

# 门控循环单元

## (Gated Recurrent Unit, GRU)

Grissom, 2019/07/12  
<http://sunbo2019.github.io>

# 门控循环单元（Gated Recurrent Unit, GRU）

整个RNN网络中，最有价值的部分在哪里？是**隐层**呀！

因为每个时刻的隐层，都包含着“当前学习到的信息”。（当然，也有一定的1~t时刻的信息，因为数据之间是有依赖关系的）

那么，如何求得当前时刻的隐层数据  $\mathbf{H}_t$  呢？

$\mathbf{H}_t$  是t时刻的隐状态矩阵，它是：

**t-1时刻的隐状态矩阵**（旧信息） + **t时刻的候选隐状态矩阵**（新信息）

按比例调制所得。我们约定取  $\mathbf{Z}_t$  个新信息， $1 - \mathbf{Z}_t$  个旧信息。也就是：

$$\mathbf{H}_t = (1 - \mathbf{Z}_t) \odot \mathbf{H}_{t-1} + \mathbf{Z}_t \odot \tilde{\mathbf{H}}_t$$

\* 将  $1 - \mathbf{Z}_t$  与  $\mathbf{Z}_t$  交换，也无伤大雅。

# 什么是 $\mathbf{Z}_t$ ？

$\mathbf{Z}_t$  是一个向量，向量元素的值可以各不相同，但必须都在0-1之间。  
也就是将来按这个百分比从信息矩阵的对应维度上取值。那具体是如何得到的呢？

$$\mathbf{Z}_t = \sigma(\mathbf{X}_t \mathbf{U}_{xz} + \mathbf{H}_{t-1} \mathbf{W}_{hz} + \mathbf{b}_z)$$

- ① 这里的  $\mathbf{X}_t$ （第t时刻的输入，向量）、 $\mathbf{H}_{t-1}$ （第t-1时刻的隐状态，矩阵）都是已知值，而权重  $\mathbf{U}_{xz}$ （矩阵）、 $\mathbf{W}_{hz}$ （矩阵）和偏置项  $\mathbf{b}_z$ （向量）都是需要我们训练出来的变量。

\* 注意：在初始时刻输入的隐层  $\mathbf{H}_0$ （矩阵），是随机初始化出来的。

- ② 激活函数sigmoid，可以保证向量Z的所有元素值都在0~1之间。  
[ 任何一个能将值映射到0~1之间的函数都可以用来作为这里的激活函数 ]

- ③  $\mathbf{Z}_t$  被称为“第t时刻的更新门（update gate）”。

# 什么是 $\tilde{\mathbf{H}}_t$ ？

$\tilde{\mathbf{H}}_t$  是一个矩阵，表示“**候选的隐层信息**”。那具体是如何得到的呢？

$$\tilde{\mathbf{H}}_t = \tanh(\mathbf{X}_t \mathbf{U}_{xh} + \mathbf{R}_t \odot \mathbf{H}_{t-1} \mathbf{W}_{hh} + \mathbf{b}_h)$$

$\mathbf{b}_h$

偏置项（向量）

$\mathbf{X}_t \mathbf{U}_{xh}$

**t时刻输入的新信息** [  $\mathbf{x}_t$  新输入的数据（向量）；  $\mathbf{U}_{xh}$  对应的权重（矩阵） ]

$\mathbf{H}_{t-1} \mathbf{W}_{hh}$

**从t-1时刻传入的隐层信息** [  $\mathbf{H}_{t-1}$  第t-1时刻的隐层信息（矩阵）；  $\mathbf{W}_{hh}$  对应的权重（矩阵） ]

$\odot$

操作符，表示“向量或矩阵中，**对应位置的元素相乘**”

$\mathbf{R}_t$

叫“**重置门（reset gate）**”，是一个向量。其元素值可以各不相同，但必须都是在**0~1**之间。  
[ 它决定了在不同维度上分别受多少旧信息的影响。 ]

