neural network. A functional stage is an operation that takes some input and transforms it into some output at the most granular level. Some examples of functional stages are the summing of inputs multiplied by weights, convolution operation (summing over a structural subset of the input where the patch values that constitute the weights are multiplied against a subset of the input having the same shape as the patch), passing input through a activation function, max pooling, softmaxing, loss calculation, etc. In order to work through back propagation, you need to first be aware of all functional stages that are a part of forward propagation.

Let's look at LSTM.



Several things to note when looking at this picture. Everything coming in from left is from the prior LSTM block (<t-1>). Everything from bottom, middle, top and right is with respect to the current time/block interval (<t>). Some articles reference LSTM with an update gate while others call it an input gate. I put both in the picture but for the calculations, I chose the uppercase gamma Γ notation to represent the output of each of the gates and chose update gate instead of input gate for reference in my calculations.

$a^{<t-1>}$ is the output and $c^{<t-1>}$ is the cell state from the previous block coming in on the left side of the block. $a^{<t>}$ is your output and $c^{<t>}$ is your resultant cell state on the right side of the block (which is passed to the next block). The sum at the bottom of the boxes (separated by a red dotted line) is the result of summing $a^{<t-1>}$ (previous output) concatenated with $x^{<t>}$ (current input) dot producted (numpy.dot) with W_f, W_u, W_c or W_o (depending on which gate you are solving). The result of the sum is passed to each gate's activation function. The forget, update (or input), and output gates have sigmoid as their activation function. $\Gamma_f^{<t>}$, $\Gamma_u^{<t>}$,$\tilde{c}^{<t>}$, $\Gamma_o^{<t>}$ are the outputs of their respective gates (after applying sigmoid or tanh to the sums). The c represents your cell state which passes along pieces of information about each input ($x^{<t>}$) to successive LSTM blocks.

So let’s start from the bottom. We take the previous output and concatenate it with the current input and dot product it with each W_x and add the bias.

$$sum_f = W_f[a^{\langle t-1\rangle}, x^{\langle t\rangle}] + b_f$$

$$sum_u = W_u[a^{\langle t-1\rangle}, x^{\langle t\rangle}] + b_u$$

$$sum_c = W_c[a^{\langle t-1\rangle}, x^{\langle t\rangle}] + b_c$$

$$sum_o = W_o[a^{\langle t-1\rangle}, x^{\langle t\rangle}] + b_o$$

The sums (shown below) will be passed to their respective activation functions to generate $\Gamma_f^{<t>}$, $\Gamma_u^{<t>}$, $\tilde{c}$, $\Gamma_o^{<t>}$

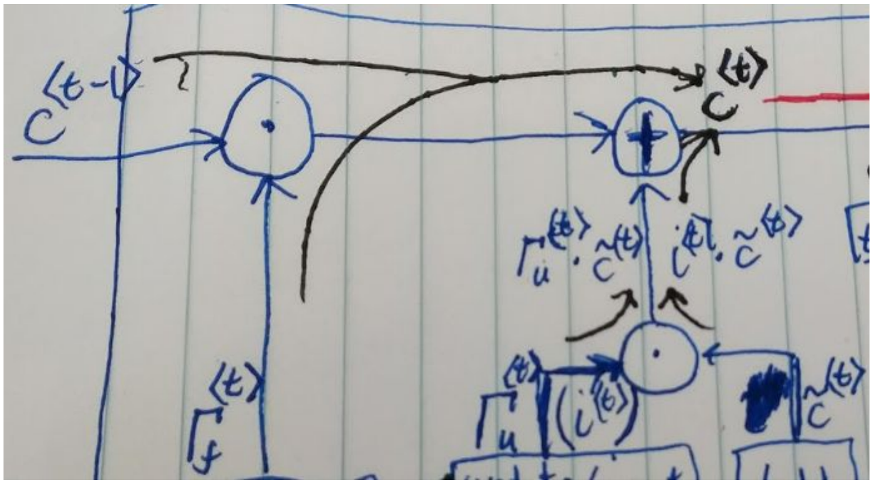
$$\Gamma_f^{\langle t\rangle} = sigmoid(sum_f)$$

$$\Gamma_u^{\langle t\rangle} = sigmoid(sum_u)$$

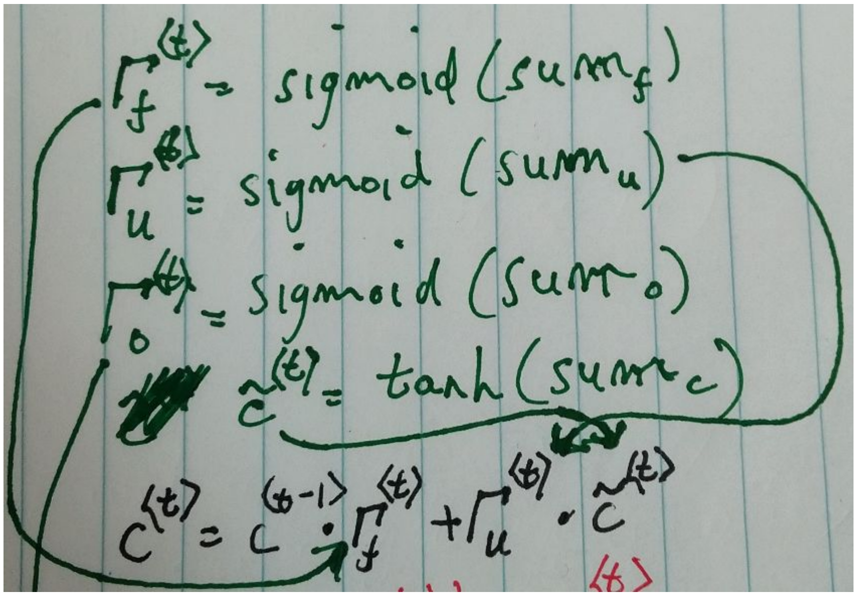
$$\Gamma_o^{\langle t\rangle} = sigmoid(sum_o)$$

$$\tilde{c}^{\langle t\rangle} = tanh(sum_c)$$

Now that we have calculated all of the gate output values, the next step is to calculate $c^{<t>}$. To calculate $c^{<t>}$ you will need to dot product $c^{<t-1>}$ with $\Gamma_f^{<t>}$ then add this to the dot product of $\Gamma_u^{<t>}$ with $\tilde{c}^t$.


$c^{\langle t-1\rangle}$
$c^{\langle t\rangle}$
$\cdot$
$+$
$\Gamma_u^{\langle t\rangle} \cdot \tilde{c}^{\langle t\rangle}$
$i^{\langle t\rangle} \cdot \tilde{c}^{\langle t\rangle}$
$\cdot$
$\Gamma_f^{\langle t\rangle}$
$\Gamma_u^{\langle t\rangle}$
$(i^{\langle t\rangle})$
$\tilde{c}^{\langle t\rangle}$

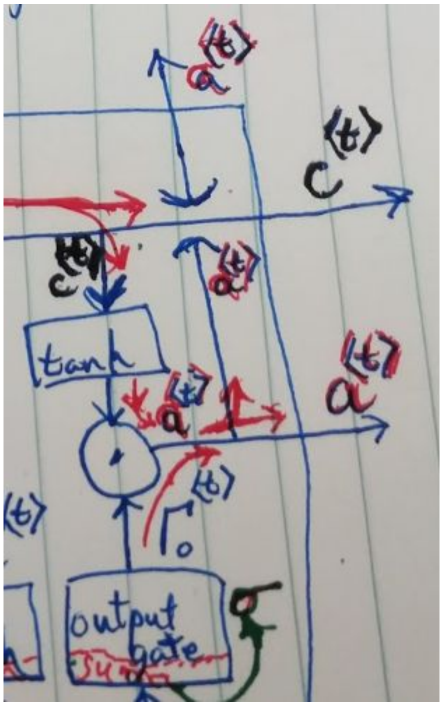
Here is what it looks like with all the variables added in.


$\Gamma_f^{\langle t\rangle}$ = sigmoid (sum$_f$)
$\Gamma_u^{\langle t\rangle}$ = sigmoid (sum$_u$)
$\Gamma_o^{\langle t\rangle}$ = sigmoid (sum$_o$)
$\tilde{c}^{\langle t\rangle}$ = tanh (sum$_c$)
$c^{\langle t\rangle} = c^{\langle t-1\rangle} \cdot \Gamma_f^{\langle t\rangle} + \Gamma_u^{\langle t\rangle} \cdot \tilde{c}^{\langle t\rangle}$
⟨t⟩

What you end up with is:

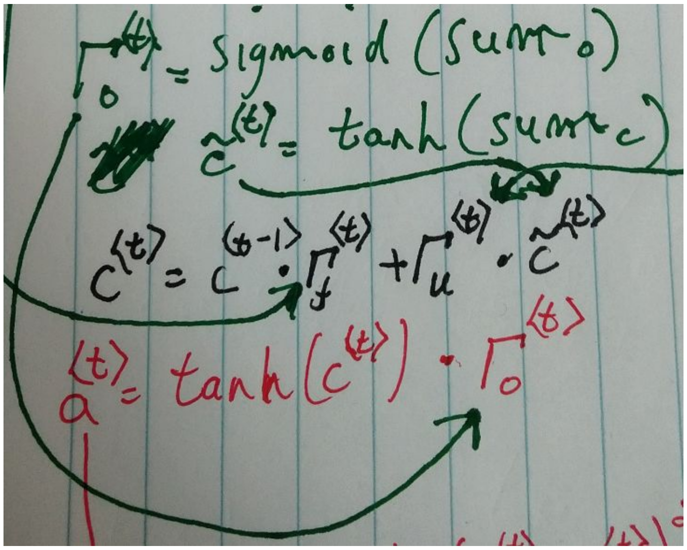
$$c^{<t>} = c^{<t-1>} \cdot \Gamma_f^{<t>} + \Gamma_u^{<t>} \cdot \tilde{c}^t$$

The next step is to calculate $a^{<t>}$.

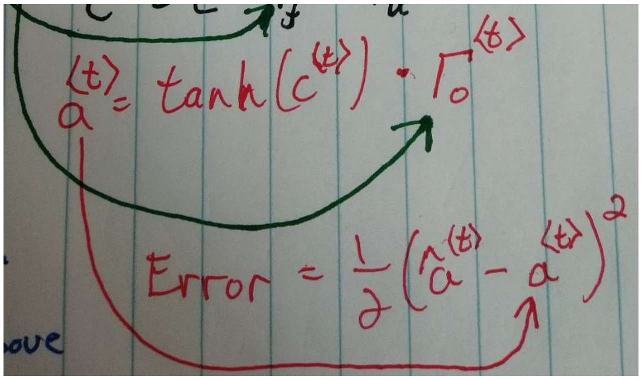


From the picture we see that $c^{<t>}$ is being passed into tanh($c^{<t>}$) and the result is multiplied with $\Gamma_o^{<t>}$ (from the bottom) giving:

$$a^{<t>} = \tanh(c^{<t>}) \cdot \Gamma_o^{<t>}$$


Γ_o^⟨t⟩ = Sigmoid(sum_o)
c̃^⟨t⟩ = tanh(sum_c)
c^⟨t⟩ = c^⟨t-1⟩ · Γ_f^⟨t⟩ + Γ_u^⟨t⟩ · c̃^⟨t⟩
a^⟨t⟩ = tanh(c^⟨t⟩) · Γ_o^⟨t⟩

Once we have $a^{<t>}$ (output) we can pass it to the error function and calculate our loss as well as pass it to the next LSTM block.



In order to be consistent with my formulas, $\hat{a}^{<t>}$ represents the correct value and $a^{<t>}$ is the predicted value from the neural network. Most papers will refer to ŷ as the predicted value and y as the correct value. I wanted to keep everything consistent with the formulas I had written so I switched which variable has the hat over it.

Once forward propagation is complete we need to use to the Chain Rule to back propagate through time.

