Vanishing Gradient

The Vanishing Gradient Problem

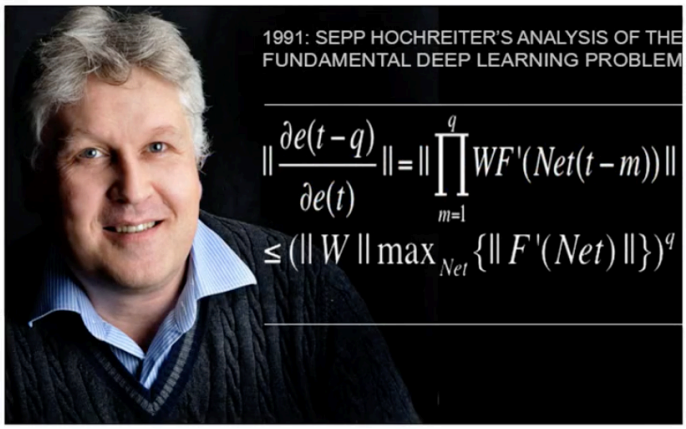


Image Source: people.idsia.ch

The Vanishing Gradient Problem

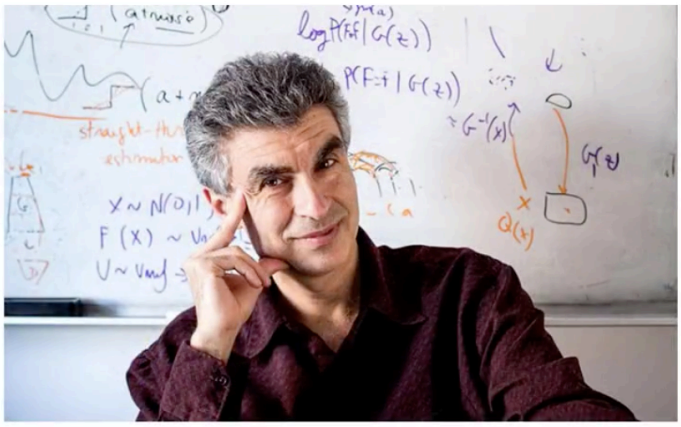


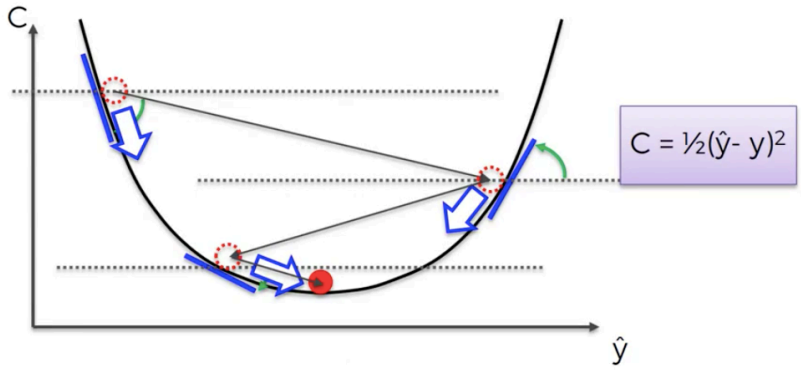
Image Source: Université Montréal

Yasha Bengio

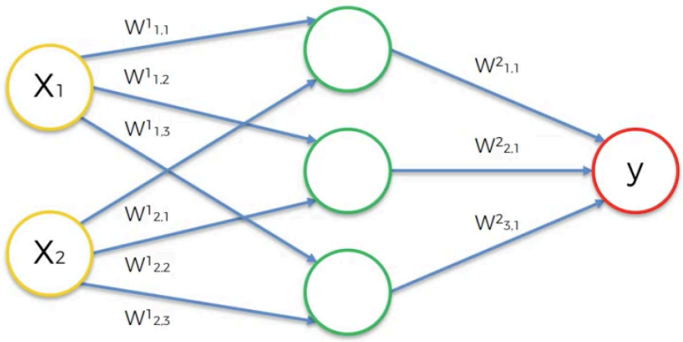
The Vanishing Gradient Problem

Image Source: recode.net

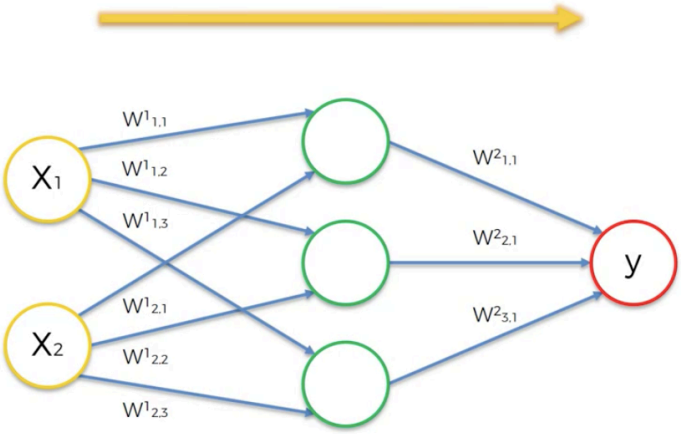
The Vanishing Gradient Problem



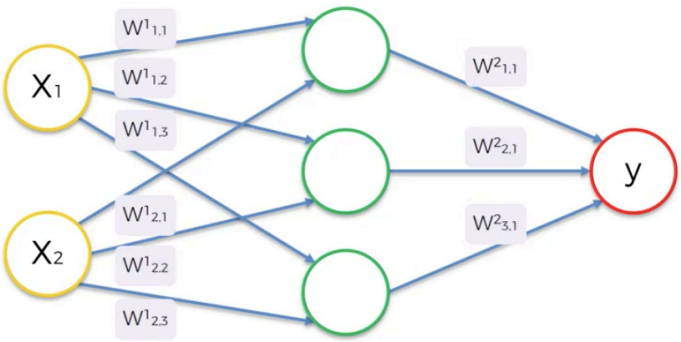
The Vanishing Gradient Problem



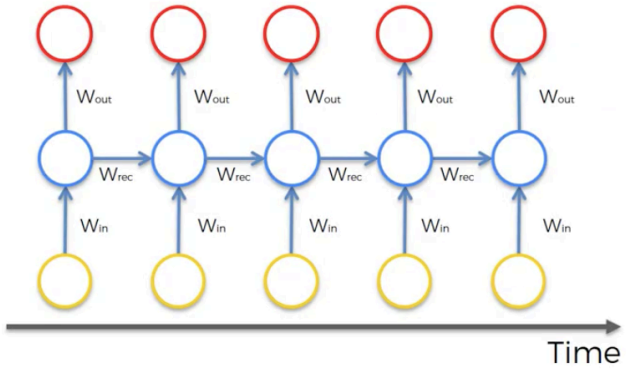
The Vanishing Gradient Problem



The Vanishing Gradient Problem

