循环神经网络与LSTM

主要内容

- 神经网络与循环神经网络
  1. 强大的功能
  2. 层级结构
  3. 多种RNN
- LSTM
  1. 长时依赖问题
  2. “记忆细胞”与状态
- LSTM变体
  1. GRU等

循环神经网络与应用

- 模仿论文(连公式都格式很正确)

For $\bigoplus_{n=1,\ldots,m}$ where $\mathcal{L}_{m_\bullet} = 0$, hence we can find a closed subset $\mathcal{H}$ in $\mathcal{H}$ and any sets $\mathcal{F}$ on X, U is a closed immersion of S, then $U \to T$ is a separated algebraic space.

Proof. Proof of (1). It also start we get

$$S = \mathrm{Spec}(R) = U \times_X U \times_X U$$

and the comparicoly in the fibre product covering we have to prove the lemma generated by $\coprod Z \times_U U \to V$. Consider the maps M along the set of points Sch_{fppf} and $U \to U$ is the fibre category of S in U in Section, **??** and the fact that any U affine, see Morphisms, Lemma **??**. Hence we obtain a scheme S and any open subset $W \subset U$ in $Sh(G)$ such that $\mathrm{Spec}(R') \to S$ is smooth or an

$$U = \bigcup U_i \times_{S_i} U_i$$

which has a nonzero morphism we may assume that f_i is of finite presentation over S. We claim that $\mathcal{O}_{X,x}$ is a scheme where $x, x', s'' \in S'$ such that $\mathcal{O}_{X,x'} \to \mathcal{O}'_{X',x'}$ is separated. By Algebra, Lemma **??** we can define a map of complexes $\mathrm{GL}_{S'}(x'/S'')$ and we win. □

To prove study we see that $\mathcal{F}|_U$ is a covering of $\mathcal{X}'$, and $\mathcal{T}_i$ is an object of $\mathcal{F}_{X/S}$ for $i > 0$ and $\mathcal{F}_p$ exists and let $\mathcal{F}_i$ be a presheaf of $\mathcal{O}_X$-modules on $\mathcal{C}$ as a $\mathcal{F}$-module. In particular $\mathcal{F} = U/\mathcal{F}$ we have to show that

$$\widetilde{M}^\bullet = \mathcal{I}^\bullet \otimes_{\mathrm{Spec}(k)} \mathcal{O}_{S,s} - i_X^{-1}\mathcal{F})$$

is a unique morphism of algebraic stacks. Note that

$$\mathrm{Arrows} = (Sch/S)^{opp}_{fppf}, (Sch/S)_{fppf}$$

and

$$V = \Gamma(S, \mathcal{O}) \longmapsto (U, \mathrm{Spec}(A))$$

is an open subset of X. Thus U is affine. This is a continuous map of X is the inverse, the groupoid scheme S.

Proof. See discussion of sheaves of sets. □

The result for prove any open covering follows from the less of Example **??**. It may replace S by $X_{spaces,étale}$ which gives an open subspace of X and T equal to S_{Zar}, see Descent, Lemma **??**. Namely, by Lemma **??** we see that R is geometrically regular over S.

Lemma 0.1. *Assume (3) and (3) by the construction in the description.*

Suppose $X = \lim |X|$ (by the formal open covering X and a single map $\underline{Proj}_X(\mathcal{A}) =$ Spec(B) over U compatible with the complex

$$Set(\mathcal{A}) = \Gamma(X, \mathcal{O}_{X,\mathcal{O}_X}).$$

*When in this case of to show that $\mathcal{Q} \to \mathcal{C}_{Z/X}$ is stable under the following result in the second conditions of (1), and (3). This finishes the proof. By Definition **??** (without element is when the closed subschemes are catenary. If T is surjective we may assume that T is connected with residue fields of S. Moreover there exists a closed subspace $Z \subset X$ of X where U in X' is proper (some defining as a closed subset of the uniqueness it suffices to check the fact that the following theorem*

(1) f is locally of finite type. Since $S =$ Spec(R) and $Y =$ Spec(R).

Proof. This is form all sheaves of sheaves on X. But given a scheme U and a surjective étale morphism $U \to X$. Let $U \cap U = \coprod_{i=1,\ldots,n} U_i$ be the scheme X over S at the schemes $X_i \to X$ and $U = \lim_i X_i$. □

The following lemma surjective restrocomposes of this implies that $\mathcal{F}_{x_0} = \mathcal{F}_{x_0} = \mathcal{F}_{X,\ldots,0}$.

Lemma 0.2. *Let X be a locally Noetherian scheme over S, $E = \mathcal{F}_{X/S}$. Set $\mathcal{I} = \mathcal{J}_1 \subset \mathcal{I}'_n$. Since $\mathcal{I}^n \subset \mathcal{I}^n$ are nonzero over $i_0 \leq \mathfrak{p}$ is a subset of $\mathcal{J}_{n,0} \circ \overline{A}_2$ works.*

Lemma 0.3. *In Situation **??**. Hence we may assume $\mathfrak{q}' = 0$.*

Proof. We will use the property we see that $\mathfrak{p}$ is the mext functor (**??**). On the other hand, by Lemma **??** we see that

$$D(\mathcal{O}_{X'}) = \mathcal{O}_X(D)$$

where K is an F-algebra where δ_{n+1} is a scheme over S. □

循环神经网络与应用

- 模仿莎士比亚的作品

```
PANDARUS:
Alas, I think he shall be come approached and the day
When little srain would be attain'd into being never fed,
And who is but a chain and subjects of his death,
I should not sleep.

Second Senator:
They are away this miseries, produced upon my soul,
Breaking and strongly should be buried, when I perish
The earth and thoughts of many states.

DUKE VINCENTIO:
Well, your wit is in the care of side and that.

Second Lord:
They would be ruled after this chamber, and
my fair nues begun out of the fact, to be conveyed,
Whose noble souls I'll have the heart of the wars.

Clown:
Come, sir, I will make did behold your worship.

VIOLA:
I'll drink it.
```