The Chain Rule

There are two main parts to the Chain Rule. The first roughly states:

$$\frac{d}{dx}\left[f\Big(g(x)\Big)\right] = f'\Big(g(x)\Big)g'(x)$$

If you are given a function f = sin(2x + 2) and looking to take the derivative of it you can decompose it to parts such that f = sin() and g = 2x+2. Applying the chain rule:

$$\frac{df}{dx}\,\text{sin(2x + 2)}$$
$$= \frac{df}{dg}\;\text{sin(g)} \quad * \qquad \frac{dg}{dx}\,\text{2x+2}$$
$$= \text{cos(g)} \quad * \qquad 2$$

now plug in what g equals and you have the answer:

$$\frac{df}{dx} = \text{2cos(2x+2)}$$

Think of g as your input which is passed to your function f that produces output f.
If we pass f to another function e and it produces output e, we can calculate another differential:

$$\frac{de}{df}$$

The output is always on top (e above) and input on the bottom (f above). What we want to know is how a change in the bottom (f) will affect the top (e). Using all of the chained functions separately we could calculate $\frac{de}{dx}$ by multiplying the previous differentials together and cancel out neighboring denominator-numerator pairs:

$$\frac{de}{df} * \frac{df}{dg} * \frac{dg}{dx} = \frac{de}{dx}$$

If you review the forward propagation steps above, you will notice that they are also just a sequence of inputs feed to a function which results in an output that is then passed to another function which repeats the process until a final output and error are produced. This means we can use the Chain Rule to calculate the derivatives of the Error with respect to each weight.

Once we have these differentials we can use the standard update formula to correct our weights during backpropagation:

$$W^{*}_{x} = W^{old}_{x} - \alpha\left(\frac{\partial Error}{\partial Wx} \right)$$

Working in Reverse Through the Functional Stages and Forks Using Partial Derivatives

For review let's look at the functional stages and forks for W_f:

1. W_f is the input to a function which then outputs sum$_f$.
2. sum_f is passed to a sigmoid function and then is output as $\Gamma_f^{<t>}$.
3. $\Gamma_f^{<t>}$ is passed as input to a function which outputs $c^{<t>}$.
4. $c^{<t>}$ forks wherein one branch $c^{<t>}$ is passed to a function that outputs $a^{<t>}$ and another branch where $c^{<t>}$ is transferred to the next block and becomes $c^{<t-1>}$ in the subsequent block and is passed to a function that outputs $c^{<t>}$ in that subsequent block. This creates a differentiable function between the current state ($c^{<t>}$) and the next state ($c^{<t+1>}$).
5. $a^{<t>}$ forks with one branch being passed as input to a function which outputs the Error and the other branch where $a^{<t>}$ is transferred to the subsequent block as $a^{<t-1>}$.

An important thing to consider are the values of $c^{<t>}$ and $a^{<t>}$ at the extreme ends and edges of the LSTM blocks. For example, what are the values for $c^{<0>}$ and $a^{<0>}$ for the beginning block (at time interval=1)? Usually $a^{<0>}$ (previous output) for the first block are zeros but $c^{<0>}$ (the initial cell state) can be zeros or if your data includes many short sequences then training the initial state can accelerate learning (https://r2rt.com/non-zero-initial-states-for-recurrent-neural-networks.html) or if your data includes a small number of long sequences then using a noisy initial state can accelerate learning.

If you have a many (inputs) going to one (output) then $\frac{\partial Error^{<t>}}{\partial a^{<t>}}$ may equal 0 at intermediate LSTM blocks because there are no calculations for the error until the last LSTM block. So the calculation for the $a^{<t>}$ fork for intermediate blocks may look like the following in intermediate blocks:

$$\left(\frac{\partial Error^{<t>}}{\partial a^{<t>}} + \frac{\partial Error^{<t+1>}}{\partial a^{<t>}} \right) = \left(0 + \frac{\partial Error^{<t+1>}}{\partial a^{<t>}} \right)$$

If you have a one (input) to many outputs then the value for $\frac{\partial Error^{<t>}}{\partial a^{<t>}}$ may be non-zero for every LSTM block producing an output. Finally, what are the values to be assigned to the fork derivatives for $c^{<t>}$ and $a^{<t>}$ that are related to the next block when you are calculating block values at the last LSTM block (the answer is 0)?

Since the forward propagation chain that we have been working on above progresses from top to bottom. When we back propagate, we will work in reverse, starting from the bottom and working our way back up to the top while calculating the derivatives of the functional stages.

Let’s calculate the derivative of the Error with respect to W_f.

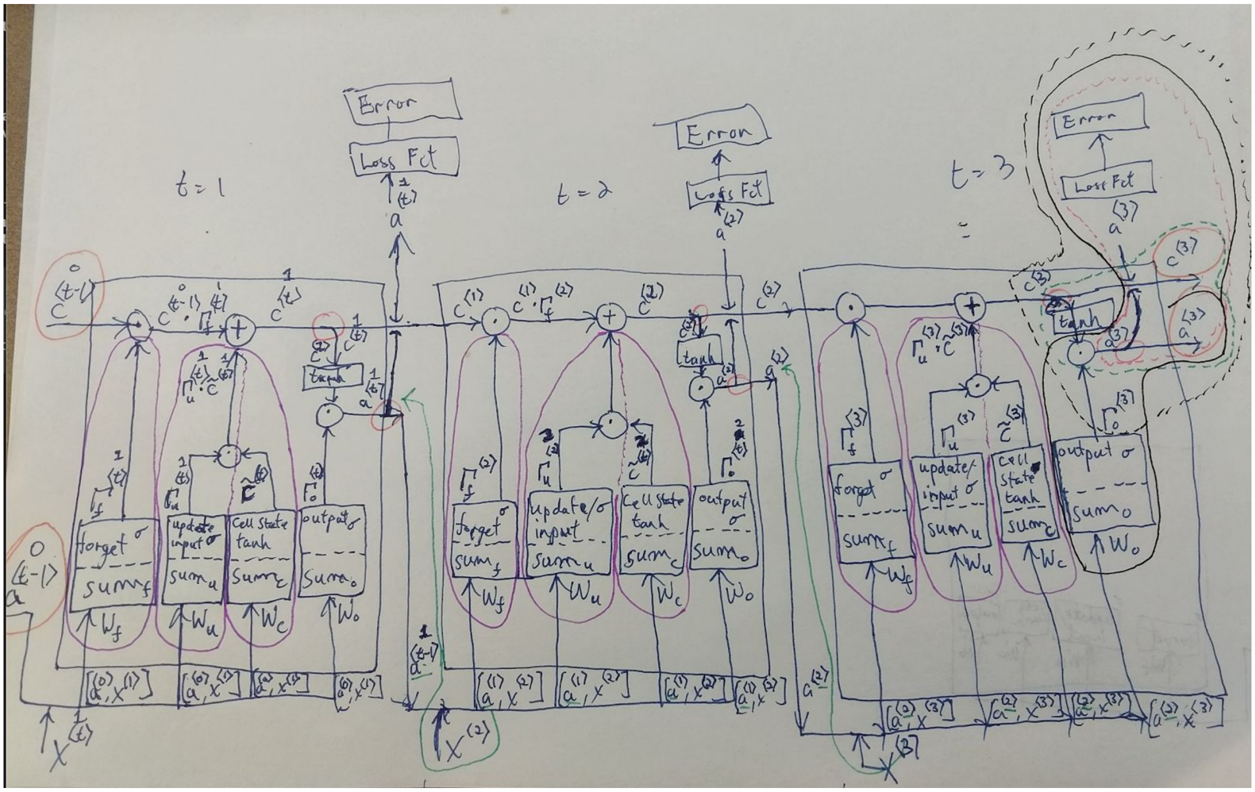
Remember that outputs to our functional stages are always on top of our representative differential part. Just so you know where we are going here is the final derivative of the Error with respect to W_f.

$$\frac{dError}{dW_f} = (((((\frac{\partial Error^{<t>}}{\partial a^{<t>}} + \frac{\partial Error^{<t+1>}}{\partial a^{<t>}}) * \frac{\partial a^{<t>}}{\partial c^{<t>}}) + \frac{\partial c^{<t+1>}}{\partial c^{<t>}}) * (\frac{\partial c^{<t>}}{\partial \Gamma_f^{<t>}} * \frac{\partial \Gamma_f^{<t>}}{\partial sum_f^{<t>}} * \frac{\partial sum_f^{<t>}}{\partial W_f})) +$$

$$((((\frac{\partial Error^{<t-1`>}}{\partial a^{<t-1>}} + \frac{\partial Error^{<t>}}{\partial a^{<t-1>}}) * \frac{\partial a^{<t-1>}}{\partial c^{<t-1>}}) + \frac{\partial c^{<t>}}{\partial c^{<t-1>}}) * (\frac{\partial c^{<t-1>}}{\partial \Gamma_f^{<t-1>}} * \boxed{\frac{\partial \Gamma_f^{<t-1>}}{\partial sum_f^{<t-1>}}} * \frac{\partial sum_f^{<t-1>}}{\partial W_f})) +$$

$$((((\frac{\partial Error^{<t-2>}}{\partial a^{<t-2>}} + \frac{\partial Error^{<t-1>}}{\partial a^{<t-2>}}) * \frac{\partial a^{<t-2>}}{\partial c^{<t-2>}}) + \frac{\partial c^{<t-1>}}{\partial c^{<t-2>}}) * (\frac{\partial c^{<t-2>}}{\partial \Gamma_f^{<t-2>}} * \frac{\partial \Gamma_f^{<t-2>}}{\partial sum_f^{<t-2>}} * \frac{\partial sum_f^{<t-2>}}{\partial W_f})))$$

At first, the formula above looks complicated but let’s look at the picture below and break things down step by step. Because there are 3 block intervals you would substitute t=3 everywhere in the three rows of the equation above. The three rows represent the three blocks intervals. We are working in reverse order from right to left in the picture below (because we are back propagating). So the first top row in the equation above is the 3rd block interval, the second row is the second block interval and the third bottom row of the equation above represents the first block interval.



Notice how each block is dependent on its future block interval when doing backpropagation. This is because we start from last block interval (rightmost block) and then move to the next previous block interval calculating

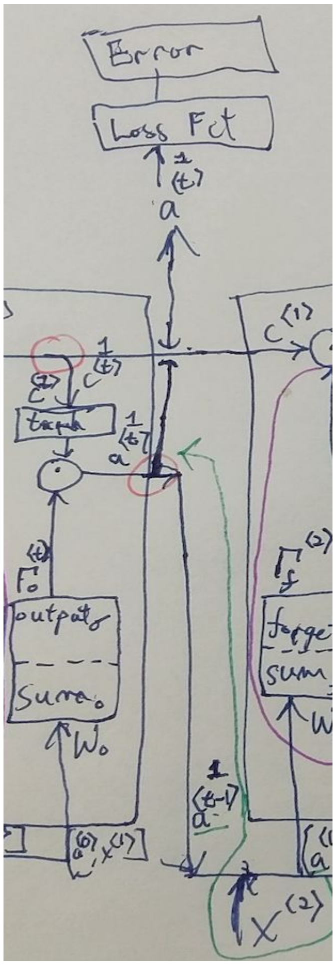
the gradients (derivatives) finally ending up at the first block interval (leftmost block). The two variables that are dependent between blocks are $c^{<t>}$ and $a^{<t>}$. $c^{<t>}$ has to go through a function before it become $c^{<t+1>}$.
However, $a^{<t>}$ is the same between blocks.

$$\frac{dError}{dW_f} = (((((\frac{\partial Error^{<t>}}{\partial a^{<t>}} + \frac{\partial Error^{<t+1>}}{\partial a^{<t>}}) * \frac{\partial a^{<t>}}{\partial c^{<t>}}) + \frac{\partial c^{<t+1>}}{\partial c^{<t>}}) * (\frac{\partial c^{<t>}}{\partial \Gamma_f^{<t>}} * \frac{\partial \Gamma_f^{<t>}}{\partial sum_f^{<t>}} * \frac{\partial sum_f^{<t>}}{\partial W_f})) +$$

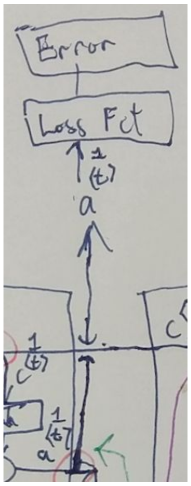
$$((((\frac{\partial Error^{<t-1'>}}{\partial a^{<t-1>}} + \frac{\partial Error^{<t>}}{\partial a^{<t-1>}}) * \frac{\partial a^{<t-1>}}{\partial c^{<t-1>}}) + \frac{\partial c^{<t>}}{\partial c^{<t-1>}}) * (\frac{\partial c^{<t-1>}}{\partial \Gamma_f^{<t-1>}} * \frac{\partial \Gamma_f^{<t-1>}}{\partial sum_f^{<t-1>}} * \frac{\partial sum_f^{<t-1>}}{\partial W_f})) +$$

$$((((\frac{\partial Error^{<t-2>}}{\partial a^{<t-2>}} + \frac{\partial Error^{<t-1>}}{\partial a^{<t-2>}}) * \frac{\partial a^{<t-2>}}{\partial c^{<t-2>}}) + \frac{\partial c^{<t-1>}}{\partial c^{<t-2>}}) * (\frac{\partial c^{<t-2>}}{\partial \Gamma_f^{<t-2>}} * \frac{\partial \Gamma_f^{<t-2>}}{\partial sum_f^{<t-2>}} * \frac{\partial sum_f^{<t-2>}}{\partial W_f})))$$

So when we look at the equation above. The parts with a blue “**+**” sign between them indicate a fork in the computation flow. There are two forks in the picture below (circled in red).