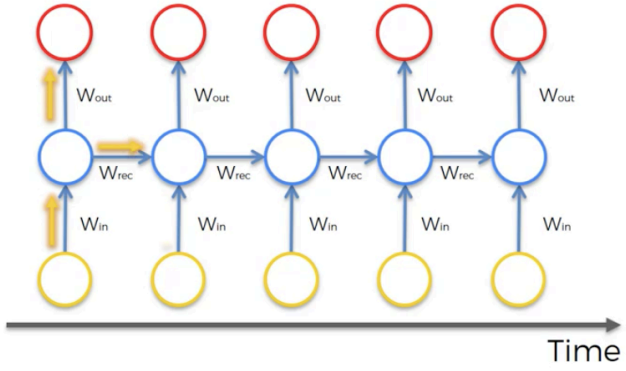


The Vanishing Gradient Problem



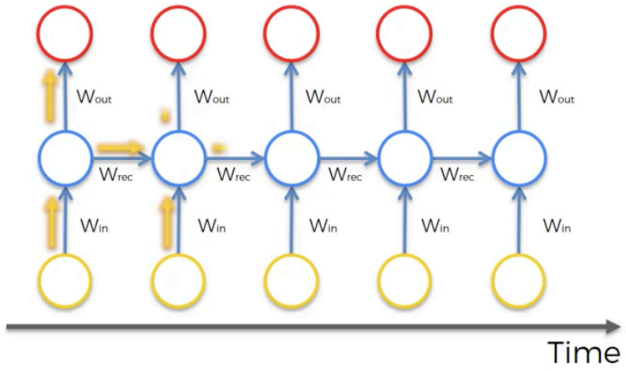
Formula Source: Razvan Pascanu et al. (2013)

The Vanishing Gradient Problem



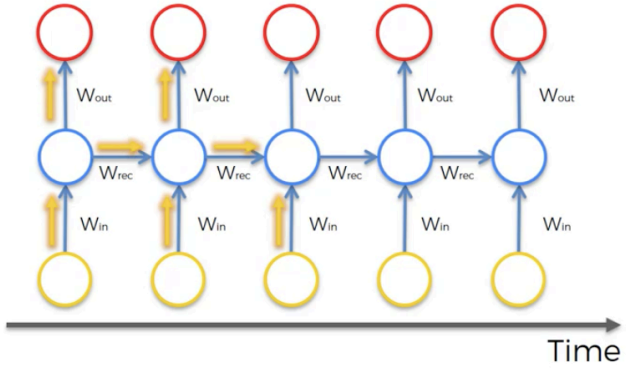
Formula Source: Razvan Pascanu et al. (2013)

The Vanishing Gradient Problem



Formula Source: Razvan Pascanu et al. (2013)

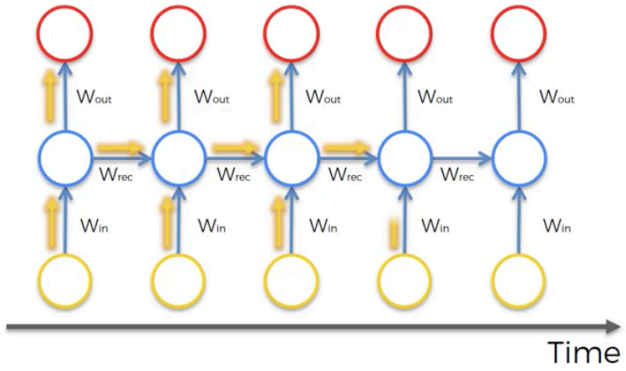
The Vanishing Gradient Problem



Formula Source: Razvan Pascanu et al. (2013)

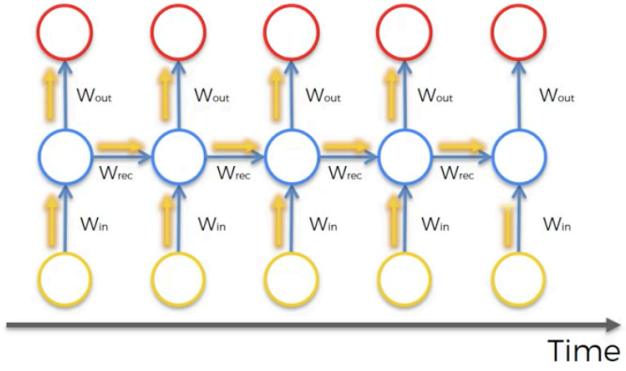
Deep Learning A-Z
© SuperDataScience

The Vanishing Gradient Problem



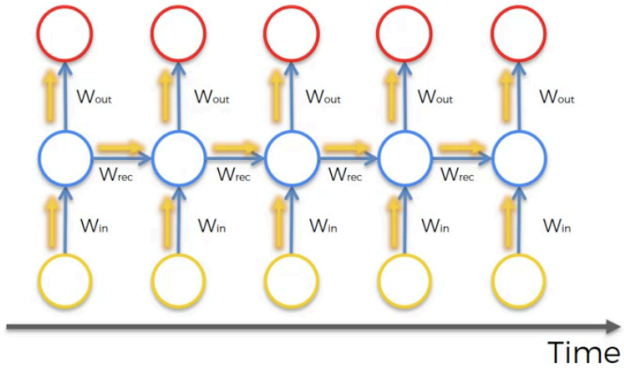
Formula Source: Razvan Pascanu et al. (2013)

