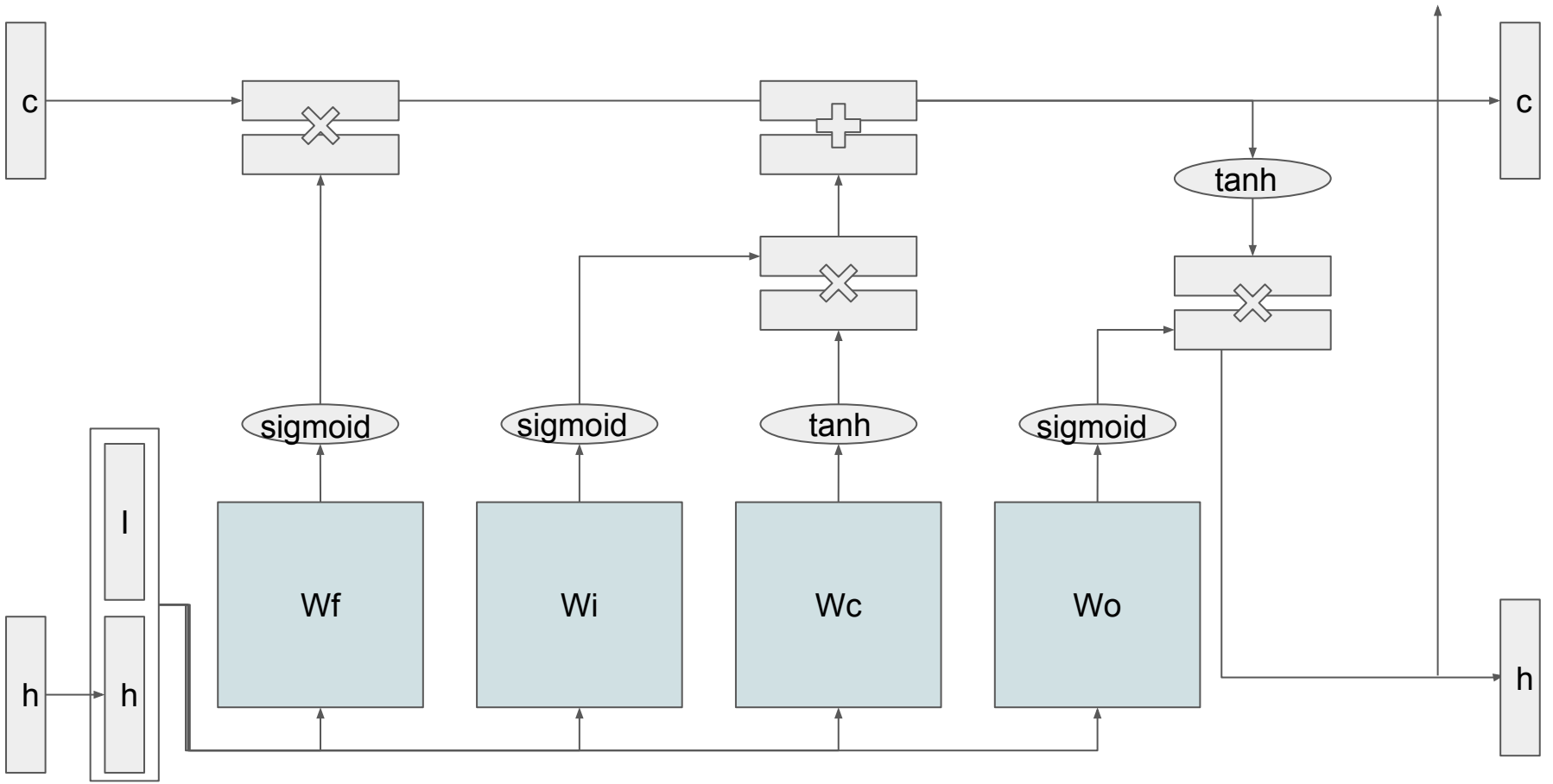
LSTM

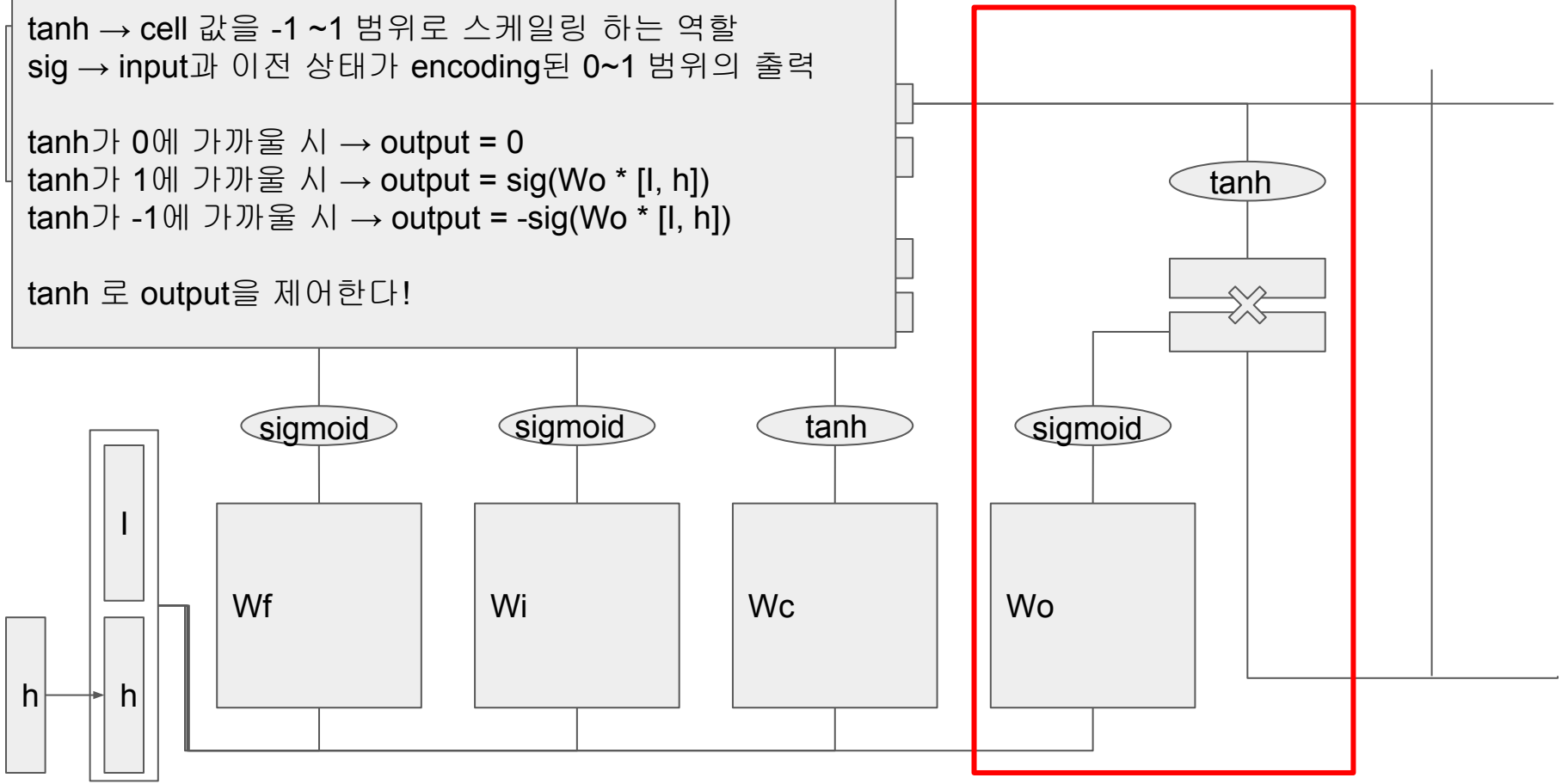

c
c
tanh
sigmoid
sigmoid
tanh
sigmoid
I
h
h
Wf
Wi
Wc
Wo
h

Output = sig(Wo * [I, h]) * tanh(cell)

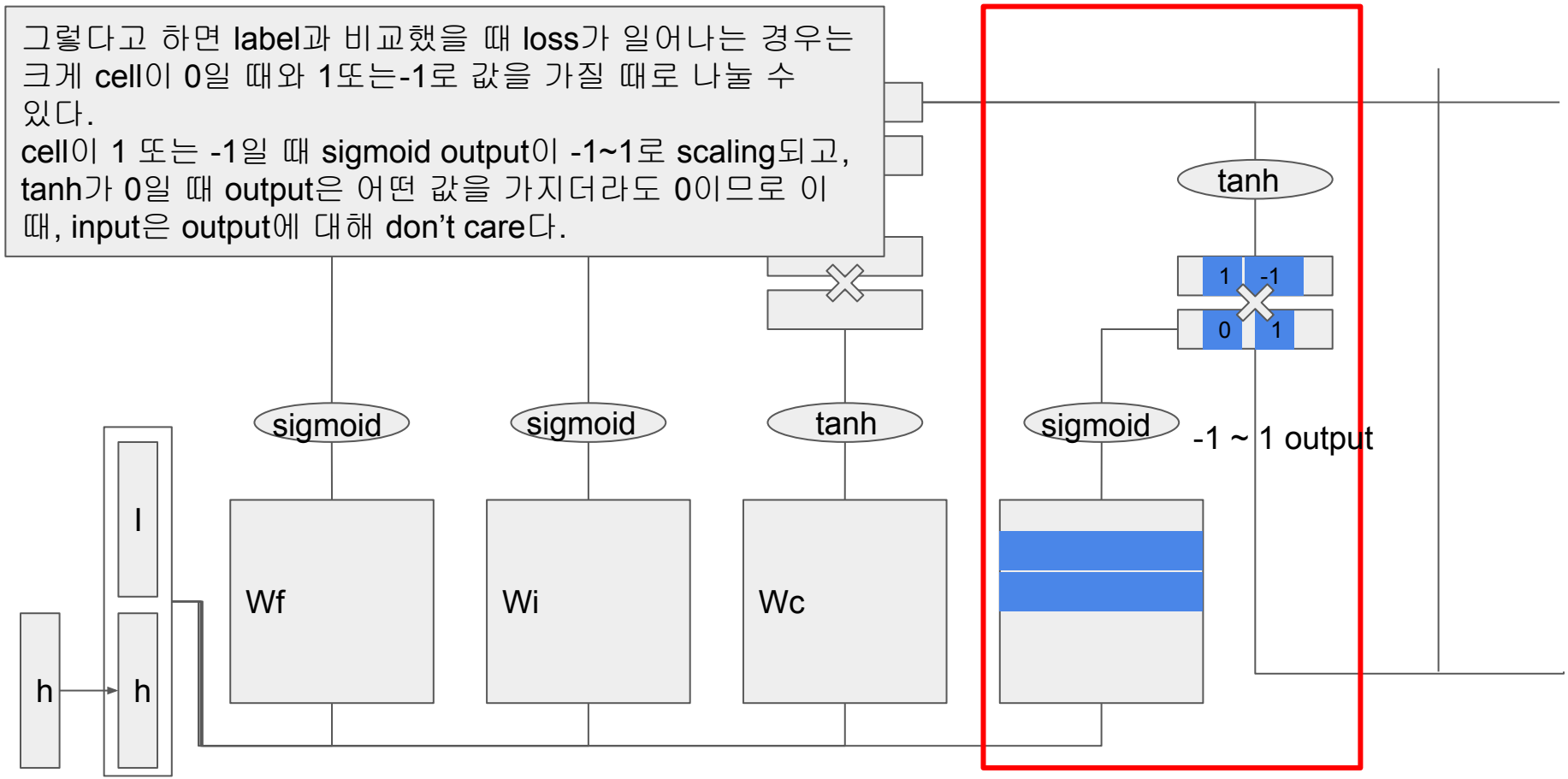
tanh → cell 값을 -1 ~1 범위로 스케일링 하는 역할
sig → input과 이전 상태가 encoding된 0~1 범위의 출력

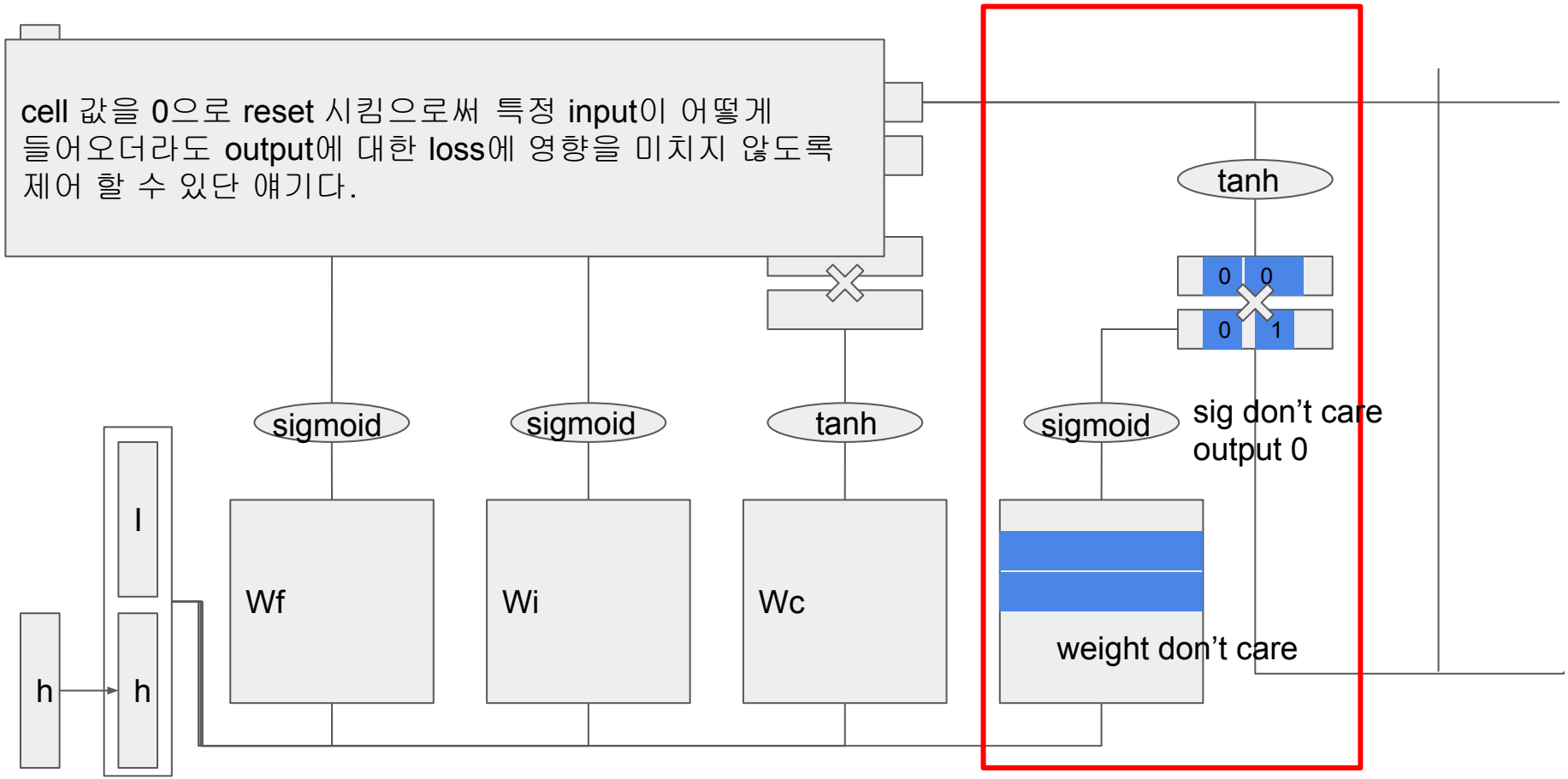
tanh가 0에 가까울 시 → output = 0
tanh가 1에 가까울 시 → output = sig(Wo * [I, h])
tanh가 -1에 가까울 시 → output = -sig(Wo * [I, h])

