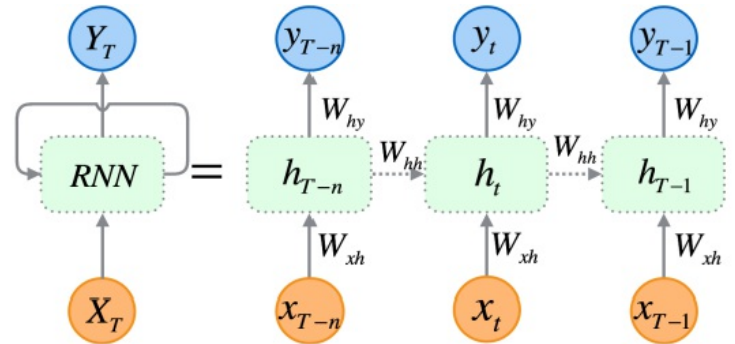
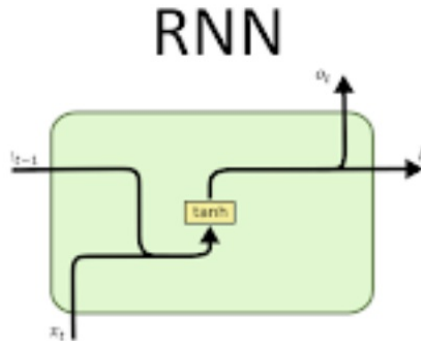


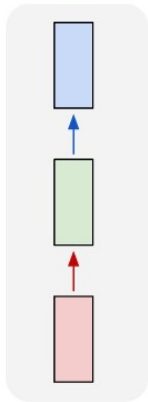
## RNN (Recurrent Neural Network)

Model Neural Network หลายๆ ตัว สร้างเท่ากับข้อมูล Time Step ( ST ) และส่ง Information ให้อัน , Weigth ให้อัน

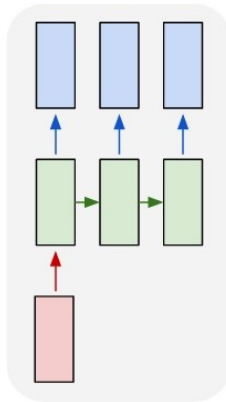


1 Standard RNN architecture and an unfolded structure with T time

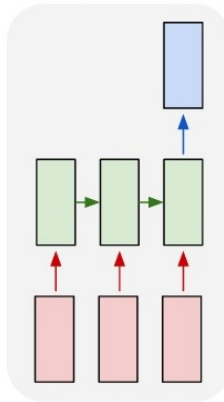
one to one



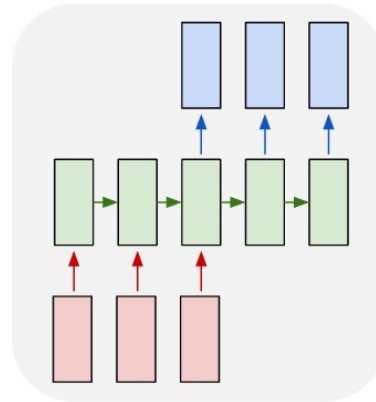
one to many



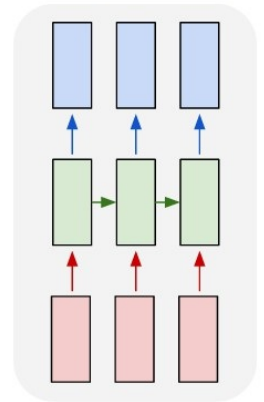
many to one



many to many



many to many



	TS_1		TS_2		TS_3	
	H	L	H	L	H	L
1	1.135	1.132	1.138	1.133	1.14	1.135
2	1.137	1.132	1.136	1.132	1.137	1.133
3	1.137	1.132	1.137	1.132	1.136	1.132
4	1.136	1.128	1.137	1.132	1.137	1.132

Y

Many to one = Target (Y)  
ตัวเดียว  
Many to Many = Target  
(Y) ตัวหลายตัว

Y

Y

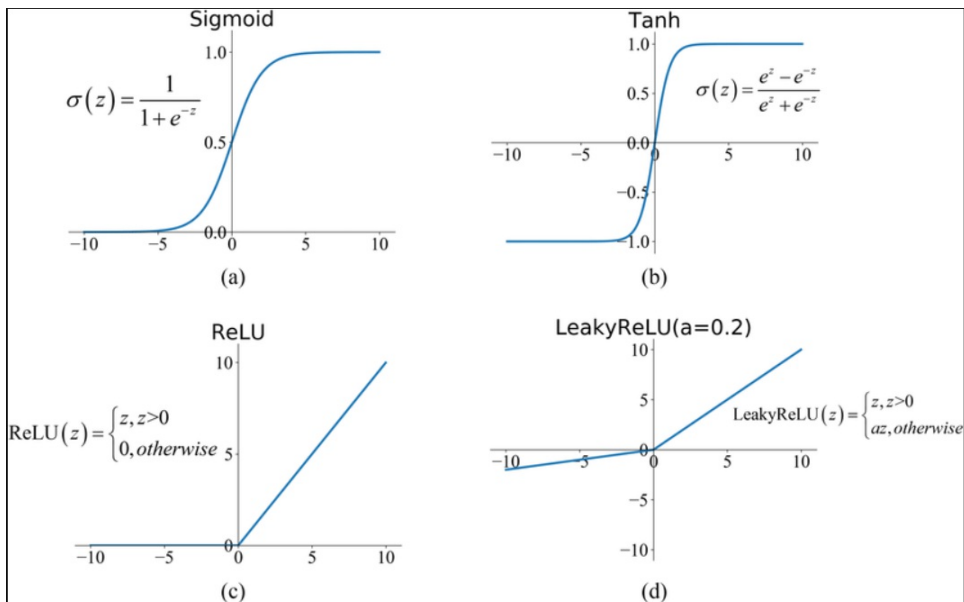
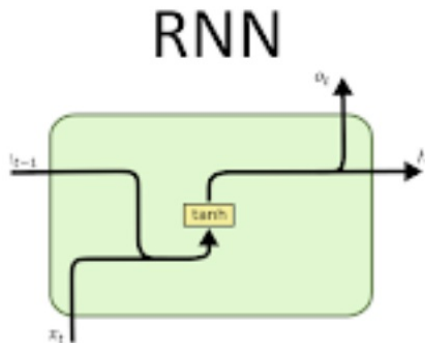
Y