Of the two circled forks, one is for $c^{<t>}$ (at the top left) and the other is for $a^{<t>}$ (below and slightly to the right). Let’s start with $a^{<t>}$ (circle that is lower and slightly to the right of the top left circle) since $c^{<t>}$ is dependent on it. When calculating the derivatives associated with a fork you add the derivative branches coming out of the fork together. The derivative that we can calculate for the upper branch of the $a^{<t>}$ fork is:

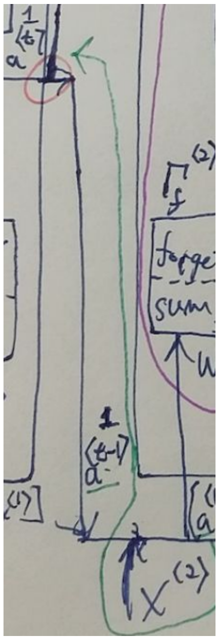


$$\frac{\partial Error^{<t>}}{\partial a^{<t>}}$$

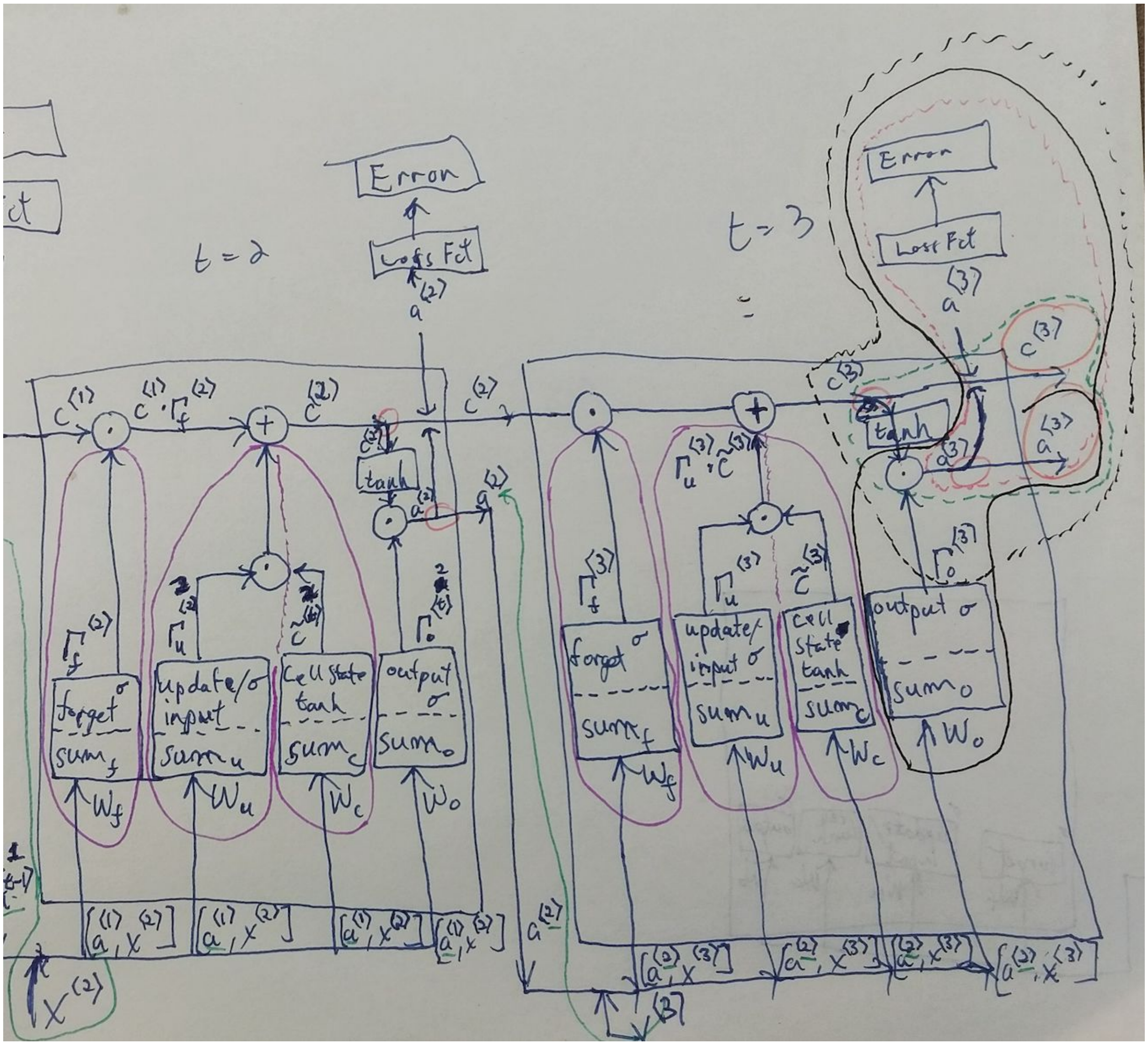
We can derive $\frac{\partial Error^{<t>}}{\partial a^{<t>}}$ from the last functional step in our forward propagation functional stages (previous section) by using the following equation used to calculate the Error.

$$Error \;=\; \tfrac{1}{2}(\widehat{a^{<t>}} - a^{<t>})^2$$

But what about the second branch which goes to the right and then down and across to the next block.



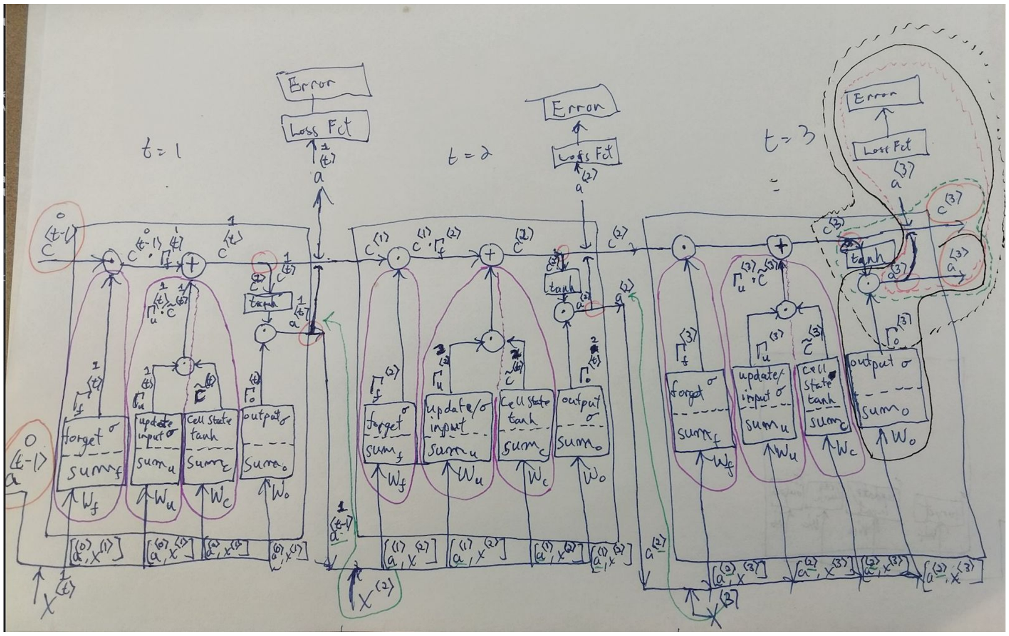
How do we get the derivative for the second branch of $a^{<t>}$? Well, we have to go back to the future to find its gradient. Let's focus in on the middle block (time interval=2).



If our frame of reference is the middle block, than to find the gradient with respect to $a^{<2>}$ we have to find a relationship where $a^{<2>}$ is passed as an input to a function and results in an output that we can calculate the gradient on. We're in luck. The next block (block in the future where time interval=3) calculates the gradient of $\frac{\partial Error^{<3>}}{\partial a^{<2>}}$ ($\frac{\partial Error^{<t+1>}}{\partial a^{<t>}}$). We can add gradient differentials when the denominators match. So our first fork at $a^{<t>}$ is

$$\left(\frac{\partial Error^{<t>}}{\partial a^{<t>}} + \frac{\partial Error^{<t+1>}}{\partial a^{<t>}} \right) = \delta a^{<t>}$$

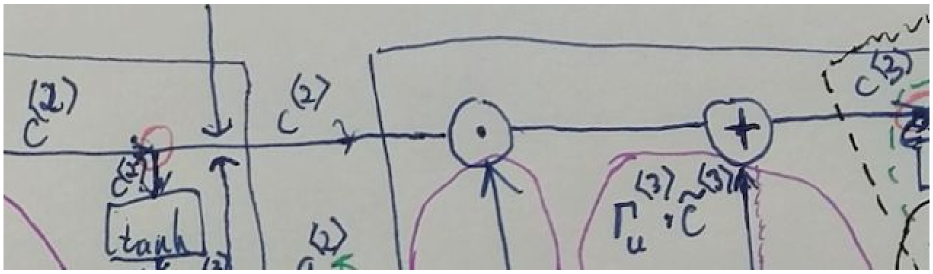
# top right side of left LSTM block



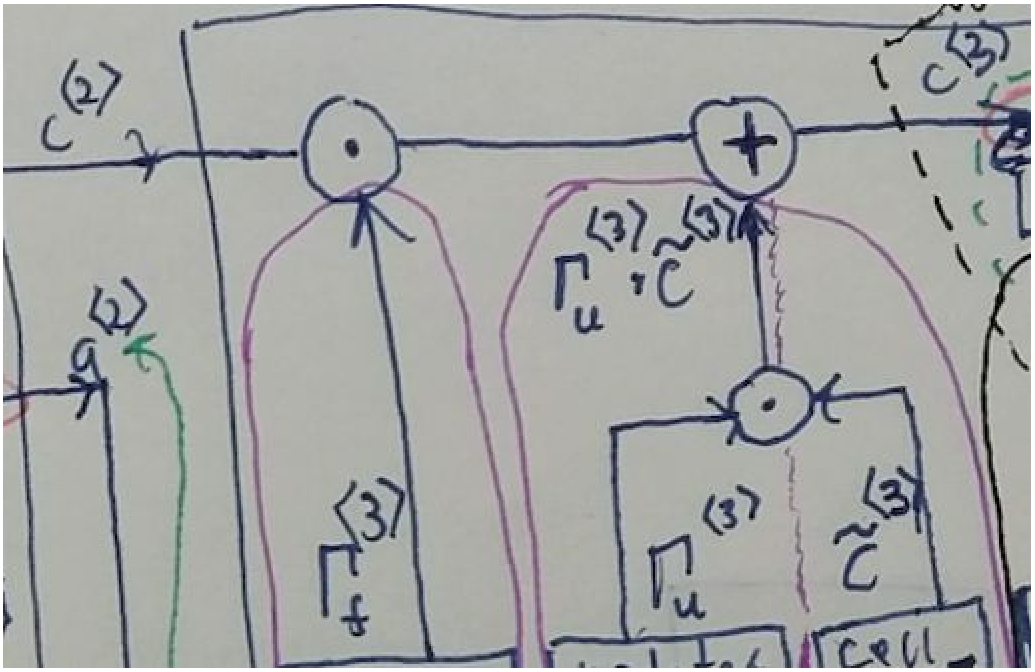
So at time interval=2 (middle block) we can add the branches for $a^{<t>}$ and get:

$$\delta a^{<2>} = \left(\frac{\partial Error^{<2>}}{\partial a^{<2>}} + \frac{\partial Error^{<3>}}{\partial a^{<2>}} \right) \text{ from } \left(\frac{\partial Error^{<t>}}{\partial a^{<t>}} + \frac{\partial Error^{<t+1>}}{\partial a^{<t>}} \right) = \delta a^{<t>}$$

Now let's look at the second fork at $c^{<2>}$ (circled in red on the left below). $c^{<2>}$ is on the far left and $c^{<3>}$ is on far right (sorry it is hard to read).