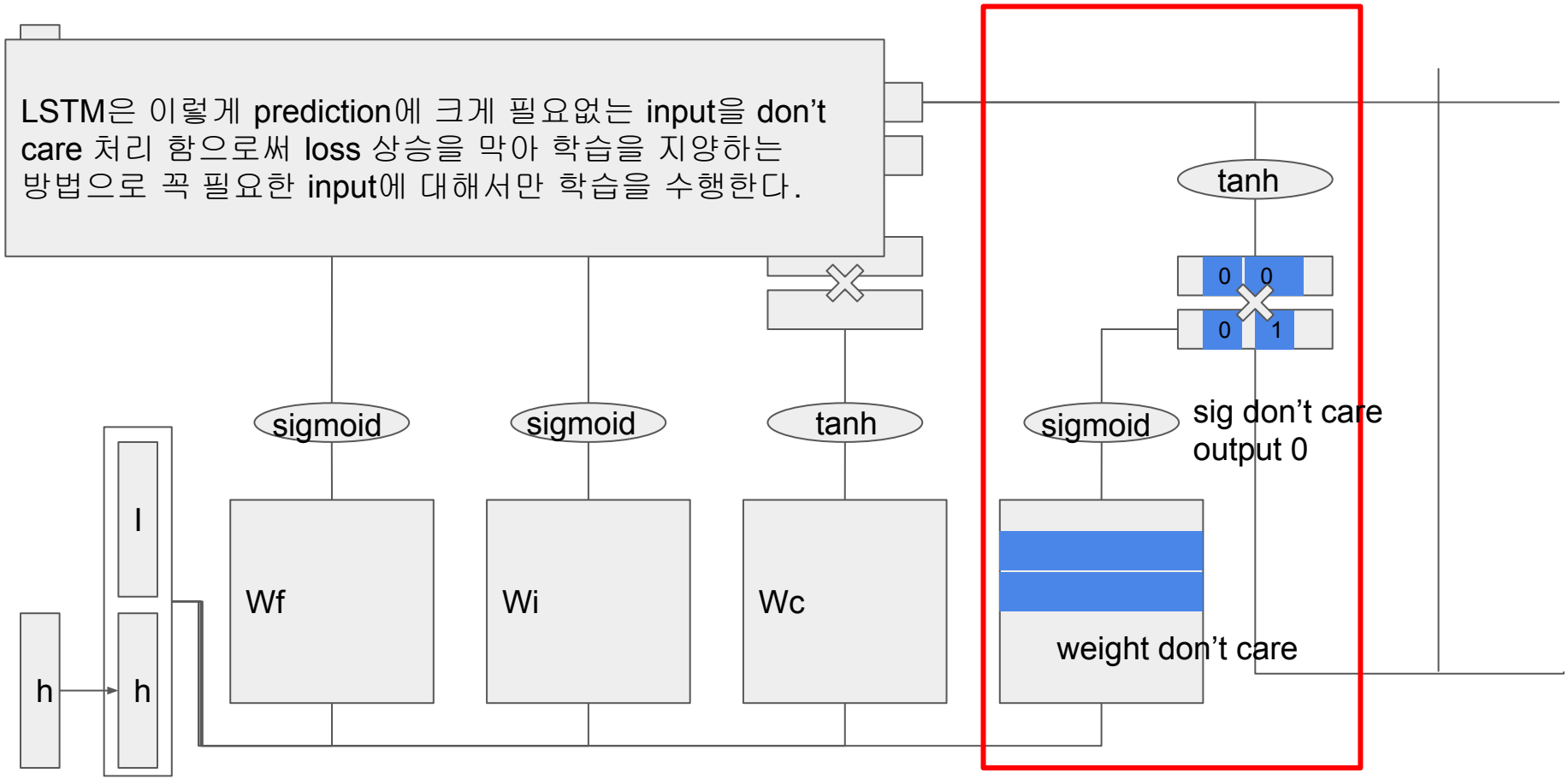
tanh 로 output을 제어한다!

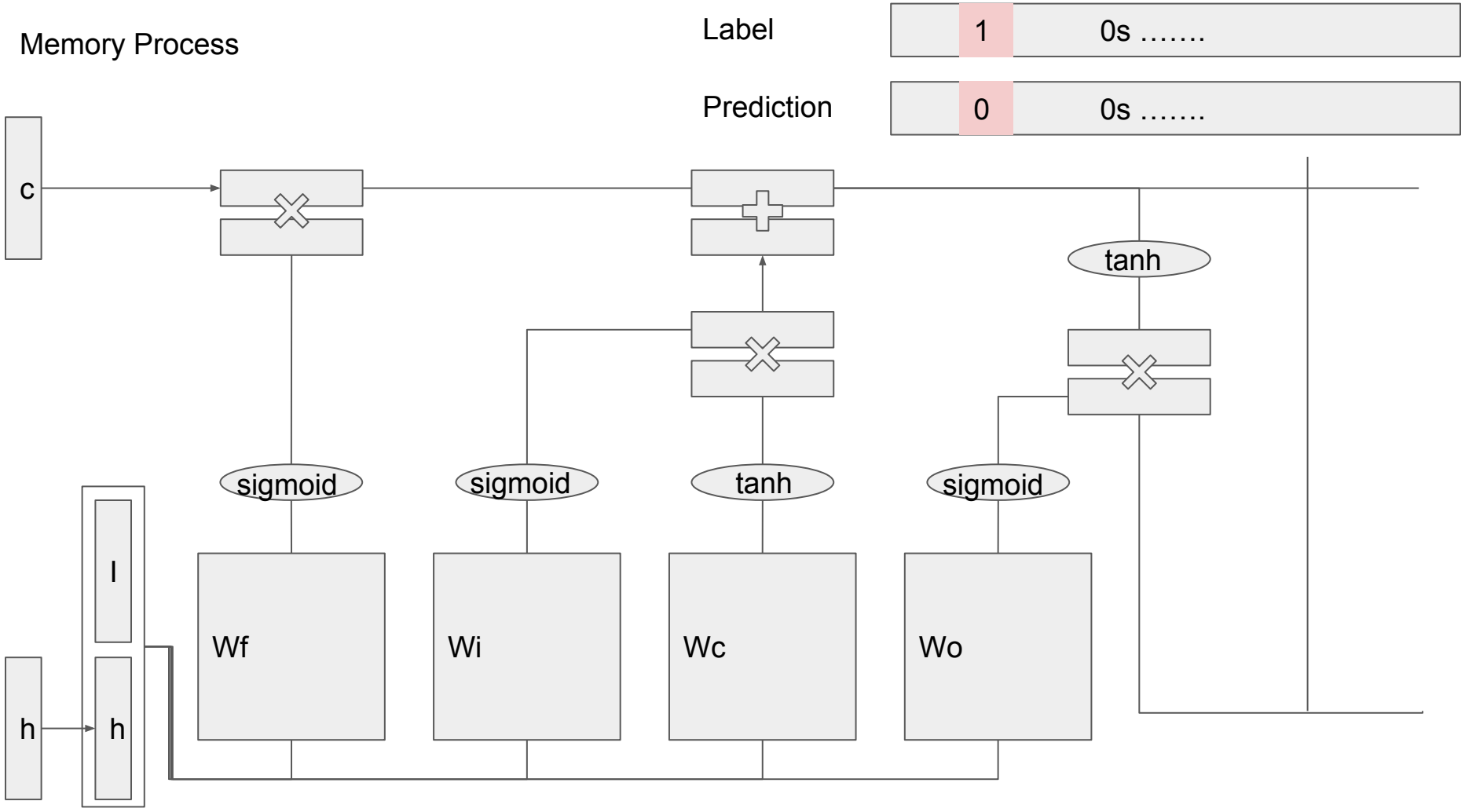


그렇다고 하면 label과 비교했을 때 loss가 일어나는 경우는 크게 cell이 0일 때와 1또는-1로 값을 가질 때로 나눌 수 있다.
cell이 1 또는 -1일 때 sigmoid output이 -1~1로 scaling되고, tanh가 0일 때 output은 어떤 값을 가지더라도 0이므로 이 때, input은 output에 대해 don't care다.

