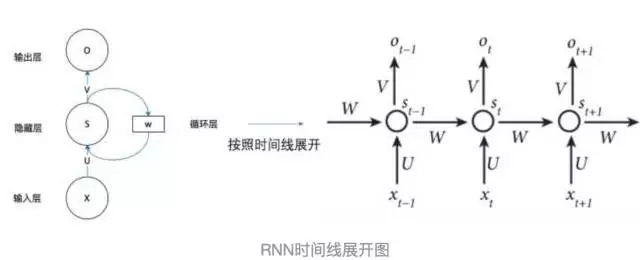
**Recurrent Neural Network**

**1. RNN的工作原理是什么？**

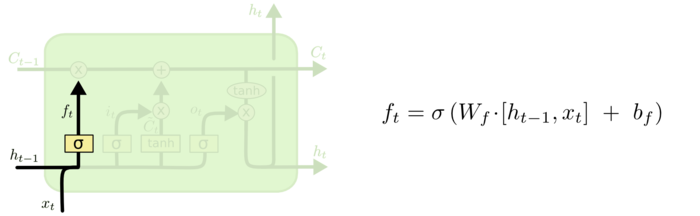
基础的神经网络只在层与层之间建立了权连接，RNN最大的不同之处就是在层之间的神经元之间也建立的权连接。



