minGRUs

> Created by Zeng Zhiwen
> Reference: [Were RNNs All We Needed?]

Traditional GRU

- **Gated Recurrent Unit**，门控循环单元，通过引入**门控机制**，能更有效地捕捉时间序列中的长期依赖关系（RNN变体）
- **核心**：
  - Hidden State Recurrence

$$\mathbf{h}_t = (1 - \mathbf{z}_t) \odot \mathbf{h}_{t-1} + \mathbf{z}_t \odot \tilde{\mathbf{h}_t} \tag{1}$$

  - Update Gate：取决于当前输入 与 前一时刻隐藏状态

$$\mathbf{z}_t = \sigma(Linear_{d_h}([x_t, \mathbf{h}_{t-1}])) \tag{2}$$

  - Reset Gate：取决于当前输入 与 前一时刻隐藏状态

$$\mathbf{r}_t = \sigma(Linear_{d_h}([x_t, \mathbf{h}_{t-1}])) \tag{3}$$

  - Candidate Hidden State：取决于遗忘门、当前输入 及 前一时刻隐藏状态

$$\tilde{\mathbf{h}_t} = tanh(Linear_{d_h}([x_t, \mathbf{r}_t \odot \mathbf{h}_{h-1}])) \tag{4}$$

- **缺陷**：无法并行，反向传播依赖时序（BPTT backpropagation through time），需保存中间隐藏状态

miniGRU

- **特点**
  - 去除遗忘门 $\mathbf{r}_t$，仅保留更新门 $\mathbf{z}_t$
  - 候选隐藏状态 $\tilde{\mathbf{h}_t}$ 变成仅取决于当前输入，且移除激活函数
- **核心**
  - Hidden State Recurrence

$$\mathbf{h}_t = (1 - \mathbf{z}_t) \odot \mathbf{h}_{t-1} + \mathbf{z}_t \odot \tilde{\mathbf{h}_t} \tag{5}$$

  - 门控参数

$$\mathbf{z}_t = \sigma(Linear_{d_h}(\mathbf{x}_t)) \tag{6}$$

  - 候选隐藏状态

$$\tilde{\mathbf{h}_t} = Linear_{d_h}(\mathbf{x}_t) \tag{7}$$

  - 由 Heinsen

#递归转通项证明

传统GRU更新公式：

$$h_t = (1 - z_t)h_{t-1} + z_t\tilde{h}_t \tag{8}$$

通过递归展开（假设初始状态为 h_0），可以得到：

$$h_t = h_0 \prod_{i=1}^{t}(1 - z_i) + \sum_{i=1}^{t}\left(z_i\tilde{h}_i \prod_{j=i+1}^{t}(1 - z_j)\right) \tag{9}$$

一般地，初始化 $\mathbf{h}_0$ 为零向量，故传统GRU递归简化为：

$$h_t = \sum_{i=1}^{t}\left(z_i\tilde{h}_i\prod_{j=i+1}^{t}(1-z_j)\right) = \sum_{i=1}^{t} z_i\tilde{h}_i\cdot\frac{\prod_{j=1}^{t}(1-z_j)}{\prod_{j=1}^{i}(1-z_j)} = \left(\prod_{j=1}^{t}(1-z_j)\right)\cdot\sum_{i=1}^{t}\frac{z_i\tilde{h}_i}{\prod_{j=1}^{i}(1-z_j)} \tag{a}$$

🚩 MinGRU的隐状态更新通过对数空间计算：

$$\log h_t = \sum_{i=1}^{t}\log\mathrm{coeff}_i + \log\sum_{i=1}^{t}\exp\left(\log(z_i\tilde{h}_i) - \sum_{j=1}^{i}\log\mathrm{coeff}_j\right) \tag{10}$$

其中：

$$\log\mathrm{coeff}_i = \log(1-z_i) \tag{11}$$

前半部分指数形式

$$\exp\left(\sum_{i=1}^{t}\log(1-z_i)\right) = \exp\left(\sum_{i=1}^{t}\log\mathrm{coeff}_i\right) = \prod_{i=1}^{t}(1-z_i) \tag{12}$$

后半部分指数形式：

$$\exp\left(\log\sum_{i=1}^{t}\exp\left(\log(z_i\tilde{h}_i) - \sum_{j=1}^{i}\log(1-z_j)\right)\right) \tag{13}$$

展开**指数和对数**：

$$\begin{aligned}
&= \sum_{i=1}^{t}\exp\left(\log(z_i\tilde{h}_i) - \sum_{j=1}^{i}\log(1-z_j)\right) \\
&= \sum_{i=1}^{t} z_i\tilde{h}_i\cdot\exp\left(-\sum_{j=1}^{i}\log(1-z_j)\right) \\
&= \sum_{i=1}^{t} z_i\tilde{h}_i\cdot\prod_{j=1}^{i}(1-z_j)^{-1} \\
&= \sum_{i=1}^{t}\frac{z_i\tilde{h}_i}{\prod_{j=1}^{i}(1-z_j)}
\end{aligned} \tag{14}$$