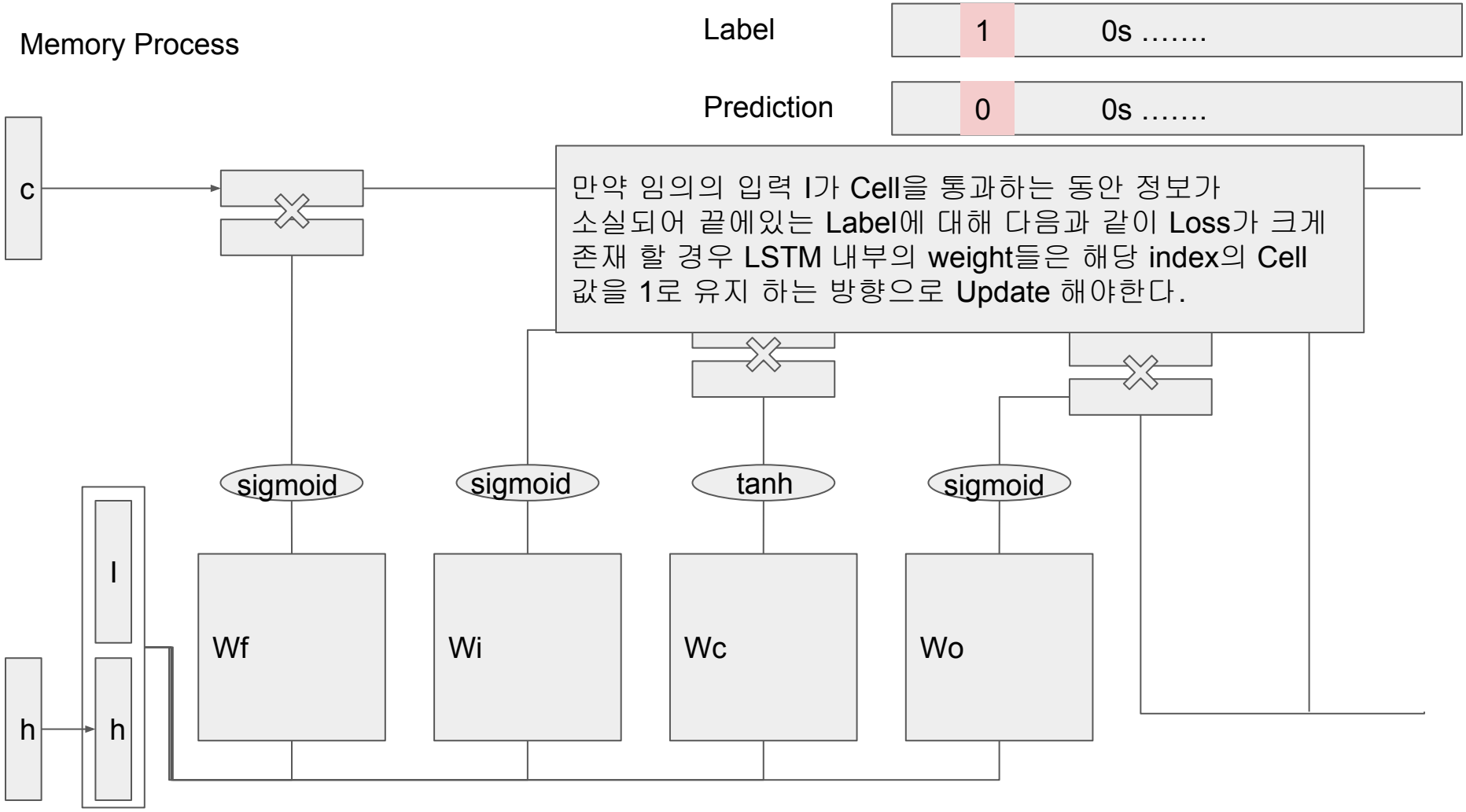


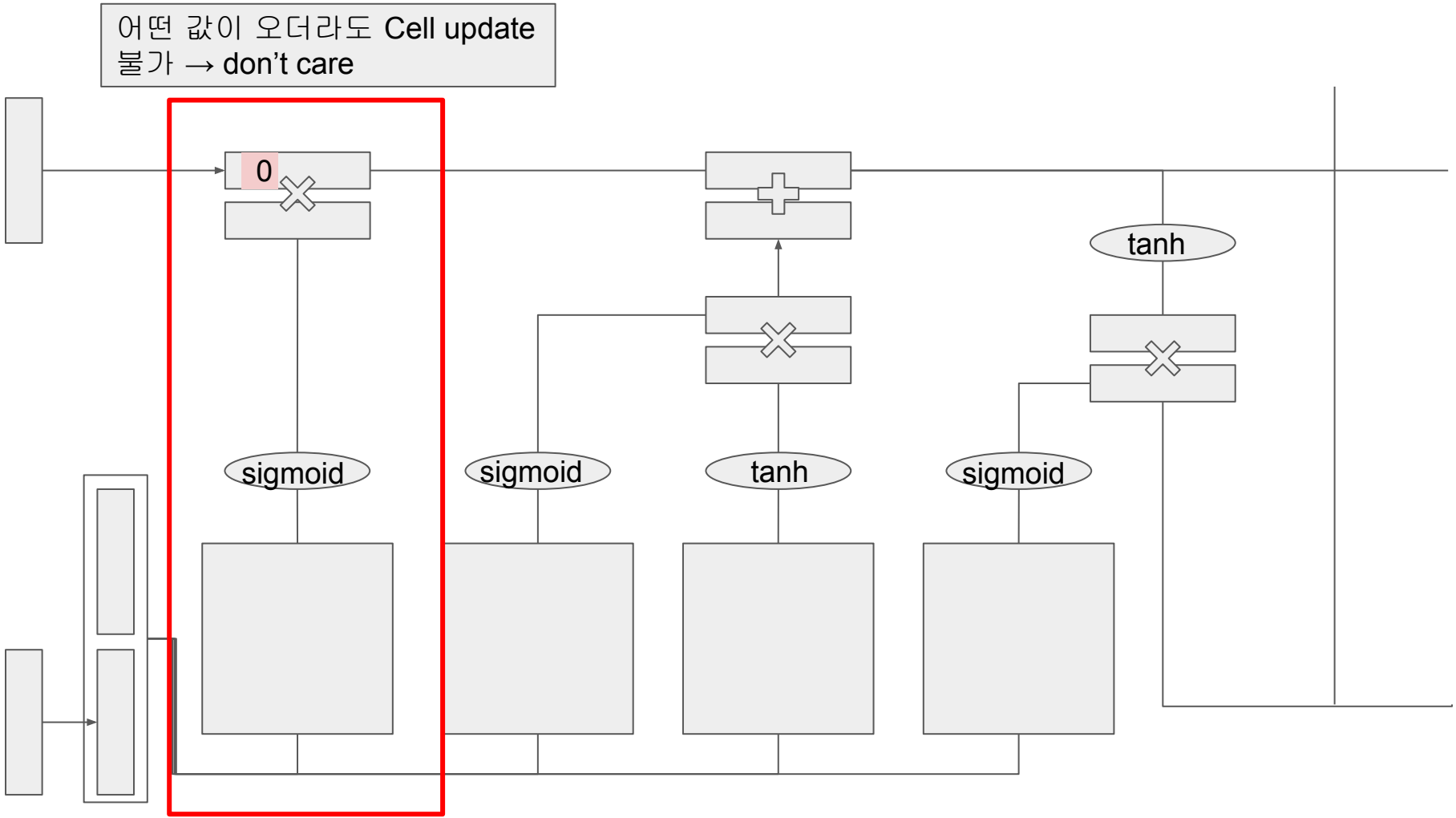

cell 값을 0으로 reset 시킴으로써 특정 input이 어떻게 들어오더라도 output에 대한 loss에 영향을 미치지 않도록 제어 할 수 있단 얘기다.
tanh
0
0
0
1
sigmoid
sigmoid
tanh
sigmoid
sig don't care
output 0
I
h
h
Wf
Wi
Wc
weight don't care

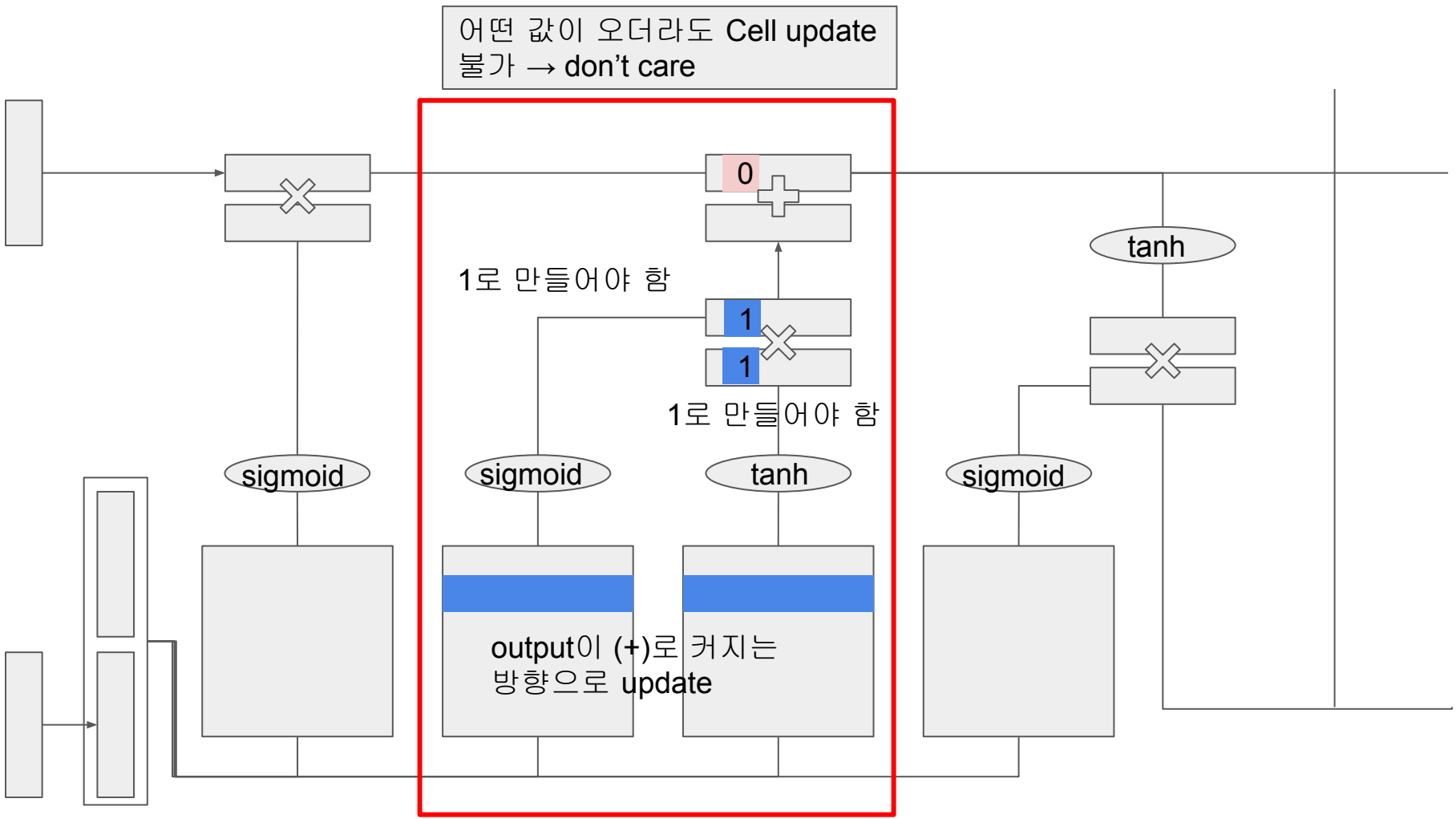

LSTM은 이렇게 prediction에 크게 필요없는 input을 don't care 처리 함으로써 loss 상승을 막아 학습을 지양하는 방법으로 꼭 필요한 input에 대해서만 학습을 수행한다.
tanh
0
0
0
1
sigmoid
sigmoid
tanh
sigmoid
sig don't care
output 0
I
h
h
Wf
Wi
Wc
weight don't care

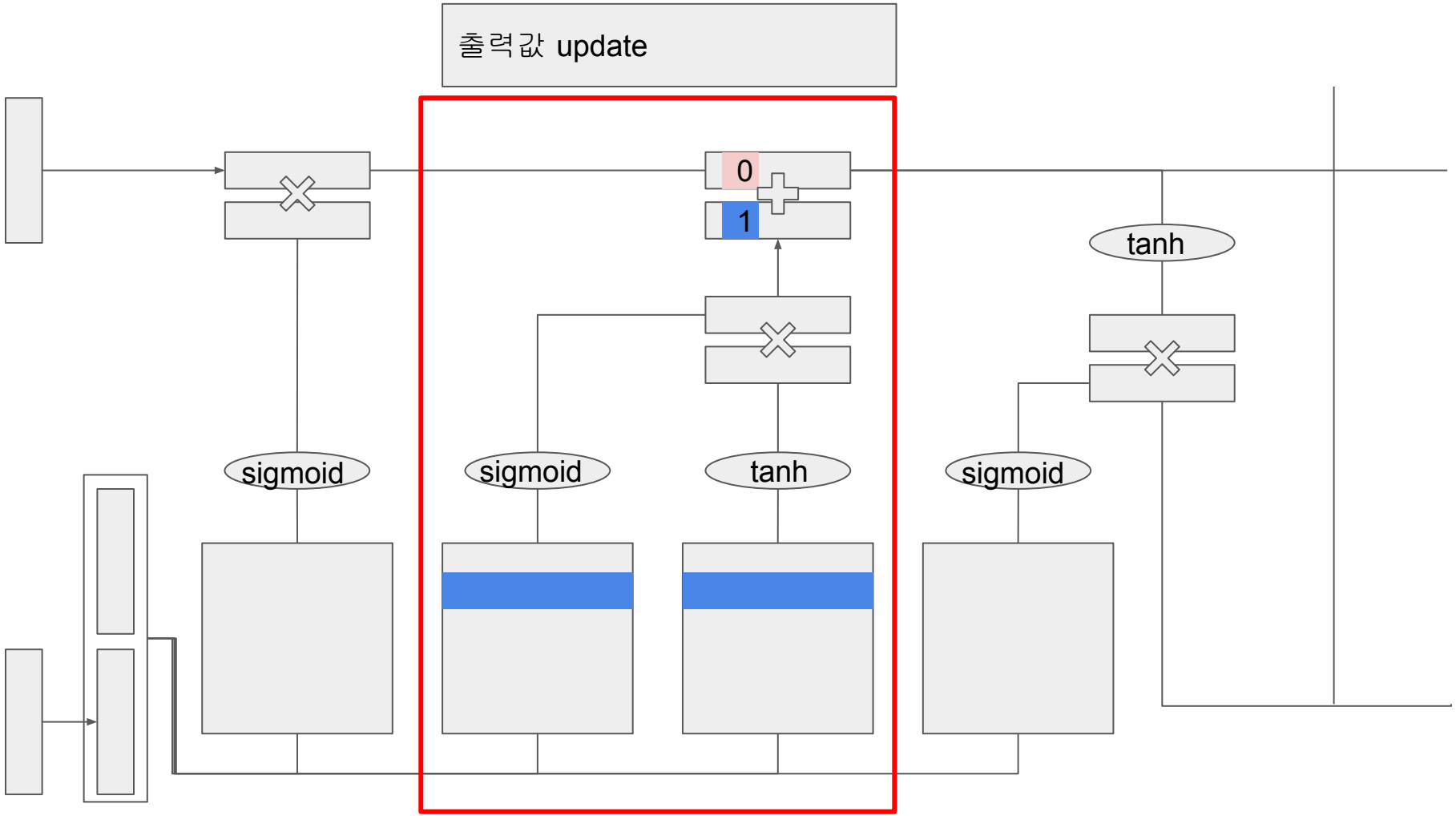

Memory Process
Label
1
0s …….
Prediction
0
0s …….
c
tanh
sigmoid
sigmoid
tanh
sigmoid
I
h
h
Wf
Wi
Wc
Wo