# 什么是 $\mathbf{R}_t$ ？

虽然叫“重置门”，但本质上和“更新门(update gate)”是一样的。即：在不同维度上产生一个0~1之间的概率，来控制接受信息的多少。

$$\mathbf{R}_t = \sigma(\mathbf{X}_t \mathbf{U}_{xr} + \mathbf{H}_{t-1} \mathbf{W}_{hr} + \mathbf{b}_r)$$

\* 注意：“控制门” / “更新门”，它们变量（ $\mathbf{U}$ 、 $\mathbf{W}$ 、 $\mathbf{V}$ ）是共享的，但各自的变量 $\mathbf{b}$ 不是共享的！

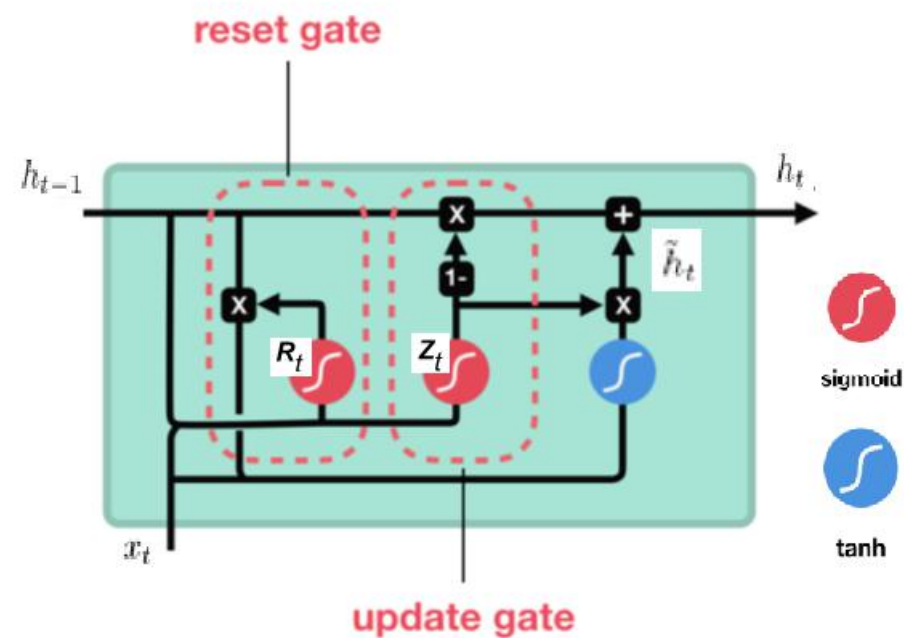
# 总结：什么是 GRU？

就是在原来RNN结构的基本上，增加了两处控制：

一处是在根据新输入的数据学习新知识时，接受多少比例的旧知识的影响。（重置门）

一处是在产生新的记忆时，新/旧知识中各占多少比例。（更新门）

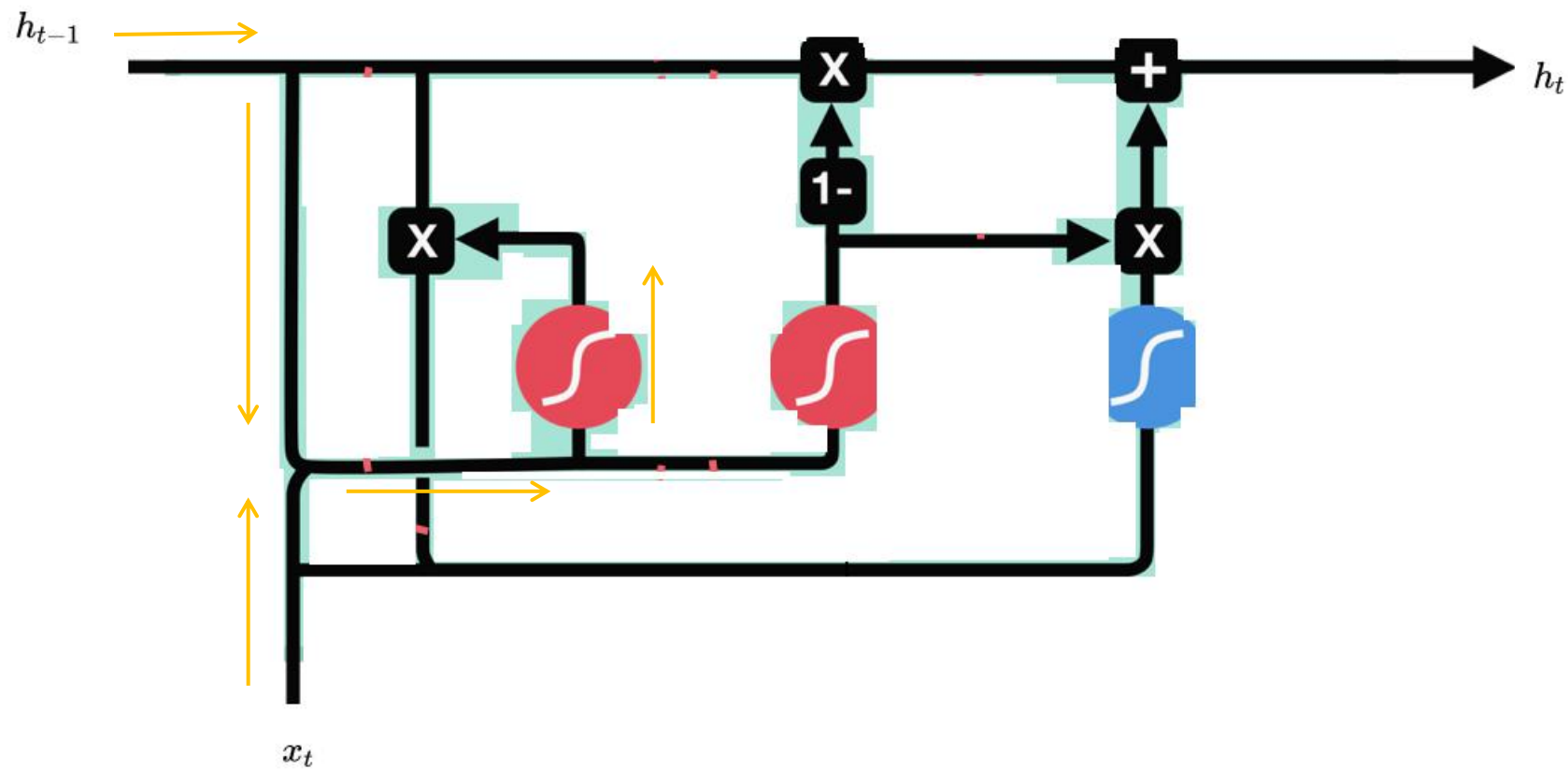
$$\begin{aligned} \mathbf{Z}_t &= \sigma(\mathbf{X}_t \mathbf{U}_{xz} + \mathbf{H}_{t-1} \mathbf{W}_{hz} + \mathbf{b}_z) \\ \downarrow \\ \mathbf{H}_t &= (1 - \mathbf{Z}_t) \odot \mathbf{H}_{t-1} + \mathbf{Z}_t \odot \tilde{\mathbf{H}}_t \\ \uparrow \\ \tilde{\mathbf{H}}_t &= \tanh(\mathbf{X}_t \mathbf{U}_{xh} + \mathbf{R}_t \odot \mathbf{H}_{t-1} \mathbf{W}_{hh} + \mathbf{b}_h) \\ \uparrow \\ \mathbf{R}_t &= \sigma(\mathbf{X}_t \mathbf{U}_{xr} + \mathbf{H}_{t-1} \mathbf{W}_{hr} + \mathbf{b}_r) \end{aligned}$$