There are two branches related to $c^{<2>}$ above. The first going horizontally passing $c^{<2>}$ to the next block's dot product operation and another path forking downward passing $c^{<2>}$ to tanh($c^{<2>}$)

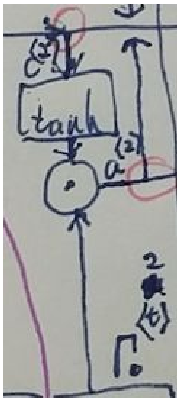


If we look at the first horizontal path in the picture above we find the following equation:

$$c^{<2>} * \Gamma_f^{<3>} + \Gamma_u^{<3>} \cdot \tilde{c}^{<3>} = c^{<3>}$$

There is an immediate relation to bridge the current block to the future block by deriving the gradient of $\frac{\partial c^{<t+1>}}{\partial c^{<t>}}$, which in our case is $\frac{\partial c^{<3>}}{\partial c^{<2>}} = \Gamma_f^{<3>}$ or $\Gamma_f^{<t+1>}$ (in generalised form).

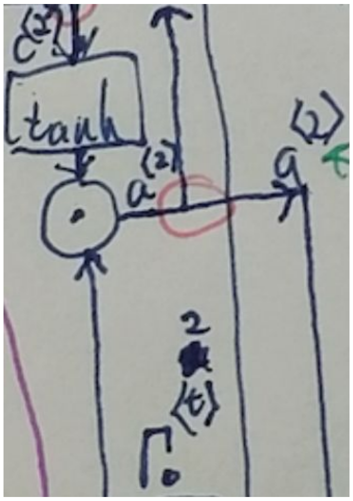
The second branch (top left red circle branching down) passes $c^{<2>}$ to tanh($c^{<2>}$) then multiplies it with $\Gamma_o^{<2>}$ (from the bottom) to output $a^{<2>}$ (right of dot circle).



Putting this all together we have an equation for the second path:

$$a^{<2>} = \tanh(c^{<2>}) * \Gamma_o^{<2>}$$

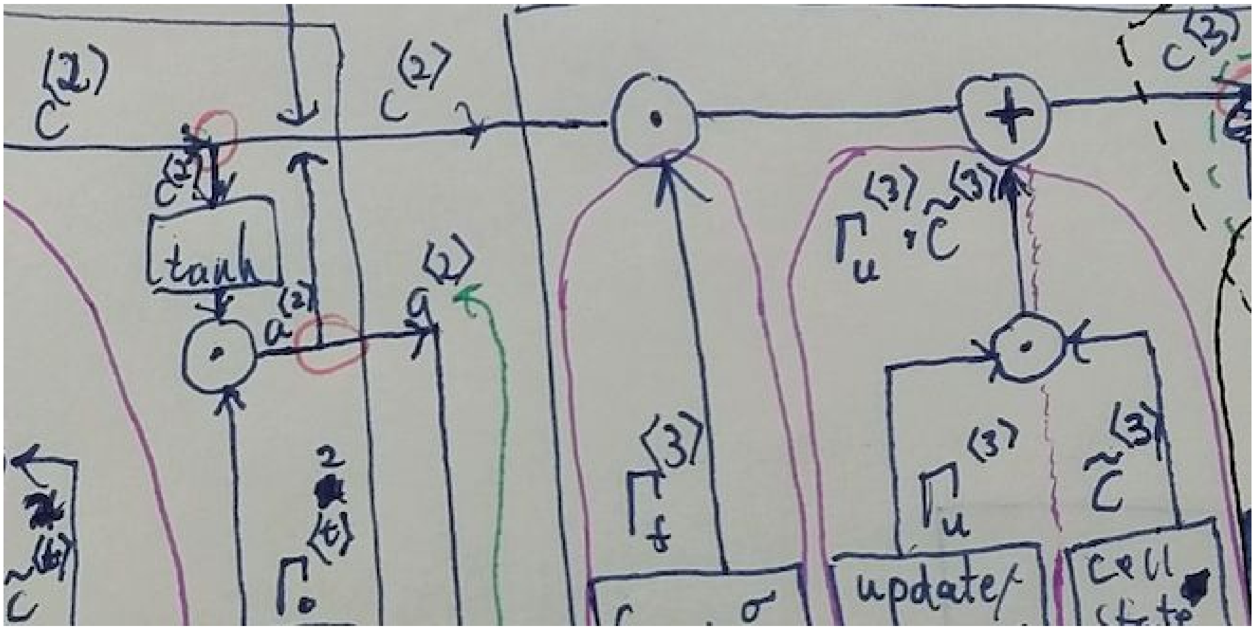
Getting the derivative of the second branch is slightly harder because you have a chain where one of the links is $a^{<2>}$ = tanh($c^{<2>}$) * $\Gamma_o^{<2>}$ and the other link is the $a^{<2>}$ fork (where we already calculated the gradient as $\delta a^{<2>}$ [$\frac{\partial Error^{<2>}}{\partial a^{<2>}}$ + $\frac{\partial Error^{<3>}}{\partial a^{<2>}}$]).



To calculate the derivative for the bottom branch of the $c^{<t>}$ fork, we first need to calculate the derivative for the new portion of the branch:

$\frac{\partial a^{<t>}}{\partial c^{<t>}}$ = $\Gamma_o^{<t>}$ * (1 - tanh($c^{<2>}$)^2)

and multiply this by the $\delta a^{<t>}$ that is chained to it (applying the Chain Rule).



So the top branch of $c^{<t>}$ looks like

$$\frac{\partial c^{<t+1>}}{\partial c^{<t>}}$$

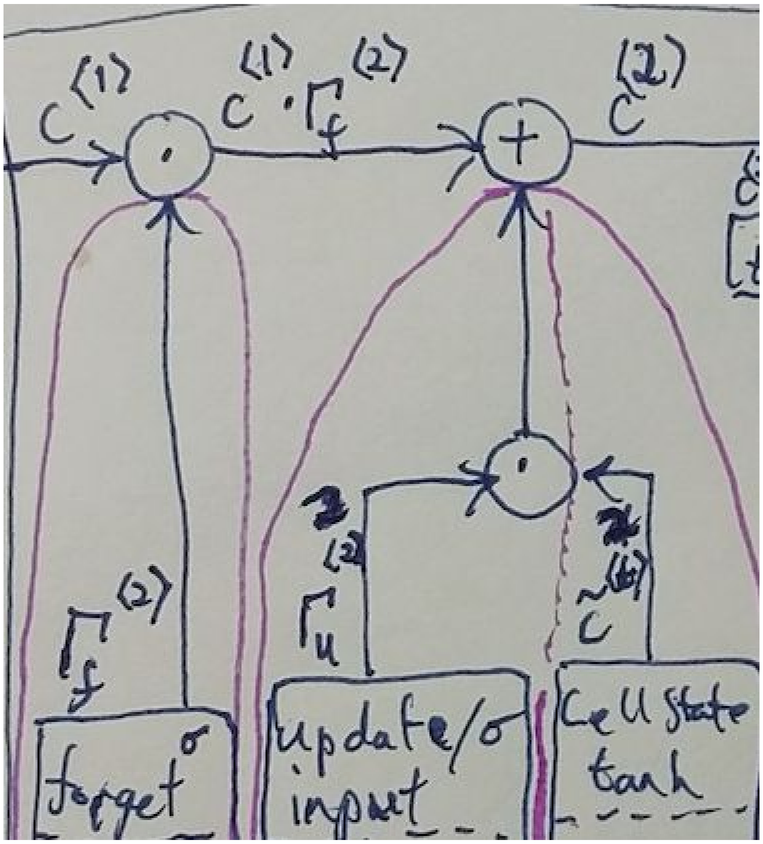
and the bottom branch for $c^{<t>}$ looks like

$$\left(\frac{\partial Error^{<t>}}{\partial a^{<t>}} + \frac{\partial Error^{<t+1>}}{\partial a^{<t>}} \right) * \frac{\partial a^{<t>}}{\partial c^{<t>}}$$

Similar to what we did for the $a^{<t>}$ fork we add the $c^{<t>}$ branches (same denominators) together to get

$$(((\frac{\partial Error^{<t>}}{\partial a^{<t>}} + \frac{\partial Error^{<t+1>}}{\partial a^{<t>}}) * \frac{\partial a^{<t>}}{\partial c^{<t>}}) + \frac{\partial c^{<t+1>}}{\partial c^{<t>}})$$

If you multiply and add the gradients together we have the derivative of the Error with respect to $c^{<t>}$ but we see need to get to W_f. Let's look at the next step along the path (in reverse):



$c^{\langle 1 \rangle}$
$c^{\langle 1 \rangle} \cdot \Gamma_f^{\langle 2 \rangle}$
$c^{\langle 2 \rangle}$
$\Gamma_f^{\langle 2 \rangle}$
$\Gamma_u^{\langle 2 \rangle}$
$\tilde{c}^{\langle 2 \rangle}$
forget σ
update/σ
input
Cell State
tanh

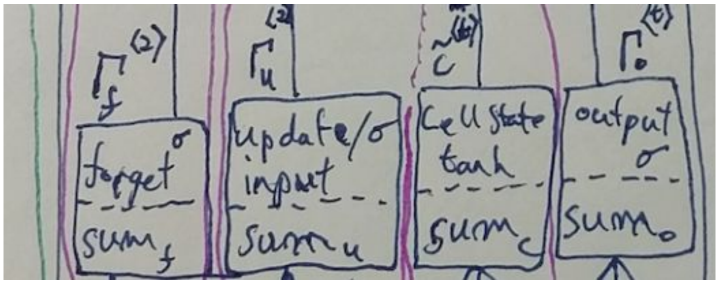
We have

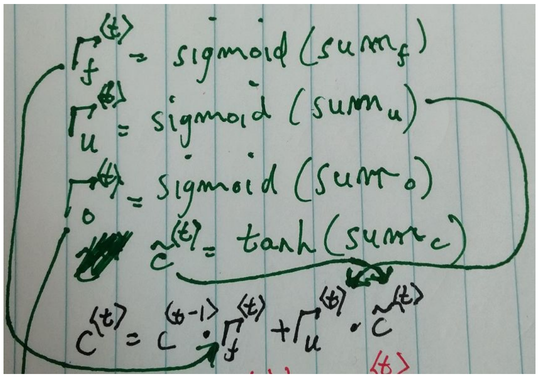
$$c^{<t>} = c^{<t-1>} \cdot \Gamma_f^{<t>} + \Gamma_u^{<t>} \cdot \tilde{c}^{<t>}$$

We need to calculate derivative of $c^{<t>}$ with respect to the gate outputs ($\Gamma_f^{<t>}$, $\Gamma_u^{<t>}$,$\tilde{c}^{<t>}$, $\Gamma_o^{<t>}$) but since we want W_f we calculate $\frac{\partial c^{<t>}}{\partial \Gamma_f^{<t>}}$.

$$\left(\left(\left(\frac{\partial Error^{<t>}}{\partial a^{<t>}} + \frac{\partial Error^{<t+1>}}{\partial a^{<t>}}\right) * \frac{\partial a^{<t>}}{\partial c^{<t>}}\right) + \frac{\partial c^{<t+1>}}{\partial c^{<t>}}\right) * \frac{\partial c^{<t>}}{\partial \Gamma_f^{<t>}}$$

As we continue along the chain in reverse the gate outputs were created from the sums through an activation function.