Y

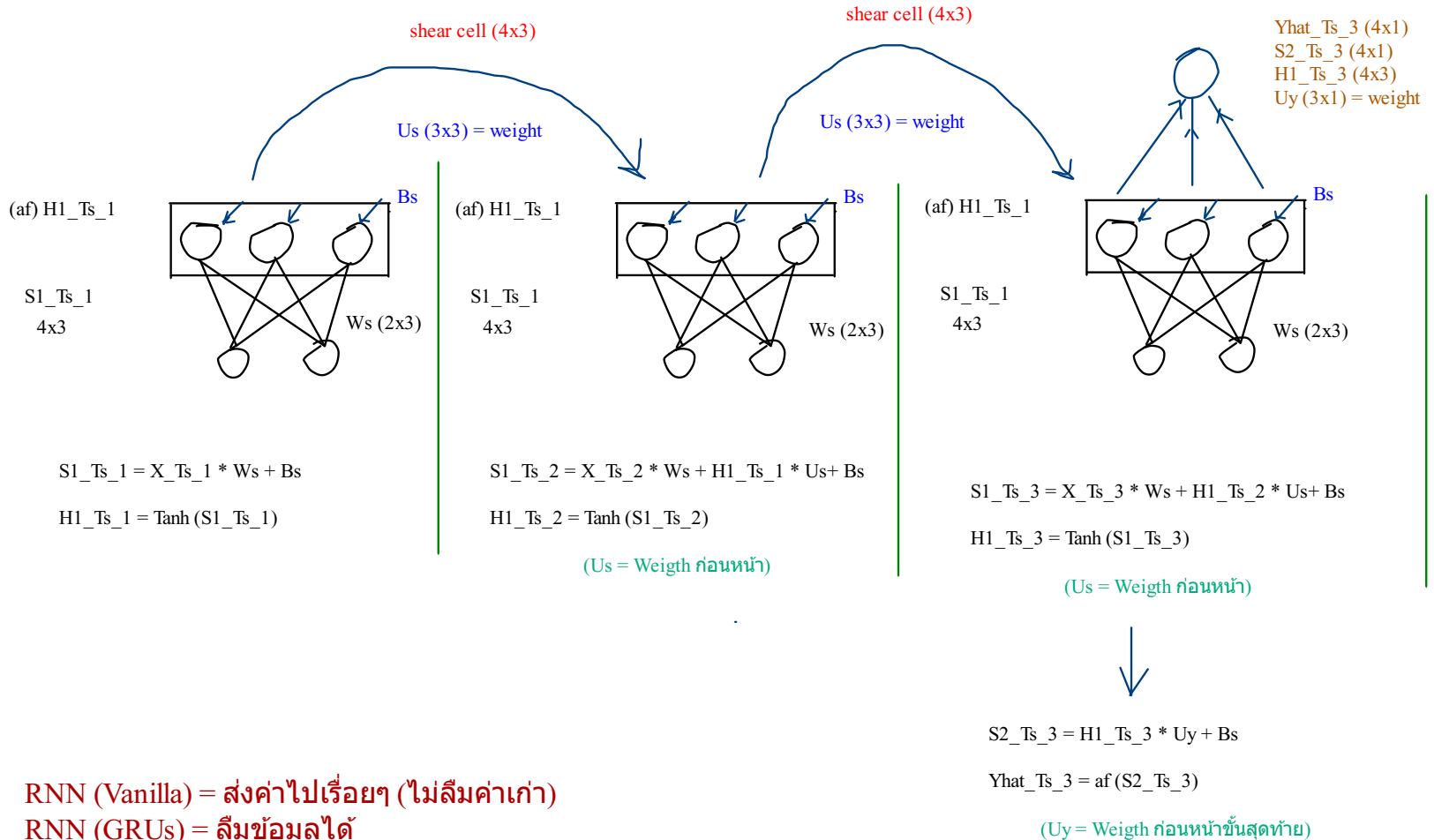
RNN จะ Share Weight ร่วมกัน --> เพื่อลด Cost

(af) ทำไมต้องใช้ Tanh

--> หลายคนเข้าใจว่า ป้องกัน exploding gradient  
(การหาความชันกราฟ แล้วทำให้ค่าใหญ่มากจนพัง)



## RNN (Vanilla)



RNN (Vanilla) = ส่งค่าไปเรื่อยๆ (ไม่ลืมหาค่าเก่า)

RNN (GRUs) = ลืมข้อมูลได้

RNN (LSTM) = ลืมข้อมูลได้ / ส่งข้ามเวลากันได้ (Memory cell)

```
regr = ai.RNN(n_ts_feature=3, n_ts_target=1, RNN_HL=[3], ANN_HL=[1], ANN_AF=['relu'])
```