式 (12) 与 式 (14) 相乘，得到MinGRU的最终隐状态为：

$$h_t^{\mathrm{MinGRU}} == \prod_{i=1}^{t}(1-z_i)\cdot\sum_{i=1}^{t}\frac{z_i\tilde{h}_i}{\prod_{j=1}^{i}(1-z_j)} \tag{b}$$

式（a）与式（b）在初始隐藏状态为零向量时完全一致。

- **算法**：我们再次考虑式（10）：

$$\log h_t = \sum_{i=1}^{t}\log\mathrm{coeff}_i + \log\sum_{i=1}^{t}\exp\left(\log(z_i\tilde{h}_i) - \sum_{j=1}^{i}\log\mathrm{coeff}_j\right)$$

$$\log h_t = \sum_{i=1}^{t}\log(1-z_i) + \log\sum_{i=1}^{t}\exp\left(\log(z_i\tilde{h}_i) - \sum_{j=1}^{i}\log(1-z_j)\right)$$

- 第一项由式（11）得知：

$$\begin{aligned}
\log\mathrm{coeff}_i &= \log(1-z_t) = \log(1-\sigma(Linear(x_t))) \\
&= \log\left(\frac{\exp(-Linear(x_t))}{1+\exp(-Linear(x_t))}\right) \\
&= -\mathrm{SoftPlus}(Linear(x_t))
\end{aligned} \tag{15}$$

- 第二项：

$$\log z_t = -\text{Softplus}(-Linear(x_t)) \tag{16}$$

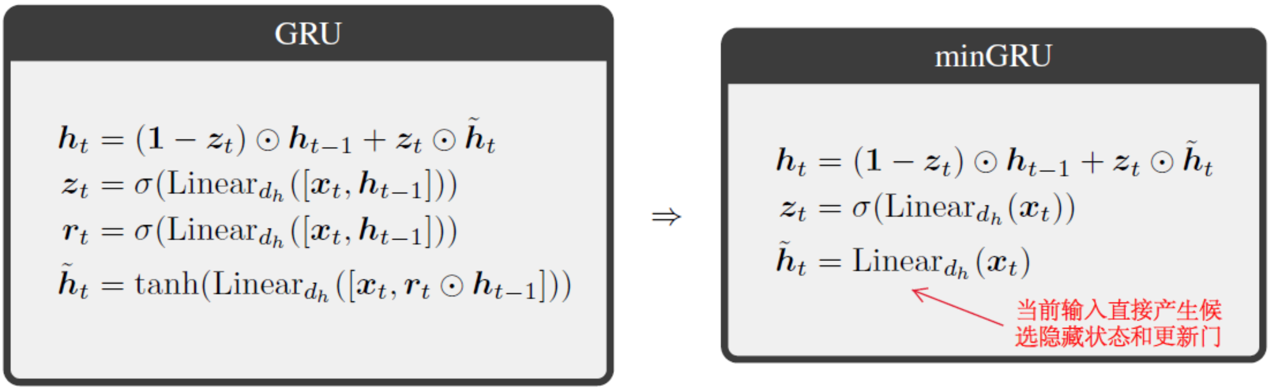
$$\log(\tilde{h}_i) = \log(g(Linear_{d_h}(x_t))) \tag{17}$$

其中，g(x)定义为：

$$g(x) = \begin{cases} x + 0.5 & \text{if } x > 0, \\ \sigma(x) & \text{otherwise.} \end{cases}$$

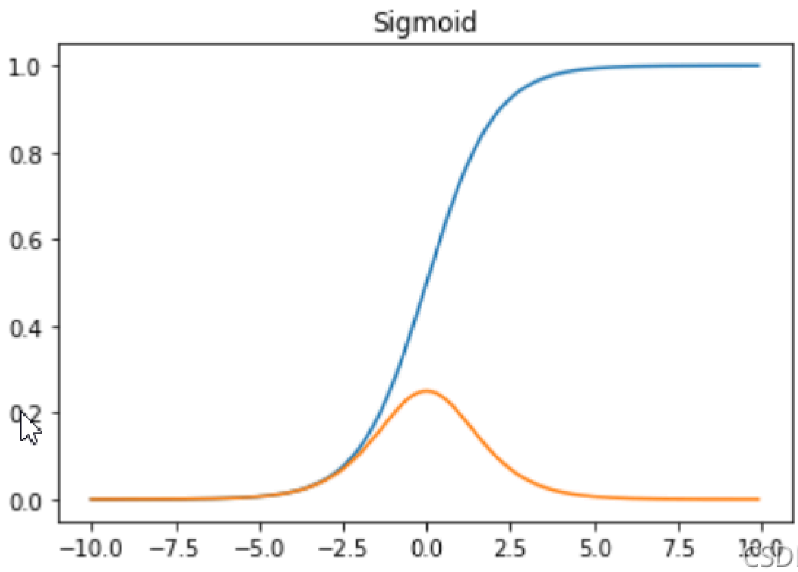
$$log(g(x)) = \begin{cases} \log(x + 0.5) & \text{if } x > 0, \\ -\text{Softplus}(-x) & \text{otherwise.} \end{cases}$$