$\Gamma_f^{<t>}$ = sigmoid (sum_f)

$\Gamma_u^{<t>}$ = sigmoid (sum_u)

$\Gamma_o^{<t>}$ = sigmoid (sum_o)

$\tilde{c}^{<t>}$ = tanh (sum_c)

$c^{<t>} = c^{<t-1>} \cdot \Gamma_f^{<t>} + \Gamma_u^{<t>} \cdot \tilde{c}^{<t>}$

Therefore we can take the derivatives of the following functions

$\Gamma_f^{<t>}$ = sigmoid (sum_f) or $\tilde{c}^{<t>}$ = tanh(sum_c)

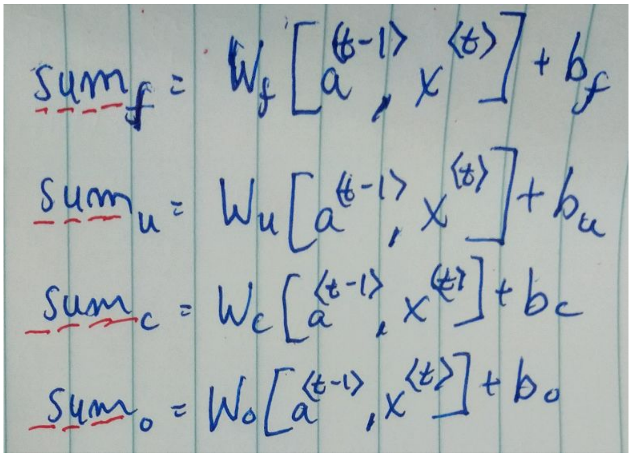
to calculate the derivatives of the gate outputs to the sums as

$$\frac{\partial \Gamma_f^{<t>}}{\partial sum_f^{<t>}} \text{ or } \frac{\partial \tilde{c}^{<t>}}{\partial sum_f^{<t>}}$$

So we can append this to the end of the chain:

$$\left(\left(\left(\frac{\partial Error^{<t>}}{\partial a^{<t>}} + \frac{\partial Error^{<t+1>}}{\partial a^{<t>}}\right) * \frac{\partial a^{<t>}}{\partial c^{<t>}}\right) + \frac{\partial c^{<t+1>}}{\partial c^{<t>}}\right) * \frac{\partial c^{<t>}}{\partial \Gamma_f^{<t>}} * \frac{\partial \Gamma_f^{<t>}}{\partial sum_f^{<t>}}$$

Finally, we can get the derivatives of the sums with respect to the W_x using the functions from the first functional stage.



$$sum_f = W_f\left[a^{\langle t-1\rangle}, x^{\langle t\rangle}\right] + b_f$$

$$sum_u = W_u\left[a^{\langle t-1\rangle}, x^{\langle t\rangle}\right] + b_u$$

$$sum_c = W_c\left[a^{\langle t-1\rangle}, x^{\langle t\rangle}\right] + b_c$$

$$sum_o = W_o\left[a^{\langle t-1\rangle}, x^{\langle t\rangle}\right] + b_o$$

Using:

$$sum_f = W_f\,[a^{<t-1>}, x^{<t>}] + b_f$$

We can calculate:

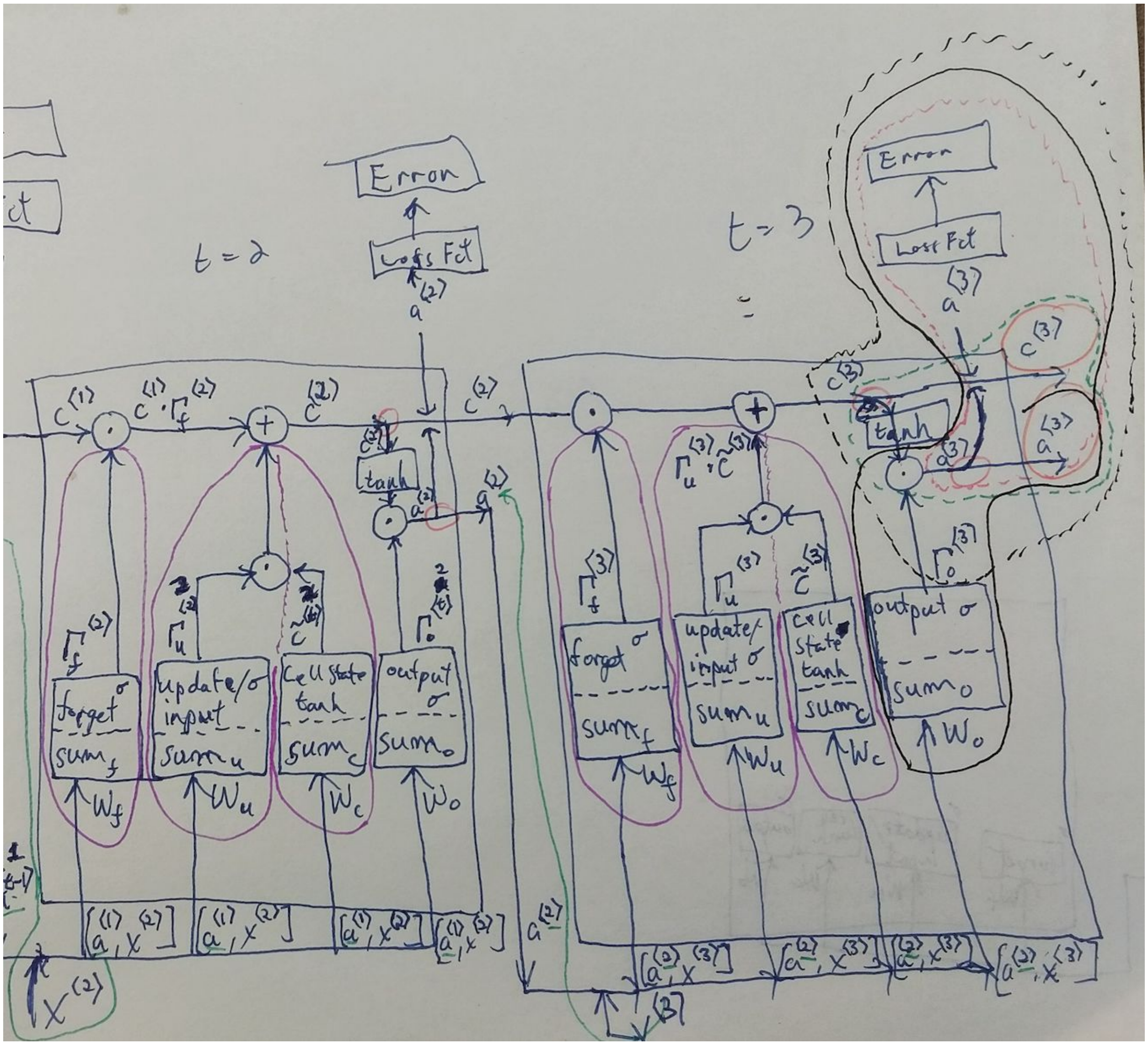
$$\frac{\partial sum_f^{<t>}}{\partial W_f}$$

Again we append this to the end of the chain:

$$\frac{\partial Error^{<t>}}{\partial W_f} = \left(\left(\left(\frac{\partial Error^{<t>}}{\partial a^{<t-1>}} + \frac{\partial Error^{<t+1>}}{\partial a^{<t>}}\right) * \frac{\partial a^{<t>}}{\partial c^{<t>}}\right) + \frac{\partial c^{<t+1>}}{\partial c^{<t>}}\right) * \frac{\partial c^{<t>}}{\partial \Gamma_f^{<t>}} * \frac{\partial \Gamma_f^{<t>}}{\partial sum_f^{<t>}} * \frac{\partial sum_f^{<t>}}{\partial W_f}$$

The above is $\frac{\partial Error^{<t>}}{\partial W_f}$ for one time interval block.

To compute $\frac{\partial Error^{<t+1>}}{\partial a^{<t>}}$ we go to the next block (t + 1) and get its $\frac{\partial Error^{<t>}}{\partial a^{<t-1>}}$ because they are equal. Let's look at the picture again and try to figure out the Error.



The only thing different is that we are going to the block in the future and will calculate the Error with respect to $a^{<t-1>}$

$$\frac{\partial Error^{<t>}}{\partial a^{<t-1>}} = (((\frac{\partial Error^{<t>}}{\partial a^{<t>}} + \frac{\partial Error^{<t+1>}}{\partial a^{<t>}}) * \frac{\partial a^{<t>}}{\partial c^{<t>}}) + \frac{\partial c^{<t+1>}}{\partial c^{<t>}}) * \frac{\partial c^{<t>}}{\partial \Gamma_f^{<t>}} * \frac{\partial \Gamma_f^{<t>}}{\partial sum_f^{<t>}} * \frac{\partial sum_f^{<t>}}{\partial a^{<t-1>}}$$

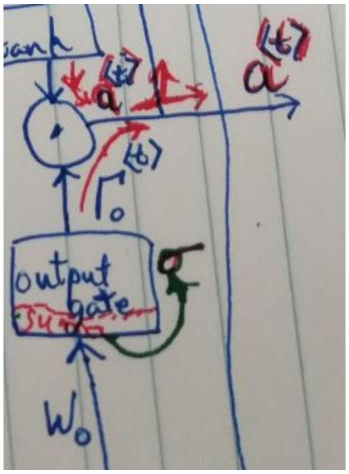
Then you can add all blocks together (t=3, t=2, t=1) to get:

$$\frac{dError}{dW_f} = (((((\frac{\partial Error^{<t>}}{\partial a^{<t>}} + \frac{\partial Error^{<t+1>}}{\partial a^{<t>}}) * \frac{\partial a^{<t>}}{\partial c^{<t>}}) + \frac{\partial c^{<t+1>}}{\partial c^{<t>}}) * (\frac{\partial c^{<t>}}{\partial \Gamma_f^{<t>}} * \frac{\partial \Gamma_f^{<t>}}{\partial sum_f^{<t>}} * \frac{\partial sum_f^{<t>}}{\partial W_f})) +$$

$$((((\frac{\partial Error^{<t-1`>}}{\partial a^{<t-1>}} + \frac{\partial Error^{<t>}}{\partial a^{<t-1>}}) * \frac{\partial a^{<t-1>}}{\partial c^{<t-1>}}) + \frac{\partial c^{<t>}}{\partial c^{<t-1>}}) * (\frac{\partial c^{<t-1>}}{\partial \Gamma_f^{<t-1>}} * \boxed{\frac{\partial \Gamma_f^{<t-1>}}{\partial sum_f^{<t-1>}}} * \frac{\partial sum_f^{<t-1>}}{\partial W_f})) +$$

$$((((\frac{\partial Error^{<t-2>}}{\partial a^{<t-2>}} + \frac{\partial Error^{<t-1>}}{\partial a^{<t-2>}}) * \frac{\partial a^{<t-2>}}{\partial c^{<t-2>}}) + \frac{\partial c^{<t-1>}}{\partial c^{<t-2>}}) * (\frac{\partial c^{<t-2>}}{\partial \Gamma_f^{<t-2>}} * \frac{\partial \Gamma_f^{<t-2>}}{\partial sum_f^{<t-2>}} * \frac{\partial sum_f^{<t-2>}}{\partial W_f})))$$

W_o is a special case. The path from $a^{<t>}$ to W_o does not go through $c^{<t>}$, therefore, the $c^{<t>}$ fork is not included in its calculations.



Everything else is the same.

$$\frac{dError}{dW_o} = \Bigg(\Bigg(\Bigg(\frac{\partial Error^{<t>}}{\partial a^{<t>}} + \frac{\partial Error^{<t+1>}}{\partial a^{<t>}} \Bigg) * \Bigg(\frac{\partial a^{<t>}}{\partial \Gamma_o^{<t>}} * \frac{\partial \Gamma_o^{<t>}}{\partial sum_o^{<t>}} * \boxed{\frac{\partial sum_o^{<t>}}{\partial W_o}} \Bigg) \Bigg) +$$

$$\Bigg(\Bigg(\frac{\partial Error^{<t-1`>}}{\partial a^{<t-1>}} + \frac{\partial Error^{<t>}}{\partial a^{<t-1>}} \Bigg) * \Bigg(\frac{\partial a^{<t-1>}}{\partial \Gamma_o^{<t-1>}} * \frac{\partial \Gamma_o^{<t-1>}}{\partial sum_o^{<t>}} * \frac{\partial sum_o^{<t-1>}}{\partial W_o} \Bigg) \Bigg) +$$

$$\Bigg(\Bigg(\frac{\partial Error^{<t-2>}}{\partial a^{<t-2>}} + \frac{\partial Error^{<t-1>}}{\partial a^{<t-2>}} \Bigg) * \Bigg(\frac{\partial a^{<t-2>}}{\partial \Gamma_o^{<t-2>}} * \frac{\partial \Gamma_o^{<t-2>}}{\partial sum_o^{<t-2>}} * \frac{\partial sum_o^{<t-2>}}{\partial W_o} \Bigg) \Bigg) \Bigg)$$

Given that we have the formula to calculate the derivative that we want, let’s plug in values.