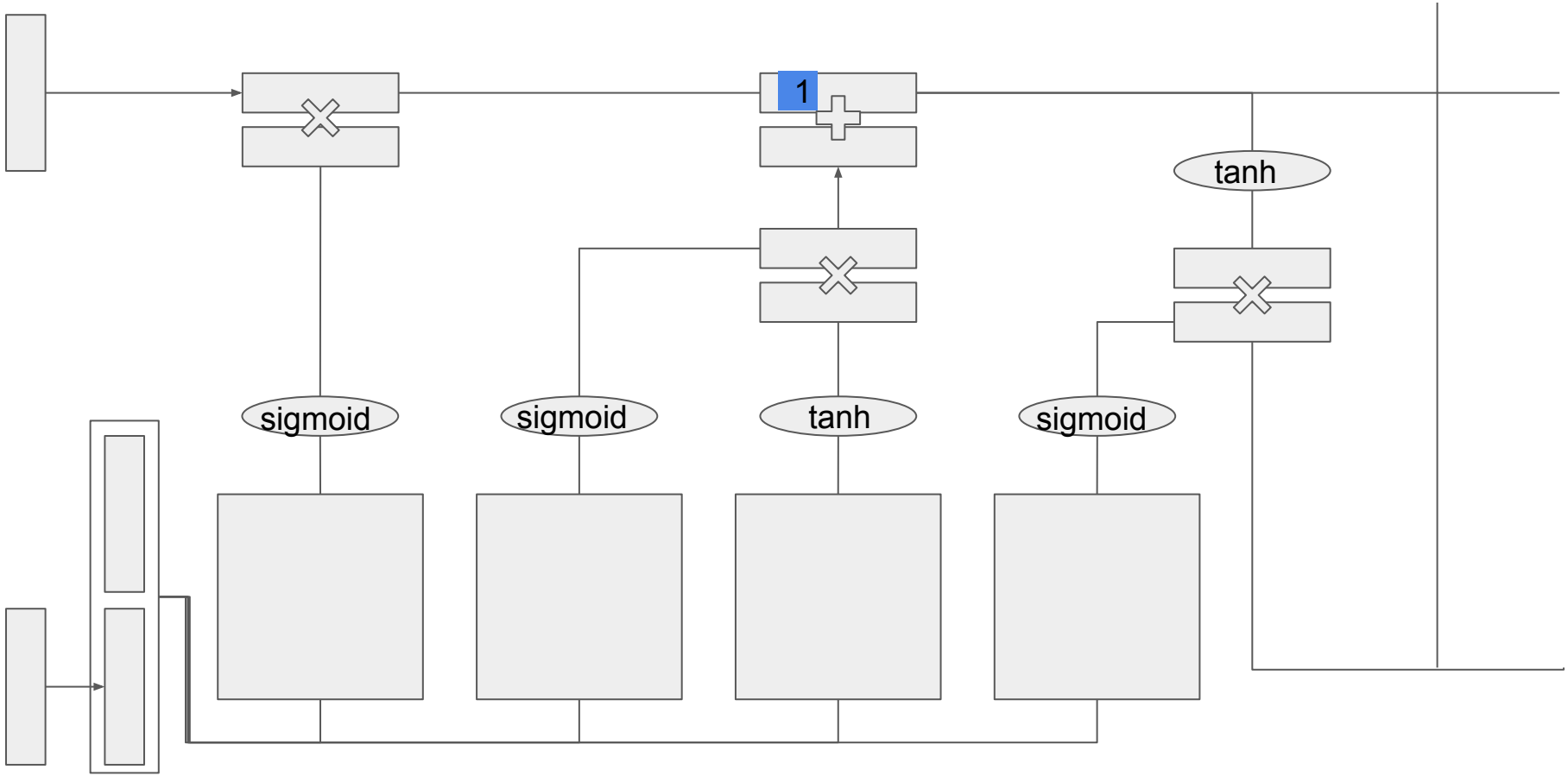
Memory Process

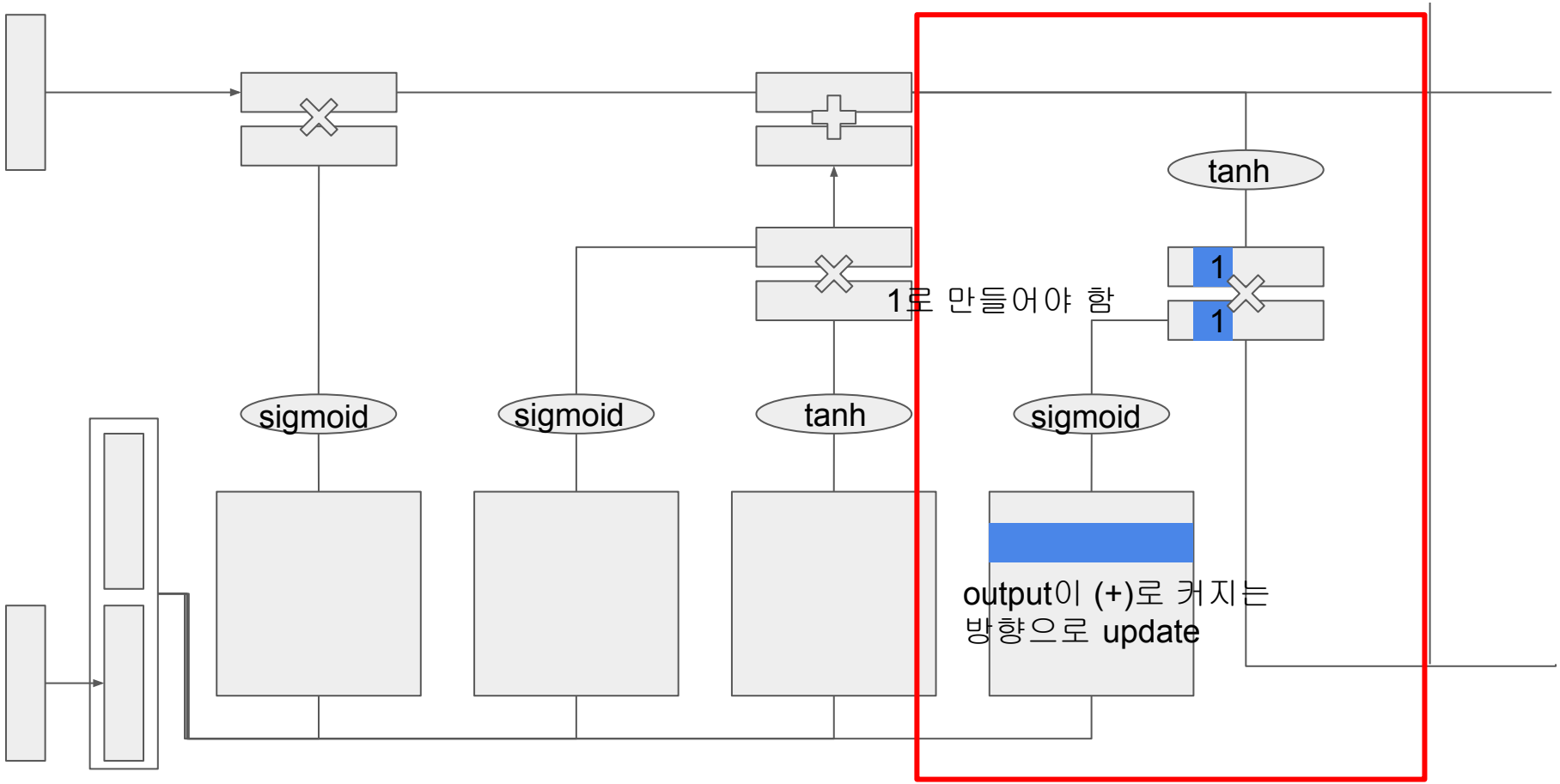


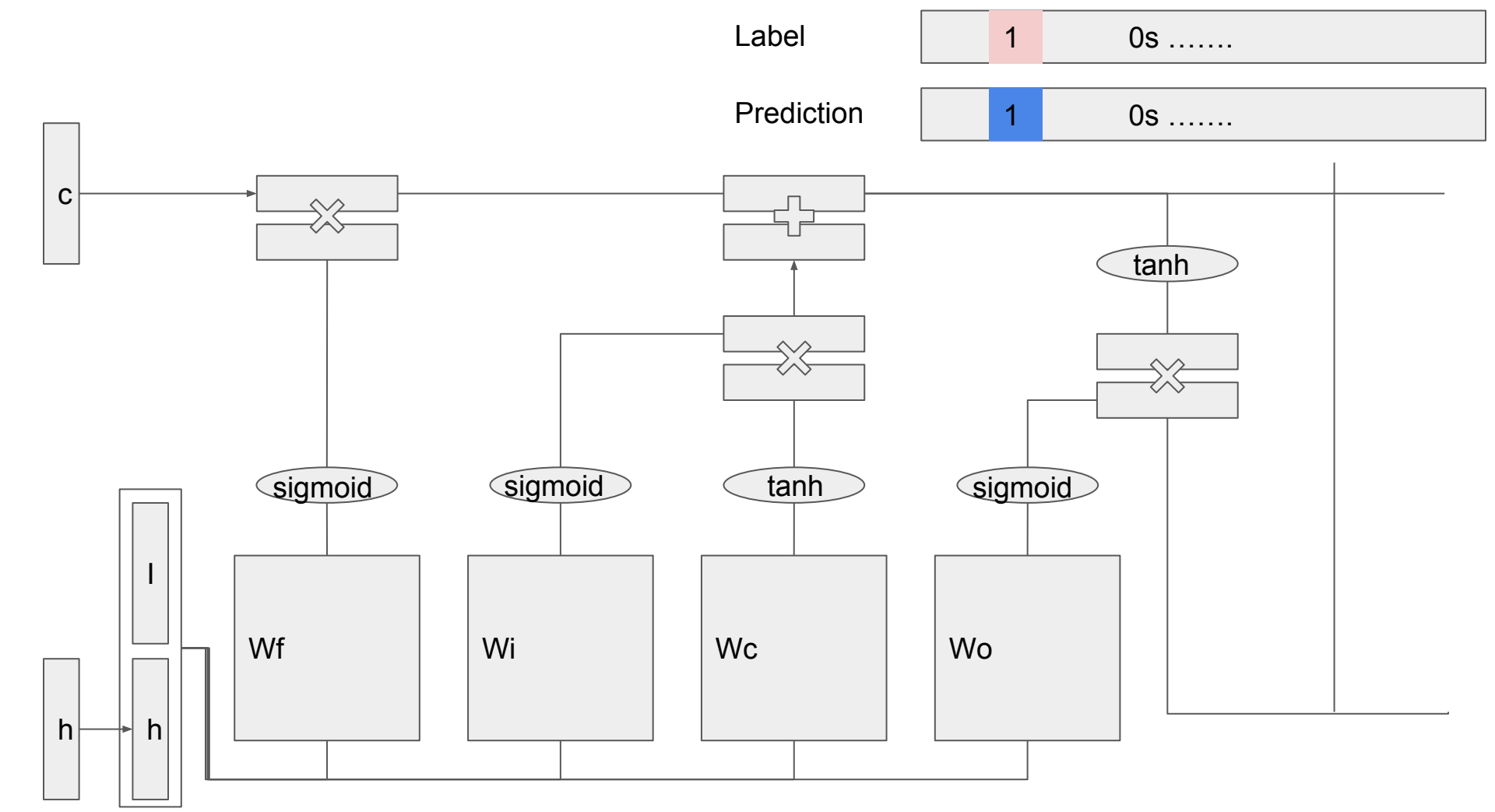

어떤 값이 오더라도 Cell update 불가 → don't care
0
sigmoid
sigmoid
tanh
sigmoid
tanh

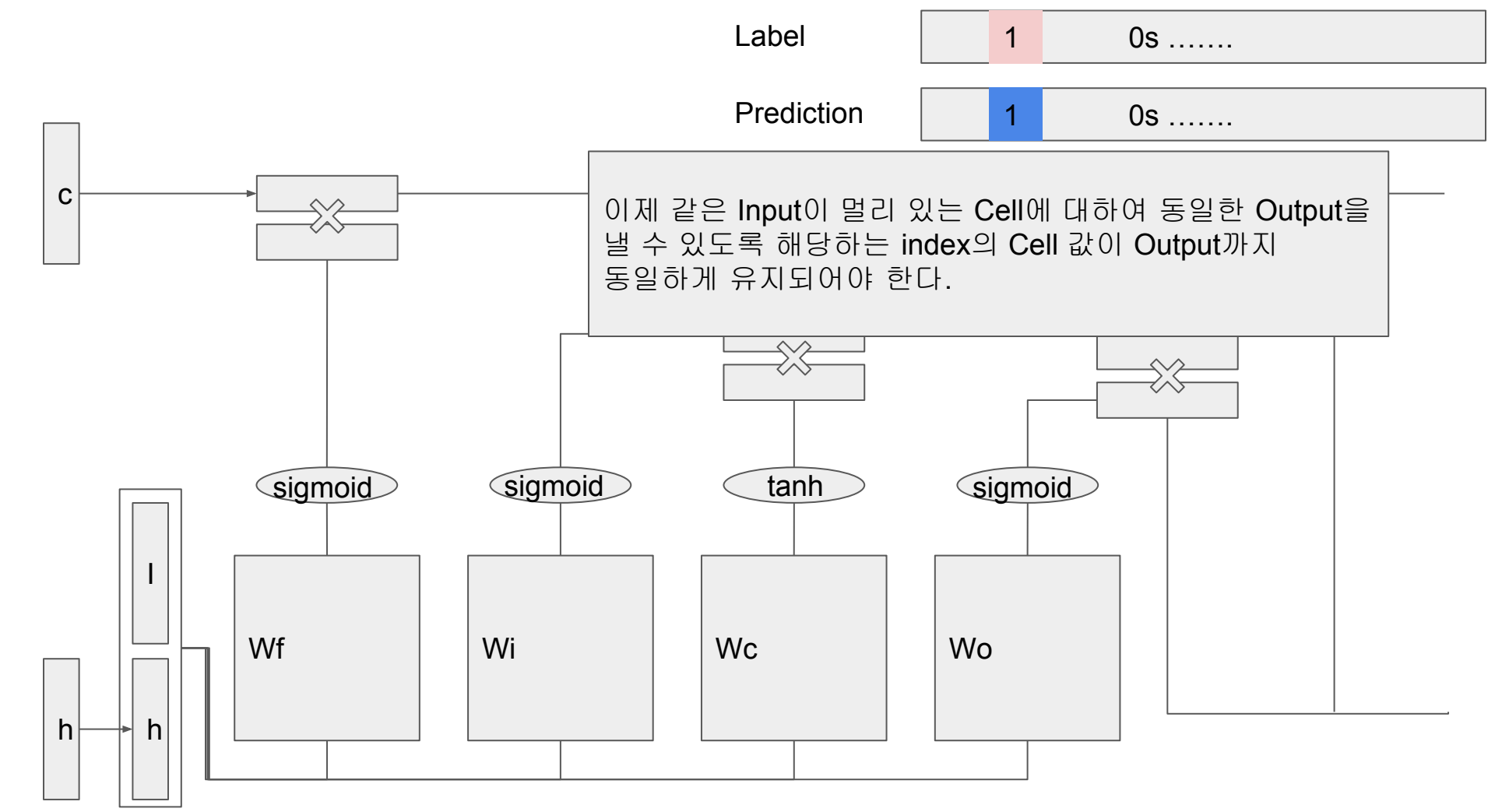

어떤 값이 오더라도 Cell update 불가 → don't care
0
1로 만들어야 함
1
1
1로 만들어야 함
tanh
sigmoid
sigmoid
tanh
sigmoid
output이 (+)로 커지는 방향으로 update