这图看起来有点绕，对吗？别怕！我来导航一下！

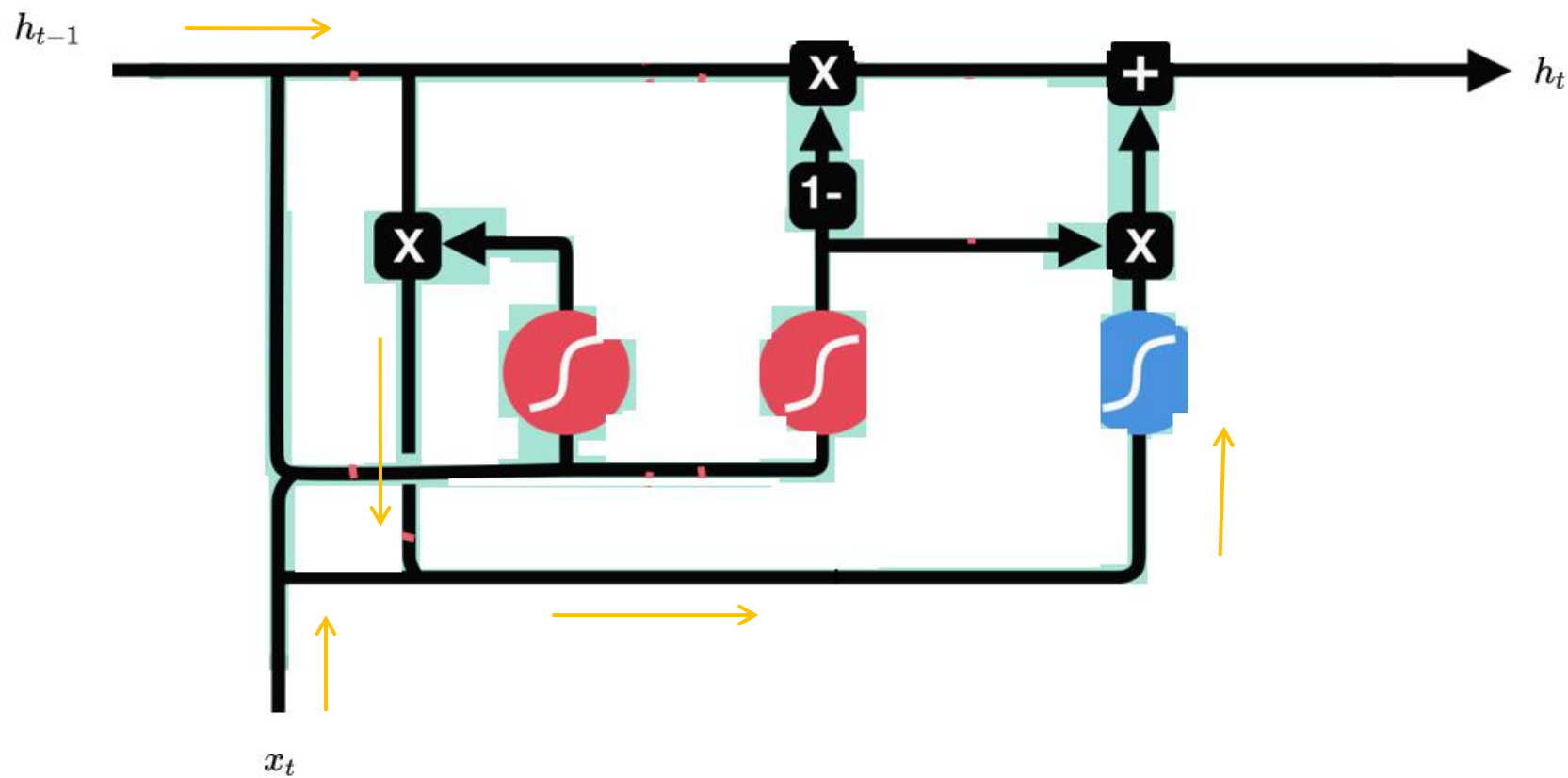
step01：计算出重置门

$$\mathbf{R}_t = \sigma(\mathbf{X}_t \mathbf{U}_{xr} + \mathbf{H}_{t-1} \mathbf{W}_{hr} + \mathbf{b}_r)$$



step02: 计算出候选**新知识** (依赖于重置门)