The Vanishing Gradient Problem



Formula Source: Razvan Pascanu et al. (2013)

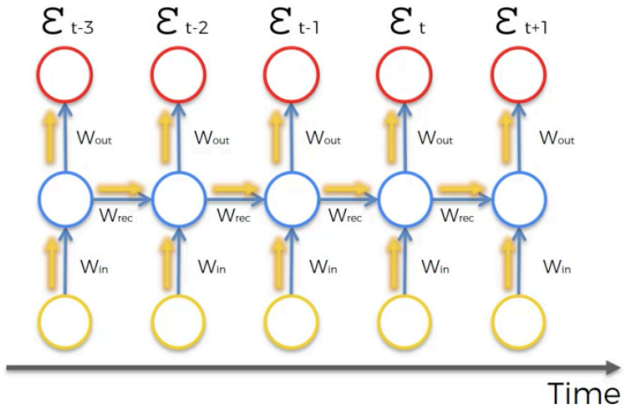
The Vanishing Gradient Problem



Formula Source: Razvan Pascanu et al. (2013)

Every single node in here is not just a node. But a representation of a whole layer. Remember that we looking at it from a different dimension.

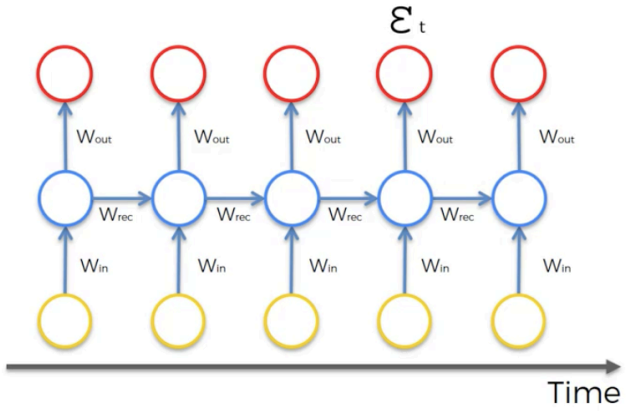
The Vanishing Gradient Problem



Formula Source: Razvan Pascanu et al. (2013)

The cost function or the error of the output.

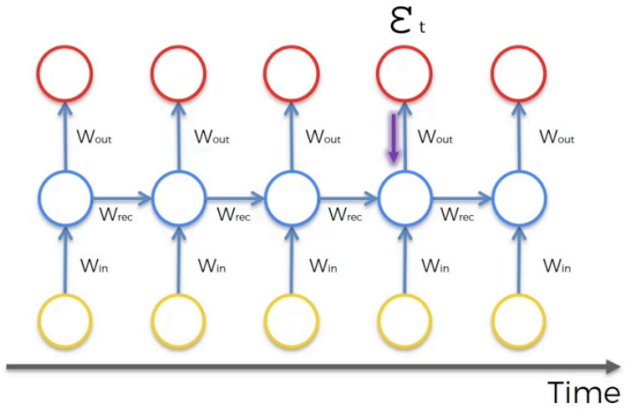
The Vanishing Gradient Problem



Formula Source: Razvan Pascanu et al. (2013)

Let's now just focus on one of the cost functions.

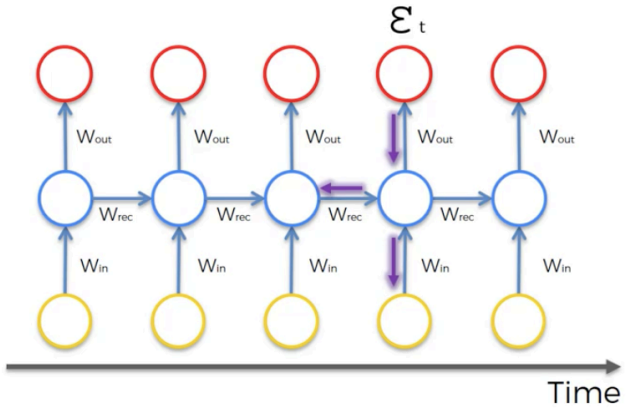
The Vanishing Gradient Problem



Formula Source: Razvan Pascanu et al. (2013)

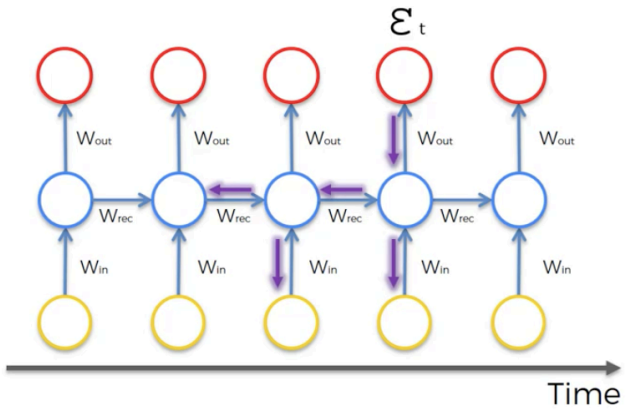
Now we want to propagate our cost function back through the network. Remember that every single neuron which participated in the calculation of the output associated with the cost function, their weight should be updated.
And it's not just the neurons below the red circle but it's all the neurons that contributed.