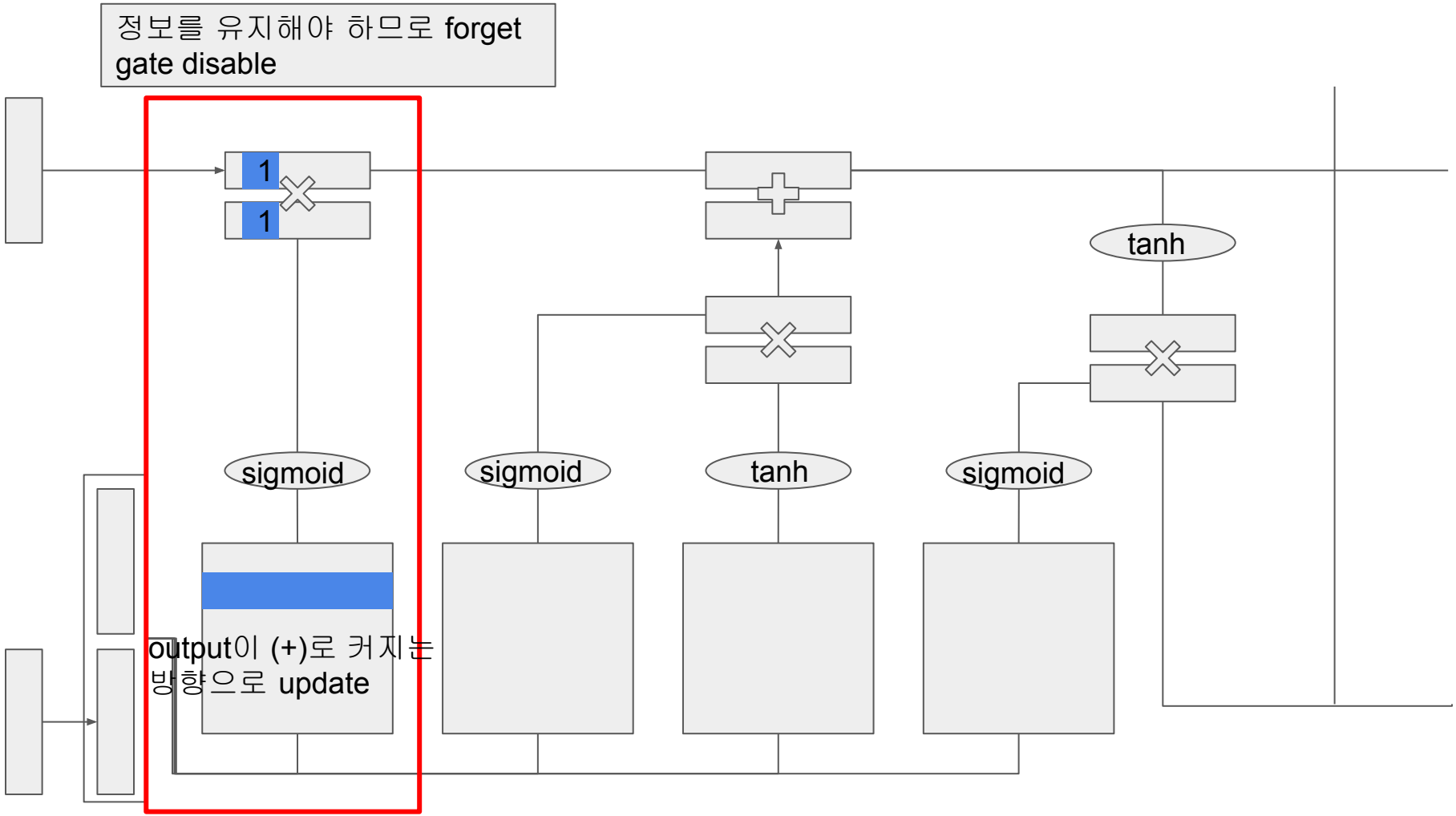

출력값 update
0
1
tanh
sigmoid
sigmoid
tanh
sigmoid

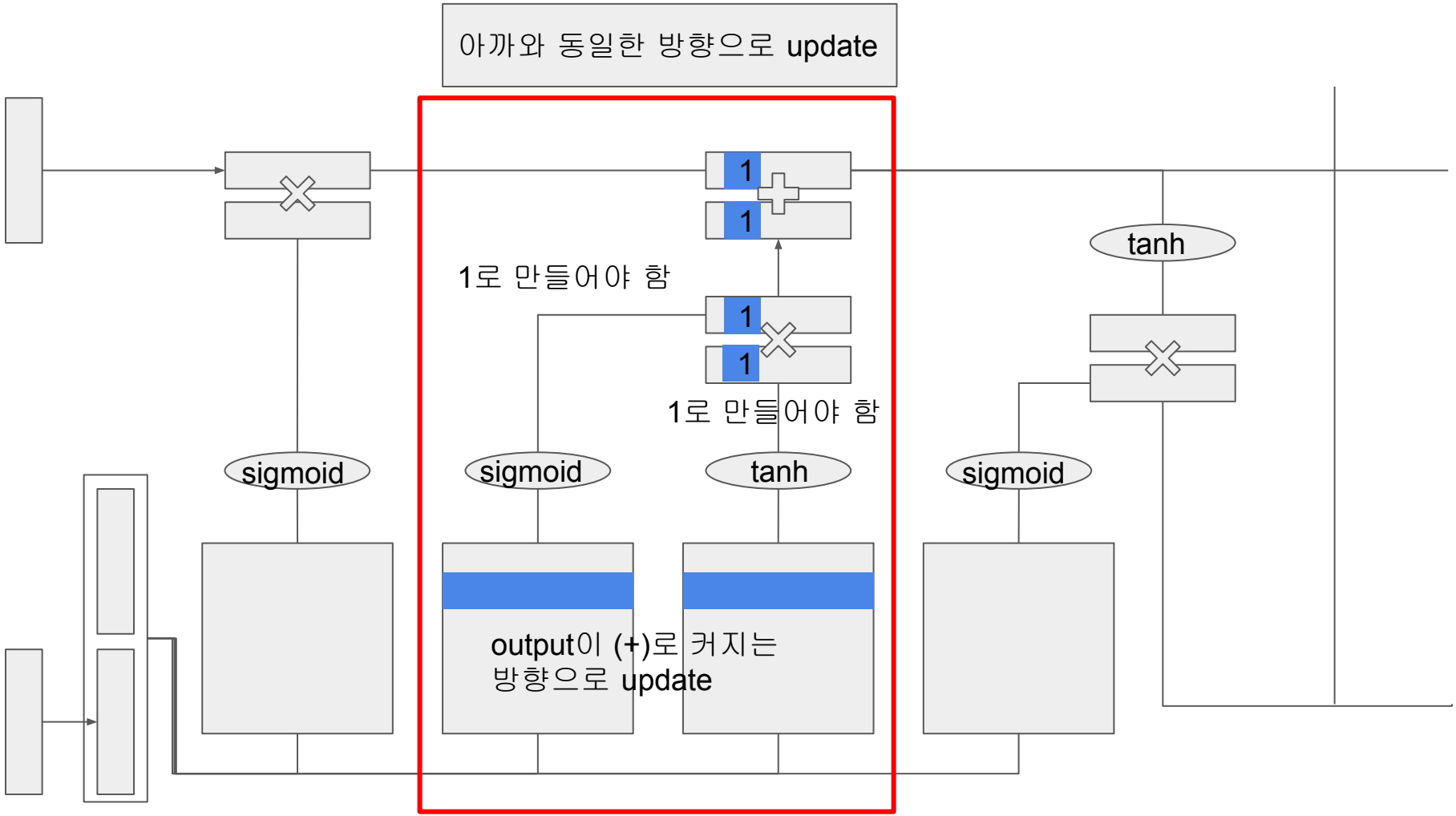

1
tanh
sigmoid
sigmoid
tanh
sigmoid

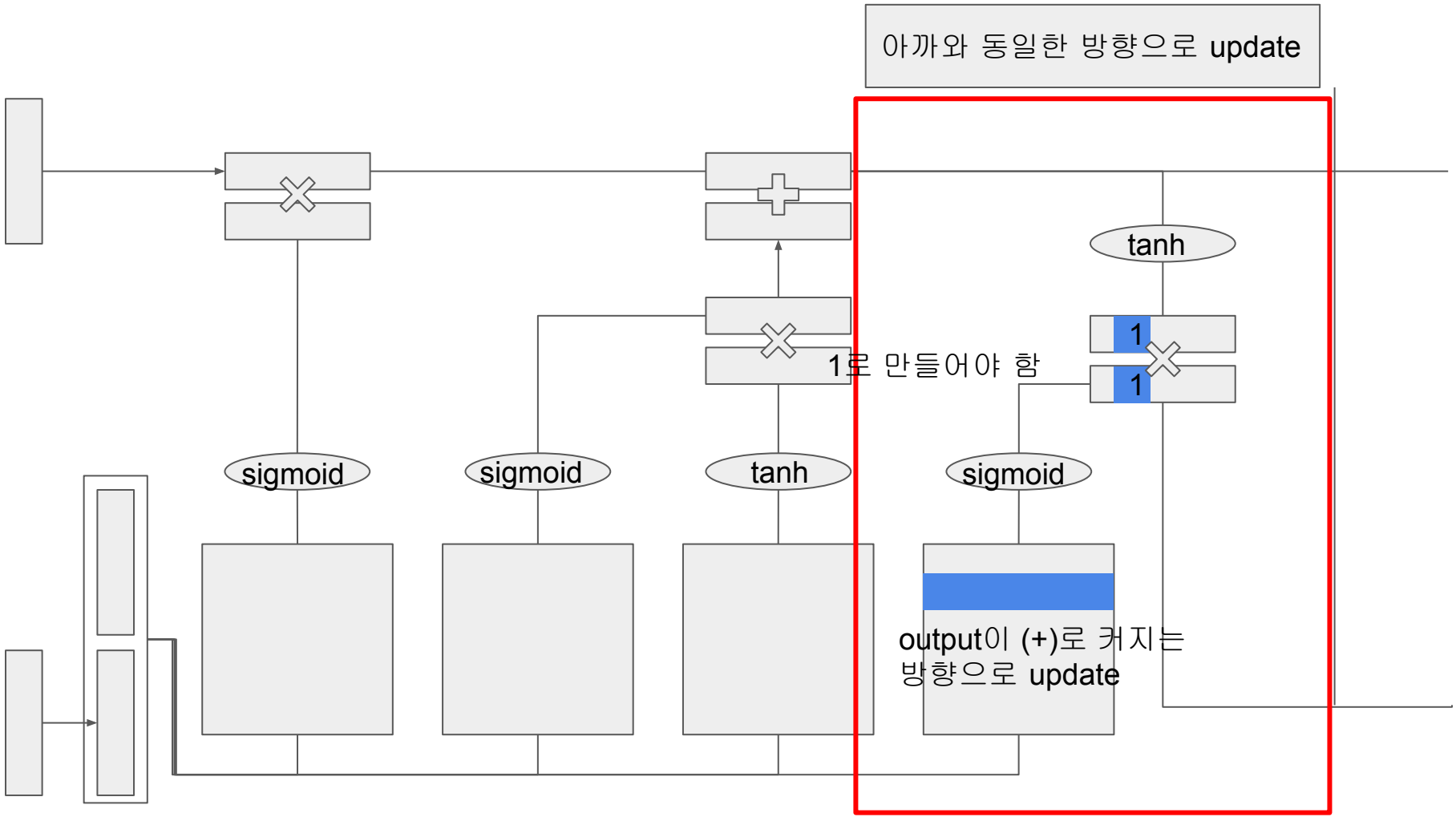

tanh
1
1
1로 만들어야 함
sigmoid
sigmoid
tanh
sigmoid
output이 (+)로 커지는
방향으로 update


Label
1
0s …….
Prediction
1
0s …….
c
tanh
sigmoid
sigmoid
tanh
sigmoid
I
h
h
Wf
Wi
Wc
Wo

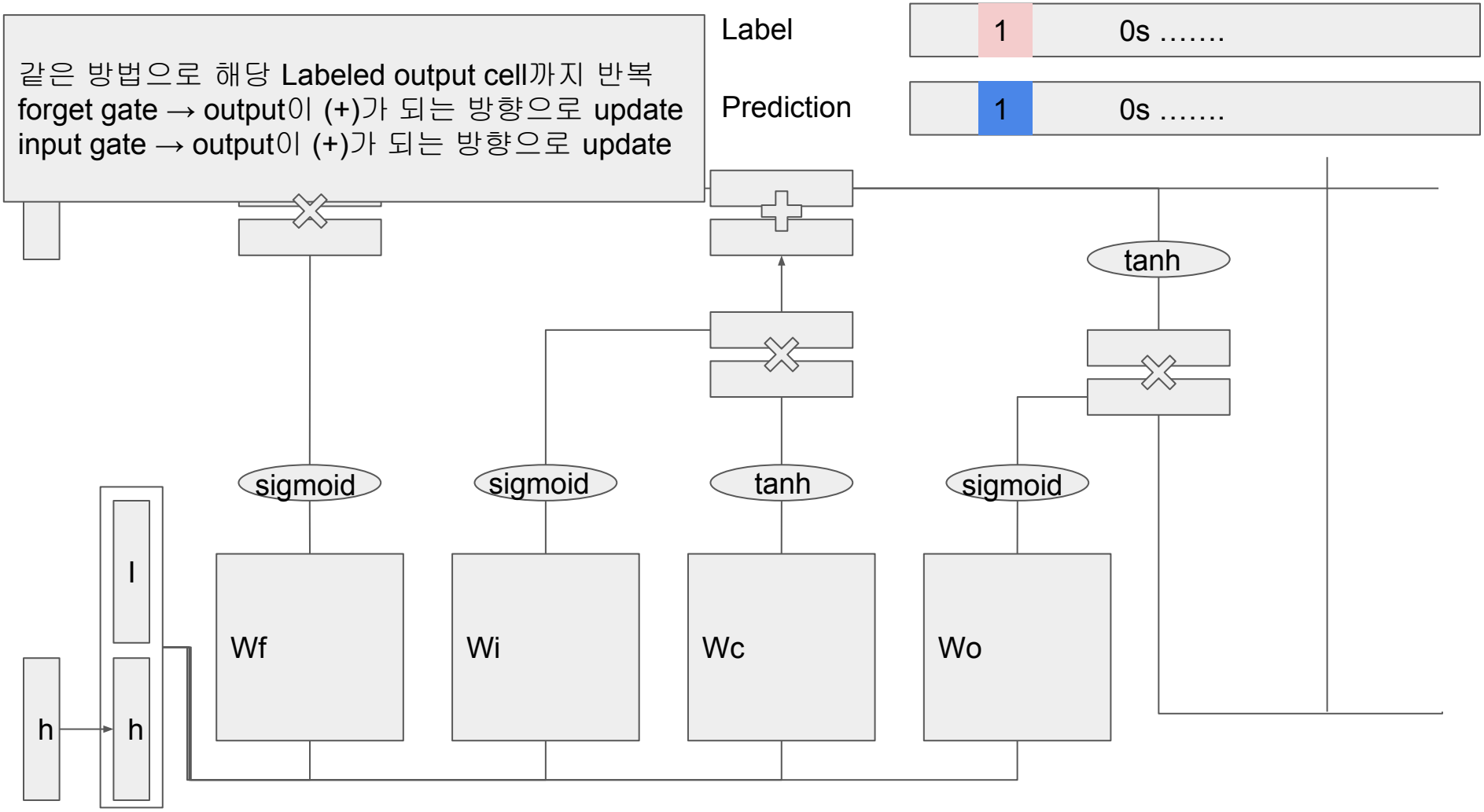

Label
1
0s …….
Prediction
1
0s …….
c
이제 같은 Input이 멀리 있는 Cell에 대하여 동일한 Output을 낼 수 있도록 해당하는 index의 Cell 값이 Output까지 동일하게 유지되어야 한다.
sigmoid
sigmoid
tanh
sigmoid
I
h
h
Wf
Wi
Wc
Wo