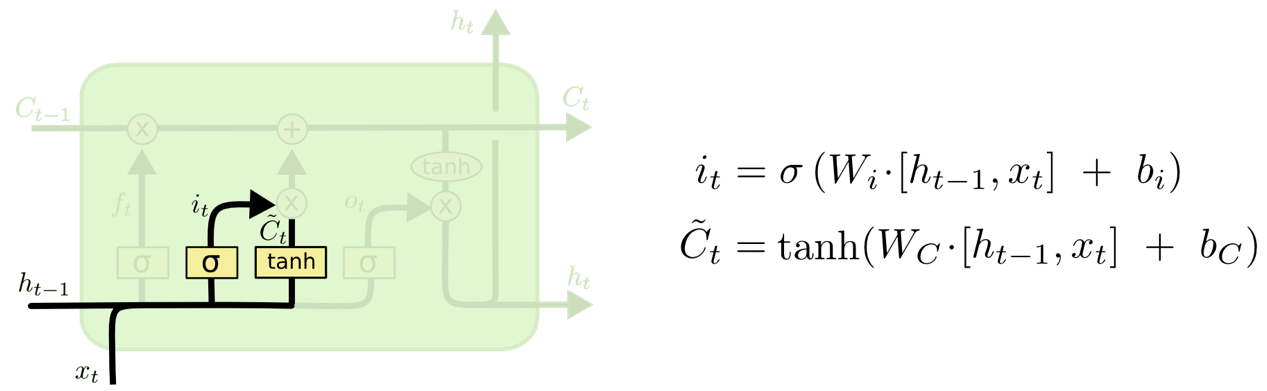
对于t时刻的前向传播算法公式如下：

其中，为激活函数，一般会选择tanh函数，b为偏置。

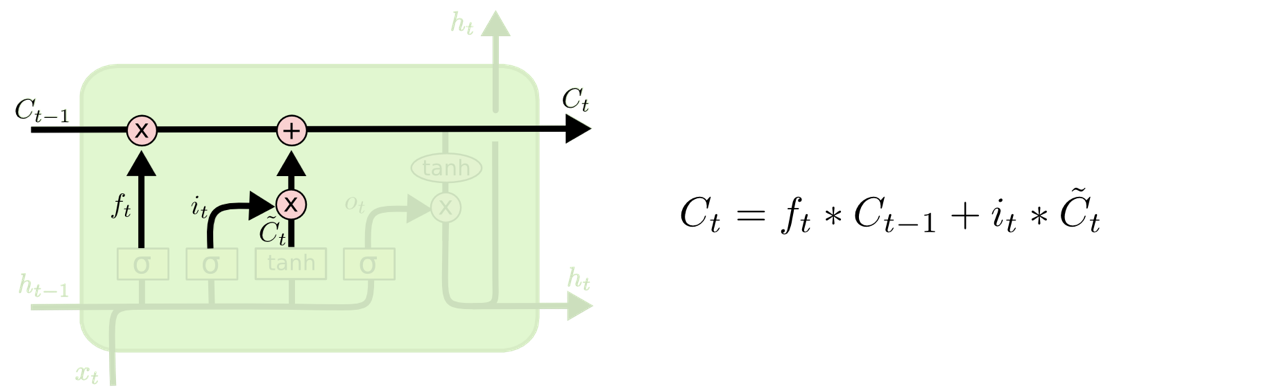
**2. LSTM的工作原理是什么？**

**遗忘门：**LSTM 中的第一步是决定我们会从细胞状态中丢弃什么信息。这个决定通过一个称为遗忘门完成。该门会读取和，输出一个在 0 到 1 之间的数值给每个在细胞状态中的数字。1 表示“完全保留”，0 表示“完全舍弃”。

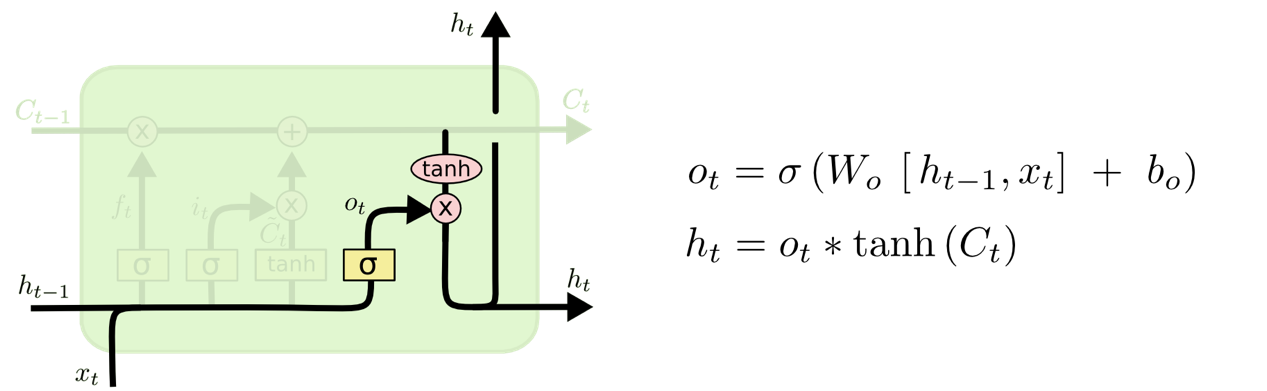
**输入门：**下一步是确定什么样的新信息被存放在细胞状态中。这里包含两个部分。第一，sigmoid 层称 “输入门层” 决定什么值我们将要更新。然后，一个tanh层创建一个新的候选值向量，会被加入到状态中。



sigmoid函数选择更新内容，tanh函数创建更新候选。



**输出门：**



**3. LSTM为什么能防止梯度消失？**

通过公式可以看出，在LSTM中，，其中是此前的信息，是当前时刻的新信息，是最终信息。可以看到和此时是线性关系，不再是RNN中的乘积关系，因此梯度在计算的时候不再是连乘关系，因此梯度在计算的时候不再是连乘关系，梯度以线性在中间节点流动，因此就会保证很长时间的记忆。

**4. RNN为什么会出现梯度消失？**

RNN之所以会梯度消失还是由于两个原因：

第一，参数的初始化，假如参数的初始都比较小，那么在BP传播的时候，那么从最后一层往前传，由于参数过小的话，那么往前传越小，一方面会造成梯度消失的问题，另一方面会造成对于时间步长过长造成信息丢失。

第二，原因是激活的函数的原因，不管是使用tanh，他的有时候 还是无法根本解决这个问题，但大多数tanh的激活函数在0附近较大，当在[5，-5]以外的时候，导数已经非常小了，就会造成梯度小时。

为啥刚开始不使用relu作为激活函数呢？刚开始在论文里，relu就是为了解决梯度消失而发明的，并且在RNN中work的非常好，并且RNN用relu和LSTM的性能差距其实非常小，3%左右，如果权重衰减调的好，基本没有差距。weight decay是放在正则项（regularization）前面的一个系数，正则项一般指示模型的复杂度，所以weight decay的作用是调节模型复杂度对损失函数的影响，若weight decay很大，则复杂的模型损失函数的值也就大。

**5. 为什么LSTM模型中既存在sigmoid又存在tanh两种激活函数？**

二者目的不一样，sigmoid 用在了各种gate上，产生0~1之间的值，这个一般只有sigmoid最直接了。tanh 用在了状态和输出上，是对数据的处理，这个用其他激活函数或许也可以。