The Vanishing Gradient Problem



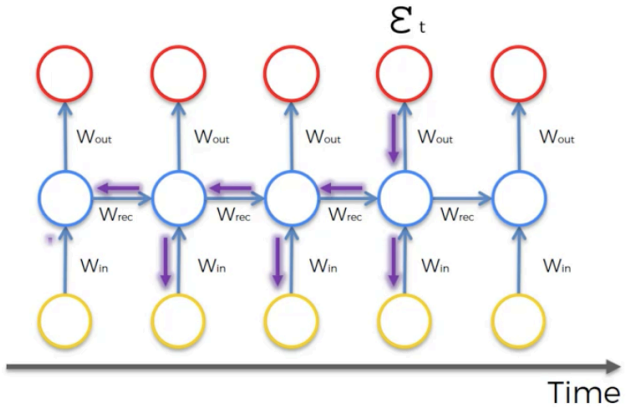
Formula Source: Razvan Pascanu et al. (2013)

The Vanishing Gradient Problem



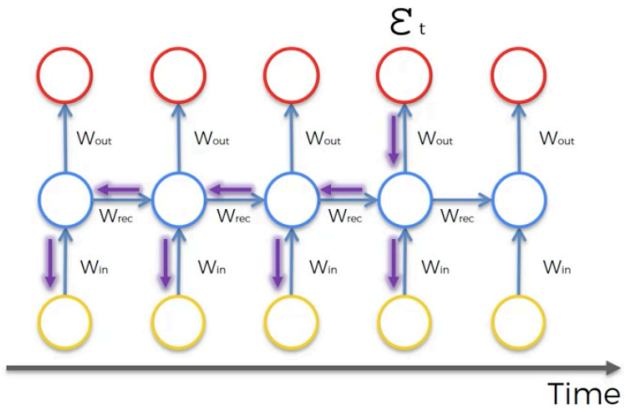
Formula Source: Razvan Pascanu et al. (2013)

The Vanishing Gradient Problem



Formula Source: Razvan Pascanu et al. (2013)

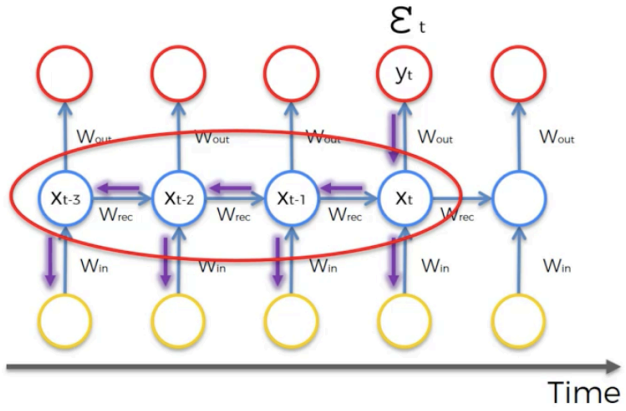
The Vanishing Gradient Problem



Formula Source: Razvan Pascanu et al. (2013)

Deep Learning A-Z
© SuperDataScience

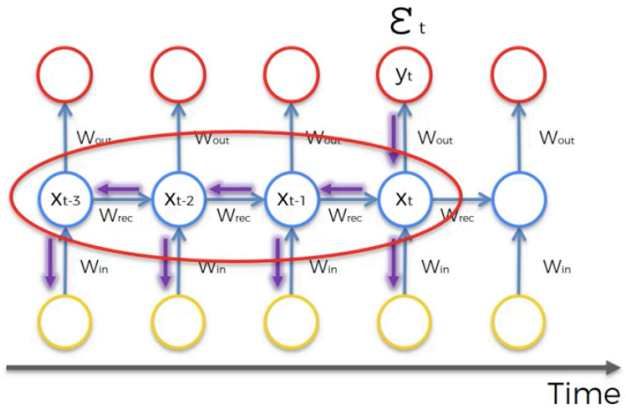
The Vanishing Gradient Problem



Formula Source: Razvan Pascanu et al. (2013)

That's where the problem lies because we have to update or propagate all the way back through these neurons

The Vanishing Gradient Problem



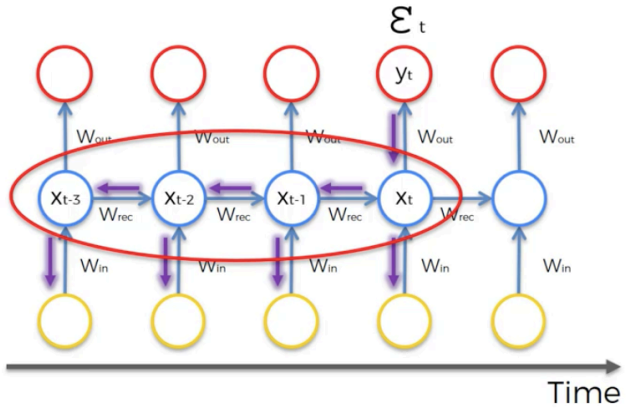
$$\frac{\partial \mathcal{E}}{\partial \theta} = \sum_{1 \le t \le T} \frac{\partial \mathcal{E}_t}{\partial \theta} \quad (3)$$

$$\frac{\partial \mathcal{E}_t}{\partial \theta} = \sum_{1 \le k \le t} \left(\frac{\partial \mathcal{E}_t}{\partial \mathbf{x}_t} \frac{\partial \mathbf{x}_t}{\partial \mathbf{x}_k} \frac{\partial^+ \mathbf{x}_k}{\partial \theta} \right) \quad (4)$$

$$\frac{\partial \mathbf{x}_t}{\partial \mathbf{x}_k} = \prod_{t \ge i > k} \frac{\partial \mathbf{x}_i}{\partial \mathbf{x}_{i-1}} = \prod_{t \ge i > k} \mathbf{W}_{rec}^T diag(\sigma'(\mathbf{x}_{i-1})) \quad (5)$$

Formula Source: Razvan Pascanu et al. (2013)

The Vanishing Gradient Problem



$$\frac{\partial \mathcal{E}}{\partial \theta} = \sum_{1 \le t \le T} \frac{\partial \mathcal{E}_t}{\partial \theta} \quad (3)$$