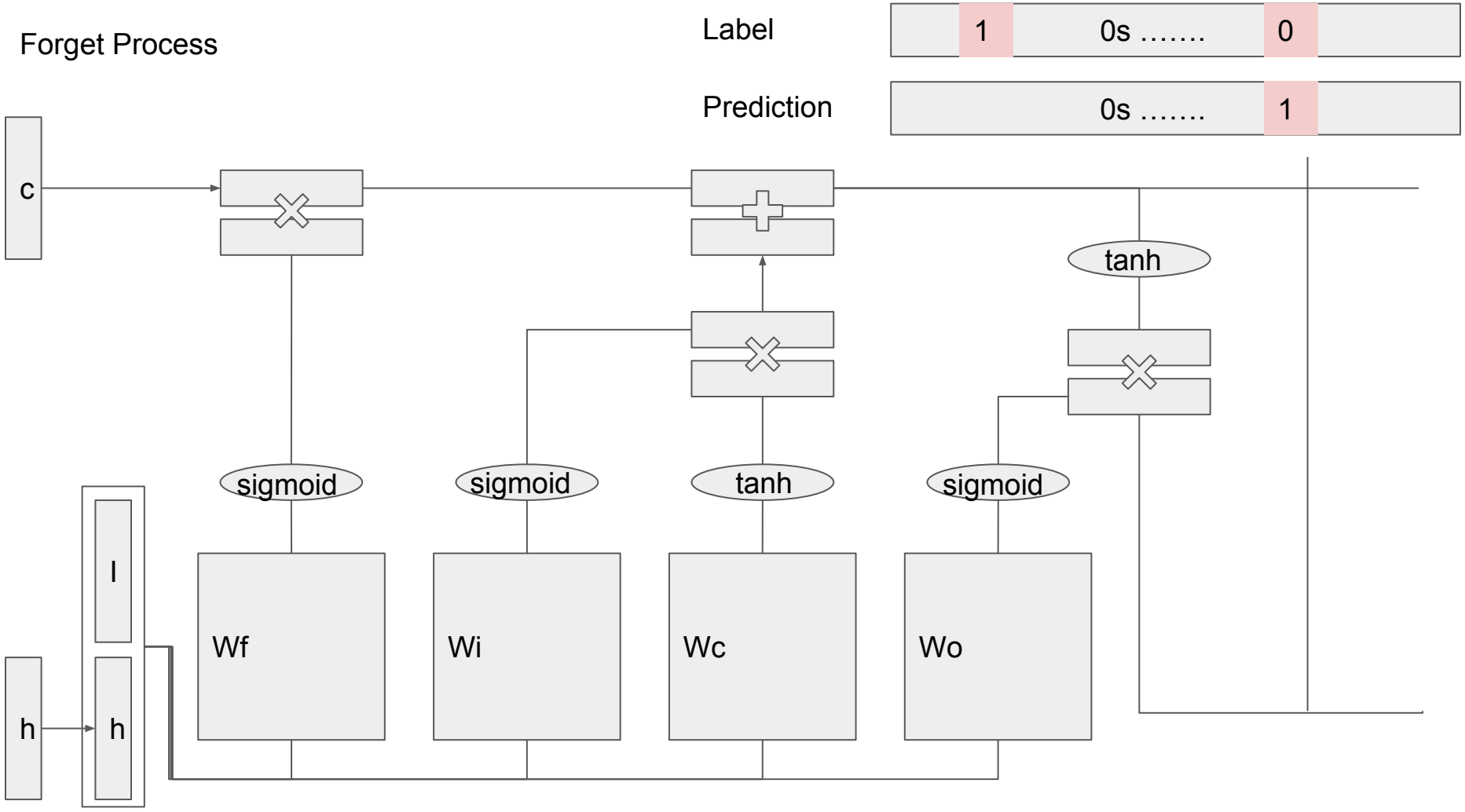

정보를 유지해야 하므로 forget gate disable
1
1
sigmoid
sigmoid
tanh
sigmoid
tanh
output이 (+)로 커지는 방향으로 update

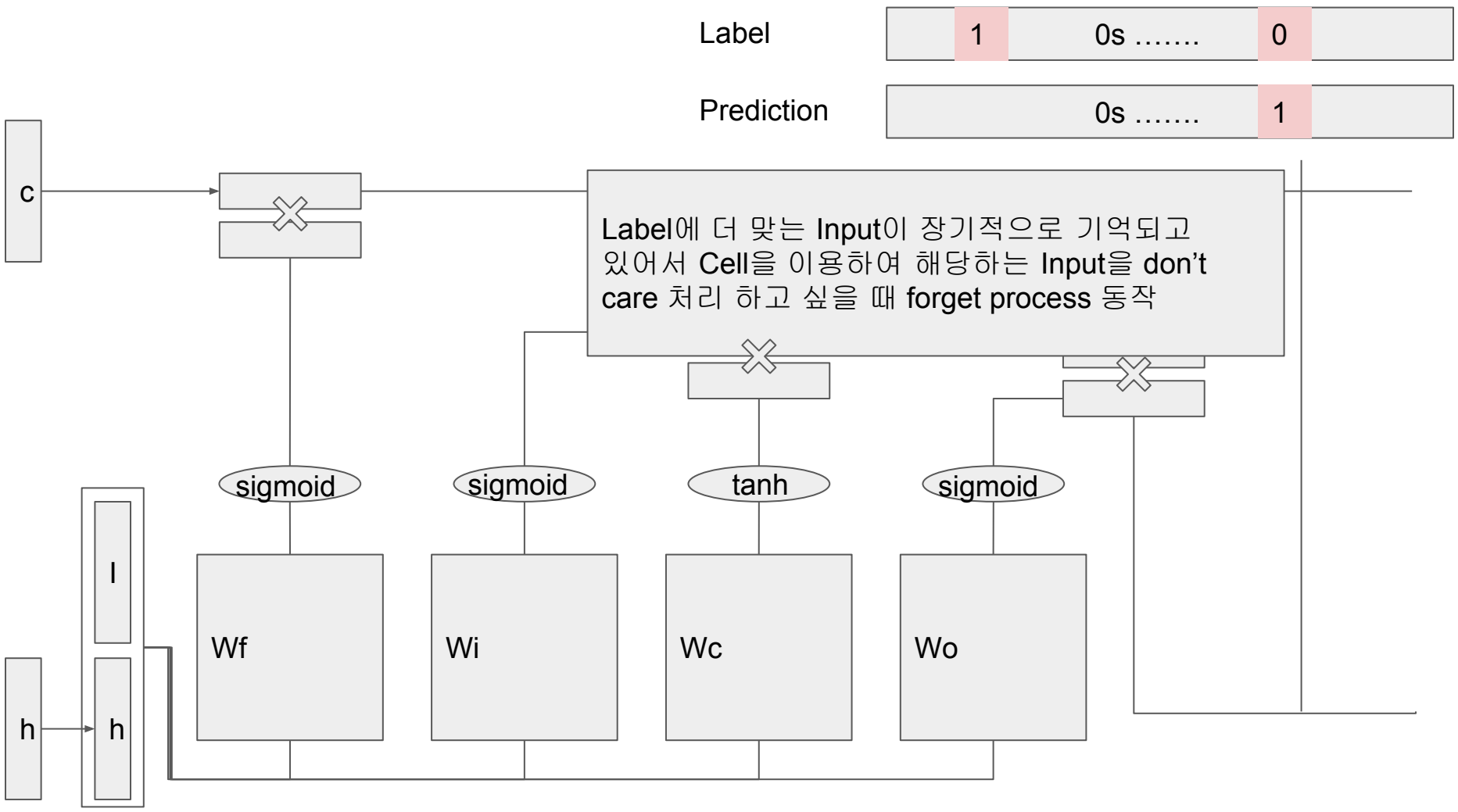

아까와 동일한 방향으로 update
1
1
1로 만들어야 함
1
1
1로 만들어야 함
tanh
sigmoid
sigmoid
tanh
sigmoid
output이 (+)로 커지는
방향으로 update

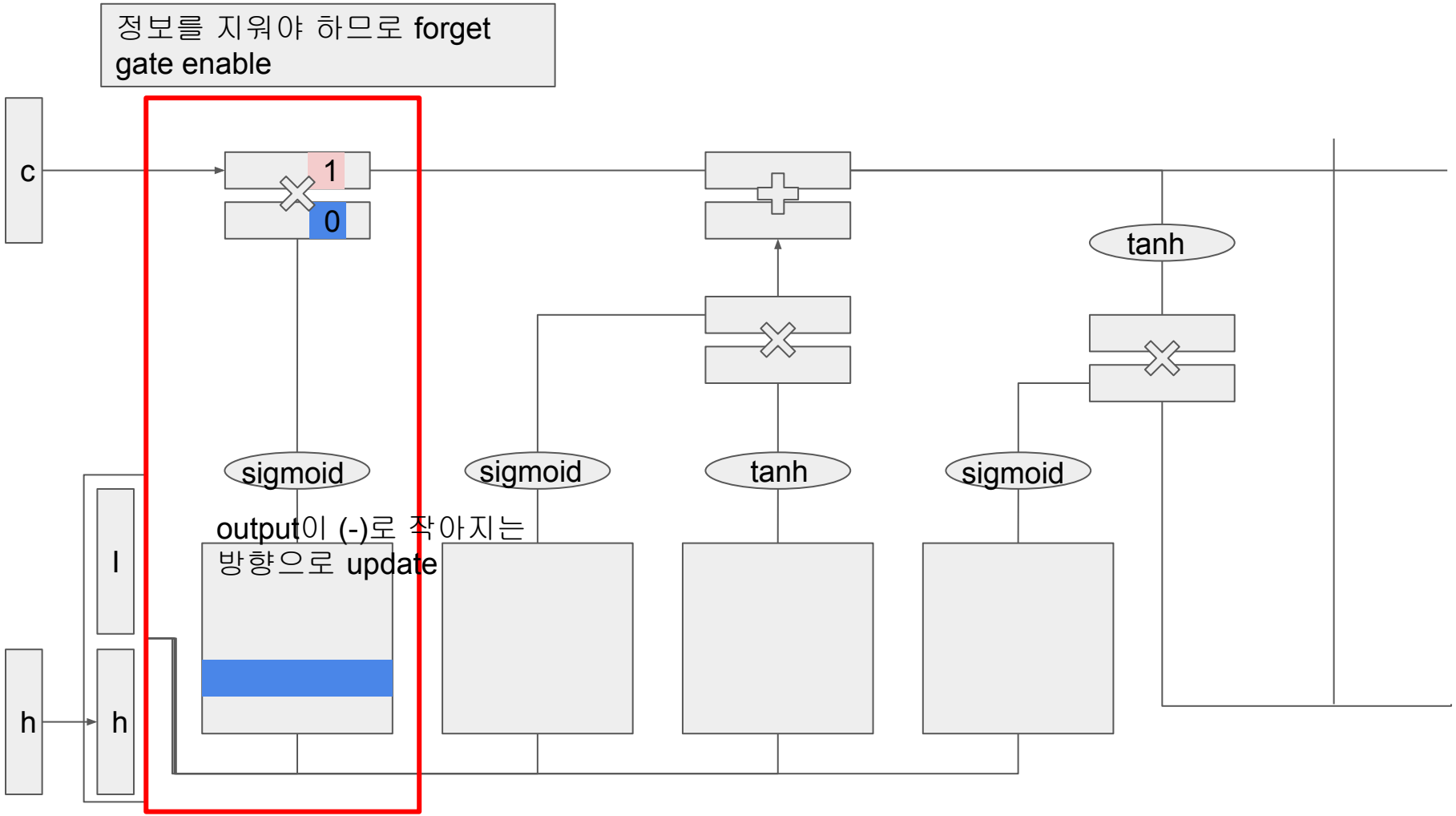

아까와 동일한 방향으로 update
tanh
1
1
1로 만들어야 함
sigmoid
sigmoid
tanh
sigmoid
output이 (+)로 커지는
방향으로 update