To derive $\frac{\partial Error}{\partial a^{<t>}}$ we have to use the Chain Rule again in a nested fashion. Given the Chain Rule as

$$\frac{d}{dx}\left[f\Big(g(x)\Big)\right] = f'\Big(g(x)\Big)g'(x)$$

we can calculate

$\frac{\partial Error}{\partial a^{<t>}}$ for $\frac{(\hat{a}^{<t>} - a^{<t>})^2}{2}$ by making f = $\frac{x^2}{2}$ and g = ($\hat{a}^{<t>}$ - $a^{<t>}$)

= $\frac{dError}{dg}$ (g^2)/2 * $\frac{dg}{da^{<t>}}$ ($\hat{a}^{<t>}$ - $a^{<t>}$)

= ((2 * g) / 2) * (-1)

substitute in for g = ($\hat{a}^{<t>}$ - $a^{<t>}$)

= (2 * ($\hat{a}^{<t>}$ - $a^{<t>}$) / 2) * (-1)

$\frac{dError}{da^{<t>}}$ = $-(\hat{a}^{<t>} - a^{<t>})$ #remember that I used $a^{<t>}$ as the predicted value from NN and $\hat{a}^{<t>}$ as correct value

$\frac{\partial Error^{<t+1>}}{\partial a^{<t>}}$ is the derivative of the next block's Error with respect to the next block's $a^{<t-1>}$. If you are at the last LSTM block the value for this is zero.

$$\left(\left(\left(-(\hat{a}^{<t>} - a^{<t>}) + \frac{\partial Error^{<t+1>}}{\partial a^{<t>}} \right) * \frac{\partial a^{<t>}}{\partial c^{<t>}} \right) + \frac{\partial c^{<t+1>}}{\partial c^{<t>}} \right) * \left(\frac{\partial c^{<t>}}{\partial \Gamma_f^{<t>}} * \frac{\partial \Gamma_f^{<t>}}{\partial sum_f^{<t>}} * \frac{\partial sum_f^{<t>}}{\partial W_f} \right)$$

Next we need to calculate

$$\frac{\partial a^{<t>}}{\partial c^{<t>}}$$

based on the function

$$a^{<t>} = \tanh(c^{<t>}) * \Gamma_o^{<t>}$$

Because we are taking partial derivatives we can consider $\Gamma_o^{<t>}$ a constant and multiply it by the derivative of tanh($c^{<t>}$). This results in

$$\Gamma_o^{<t>} * (1-\tanh(c^{<t>})^2)$$

$$\frac{dError}{dW_f} = \left(\left(\left(-(\hat{a}^{<t>} - a^{<t>}) + \frac{\partial Error^{<t+1>}}{\partial a^{<t>}} \right) * \left(\Gamma_o^{<t>} * (1-\tanh(c^{<t>})^2) \right) \right) + \frac{\partial c^{<t+1>}}{\partial c^{<t>}} \right) * \left(\frac{\partial c^{<t>}}{\partial \Gamma_f^{<t>}} * \frac{\partial \Gamma_f^{<t>}}{\partial sum_f^{<t>}} * \frac{\partial sum_f^{<t>}}{\partial W_f} \right)$$

The relation we have between $c^{<t>}$ and $c^{<t-1>}$ is

$$c^{<t>} = c^{<t-1>} \cdot \Gamma_f^{<t>} + \Gamma_u^{<t>} \cdot \tilde{c}^{<t>}$$

or if you look at the next block with reference to the previous block to it

$$c^{<t+1>} = c^{<t>} \cdot \Gamma_f^{<t+1>} + \Gamma_u^{<t+1>} \cdot \tilde{c}^{<t+1>}$$

Therefore

$$\frac{\partial c^{<t+1>}}{\partial c^{<t>}} = \Gamma_f^{<t+1>}$$

Now we have:

$$\frac{dError}{dW_f} = \left(\left(\left(-(\hat{a}^{<t>} - a^{<t>}) + \frac{\partial Error^{<t+1>}}{\partial a^{<t>}} \right) * \left(\Gamma_o^{<t>} * (1-\tanh(c^{<t>})^2) \right) \right) + \Gamma_f^{<t+1>} \right) * \left(\frac{\partial c^{<t>}}{\partial \Gamma_f^{<t>}} * \frac{\partial \Gamma_f^{<t>}}{\partial sum_f^{<t>}} * \frac{\partial sum_f^{<t>}}{\partial W_f} \right)$$

To calculate $\frac{\partial c^{<t>}}{\partial \Gamma_f^{<t>}}$ we need to use

$$c^{<t>} = c^{<t-1>} \cdot \Gamma_f^{<t>} + \Gamma_u^{<t>} \cdot \tilde{c}^{<t>}$$

Treat

$$\frac{\partial c^{<t>}}{\partial \Gamma_f^{<t>}} = c^{<t-1>}$$

$$\frac{dError}{dW_f} = (((-(\hat{a}^{<t>} - a^{<t>}) + \frac{\partial Error^{<t+1>}}{\partial a^{<t>}}) * (\Gamma_o^{<t>} * (1-\tanh(c^{<t>})^2))) + \Gamma_f^{<t+1>}) * (c^{<t-1>} * \frac{\partial \Gamma_f^{<t>}}{\partial sum_f^{<t>}} * \frac{\partial sum_f^{<t>}}{\partial W_f})$$

To calculate $\frac{\partial \Gamma_f^{<t>}}{\partial sum_o}$ we need to look at the function at outputs $\Gamma_f^{<t>}$: $\Gamma_f^{<t>}$ = sigmoid(sum_f). Here the derivative of the sigmoid function is sigmoid(sum_f) * (1 - sigmoid(sum_f)). Since $\Gamma_f^{<t>}$ = sigmoid(sum_o) we can further simplify

$$\frac{\partial \Gamma_f^{<t>}}{\partial sum_f} = \Gamma_f^{<t>} * (1 - \Gamma_f^{<t>})$$

$$\frac{dError}{dW_f} = (((-(\hat{a}^{<t>} - a^{<t>}) + \frac{\partial Error^{<t+1>}}{\partial a^{<t>}}) * (\Gamma_o^{<t>} * (1-\tanh(c^{<t>})^2))) + \Gamma_f^{<t+1>}) * (c^{<t-1>} * (\Gamma_f^{<t>} * (1 - \Gamma_f^{<t>})) * \frac{\partial sum_f^{<t>}}{\partial W_f})$$

What is left is $\frac{\partial sum_f}{\partial W_f}$ which can be derived from $sum_f = W_f\left[a^{<t-1>}, x^{<t>}\right] + b_f$. Here the derivative of the function with respect to W_f leaves $\left[a^{<t-1>}, x^{<t>}\right]$. Therefore finally we have:

$$\frac{dError}{dW_f} = (((-(\hat{a}^{<t>} - a^{<t>}) + \frac{\partial Error^{<t+1>}}{\partial a^{<t>}}) * (\Gamma_o^{<t>} * (1-\tanh(c^{<t>})^2))) + \Gamma_f^{<t+1>}) * (c^{<t-1>} * (\Gamma_f^{<t>} * (1 - \Gamma_f^{<t>})) * \left[a^{<t-1>}, x^{<t>}\right])$$

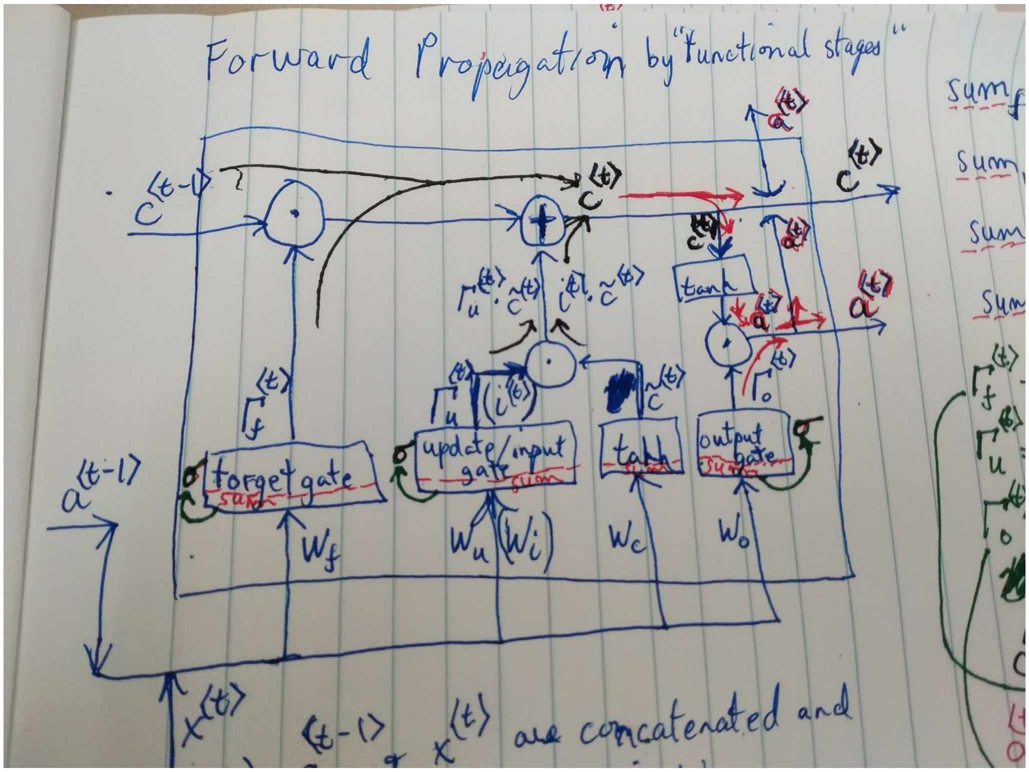
If we wanted to update the bias b_f then everything is similar except for the last part.

$$((((\frac{\partial Error^{<t>}}{\partial a^{<t>}} + \frac{\partial Error^{<t+1>}}{\partial a^{<t>}}) * \frac{\partial a^{<t>}}{\partial c^{<t>}}) + \frac{\partial c^{<t+1>}}{\partial c^{<t>}}) * (\frac{\partial c^{<t>}}{\partial \Gamma_f^{<t>}} * \frac{\partial \Gamma_f^{<t>}}{\partial sum_f^{<t>}} * \frac{\partial sum_f^{<t>}}{\partial b_f}))$$

What is left is $\frac{\partial sum_f}{\partial b_f}$ which can be derived from $sum_f = W_f\left[a^{<t-1>}, x^{<t>}\right] + b_f$. Here the derivative of the function with respect to b_f is 1 because when we take partial derivatives, $W_f\left[a^{<t-1>}, x^{<t>}\right]$ is considered a constant and a derivative of a constant is zero leaving the derivative of b_f which is equal to 1 . Therefore finally we have:

$$\frac{dError}{db_f} = (((-(\hat{a}^{<t>} - a^{<t>}) + \frac{\partial Error^{<t+1>}}{\partial a^{<t>}}) * (\Gamma_o^{<t>} * (1-\tanh(c^{<t>})^2))) + \Gamma_f^{<t+1>}) * (c^{<t-1>} * (\Gamma_f^{<t>} * (1 - \Gamma_f^{<t>})) * 1)$$

All parameters share the same initial path from Error to $a^{<t>}$ however they take different paths toward $c^{<t>}$ to eventually reach their respective gates.