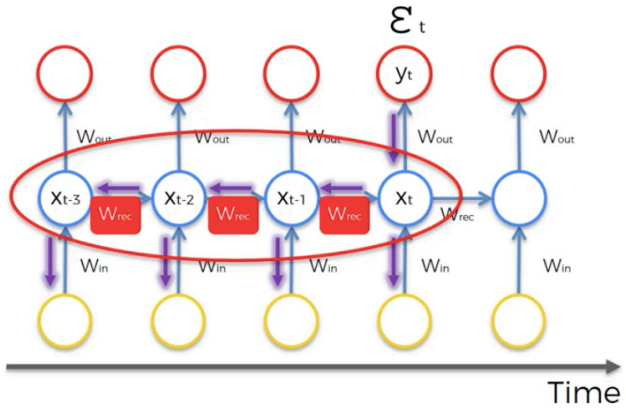
$$\frac{\partial \mathcal{E}_t}{\partial \theta} = \sum_{1 \le k \le t} \left(\frac{\partial \mathcal{E}_t}{\partial \mathbf{x}_t} \frac{\partial \mathbf{x}_t}{\partial \mathbf{x}_k} \frac{\partial^+ \mathbf{x}_k}{\partial \theta} \right) \quad (4)$$

$$\frac{\partial \mathbf{x}_t}{\partial \mathbf{x}_k} = \prod_{t \ge i > k} \frac{\partial \mathbf{x}_i}{\partial \mathbf{x}_{i-1}} = \prod_{t \ge i > k} \mathbf{W}_{rec}^T diag(\sigma'(\mathbf{x}_{i-1})) \quad (5)$$

Formula Source: Razvan Pascanu et al. (2013)

In here W_{rec} stands for weight and recurring and that's the weight used to connect the hidden layers to themselves in the unrolled temporal loop

The Vanishing Gradient Problem



$$\frac{\partial \mathcal{E}}{\partial \theta} = \sum_{1 \le t \le T} \frac{\partial \mathcal{E}_t}{\partial \theta} \quad (3)$$

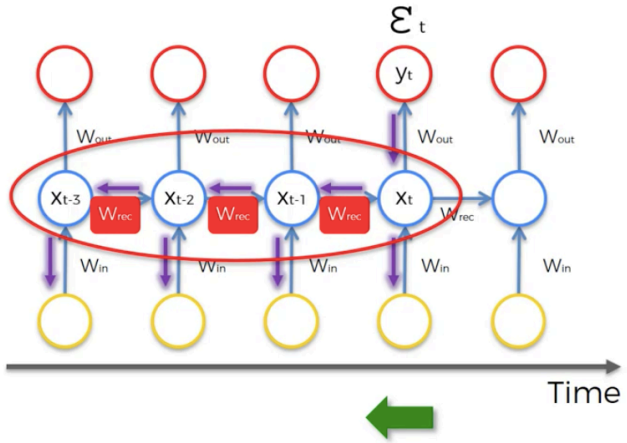
$$\frac{\partial \mathcal{E}_t}{\partial \theta} = \sum_{1 \le k \le t} \left(\frac{\partial \mathcal{E}_t}{\partial \mathbf{x}_t} \frac{\partial \mathbf{x}_t}{\partial \mathbf{x}_k} \frac{\partial^+ \mathbf{x}_k}{\partial \theta} \right) \quad (4)$$

$$\frac{\partial \mathbf{x}_t}{\partial \mathbf{x}_k} = \prod_{t \ge i > k} \frac{\partial \mathbf{x}_i}{\partial \mathbf{x}_{i-1}} = \prod_{t \ge i > k} \mathbf{W}_{rec}^T diag(\sigma'(\mathbf{x}_{i-1})) \quad (5)$$

Formula Source: Razvan Pascanu et al. (2013)

In order to go from X_{t-3} to X_{t-2}, we have to apply W_{rec} (multiply X_{t-3} by W_{rec} between them) and In order to go from X_{t-2} to X_{t-1}, we have to apply W_{rec} (multiply X_{t-2} by W_{rec} between them) and so on. Remember that we are multiplying by the same weight every time.
The multiplication comes from the ∏ (pi)

The Vanishing Gradient Problem



$$\frac{\partial \mathcal{E}}{\partial \theta} = \sum_{1 \le t \le T} \frac{\partial \mathcal{E}_t}{\partial \theta} \quad (3)$$

$$\frac{\partial \mathcal{E}_t}{\partial \theta} = \sum_{1 \le k \le t} \left(\frac{\partial \mathcal{E}_t}{\partial \mathbf{x}_t} \frac{\partial \mathbf{x}_t}{\partial \mathbf{x}_k} \frac{\partial^+ \mathbf{x}_k}{\partial \theta} \right) \quad (4)$$

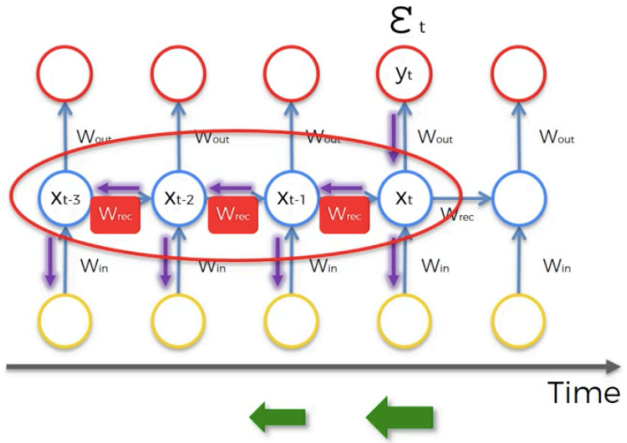
$$\frac{\partial \mathbf{x}_t}{\partial \mathbf{x}_k} = \prod_{t \ge i > k} \frac{\partial \mathbf{x}_i}{\partial \mathbf{x}_{i-1}} = \prod_{t \ge i > k} \mathbf{W}_{rec}^T diag(\sigma'(\mathbf{x}_{i-1})) \quad (5)$$

Formula Source: Razvan Pascanu et al. (2013)

Deep Learning A-Z © SuperDataScience

weight are usually assigned at the start of neural network to a random value close to 0. so because at each step we are multiplying them, the more we multiply them, the more small the value gets.