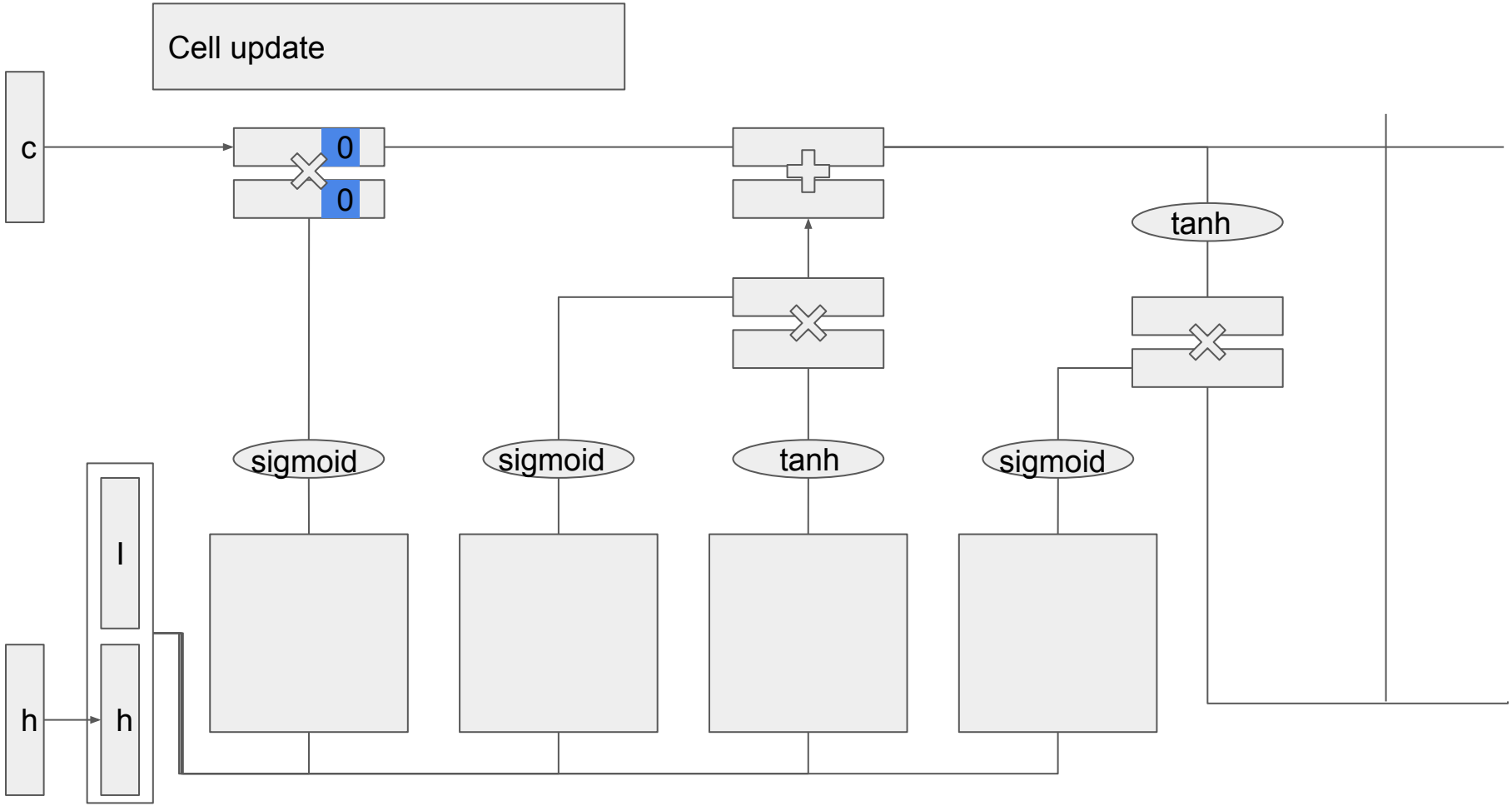
같은 방법으로 해당 Labeled output cell까지 반복
forget gate → output이 (+)가 되는 방향으로 update
input gate → output이 (+)가 되는 방향으로 update

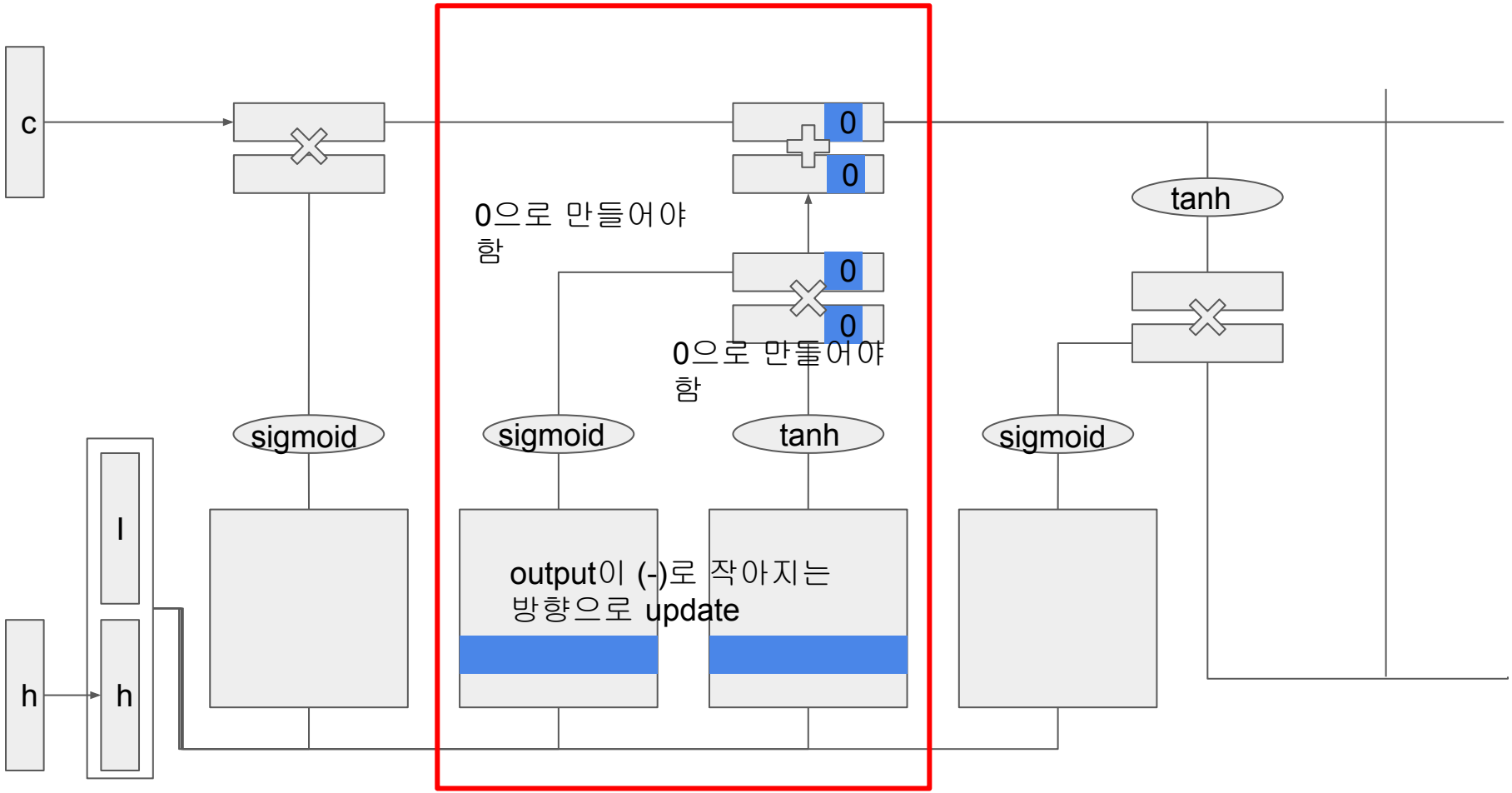



Forget Process
Label
1
0s …….
0
Prediction
0s …….
1
c
tanh
sigmoid
sigmoid
tanh
sigmoid
I
h
h
Wf
Wi
Wc
Wo

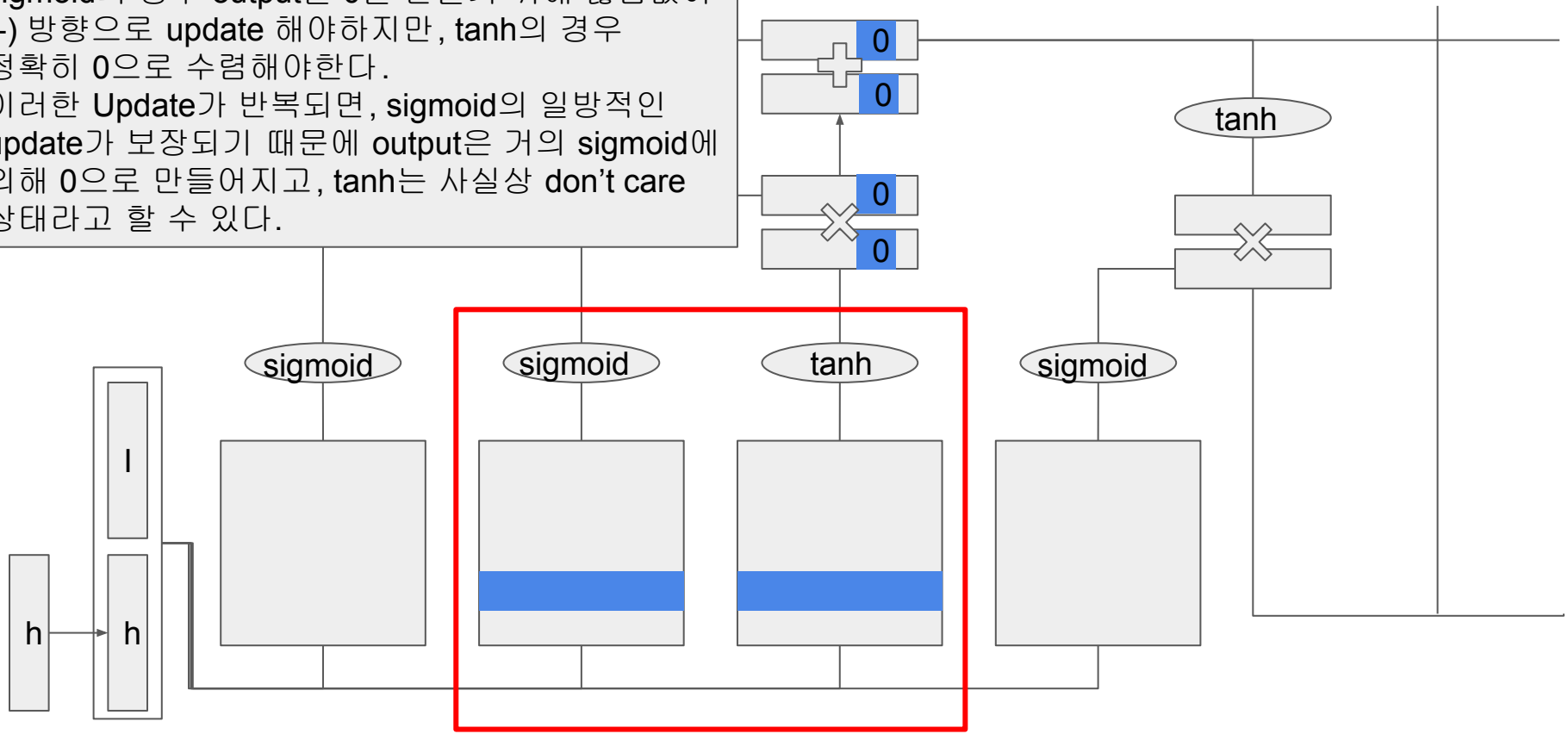

Label
1
0s …….
0
Prediction
0s …….
1
c
Label에 더 맞는 Input이 장기적으로 기억되고 있어서 Cell을 이용하여 해당하는 Input을 don't care 처리 하고 싶을 때 forget process 동작
sigmoid
sigmoid
tanh
sigmoid
I
h
h
Wf
Wi
Wc
Wo


정보를 지워야 하므로 forget gate enable
c
1
0
tanh
sigmoid
sigmoid
tanh
sigmoid
output이 (-)로 작아지는 방향으로 update
I
h
h