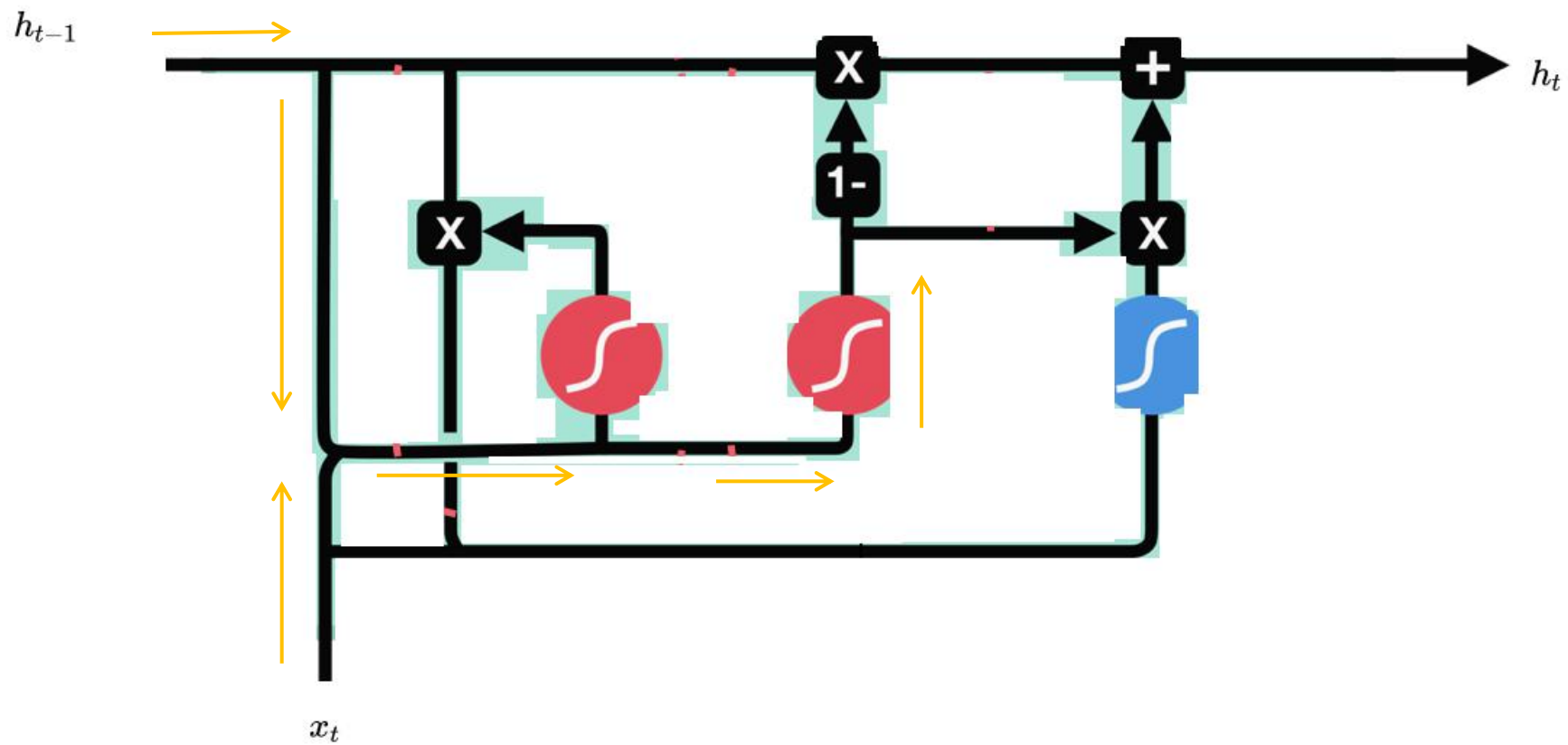
$$\tilde{\mathbf{H}}_t = \tanh(\mathbf{X}_t \mathbf{U}_{xh} + \mathbf{R}_t \odot \mathbf{H}_{t-1} \mathbf{W}_{hh} + \mathbf{b}_h)$$





step03: 计算出更新门

$$\mathbf{z}_t = \sigma(\mathbf{X}_t \mathbf{U}_{xz} + \mathbf{H}_{t-1} \mathbf{W}_{hz} + \mathbf{b}_z)$$



step04：计算出新、旧知识的混合记忆（即：当前时刻的隐层输出）

$$\mathbf{H}_t = (1 - \mathbf{Z}_t) \odot \mathbf{H}_{t-1} + \mathbf{Z}_t \odot \tilde{\mathbf{H}}_t$$

