Mamba

2024, 4, 10

Why not Transformer

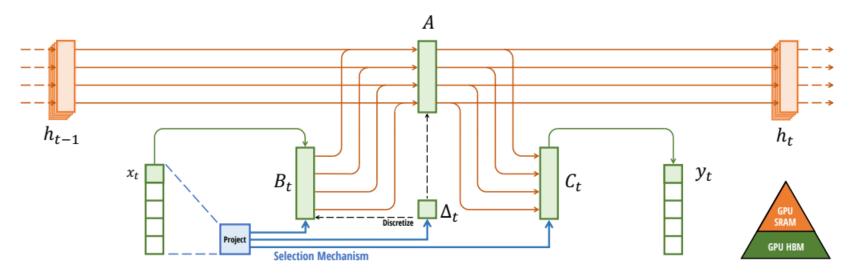
- □ Attention是O(n²)复杂度
- □ 无法见到窗口外的内容(拓展窗口会二次方扩大计算量)
- □ 传统RNN是O(n)且理论上可以见到前面所有内容(其实 会遗忘)

Mamba结构

- □ 类似RNN结构, h(t)依赖于h(t-1)
- □ 利用selection实现过去的存储x(t)
- □ 单独的输出函数y(t)=Ch(t)

Selective State Space Model

with Hardware-aware State Expansion



$$h'(t) = Ah(t) + Bx(t)$$
 (1a)

$$h_t = \overline{A}h_{t-1} + \overline{B}x_t \qquad (2a)$$

$$\overline{K} = (C\overline{B}, C\overline{AB}, ..., C\overline{A}^k \overline{B}, ...)$$
 (3a)

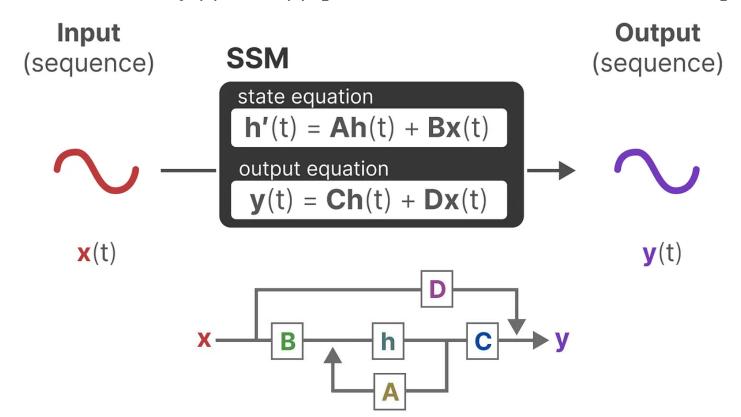
$$y(t) = Ch(t)$$

$$y_t = Ch_t$$

$$y = x * \overline{K} \tag{3b}$$

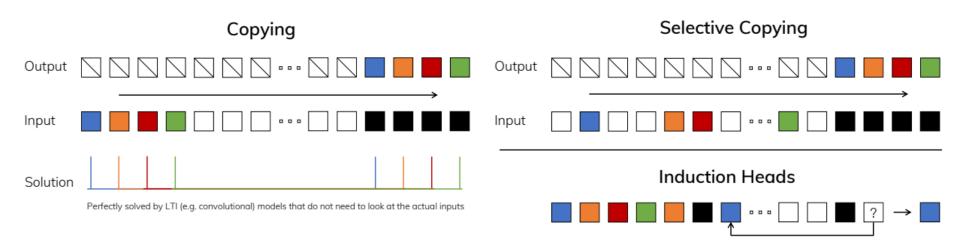
SSM是什么

- □ State Space Models(状态机)
- □ 依赖输入序列x(t)
- □ 状态函数: h(t)=Ah(t)+Bx(t)
- □ 输出函数: y(t)=Ch(t) [加上可能的short connection]