循环神经网络与应用

模仿小四的作品

每个人，闭上眼睛的时候，才能真正面对光明

他们在吱呀作响的船舷上，静静看着世界，没有痛苦的声音，碎裂的海洋里摇晃出阵阵沉默，吞噬过来。他们的躯体，一点，一点，逐渐暗淡在

你们虔诚的看着远方，我抬起头，不经意间，目光划过你们的面庞，上面淡淡的倔强印，那么坚强

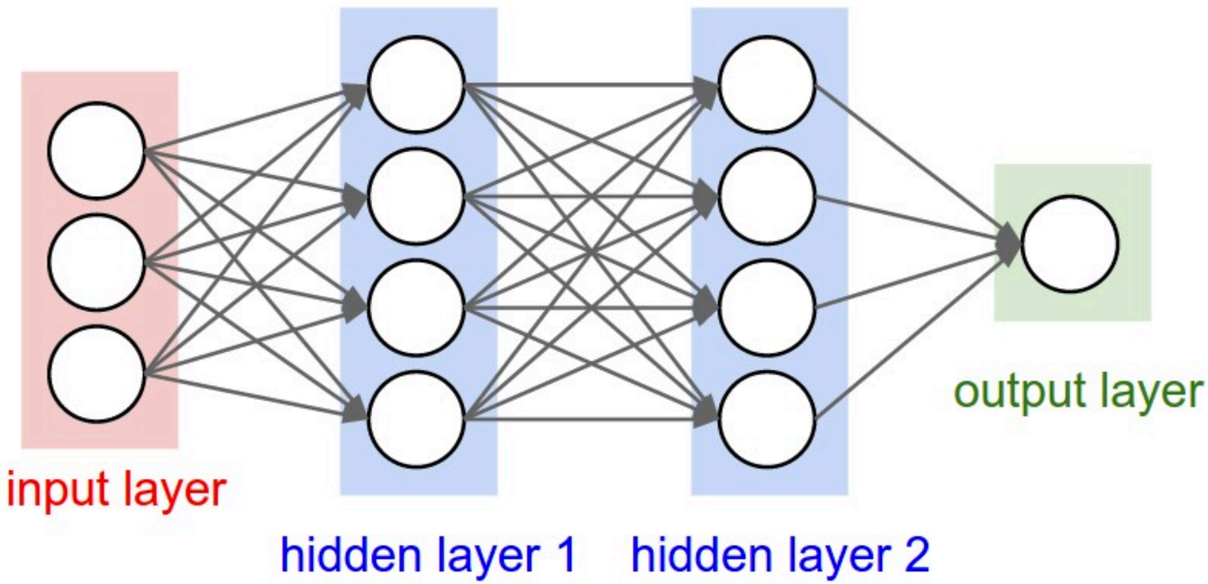
尘世凡间
沉睡亿万光年的年轻战士
萦绕不散的寂寞烟云中
静候在末世岛屿之上
守候，女王何时归来
你的目光延向她迟归的方向
缓缓推进的海浪
这最后一夜
荡漾

循环神经网络与应用

- 看图说话

神经网络到循环神经网络

- 我们知道神经网络结构如下



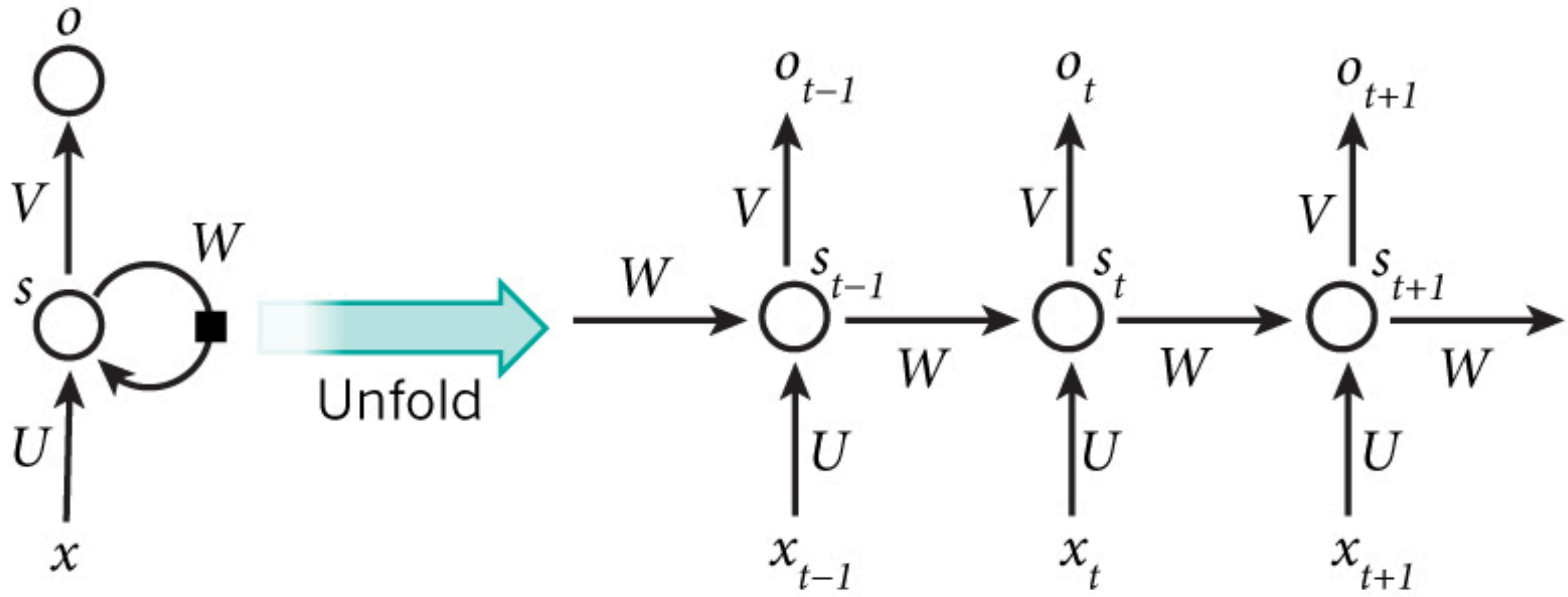
- 那循环神经网络和它是什么关系呢?