The Vanishing Gradient Problem



$$\frac{\partial \mathcal{E}}{\partial \theta} = \sum_{1 \le t \le T} \frac{\partial \mathcal{E}_t}{\partial \theta} \quad (3)$$

$$\frac{\partial \mathcal{E}_t}{\partial \theta} = \sum_{1 \le k \le t} \left(\frac{\partial \mathcal{E}_t}{\partial \mathbf{x}_t} \frac{\partial \mathbf{x}_t}{\partial \mathbf{x}_k} \frac{\partial^+ \mathbf{x}_k}{\partial \theta} \right) \quad (4)$$

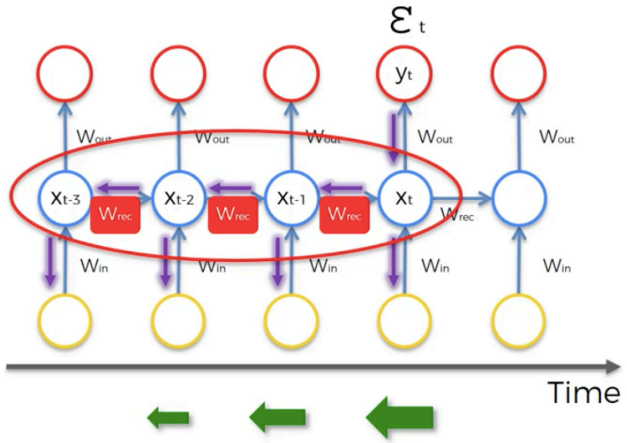
$$\frac{\partial \mathbf{x}_t}{\partial \mathbf{x}_k} = \prod_{t \ge i > k} \frac{\partial \mathbf{x}_i}{\partial \mathbf{x}_{i-1}} = \prod_{t \ge i > k} \mathbf{W}_{rec}^T diag(\sigma'(\mathbf{x}_{i-1})) \quad (5)$$

Formula Source: Razvan Pascanu et al. (2013)

Deep Learning A-Z © SuperDataScience

By moving backward (from right to left), our gradient becomes less and less. This means to the neural network that a vanishing gradient will happen. As the gradient goes back through network, it's used to update the weights. So the lower the gradient is, the harder (slower) it is for network to update the weights. The higher the gradient is, the faster it updates the weights.

The Vanishing Gradient Problem



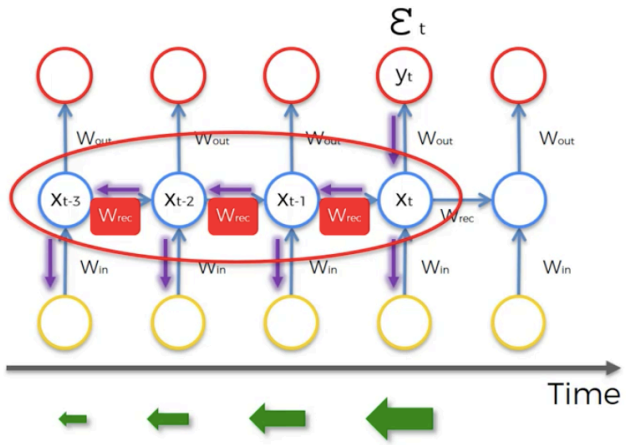
$$\frac{\partial \mathcal{E}}{\partial \theta} = \sum_{1 \leq t \leq T} \frac{\partial \mathcal{E}_t}{\partial \theta} \quad (3)$$

$$\frac{\partial \mathcal{E}_t}{\partial \theta} = \sum_{1 \leq k \leq t} \left(\frac{\partial \mathcal{E}_t}{\partial \mathbf{x}_t} \frac{\partial \mathbf{x}_t}{\partial \mathbf{x}_k} \frac{\partial^+ \mathbf{x}_k}{\partial \theta} \right) \quad (4)$$

$$\frac{\partial \mathbf{x}_t}{\partial \mathbf{x}_k} = \prod_{t \geq i > k} \frac{\partial \mathbf{x}_i}{\partial \mathbf{x}_{i-1}} = \prod_{t \geq i > k} \mathbf{W}_{rec}^T diag(\sigma'(\mathbf{x}_{i-1})) \quad (5)$$

Formula Source: Razvan Pascanu et al. (2013)

The Vanishing Gradient Problem



$$\frac{\partial \mathcal{E}}{\partial \theta} = \sum_{1 \leq t \leq T} \frac{\partial \mathcal{E}_t}{\partial \theta} \quad (3)$$

$$\frac{\partial \mathcal{E}_t}{\partial \theta} = \sum_{1 \leq k \leq t} \left(\frac{\partial \mathcal{E}_t}{\partial \mathbf{x}_t} \frac{\partial \mathbf{x}_t}{\partial \mathbf{x}_k} \frac{\partial^+ \mathbf{x}_k}{\partial \theta} \right) \quad (4)$$

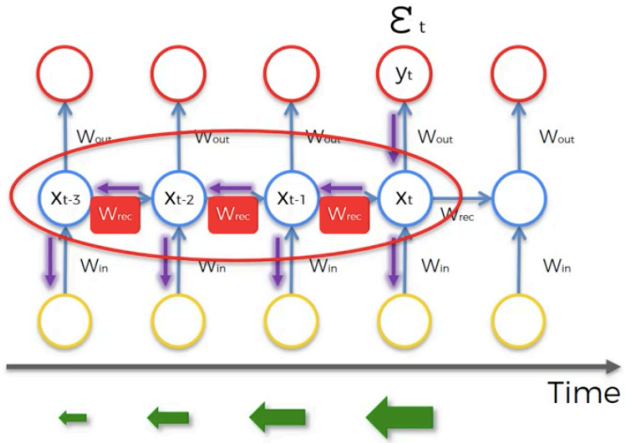
$$\frac{\partial \mathbf{x}_t}{\partial \mathbf{x}_k} = \prod_{t \geq i > k} \frac{\partial \mathbf{x}_i}{\partial \mathbf{x}_{i-1}} = \prod_{t \geq i > k} \mathbf{W}_{rec}^T diag(\sigma'(\mathbf{x}_{i-1})) \quad (5)$$