通常，用于计算 z_t 和 $\tilde{h}_t$ 的 x_t 来源于序列原始特征沿通道对分后的两个序列向量。



Sigmoid

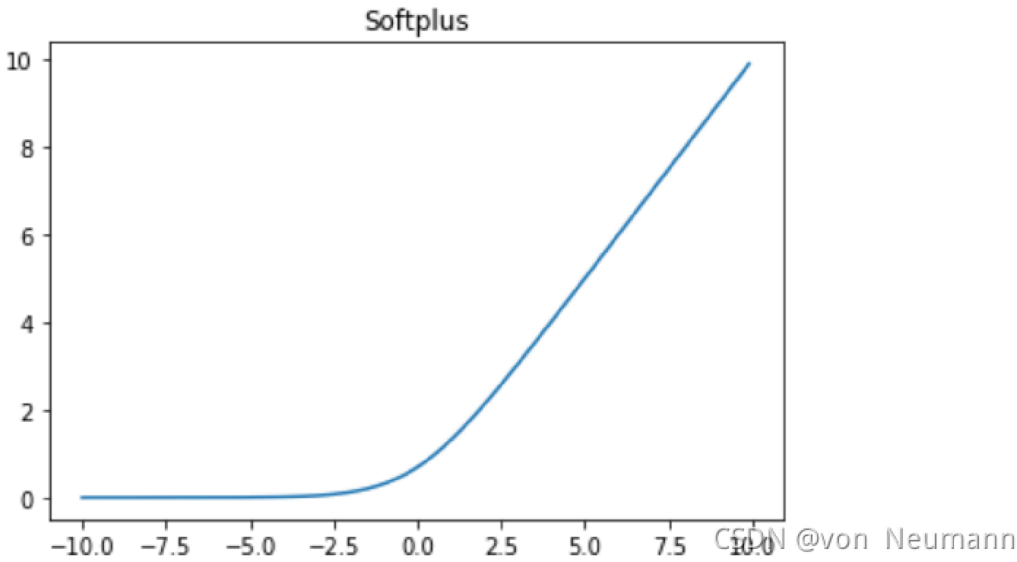
$$S(x) = \frac{1}{1 + e^{-x}}$$



Sigmoid 及其导数

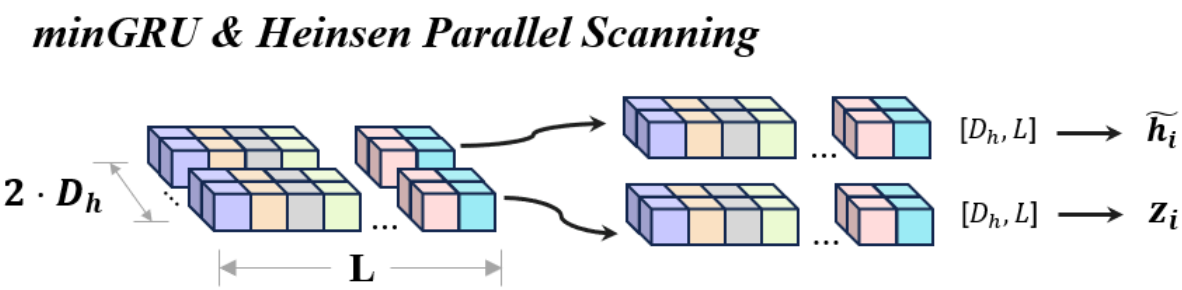
Softplus

$$S(x) = \log(1 + e^x)$$



本框架下的minGRU结构

- **作用：** 捕捉融合（通道拼接）的音频与图谱序列的时序特征



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

$$\log h_t = \sum_{i=1}^{t} \log(1 - z_i) + \log \sum_{i=1}^{t} \exp\left(\log(z_i\tilde{h}_i) - \sum_{j=1}^{i} \log(1 - z_j)\right)$$

- **输入：** 音频特征序列 $A \in [D_A, L]$ 、投影后的噪声图谱特征序列 $X_t \in [D_h, L]$
- **拼接扩展：** 沿通道维度拼接 $AX \in [D_A + D_h, L]$，并映射至 $[2 \cdot D_h, L]$
- **拆分运算：** 沿通道平分为 $h \in [D_h, L]$ 与 $g \in [D_h, L]$，分别用以产生候选状态 $\tilde{h}_i$ 与 更新门控 z_i ，根据式（15)、式（16)、式（17）并行计算出每一轮迭代的系数，后根据式（11）并行计算每一轮的隐藏状态输出 h_i
- **可选：** 通过翻转序列，捕捉反向特征