循环神经网络

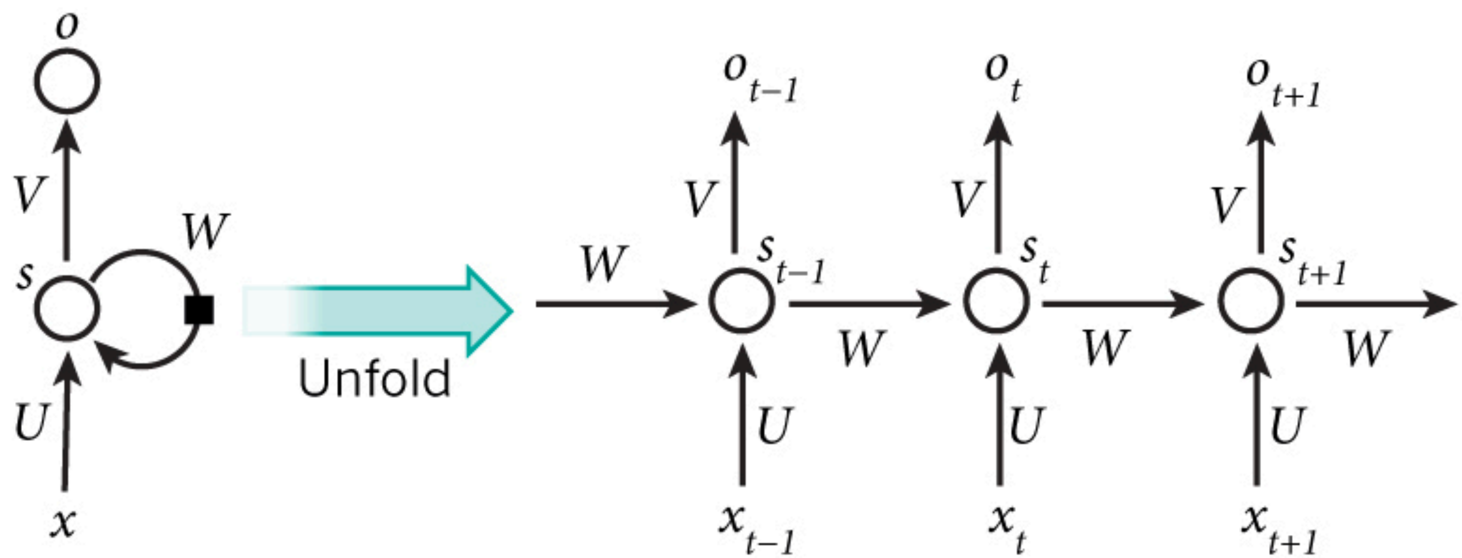
- 为什么有BP神经网络，CNN，还要RNN？
  - 传统神经网络(包括CNN)，输入和输出都是互相独立的。
    - 图像上的猫和狗是分隔开的，但有些任务，后续的输出和之前的内容是相关的。
    - “我是中国人，我的母语是___”
  - RNN引入“记忆”的概念
    - 循环2字来源于其每个元素都执行相同的任务。
    - 但是输出依赖于 输入 和 “记忆”

循环神经网络之 结构

- 简单来看，把序列按时间展开



循环神经网络之 结构



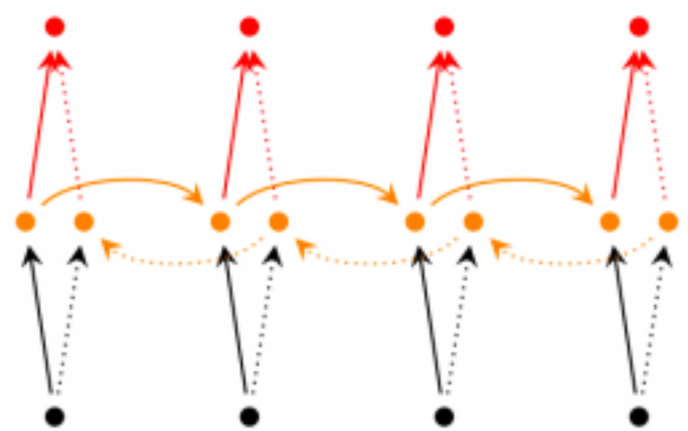
- X_t是时间t处的输入
- S_t是时间t处的“记忆”，$S_t=f(UX_t+WS_{t-1})$，f可以是tanh等
- O_t是时间t出的输出，比如是预测下个词的话，可能是softmax输出的属于每个候选词的概率

循环神经网络之 结构细节

- 可以把隐状态S_t视作“记忆体”，捕捉了之前时间点上的信息。
- 输出O_t由当前时间及之前所有的“记忆”共同计算得到。
- 很可惜，实际应用中，S_t并不能捕捉和保留之前所有信息（记忆有限？）
- 不同于CNN，这里的RNN其实整个神经网络都共享一组参数（U,V,W），极大减小了需要训练和预估的参数量
- 图中的O_t在有些任务下是不存在的，比如文本情感分析，其实只需要最后的output结果就行

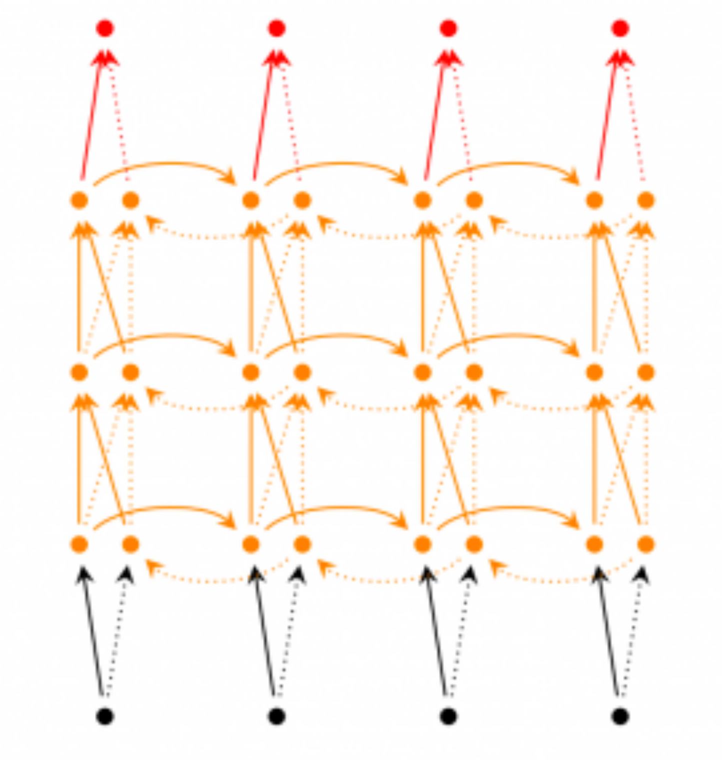
不同类型的RNN

- 双向RNN
  - 有些情况下，当前的输出不只依赖于之前的序列元素，还可能依赖之后的序列元素
  - 比如从一段话踢掉部分词，让你补全
  - 直观理解：2个RNN叠加