Forward Propagation by "functional stages"
$c^{\langle t-1\rangle}$
$c^{\langle t\rangle}$
$a^{\langle t\rangle}$
$\Gamma_u^{\langle t\rangle} \cdot \tilde{c}^{\langle t\rangle}$
$i^{\langle t\rangle} \cdot \tilde{c}^{\langle t\rangle}$
tanh
$\Gamma_f^{\langle t\rangle}$
$\Gamma_u^{\langle t\rangle}$
$(i^{\langle t\rangle})$
$\tilde{c}^{\langle t\rangle}$
$\Gamma_o^{\langle t\rangle}$
$a^{\langle t-1\rangle}$
forget gate
sum
update/input gate
sum
tanh
output gate
sum
W_f
W_u
(W_i)
W_c
W_o
$x^{\langle t\rangle}$
$a^{\langle t-1\rangle}$ & $x^{\langle t\rangle}$ are concatenated and
sum$_f$
sum
sum
sum

Let's now look at the other paths to W_f, W_u, W_c from Error (given t = 3 timesteps/parts).

$$\frac{dError}{dW_f} = (((((\frac{\partial Error^{<t>}}{\partial a^{<t>}} + \frac{\partial Error^{<t+1>}}{\partial a^{<t>}}) * \frac{\partial a^{<t>}}{\partial c^{<t>}}) + \frac{\partial c^{<t+1>}}{\partial c^{<t>}}) * (\frac{\partial c^{<t>}}{\partial \Gamma_f^{<t>}} * \frac{\partial \Gamma_f^{<t>}}{\partial sum_f^{<t>}} * \frac{\partial sum_o^{<t>}}{\partial W_f}))$$

$$+ ((((\frac{\partial Error^{<t-1`>}}{\partial a^{<t-1>}} + \frac{\partial Error^{<t>}}{\partial a^{<t-1>}}) * \frac{\partial a^{<t-1>}}{\partial c^{<t-1>}}) + \frac{\partial c^{<t>}}{\partial c^{<t-1>}}) * (\frac{\partial c^{<t-1>}}{\partial \Gamma_f^{<t-1>}} * \frac{\partial \Gamma_f^{<t-1>}}{\partial sum_f^{<t-1>}} * \frac{\partial sum_o^{<t-1>}}{\partial W_f}))$$

$$+ ((((\frac{\partial Error^{<t-2>}}{\partial a^{<t-2>}} + \frac{\partial Error^{<t-1>}}{\partial a^{<t-2>}}) * \frac{\partial a^{<t-2>}}{\partial c^{<t-2>}}) + \frac{\partial c^{<t-1>}}{\partial c^{<t-2>}}) * (\frac{\partial c^{<t-2>}}{\partial \Gamma_f^{<t-2>}} * \frac{\partial \Gamma_f^{<t-2>}}{\partial sum_f^{<t-2>}} * \frac{\partial sum_o^{<t-2>}}{\partial W_f})))$$

$$\frac{dError}{dW_u} = (((((\frac{\partial Error^{<t>}}{\partial a^{<t>}} + \frac{\partial Error^{<t+1>}}{\partial a^{<t>}}) * \frac{\partial a^{<t>}}{\partial c^{<t>}}) + \frac{\partial c^{<t+1>}}{\partial c^{<t>}}) * (\frac{\partial c^{<t>}}{\partial \Gamma_u^{<t>}} * \frac{\partial \Gamma_u^{<t>}}{\partial sum_u^{<t>}} * \frac{\partial sum_u^{<t>}}{\partial W_u}))$$

$$+ ((((\frac{\partial Error^{<t-1`>}}{\partial a^{<t-1>}} + \frac{\partial Error^{<t>}}{\partial a^{<t-1>}}) * \frac{\partial a^{<t-1>}}{\partial c^{<t-1>}}) + \frac{\partial c^{<t>}}{\partial c^{<t-1>}}) * (\frac{\partial c^{<t-1>}}{\partial \Gamma_u^{<t-1>}} * \frac{\partial \Gamma_u^{<t-1>}}{\partial sum_u^{<t-1>}} * \frac{\partial sum_u^{<t-1>}}{\partial W_u}))$$

$$+ ((((\frac{\partial Error^{<t-2>}}{\partial a^{<t-2>}} + \frac{\partial Error^{<t-1>}}{\partial a^{<t-2>}}) * \frac{\partial a^{<t-2>}}{\partial c^{<t-2>}}) + \frac{\partial c^{<t-1>}}{\partial c^{<t-2>}}) * (\frac{\partial c^{<t-2>}}{\partial \Gamma_u^{<t-2>}} * \frac{\partial \Gamma_u^{<t-2>}}{\partial sum_u^{<t-2>}} * \frac{\partial sum_u^{<t-2>}}{\partial W_u})))$$

$$
\begin{aligned}
\frac{dError}{dW_c} = (((((&\frac{\partial Error^{<t>}}{\partial a^{<t>}} + \frac{\partial Error^{<t+1>}}{\partial a^{<t>}}) * \frac{\partial a^{<t>}}{\partial c^{<t>}}) + \frac{\partial c^{<t+1>}}{\partial c^{<t>}}) * (\frac{\partial c^{<t>}}{\partial \Gamma_c^{<t>}} * \frac{\partial c^{\sim <t>}}{\partial sum_c^{<t>}} * \frac{\partial sum_c^{<t>}}{\partial W_c})) \\
+ \quad ((((&\frac{\partial Error^{<t-1`>}}{\partial a^{<t-1>}} + \frac{\partial Error^{<t>}}{\partial a^{<t-1>}}) * \frac{\partial a^{<t-1>}}{\partial c^{<t-1>}}) + \frac{\partial c^{<t>}}{\partial c^{<t-1>}}) * (\frac{\partial c^{<t-1>}}{\partial \Gamma_c^{<t-1>}} * \frac{\partial c^{\sim <t-1>}}{\partial sum_c^{<t-1>}} * \frac{\partial sum_{cu}^{<t-1>}}{\partial W_c})) \\
+ \quad ((((&\frac{\partial Error^{<t-2>}}{\partial a^{<t-2>}} + \frac{\partial Error^{<t-1>}}{\partial a^{<t-2>}}) * \frac{\partial a^{<t-2>}}{\partial c^{<t-2>}}) + \frac{\partial c^{<t-1>}}{\partial c^{<t-2>}}) * (\frac{\partial c^{<t-2>}}{\partial \Gamma_c^{<t-2>}} * \frac{\partial c^{\sim <t-2>}}{\partial sum_c^{<t-2>}} * \frac{\partial sum_c^{<t-2>}}{\partial W_c})))
\end{aligned}
$$

$$
\begin{aligned}
\frac{dError}{dW_o} = (((&\frac{\partial Error^{<t>}}{\partial a^{<t>}} + \frac{\partial Error^{<t+1>}}{\partial a^{<t>}}) * (\frac{\partial a^{<t>}}{\partial \Gamma_o^{<t>}} * \frac{\partial \Gamma_o^{<t>}}{\partial sum_o^{<t>}} * \frac{\partial sum_o^{<t>}}{\partial W_o})) + \\
((&\frac{\partial Error^{<t-1`>}}{\partial a^{<t-1>}} + \frac{\partial Error^{<t>}}{\partial a^{<t-1>}}) * (\frac{\partial a^{<t-1>}}{\partial \Gamma_o^{<t-1>}} * \frac{\partial \Gamma_o^{<t-1>}}{\partial sum_o^{<t-1>}} * \frac{\partial sum_o^{<t-1>}}{\partial W_o})) + \\
((&\frac{\partial Error^{<t-2>}}{\partial a^{<t-2>}} + \frac{\partial Error^{<t-1>}}{\partial a^{<t-2>}}) * (\frac{\partial a^{<t-2>}}{\partial \Gamma_o^{<t-2>}} * \frac{\partial \Gamma_o^{<t-2>}}{\partial sum_o^{<t-2>}} * \frac{\partial sum_o^{<t-2>}}{\partial W_o})))
\end{aligned}
$$

$$
\begin{aligned}
\frac{dError}{db_f} = (((((&\frac{\partial Error^{<t>}}{\partial a^{<t>}} + \frac{\partial Error^{<t+1>}}{\partial a^{<t>}}) * \frac{\partial a^{<t>}}{\partial c^{<t>}}) + \frac{\partial c^{<t+1>}}{\partial c^{<t>}}) * (\frac{\partial c^{<t>}}{\partial \Gamma_f^{<t>}} * \frac{\partial \Gamma_f^{<t>}}{\partial sum_f^{<t>}} * \frac{\partial sum_o^{<t>}}{\partial b_f})) \\
+ \quad ((((&\frac{\partial Error^{<t-1`>}}{\partial a^{<t-1>}} + \frac{\partial Error^{<t>}}{\partial a^{<t-1>}}) * \frac{\partial a^{<t-1>}}{\partial c^{<t-1>}}) + \frac{\partial c^{<t>}}{\partial c^{<t-1>}}) * (\frac{\partial c^{<t-1>}}{\partial \Gamma_f^{<t-1>}} * \frac{\partial \Gamma_f^{<t-1>}}{\partial sum_f^{<t-1>}} * \frac{\partial sum_o^{<t-1>}}{\partial b_f})) \\
+ \quad ((((&\frac{\partial Error^{<t-2>}}{\partial a^{<t-2>}} + \frac{\partial Error^{<t-1>}}{\partial a^{<t-2>}}) * \frac{\partial a^{<t-2>}}{\partial c^{<t-2>}}) + \frac{\partial c^{<t-1>}}{\partial c^{<t-2>}}) * (\frac{\partial c^{<t-2>}}{\partial \Gamma_f^{<t-2>}} * \frac{\partial \Gamma_f^{<t-2>}}{\partial sum_f^{<t-2>}} * \frac{\partial sum_o^{<t-2>}}{\partial b_f})))
\end{aligned}
$$

$$
\begin{aligned}
\frac{dError}{db_u} = (((((&\frac{\partial Error^{<t>}}{\partial a^{<t>}} + \frac{\partial Error^{<t+1>}}{\partial a^{<t>}}) * \frac{\partial a^{<t>}}{\partial c^{<t>}}) + \frac{\partial c^{<t+1>}}{\partial c^{<t>}}) * (\frac{\partial c^{<t>}}{\partial \Gamma_u^{<t>}} * \frac{\partial \Gamma_u^{<t>}}{\partial sum_u^{<t>}} * \frac{\partial sum_u^{<t>}}{\partial b_u})) \\
+ \quad ((((&\frac{\partial Error^{<t-1`>}}{\partial a^{<t-1>}} + \frac{\partial Error^{<t>}}{\partial a^{<t-1>}}) * \frac{\partial a^{<t-1>}}{\partial c^{<t-1>}}) + \frac{\partial c^{<t>}}{\partial c^{<t-1>}}) * (\frac{\partial c^{<t-1>}}{\partial \Gamma_u^{<t-1>}} * \frac{\partial \Gamma_u^{<t-1>}}{\partial sum_u^{<t-1>}} * \frac{\partial sum_u^{<t-1>}}{\partial b_u})) \\
+ \quad ((((&\frac{\partial Error^{<t-2>}}{\partial a^{<t-2>}} + \frac{\partial Error^{<t-1>}}{\partial a^{<t-2>}}) * \frac{\partial a^{<t-2>}}{\partial c^{<t-2>}}) + \frac{\partial c^{<t-1>}}{\partial c^{<t-2>}}) * (\frac{\partial c^{<t-2>}}{\partial \Gamma_u^{<t-2>}} * \frac{\partial \Gamma_u^{<t-2>}}{\partial sum_u^{<t-2>}} * \frac{\partial sum_u^{<t-2>}}{\partial b_u})))
\end{aligned}
$$