Formula Source: Razvan Pascanu et al. (2013)

Deep Learning A-Z

© SuperDataScience

The Vanishing Gradient Problem



$$\frac{\partial \mathcal{E}}{\partial \theta} = \sum_{1 \le t \le T} \frac{\partial \mathcal{E}_t}{\partial \theta} \quad (3)$$

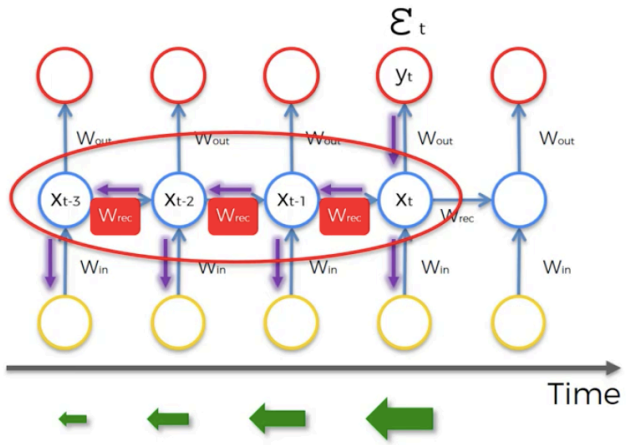
$$\frac{\partial \mathcal{E}_t}{\partial \theta} = \sum_{1 \le k \le t} \left(\frac{\partial \mathcal{E}_t}{\partial \mathbf{x}_t} \frac{\partial \mathbf{x}_t}{\partial \mathbf{x}_k} \frac{\partial^+ \mathbf{x}_k}{\partial \theta} \right) \quad (4)$$

$$\frac{\partial \mathbf{x}_t}{\partial \mathbf{x}_k} = \prod_{t \ge i > k} \frac{\partial \mathbf{x}_i}{\partial \mathbf{x}_{i-1}} = \prod_{t \ge i > k} \mathbf{W}_{rec}^T diag(\sigma'(\mathbf{x}_{i-1})) \quad (5)$$

W_{rec} ~ small

Formula Source: Razvan Pascanu et al. (2013)

The Vanishing Gradient Problem



$$\frac{\partial \mathcal{E}}{\partial \theta} = \sum_{1 \le t \le T} \frac{\partial \mathcal{E}_t}{\partial \theta} \quad (3)$$

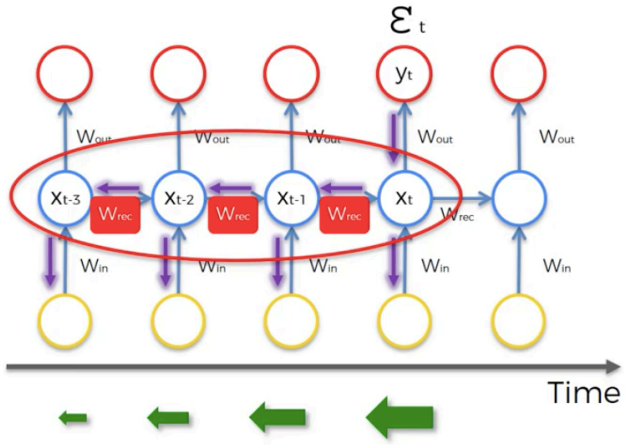
$$\frac{\partial \mathcal{E}_t}{\partial \theta} = \sum_{1 \le k \le t} \left(\frac{\partial \mathcal{E}_t}{\partial \mathbf{x}_t} \frac{\partial \mathbf{x}_t}{\partial \mathbf{x}_k} \frac{\partial^+ \mathbf{x}_k}{\partial \theta} \right) \quad (4)$$

$$\frac{\partial \mathbf{x}_t}{\partial \mathbf{x}_k} = \prod_{t \ge i > k} \frac{\partial \mathbf{x}_i}{\partial \mathbf{x}_{i-1}} = \prod_{t \ge i > k} \mathbf{W}_{rec}^T diag(\sigma'(\mathbf{x}_{i-1})) \quad (5)$$

W_{rec} ~ small ⇨ Vanishing

Formula Source: Razvan Pascanu et al. (2013)

The Vanishing Gradient Problem



$$\frac{\partial \mathcal{E}}{\partial \theta} = \sum_{1 \le t \le T} \frac{\partial \mathcal{E}_t}{\partial \theta} \quad (3)$$

$$\frac{\partial \mathcal{E}_t}{\partial \theta} = \sum_{1 \le k \le t} \left(\frac{\partial \mathcal{E}_t}{\partial \mathbf{x}_t} \frac{\partial \mathbf{x}_t}{\partial \mathbf{x}_k} \frac{\partial^+ \mathbf{x}_k}{\partial \theta} \right) \quad (4)$$

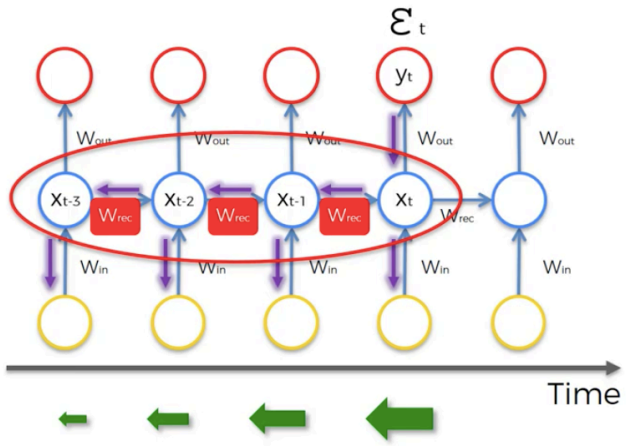
$$\frac{\partial \mathbf{x}_t}{\partial \mathbf{x}_k} = \prod_{t \ge i > k} \frac{\partial \mathbf{x}_i}{\partial \mathbf{x}_{i-1}} = \prod_{t \ge i > k} \mathbf{W}_{rec}^T diag(\sigma'(\mathbf{x}_{i-1})) \quad (5)$$