不同类型的RNN

- 深层双向RNN
  - 和双向RNN的区别是每一步/每个时间点我们设定多层结构



循环神经网络之LSTM

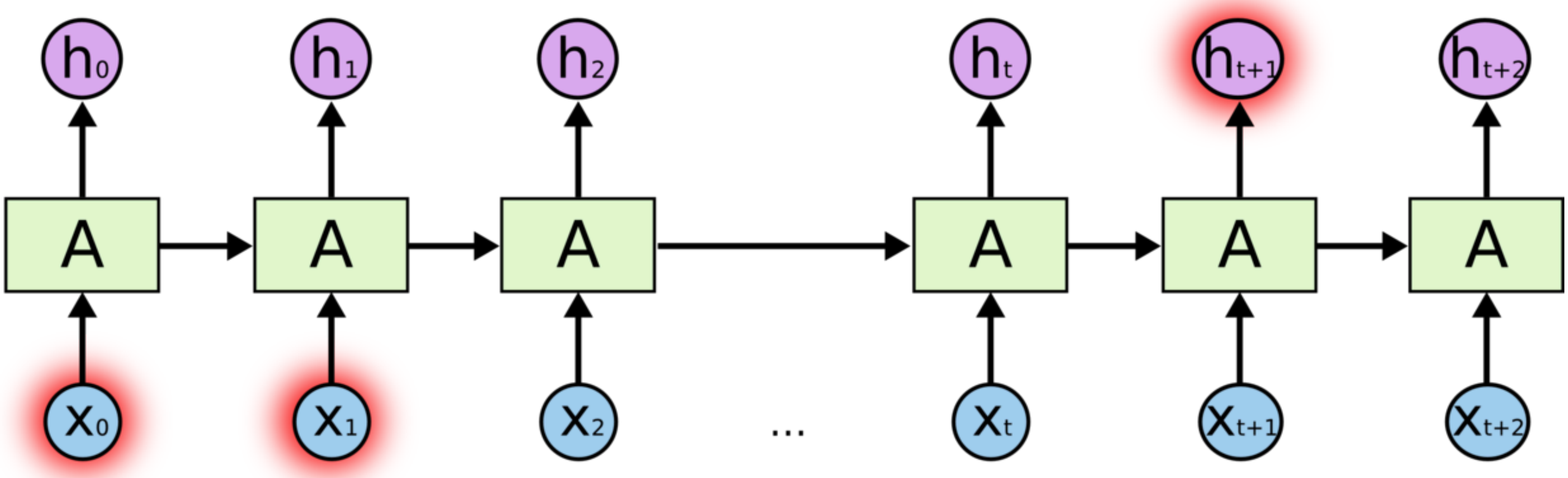
- 前面提到的RNN解决了，对之前的信息保存的问题
- 但是！从在长期依赖的问题。
  - 看电影的时候，某些情节的推断需要依赖很久以前的一些细节。
  - 很多其他的任务也一样。
  - 很可惜随着时间间隔不断增大时，RNN 会丧失学习到连接如此远的信息的能力。
  - 也就是说，记忆容量有限，一本书从头到尾一字不漏的去记，肯定离得越远的东西忘得越多。
  - 怎么办：LSTM