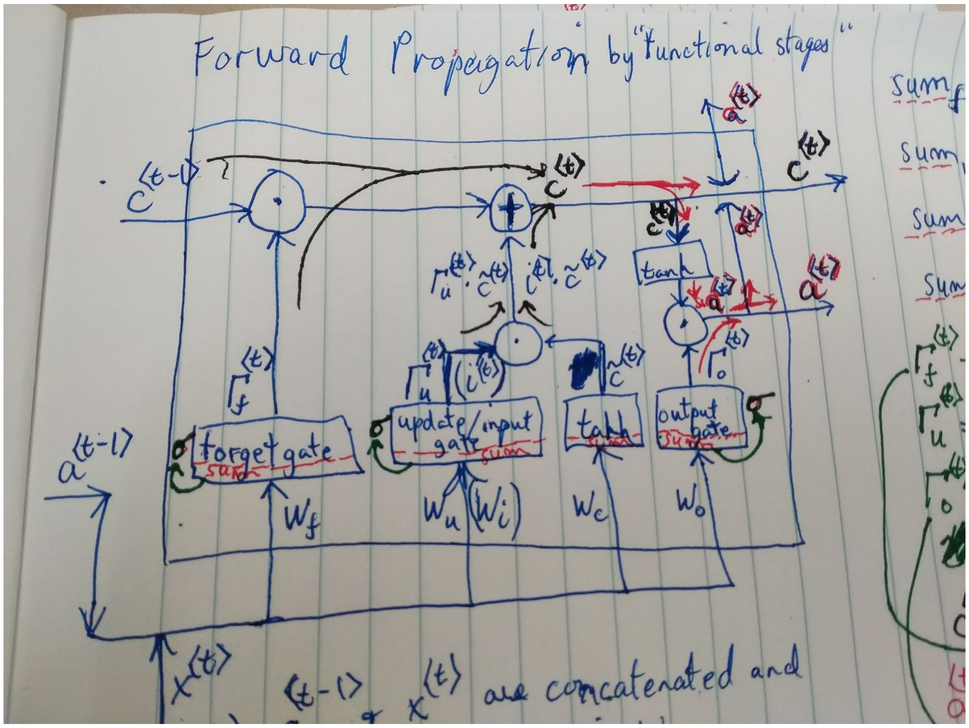
$$
\begin{aligned}
\frac{dError}{db_c} = (((((&\frac{\partial Error^{<t>}}{\partial a^{<t>}} + \frac{\partial Error^{<t+1>}}{\partial a^{<t>}}) * \frac{\partial a^{<t>}}{\partial c^{<t>}}) + \frac{\partial c^{<t+1>}}{\partial c^{<t>}}) * (\frac{\partial c^{<t>}}{\partial \Gamma_c^{<t>}} * \frac{\partial c^{\sim <t>}}{\partial sum_c^{<t>}} * \frac{\partial sum_c^{<t>}}{\partial b_c})) \\
+ \quad ((((&\frac{\partial Error^{<t-1`>}}{\partial a^{<t-1>}} + \frac{\partial Error^{<t>}}{\partial a^{<t-1>}}) * \frac{\partial a^{<t-1>}}{\partial c^{<t-1>}}) + \frac{\partial c^{<t>}}{\partial c^{<t-1>}}) * (\frac{\partial c^{<t-1>}}{\partial \Gamma_c^{<t-1>}} * \frac{\partial c^{\sim <t-1>}}{\partial sum_c^{<t-1>}} * \frac{\partial sum_{cu}^{<t-1>}}{\partial b_c})) \\
+ \quad ((((&\frac{\partial Error^{<t-2>}}{\partial a^{<t-2>}} + \frac{\partial Error^{<t-1>}}{\partial a^{<t-2>}}) * \frac{\partial a^{<t-2>}}{\partial c^{<t-2>}}) + \frac{\partial c^{<t-1>}}{\partial c^{<t-2>}}) * (\frac{\partial c^{<t-2>}}{\partial \Gamma_c^{<t-2>}} * \frac{\partial c^{\sim <t-2>}}{\partial sum_c^{<t-2>}} * \frac{\partial sum_c^{<t-2>}}{\partial b_c})))
\end{aligned}
$$

$$\frac{dError}{db_o} = (((\frac{\partial Error^{<t>}}{\partial a^{<t>}} + \frac{\partial Error^{<t+1>}}{\partial a^{<t>}}) * (\frac{\partial a^{<t>}}{\partial \Gamma_o^{<t>}} * \frac{\partial \Gamma_o^{<t>}}{\partial sum_o^{<t>}} * \frac{\partial sum_o^{<t>}}{\partial b_o})) +$$

$$((\frac{\partial Error^{<t-1`>}}{\partial a^{<t-1>}} + \frac{\partial Error^{<t>}}{\partial a^{<t-1>}}) * (\frac{\partial a^{<t-1>}}{\partial \Gamma_o^{<t-1>}} * \frac{\partial \Gamma_o^{<t-1>}}{\partial sum_o^{<t-1>}} * \frac{\partial sum_o^{<t-1>}}{\partial b_o})) +$$

$$((\frac{\partial Error^{<t-2>}}{\partial a^{<t-2>}} + \frac{\partial Error^{<t-1>}}{\partial a^{<t-2>}}) * (\frac{\partial a^{<t-2>}}{\partial \Gamma_o^{<t-2>}} * \frac{\partial \Gamma_o^{<t-2>}}{\partial sum_o^{<t-2>}} * \frac{\partial sum_o^{<t-2>}}{\partial b_o})))$$

To backpropagate to the previous block we have to figure out the Error with respect to $a^{<t-1>}$ and $c^{<t-1>}$. The key to understanding this is to look at where in the picture to $a^{<t-1>}$ and $c^{<t-1>}$ occur.



The previous output, $a^{<t-1>}$**,** is concatenated to the current input $x^{<t>}$ and effects every gate, therefore all of the partial derivatives with respect to the gates needs to be accounted for. The W_x and b_x parameters are summed across time intervals because they are reused across different t intervals. $a^{<t-1>}$ is reused in a single block across all gates but the output value is different for each block and needs to be recalculated for every block.

$$\frac{dError^{<t>}}{da^{<t-1>}} = (((((\frac{\partial Error^{<t>}}{\partial a^{<t>}} + \frac{\partial Error^{<t+1>}}{\partial a^{<t>}}) * \frac{\partial a^{<t>}}{\partial c^{<t>}}) + \frac{\partial c^{<t+1>}}{\partial c^{<t>}}) * (\frac{\partial c^{<t>}}{\partial \Gamma_f^{<t>}} * \frac{\partial \Gamma_f^{<t>}}{\partial sum_f^{<t>}} * \frac{\partial sum_o^{<t>}}{\partial a^{<t-1>}}))$$

$$+ ((((\frac{\partial Error^{<t>}}{\partial a^{<t>}} + \frac{\partial Error^{<t+1>}}{\partial a^{<t>}}) * \frac{\partial a^{<t>}}{\partial c^{<t>}}) + \frac{\partial c^{<t+1>}}{\partial c^{<t>}}) * (\frac{\partial c^{<t>}}{\partial \Gamma_c^{<t>}} * \frac{\partial c^{\sim <t>}}{\partial sum_c} * \frac{\partial sum_c}{\partial a^{<t-1>}}))$$

$$+ ((((\frac{\partial Error^{<t>}}{\partial a^{<t>}} + \frac{\partial Error^{<t+1>}}{\partial a^{<t>}}) * \frac{\partial a^{<t>}}{\partial c^{<t>}}) + \frac{\partial c^{<t+1>}}{\partial c^{<t>}}) * (\frac{\partial c^{<t>}}{\partial \Gamma_u^{<t>}} * \frac{\partial \Gamma_u^{<t>}}{\partial sum_u} * \frac{\partial sum_u}{\partial a^{<t-1>}}))$$

$$+ ((\frac{\partial Error^{<t>}}{\partial a^{<t>}} + \frac{\partial Error^{<t+1>}}{\partial a^{<t>}}) * (\frac{\partial a^{<t>}}{\partial \Gamma_o^{<t>}} * \frac{\partial \Gamma_o^{<t>}}{\partial sum_o^{<t>}} * \frac{\partial sum_o^{<t>}}{\partial a^{<t-1>}})))$$

The above calculation has to be done for every LSTM block except the first block. The $\frac{dError^{<t>}}{da^{<t-1>}}$ value is used when moving backwards to the previous block (t=3 to t=2) in place of $\frac{\partial Error^{<t+1>}}{\partial a^{<t>}}$.

$\frac{dError^{<t>}}{da^{<t-1>}} = \frac{dError^{<3>}}{da^{<2>}}$ that we calculated at t = 3 is equal to

$\frac{\partial Error^{<t+1>}}{\partial a^{<t>}} = \frac{dError^{<2+1>}}{da^{<2>}} = \frac{dError^{<3>}}{da^{<2>}}$ at t = 2

At every LSTM block we also need to calculate the derivative of $c^{<t>}$ to $c^{<t-1>}$ ($\frac{\partial c^{<t>}}{\partial c^{<t-1>}}$) to provide to the next block. At t=3 (our starting point for back propagation) $\frac{\partial c^{<t+1>}}{\partial c^{<t>}}$ = 0 so it drops out.

$$\begin{aligned}\frac{dError}{dW_f} = (((((\frac{\partial Error^{<t>}}{\partial a^{<t>}} + \frac{\partial Error^{<t+1>}}{\partial a^{<t>}}) * \frac{\partial a^{<t>}}{\partial c^{<t>}}) + 0) * (\frac{\partial c^{<t>}}{\partial \Gamma_f^{<t>}} * \frac{\partial \Gamma_f^{<t>}}{\partial sum_f^{<t>}} * \frac{\partial sum_o^{<t>}}{\partial W_f})) \\ + \quad ((((\frac{\partial Error^{<t-1`>}}{\partial a^{<t-1>}} + \frac{\partial Error^{<t>}}{\partial a^{<t-1>}}) * \frac{\partial a^{<t-1>}}{\partial c^{<t-1>}}) + \frac{\partial c^{<t>}}{\partial c^{<t-1>}}) * (\frac{\partial c^{<t-1>}}{\partial \Gamma_f^{<t-1>}} * \frac{\partial \Gamma_f^{<t-1>}}{\partial sum_f^{<t-1>}} * \frac{\partial sum_o^{<t-1>}}{\partial W_f})) \\ + \quad ((((\frac{\partial Error^{<t-2>}}{\partial a^{<t-2>}} + \frac{\partial Error^{<t-1>}}{\partial a^{<t-2>}}) * \frac{\partial a^{<t-2>}}{\partial c^{<t-2>}}) + \frac{\partial c^{<t-1>}}{\partial c^{<t-2>}}) * (\frac{\partial c^{<t-2>}}{\partial \Gamma_f^{<t-2>}} * \frac{\partial \Gamma_f^{<t-2>}}{\partial sum_f^{<t-2>}} * \frac{\partial sum_o^{<t-2>}}{\partial W_f})))\end{aligned}$$

But when we are at the middle row (second time interval) above (we still plugin 3 for t everywhere in the equation above but $a^{<1>}$ means the output at time t=1, $a^{<2>}$ means the output at time t=2, etc.) we need the gradient that we calculated in the previous back propagated block (technically in the future because start at time = 3 [the last block] we calculated $\frac{\partial c^{<t>}}{\partial c^{<t-1>}} = \frac{\partial c^{<3>}}{\partial c^{<2>}}$ when processing the 3rd block at the third time interval.

In the first row from the equation above (representing the third time interval) we were calculating

$$((((\frac{\partial Error^{<t>}}{\partial a^{<t>}} + \frac{\partial Error^{<t+1>}}{\partial a^{<t>}}) * \frac{\partial a^{<t>}}{\partial c^{<t>}}) + \frac{\partial c^{<t+1>}}{\partial c^{<t>}}) * (\frac{\partial c^{<t>}}{\partial \Gamma_f^{<t>}} * \frac{\partial \Gamma_f^{<t>}}{\partial sum_f^{<t>}} * \frac{\partial sum_o^{<t>}}{\partial W_f}))$$

The problem is that there is no 4th time interval so

$$\frac{\partial c^{<t+1>}}{\partial c^{<t>}} = \frac{\partial c^{<4>}}{\partial c^{<3>}} = 0$$

But at the second time interval (middle row from above)

$$+ \quad ((((\frac{\partial Error^{<t-1`>}}{\partial a^{<t-1>}} + \frac{\partial Error^{<t>}}{\partial a^{<t-1>}}) * \frac{\partial a^{<t-1>}}{\partial c^{<t-1>}}) + \frac{\partial c^{<t>}}{\partial c^{<t-1>}}) * (\frac{\partial c^{<t-1>}}{\partial \Gamma_f^{<t-1>}} * \frac{\partial \Gamma_f^{<t-1>}}{\partial sum_f^{<t-1>}} * \frac{\partial sum_o^{<t-1>}}{\partial W_f}))$$

$\frac{\partial c^{<t>}}{\partial c^{<t-1>}}$ = $\frac{\partial c^{<3>}}{\partial c^{<2>}}$ which was processed when doing calculations for 3rd interval block(technically in the future). The whole point of this is help you understand that even through it seem like you are calculating $\frac{\partial c^{<t>}}{\partial c^{<t-1>}}$ for a future value, it isn't. You are not going back in time just calculating gradients at different time intervals relative to each block.

Conclusion

You now understand backpropagation through LSTM. The key takeaways to understanding all back propagation is
1. Understanding the individual functional stages of forward propagation
2. Understanding the Chain Rule
3. Applying the Chain Rule and forking to the functional stages in reverse to calculate the appropriate partial derivatives.