

- 传统RNN的内部结构
  - $x_t$ 是输入， $h_t$ 是当前输出，也是下一个的输入
  - 内部工作原理：
    - 当前输入 $x_t$ 和上一个时态的输出 $h_{t-1}$ 进行 concatenate(拼接)，然后通过一个全连接层，该层使用 tanh 作为激活函数，然后得到输出 $h_t$ 。
- LSTM的内部结构：
  - $x_t$ 是输入， $h_t$ 是输出， $C_t$ 是状态。
  - 遗忘门：
    - $f_t = \text{sigmoid}(W_f[h_{t-1}, x_t] + b_f)$
    - ps(sigmoid激活函数):将数值压缩在0-1之间。
  - 输入门：
    - $i_t = \text{sigmoid}(W_i[h_{t-1}, x_t] + b_i)$
    - $C_{t\_} = \tanh(W_c[h_{t-1}, x_t] + b_c)$
    - 更新计算公式（得出 $C_t$ 的公式）： $C_t = f_t * C_{t-1} + i_t * C_{t\_}$
  - 输出门
    - $O_t = \text{sigmoid}(W_o[h_{t-1}, x_t] + b_o)$
    - 更新计算公式（得出 $h_t$ 的公式） $h_t = O_t * \tanh(C_t)$
- GRU模型：门控循环单元模型
  - $x_t$ 是输入， $h_t$ 是输出
  - 更新门：
    - $r_t = \text{sigmoid}(W_r[h_{t-1}, x_t])$
    - 所得值作用在 $h_{t-1}$ 上，确定上一个时态多少信息被这个时态所用，类似于LSTM的遗忘门。
  - 重置门
    - $z_t = \text{sigmoid}(W_z[h_{t-1}, x_t])$

- 重置门作用到 $h_{t-1}$ 和 $h_t$ 上，重置之前的计算，当门值接近1时则输出就是新的 $h_t$ ，反之则上 $h_{t-1}$ 。
- $h_t = \tanh(W[rt * h_{t-1}, X_t])$
- $h_t = (1 - Z_t) * h_{t-1} + Z_t * h_t$
- GRU 与其他 RNN 系列模型的区别？
  - GRU 输入输出的结构与普通的 RNN 相似，其中的内部思想与 LSTM 相似。
  - 与 LSTM 相比，GRU 内部少了一个“门控”，参数比 LSTM 少，但是却也能够达到与 LSTM 相当的功能。考虑到硬件的计算能力和时间成本，因而很多时候我们也会选择更加“实用”的 GRU 啦。
- LSTM 中激活函数区别？
  - 门控的激活函数为 sigmoid；
  - 输出的激活函数为 tanh 函数；

以上内容整理于 [幕布文档](#)