循环神经网络之LSTM

- LSTM是RNN一种，大体结构几乎一样。区别是？
  - 它的“记忆细胞”改造过。
  - 该记的信息会一直传递，不该记的会被“门”截断。

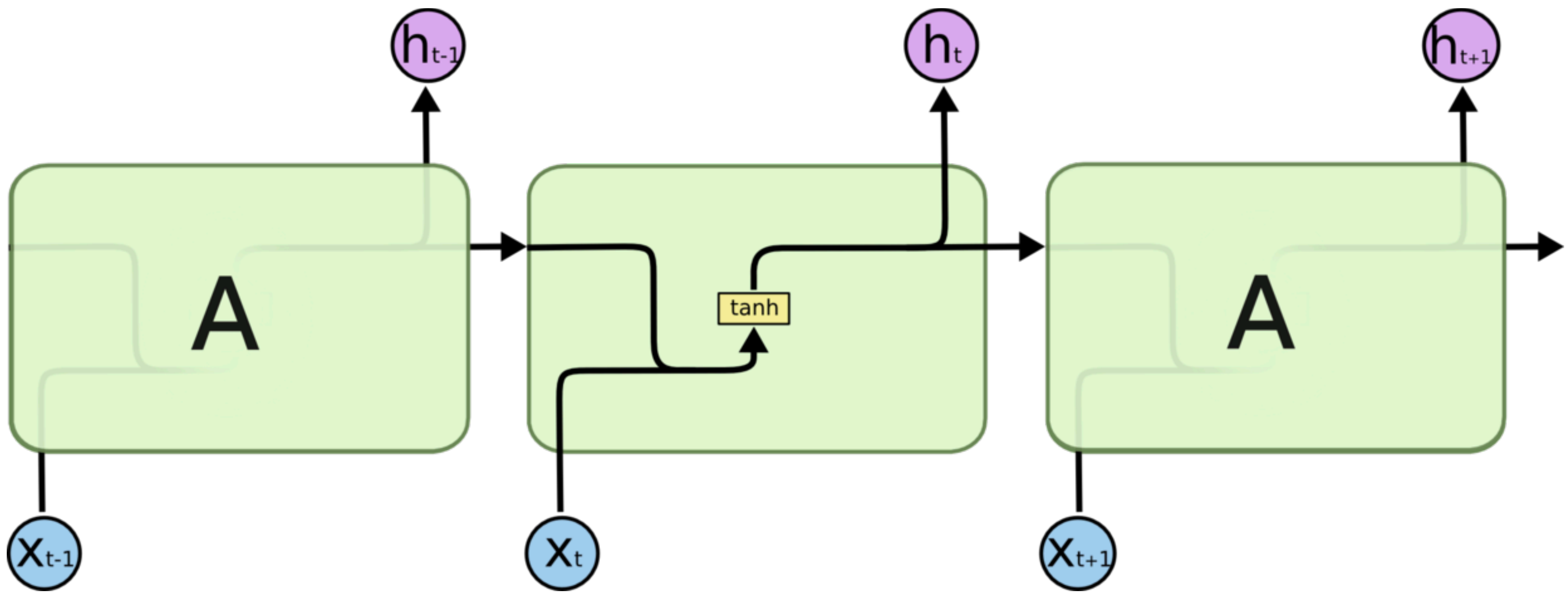
循环神经网络之LSTM

- 之前提到的RNN结构如下



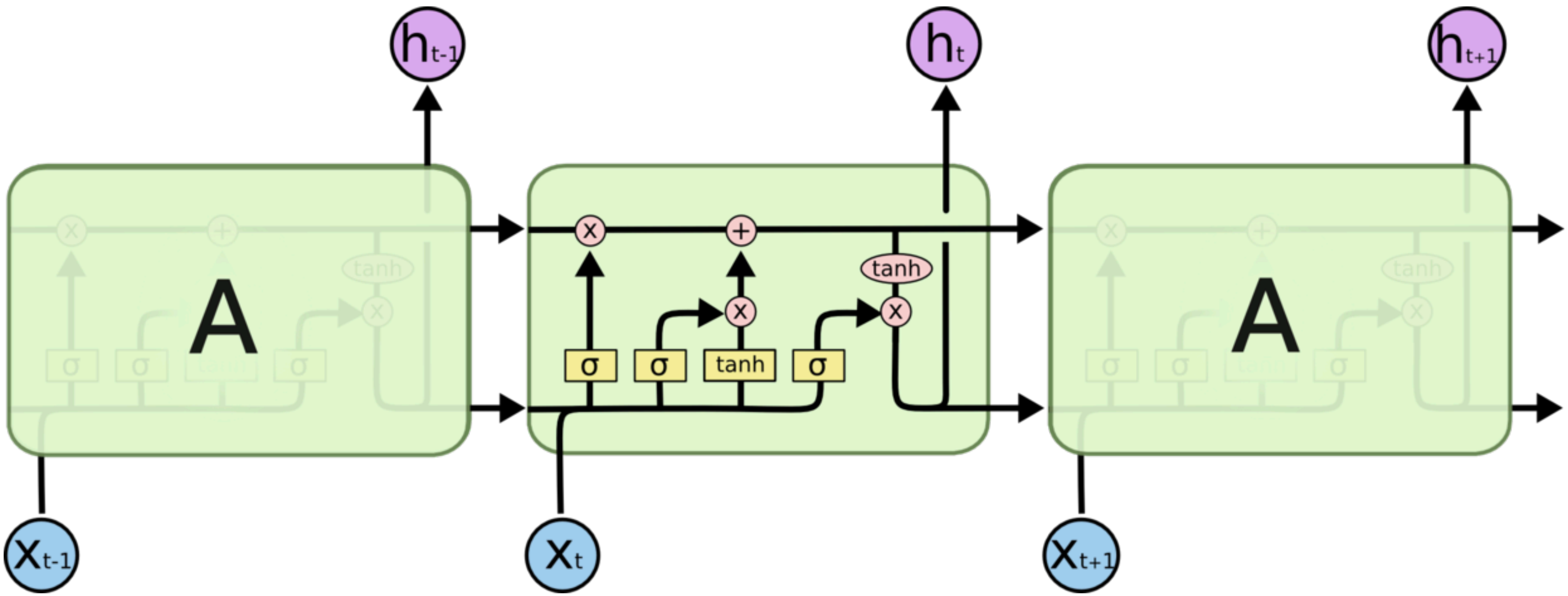
循环神经网络之 LSTM

- 咱们把“记忆细胞”表示得炫酷一点



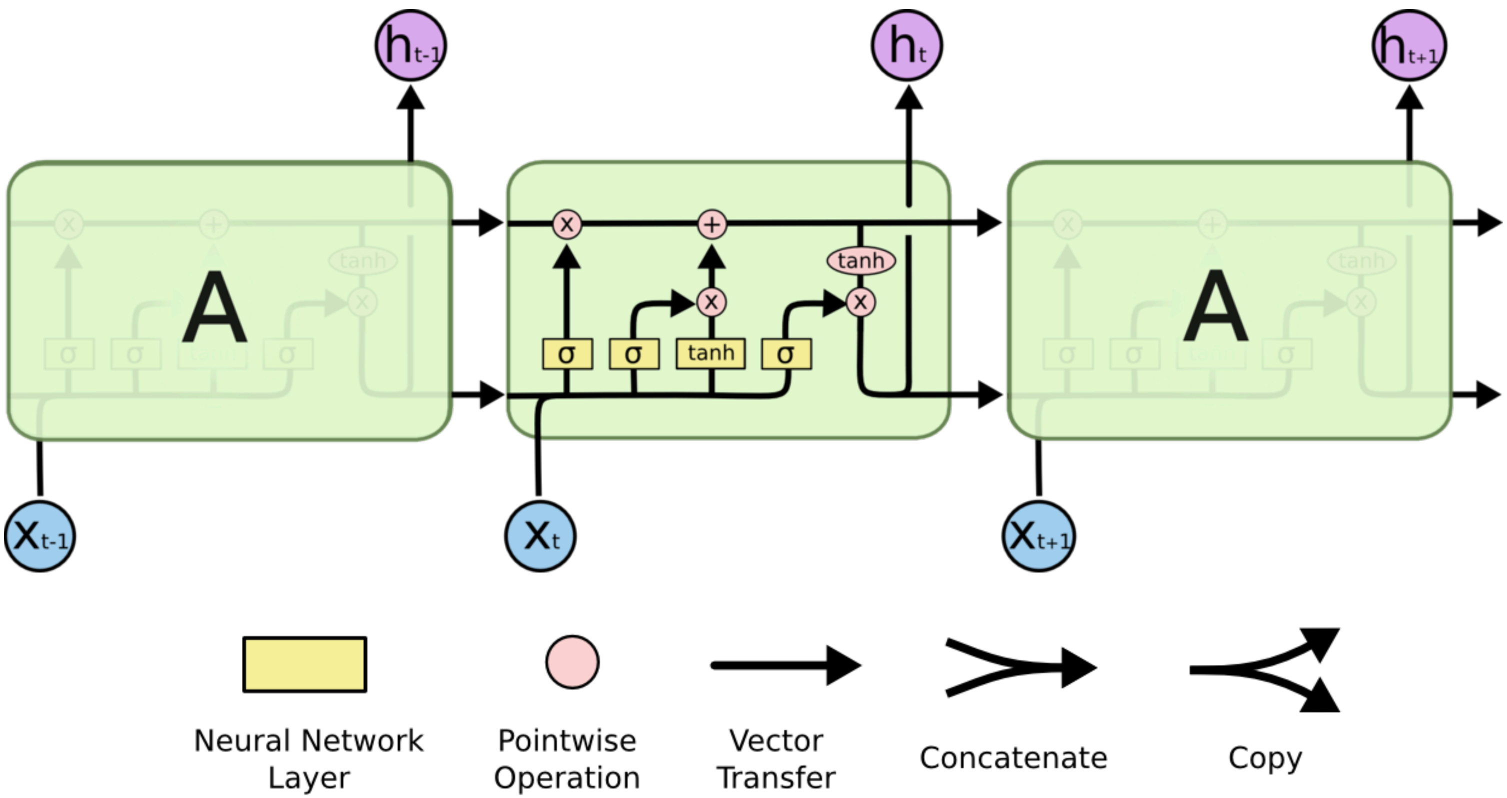
循环神经网络之 LSTM

- LSTM呢？
  - “记忆细胞”变得稍微复杂了一点点



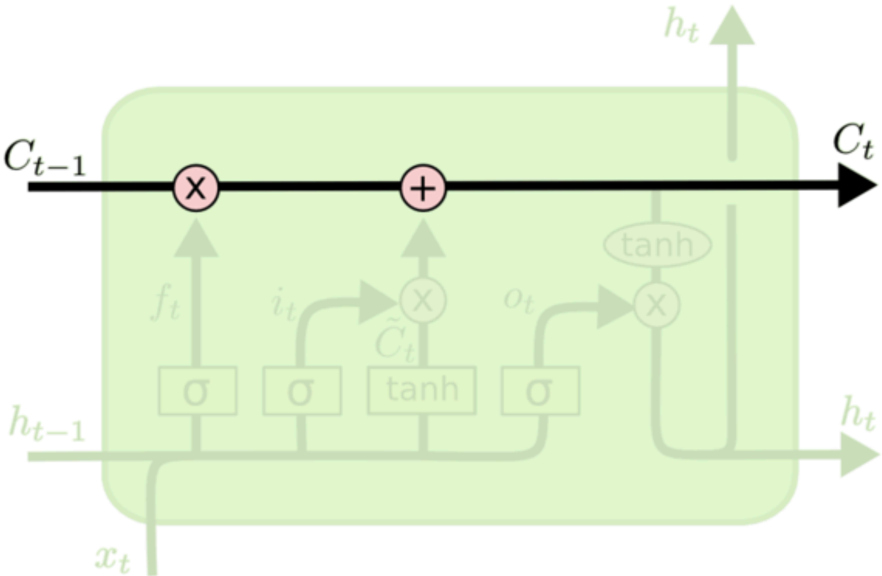
循环神经网络之 LSTM