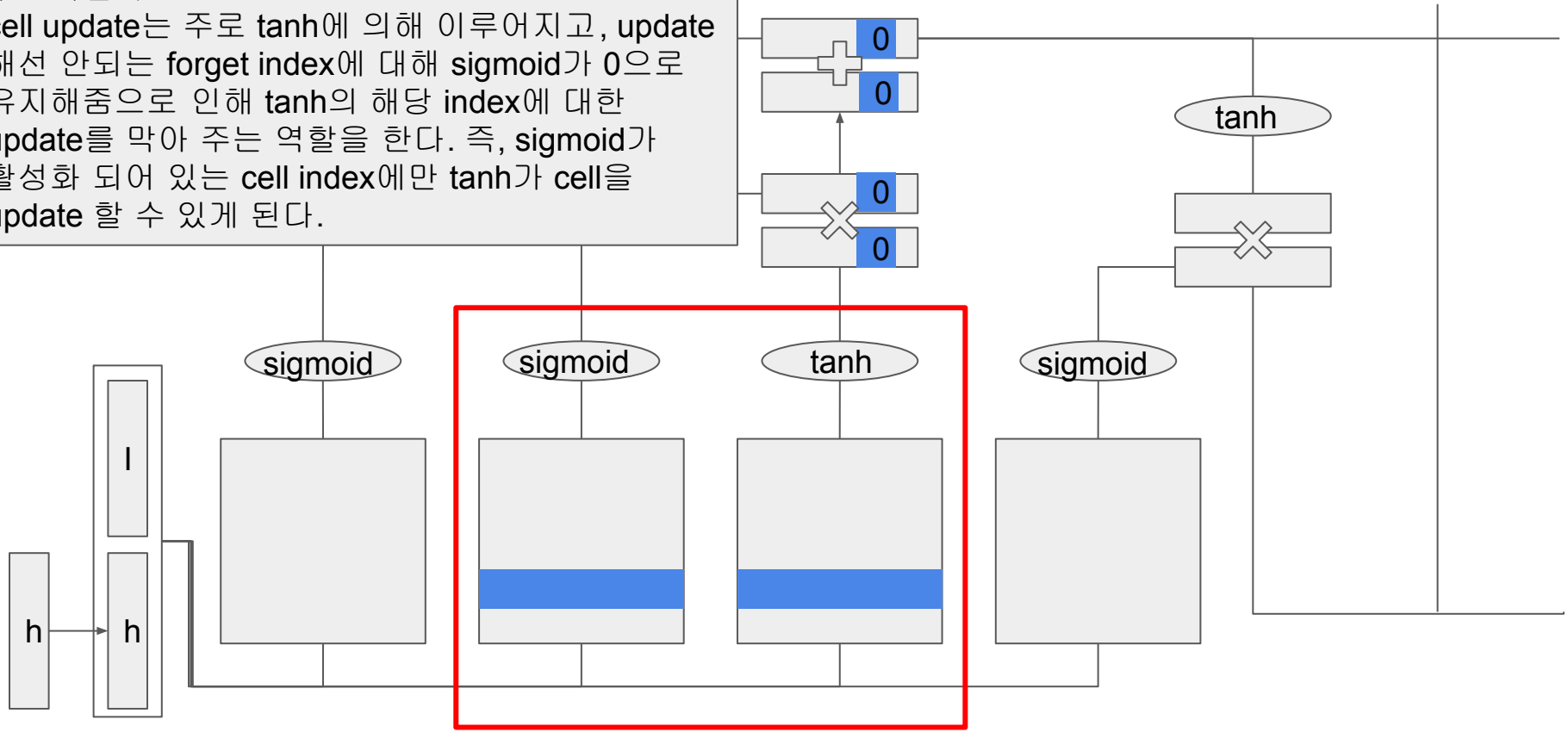

Cell update
c
0
0
tanh
sigmoid
sigmoid
tanh
sigmoid
I
h
h

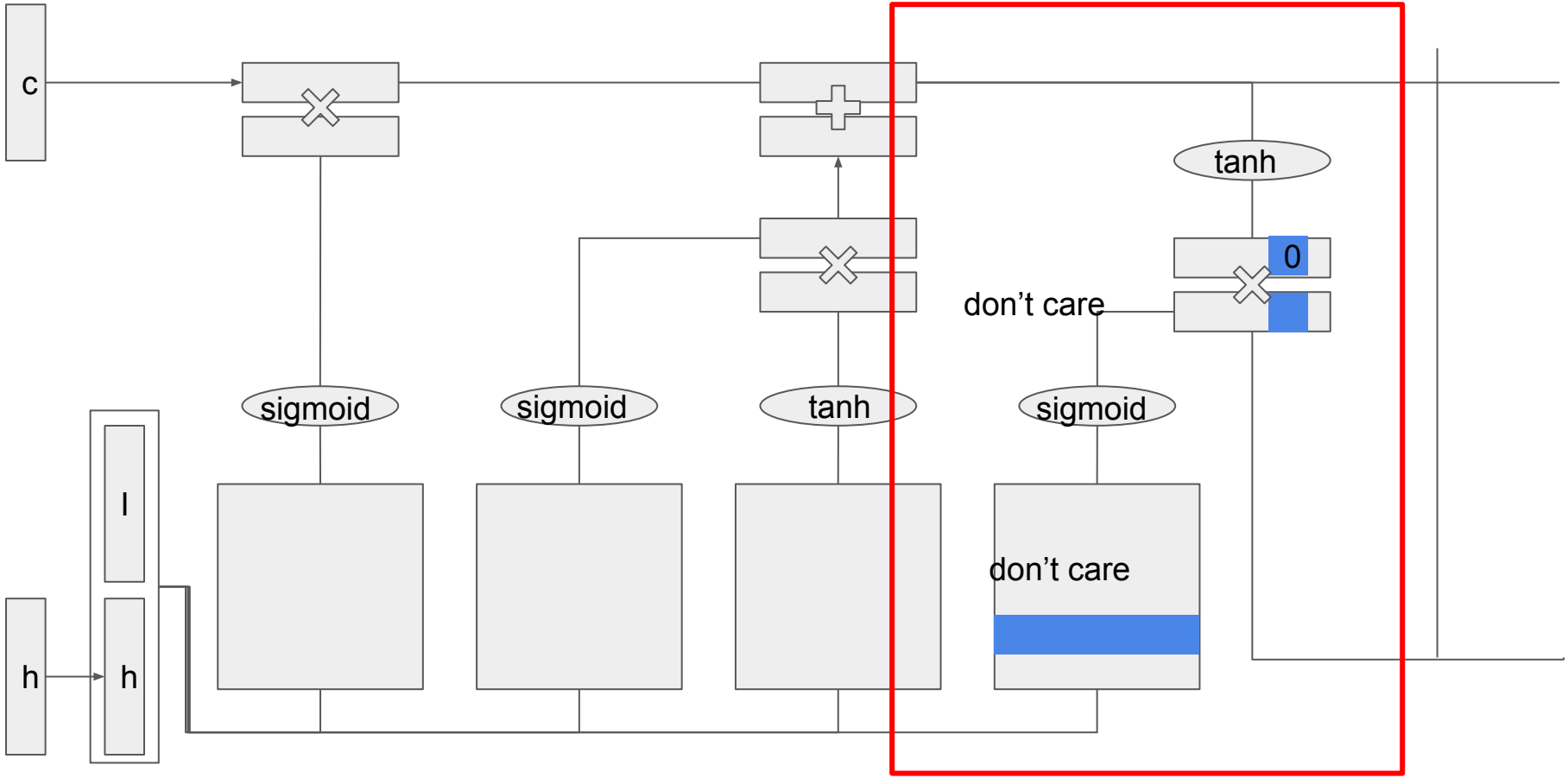

c
0
0
0으로 만들어야 함
0
0
0으로 만들어야 함
tanh
sigmoid
sigmoid
tanh
sigmoid
l
h
h
output이 (-)로 작아지는 방향으로 update

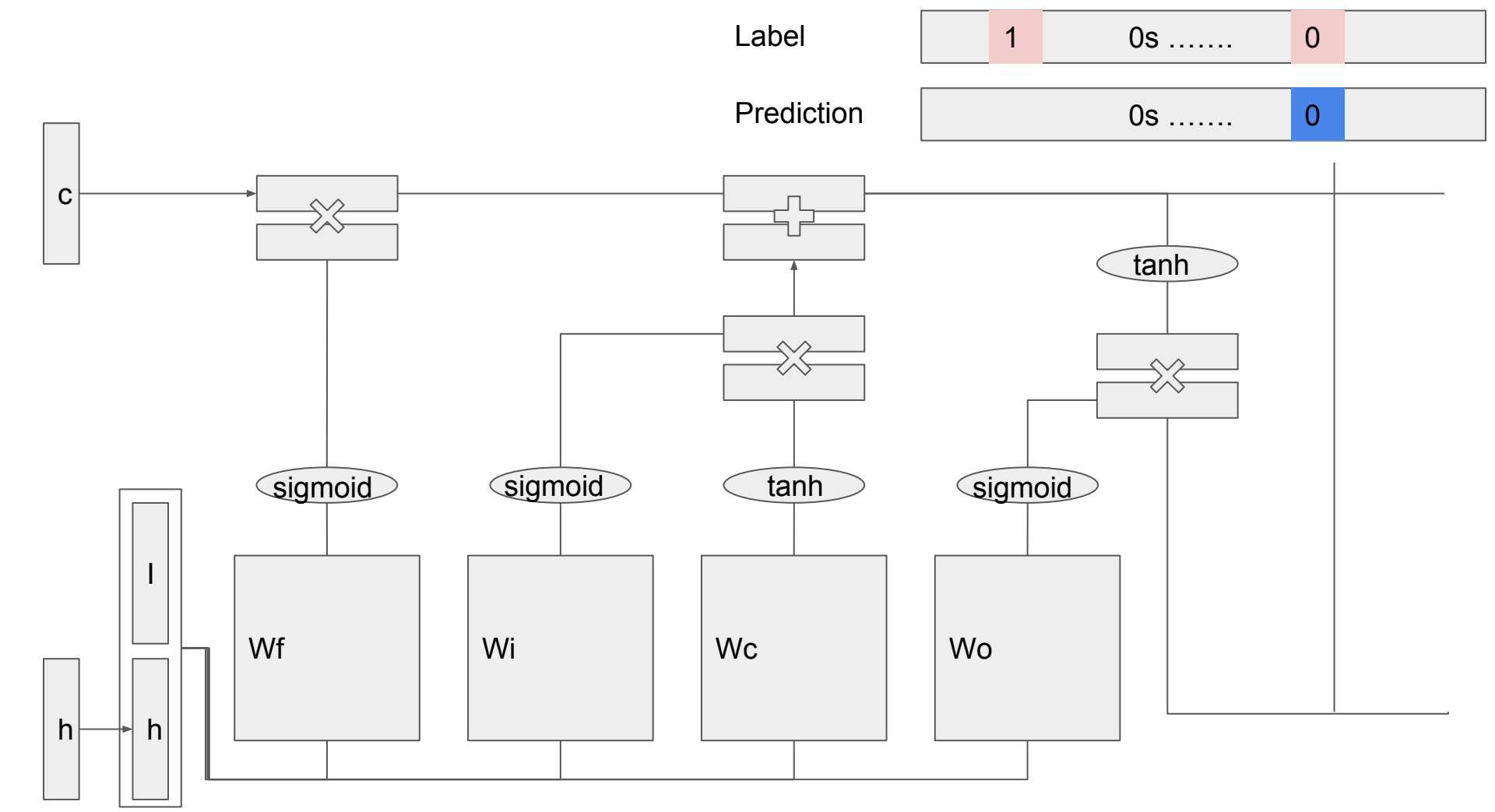
sigmoid의 경우 output을 0을 만들기 위해 끊임없이 (-) 방향으로 update 해야하지만, tanh의 경우 정확히 0으로 수렴해야한다.
이러한 Update가 반복되면, sigmoid의 일방적인 update가 보장되기 때문에 output은 거의 sigmoid에 의해 0으로 만들어지고, tanh는 사실상 don't care 상태라고 할 수 있다.

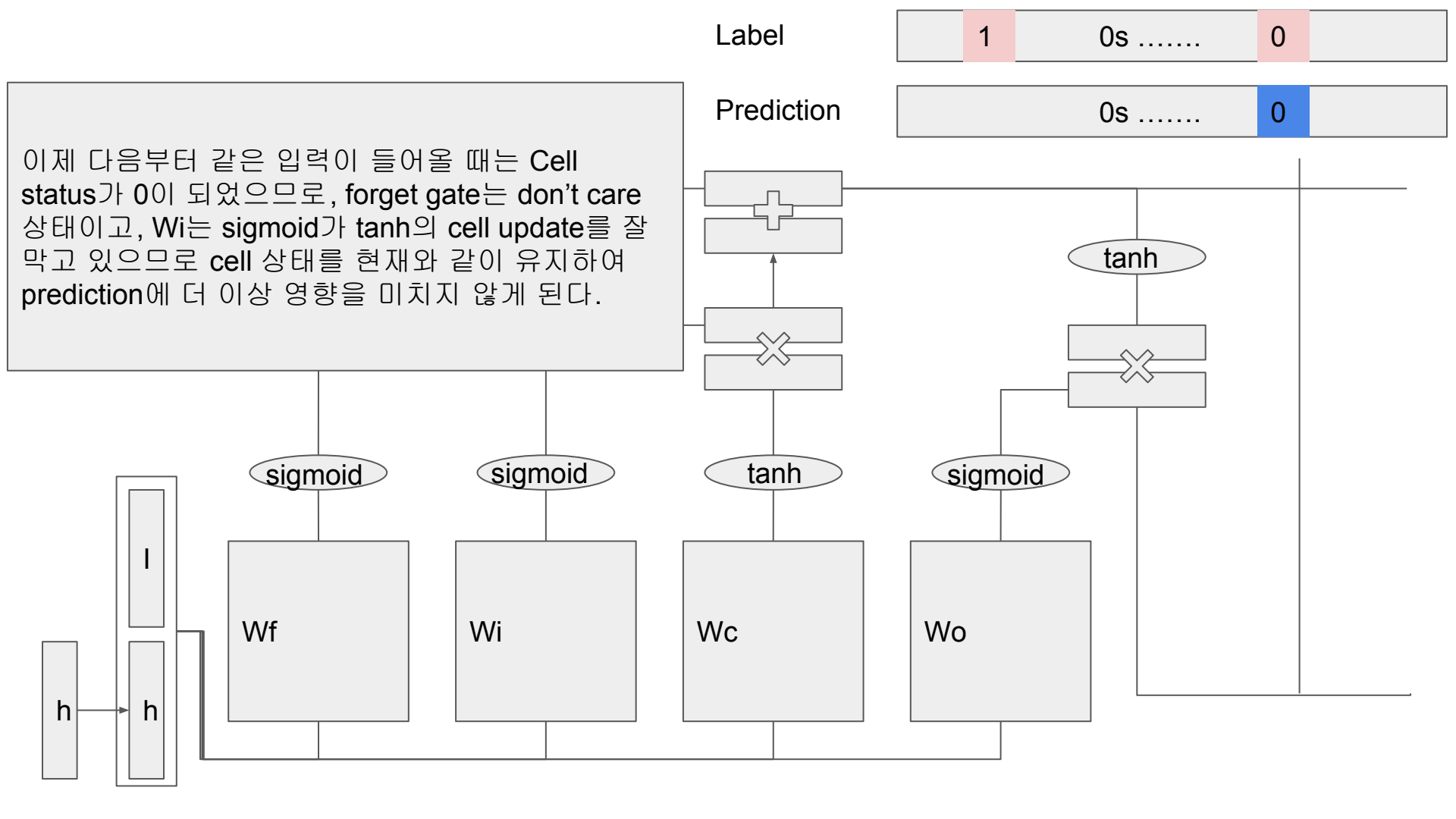


이러한 부분에서 sigmoid와 tanh의 역할이 두드러진다.
cell update는 주로 tanh에 의해 이루어지고, update 해선 안되는 forget index에 대해 sigmoid가 0으로 유지해줌으로 인해 tanh의 해당 index에 대한 update를 막아 주는 역할을 한다. 즉, sigmoid가 활성화 되어 있는 cell index에만 tanh가 cell을 update 할 수 있게 된다.




c
tanh
0
don’t care
sigmoid
sigmoid
tanh
sigmoid
l
h
h
don’t care


Label
1
0s …….
0
Prediction
0s …….
0
c
tanh
sigmoid
sigmoid
tanh
sigmoid
I
h
h
Wf
Wi
Wc
Wo


Label
1
0s …….
0
Prediction
0s …….
0
이제 다음부터 같은 입력이 들어올 때는 Cell status가 0이 되었으므로, forget gate는 don't care 상태이고, Wi는 sigmoid가 tanh의 cell update를 잘 막고 있으므로 cell 상태를 현재와 같이 유지하여 prediction에 더 이상 영향을 미치지 않게 된다.
tanh
sigmoid
sigmoid
tanh
sigmoid
I
h
h
Wf
Wi
Wc
Wo