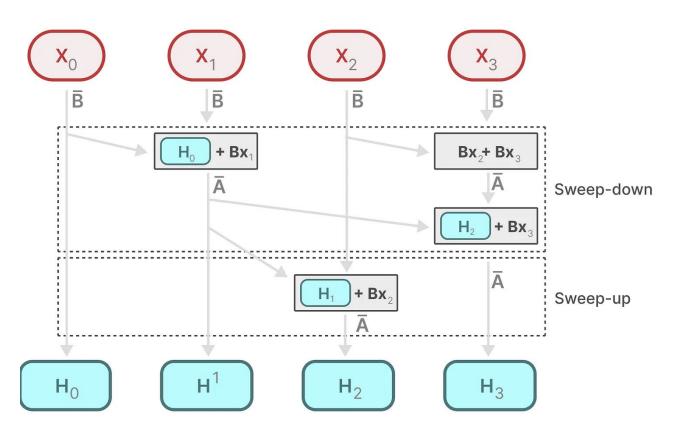
第二个要点: selection

- □ RNN结构是滑动窗口,没有attention作用(平均权重)
- □ 使用selection方法,挑选"有价值"历史输入作为x(t)



第三个要点:硬件友好并行

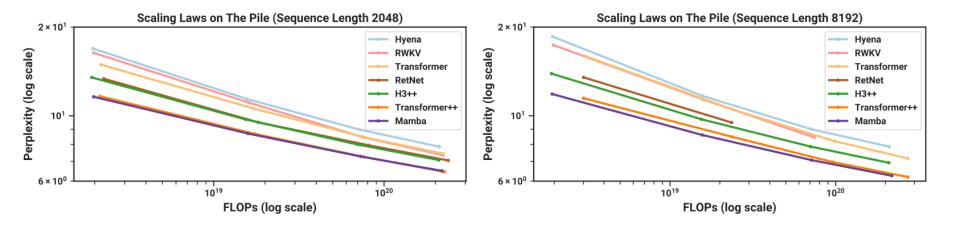
- □ 类似RNN结构(依赖t-1)不能直接并行
- □ 并行扫描技术(某种cache和广播操作)



Parallel computation O(n/t)

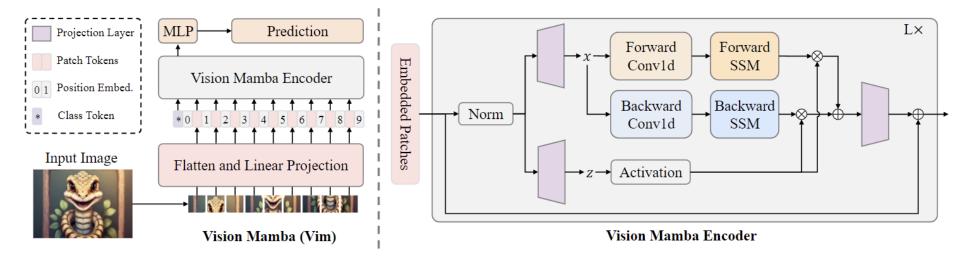
实验结果

- □ 同FLOPS性能最优
- □ 参考: https://blog.csdn.net/v_JULY_v/article/details/134923301



Vision Mamba (ViM)

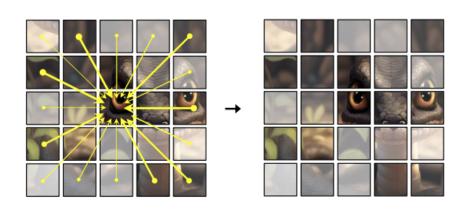
- □ 一比一复刻ViT
- □ 乏善可陈



VMamba

- □ 类似于swin思路
- □ 从两边到中间,扫四次,降低运算量

(a) Attention $O(N^2) \text{ complexity}$



(b) O(N) complexity +

VMamba

- □ 有CNN的localization性
- □ 又带一点全局性

6	7	8	9
Α	В	Α	9
8	7	6	5
4	3	2	1



6	10	8	4
7	11	7	3
8	10	6	2
9	9	5	1



12	17	16	13
17	22	17	12
16	17	12	7
13	12	7	2