- 图太复杂，细节看不懂？别着急，我们解释解释。



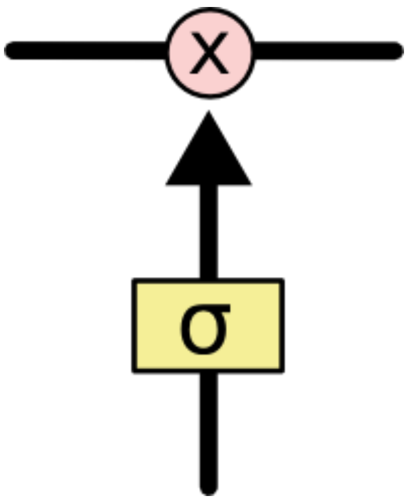
循环神经网络之LSTM

- LSTM关键：“细胞状态”
  - 细胞状态类似于传送带。直接在整个链上运行，只有一些少量的线性交互。信息在上面流传保持不变会很容易。



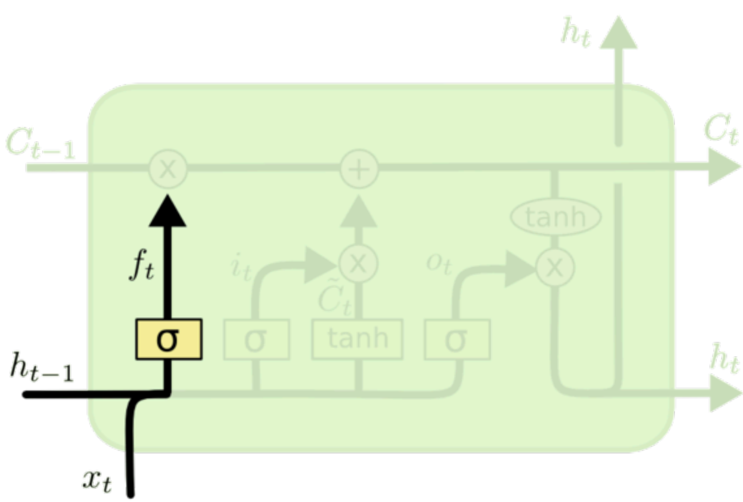
循环神经网络之LSTM

- LSTM怎么控制“细胞状态”？
  - 通过“门”让信息选择性通过，来去除或者增加信息到细胞状态
  - 包含一个sigmoid神经网络层 和 一个pointwise乘法操作
  - Sigmoid 层输出0到1之间的概率值，描述每个部分有多少量可以通过。0代表“不许任何量通过”，1就指“允许任意量通过”