W_{rec} ~ small ⇨ Vanishing

W_{rec} ~ large ⇨

Formula Source: Razvan Pascanu et al. (2013)

The Vanishing Gradient Problem



$$\frac{\partial \mathcal{E}}{\partial \theta} = \sum_{1 \le t \le T} \frac{\partial \mathcal{E}_t}{\partial \theta} \quad (3)$$

$$\frac{\partial \mathcal{E}_t}{\partial \theta} = \sum_{1 \le k \le t} \left(\frac{\partial \mathcal{E}_t}{\partial \mathbf{x}_t} \frac{\partial \mathbf{x}_t}{\partial \mathbf{x}_k} \frac{\partial^+ \mathbf{x}_k}{\partial \theta} \right) \quad (4)$$

$$\frac{\partial \mathbf{x}_t}{\partial \mathbf{x}_k} = \prod_{t \ge i > k} \frac{\partial \mathbf{x}_i}{\partial \mathbf{x}_{i-1}} = \prod_{t \ge i > k} \mathbf{W}_{rec}^T diag(\sigma'(\mathbf{x}_{i-1})) \quad (5)$$

W_{rec} ~ small ⇨ Vanishing

W_{rec} ~ large ⇨ Exploding

Formula Source: Razvan Pascanu et al. (2013)

Deep Learning A-Z © SuperDataScience

The large is also the same, but it explodes. For example, in large, at step one we going to have 100. by step two, we're going to have 1000 and so on.

The Vanishing Gradient Problem

Solutions:

The Vanishing Gradient Problem

Solutions:

1. Exploding Gradient

The Vanishing Gradient Problem

Solutions:

1. Exploding Gradient
   - Truncated Backpropagation

To stop backpropagating at a certain point but it's not optimal. Although, if we don't stop at a certain point we get an irrelevant network so it's better than the original approach.

The Vanishing Gradient Problem

Solutions:

1. Exploding Gradient
   - Truncated Backpropagation
   - Penalties

Gradient to being penalize and being artificially reduce.

The Vanishing Gradient Problem

Solutions:

1. Exploding Gradient
   - Truncated Backpropagation
   - Penalties
   - Gradient Clipping

Having a maximum limit for the gradient so our gradient never go over this value. And if it's over that value then it will stay at that value.

The Vanishing Gradient Problem

Solutions:

1. Exploding Gradient
   - Truncated Backpropagation
   - Penalties
   - Gradient Clipping

2. Vanishing Gradient

The Vanishing Gradient Problem

Solutions:

1. Exploding Gradient
   - Truncated Backpropagation
   - Penalties
   - Gradient Clipping
2. Vanishing Gradient
   - Weight Initialization

Smartly initialising the weights to minimize the potential for vanishing gradient

The Vanishing Gradient Problem

Solutions:

1. Exploding Gradient
   - Truncated Backpropagation
   - Penalties
   - Gradient Clipping
2. Vanishing Gradient
   - Weight Initialization
   - Echo State Networks

Not in this course but they're design to solve the vanishing gradient problem.

The Vanishing Gradient Problem

Solutions:

1. Exploding Gradient
   - Truncated Backpropagation
   - Penalties
   - Gradient Clipping
2. Vanishing Gradient
   - Weight Initialization
   - Echo State Networks
   - Long Short-Term Memory Networks (LSTMs)

This method is extremely popular and go to for implementing.

The Vanishing Gradient Problem

Solutions:

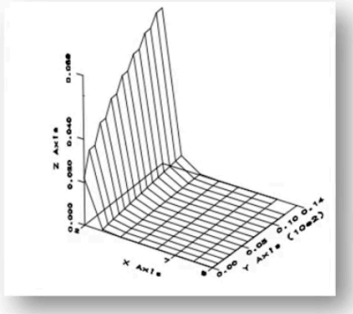
1. Exploding Gradient
   - Truncated Backpropagation
   - Penalties
   - Gradient Clipping
2. Vanishing Gradient
   - Weight Initialization
   - Echo State Networks
   - Long Short-Term Memory Networks (LSTMs)

The Vanishing Gradient Problem

Additional Reading:

Untersuchungen zu dynamischen neuronalen Netzen

By Sepp (Josef) Hochreiter (1991)



Link:

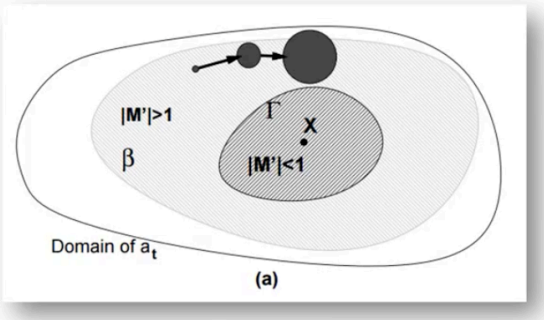
http://people.idsia.ch/~juergen/SeppHochreiter1991ThesisAdvisorSchmidhuber.pdf

The Vanishing Gradient Problem

Additional Reading:

Learning Long-Term Dependencies with Gradient Descent is Difficult

By Yoshua Bengio et al. (1994)



Link:

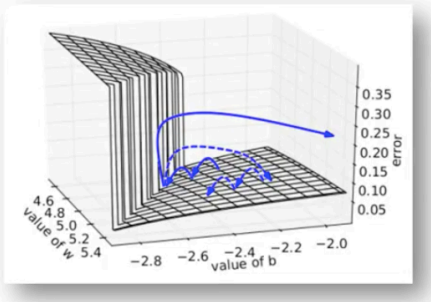
http://www-dsi.ing.unifi.it/~paolo/ps/tnn-94-gradient.pdf

The Vanishing Gradient Problem

Additional Reading:

On the difficulty of training recurrent neural networks

By Razvan Pascanu et al. (2013)



Link:

http://www.jmlr.org/proceedings/papers/v28/pascanu13.pdf