LSTM的几个关键“门”与操作

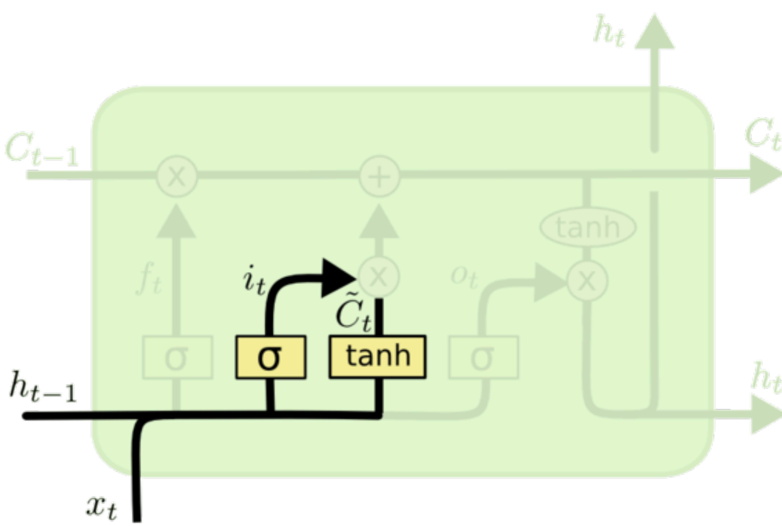
- 第1步：决定从“细胞状态”中丢弃什么信息 => “忘记门”
- 比如完形填空中填“他”或者“她”的问题，细胞状态可能包含当前主语的类别，当我们看到新的代词，我们希望忘记旧的代词。



$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \ + \ b_f\right)$$

LSTM的几个关键“门”与操作

- 第2步：决定放什么新信息到“细胞状态”中
  - ❶ Sigmoid层决定什么值需要更新
  - ❷ Tanh层创建一个新的候选值向量 $\tilde{C}_t$
  - ❸ 上述2步是为状态更新做准备

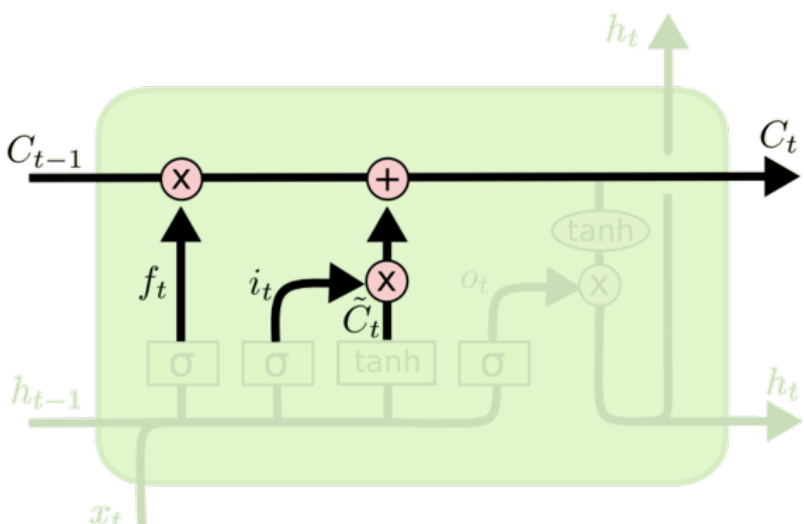


$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \ + \ b_i\right)$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] \ + \ b_C)$$

LSTM的几个关键“门”与操作

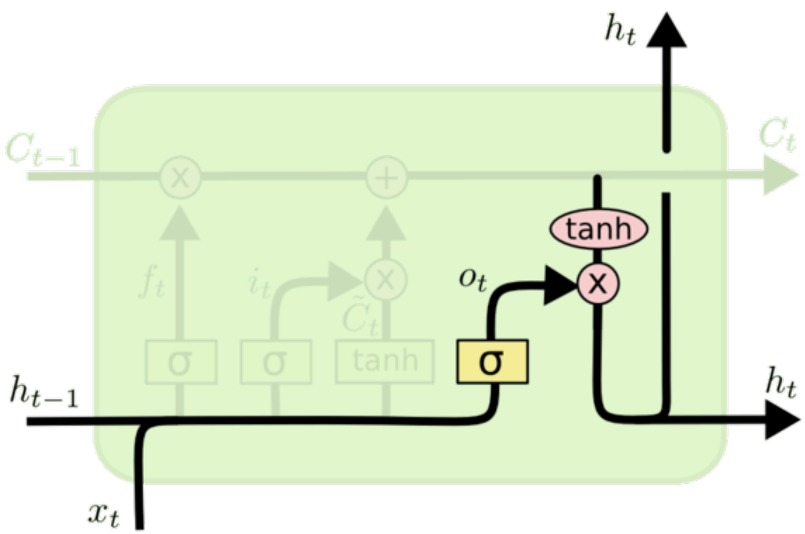
- 第3步：更新“细胞状态”
  1. 更新C_{t-1}为C_t
  2. 把旧状态与f_t相乘，丢弃掉我们确定需要丢弃的信息
  3. 加上$i_t * \tilde{C}_t$。这就是新的候选值，根据我们决定更新每个状态的程度进行变化。



$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

LSTM的几个关键“门”与操作

- 第4步：基于“细胞状态”得到输出
  1. 首先运行一个sigmoid 层来确定细胞状态的哪个部分将输出
  2. 接着用tanh处理细胞状态(得到一个在-1到1之间的值)，再将它和sigmoid门的输出相乘，输出我们确定输出的那部分。
  3. 比如我们可能需要单复数信息来确定输出“他”还是“他们”

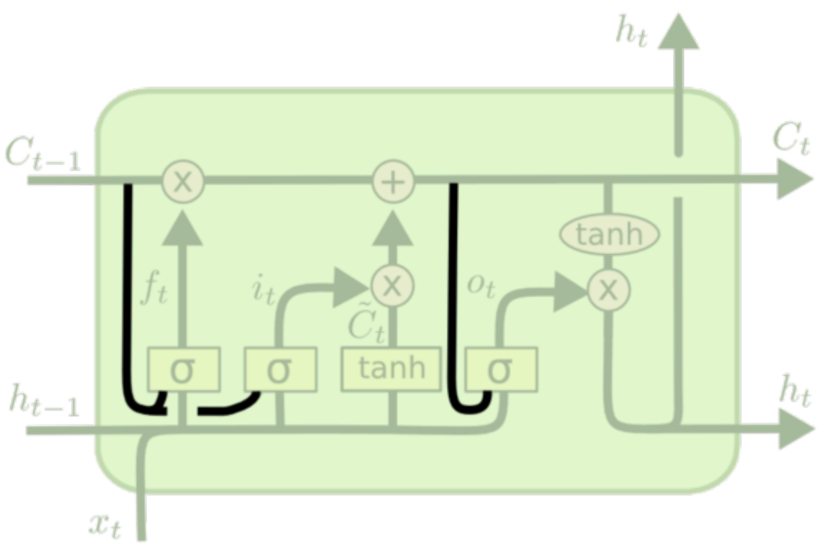


$$o_t = \sigma\left(W_o\ [h_{t-1}, x_t]\ +\ b_o\right)$$

$$h_t = o_t * \tanh\left(C_t\right)$$

LSTM的变体

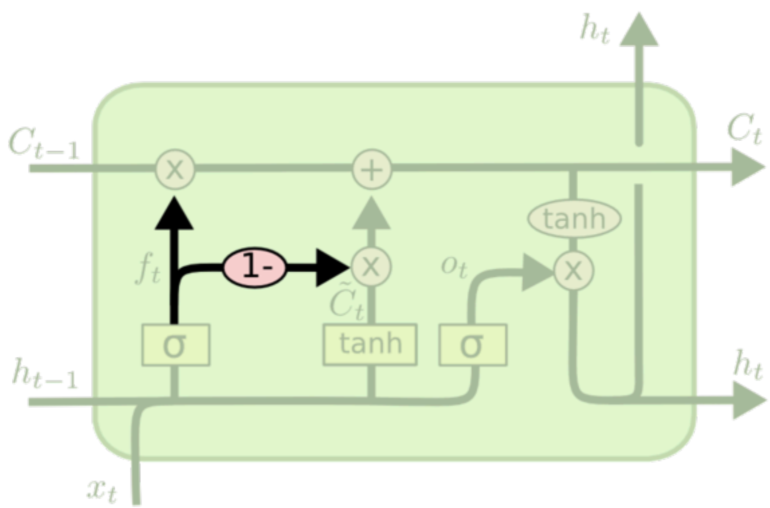
- 变种1
  - 增加“peephole connection”
  - 让 门层 也会接受细胞状态的输入。



$$f_t = \sigma\left(W_f \cdot [\boldsymbol{C_{t-1}}, h_{t-1}, x_t] \ + \ b_f\right)$$
$$i_t = \sigma\left(W_i \cdot [\boldsymbol{C_{t-1}}, h_{t-1}, x_t] \ + \ b_i\right)$$
$$o_t = \sigma\left(W_o \cdot [\boldsymbol{C_t}, h_{t-1}, x_t] \ + \ b_o\right)$$

LSTM的变体

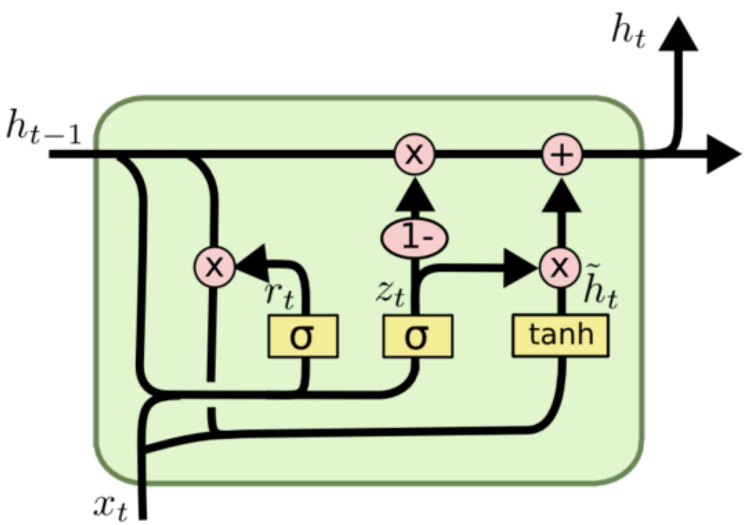
- 变种2
  - 通过使用 coupled 忘记和输入门
  - 之前是分开确定需要忘记和添加的信息，这里是一同做出决定。



$$C_t = f_t * C_{t-1} + (1 - f_t) * \tilde{C}_t$$

LSTM的变体

- 变种3：Gated Recurrent Unit (GRU)，2014年提出
  - 将忘记门和输入门合成了一个单一的 更新门
  - 同样还混合了细胞状态和隐藏状态，和其他一些改动。
  - 比标准LSTM简单。



$$z_t = \sigma\left(W_z \cdot [h_{t-1}, x_t]\right)$$

$$r_t = \sigma\left(W_r \cdot [h_{t-1}, x_t]\right)$$

$$\tilde{h}_t = \tanh\left(W \cdot [r_t * h_{t-1}, x_t]\right)$$

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t$$

LSTM比较？

- 2015的paper《LSTM: A Search Space Odyssey》中，对各种变体做了对比，发现其实本质上它们大同小异。
- 2015的论文《An Empirical Exploration of Recurrent Network Architectures》中，google和facebook的大神尝试了1w+种RNN架构，发现并非所有任务上LSTM都表现最好。
- 现在有更多的RNN研究方向，比如attention model和Grid LSTM等等