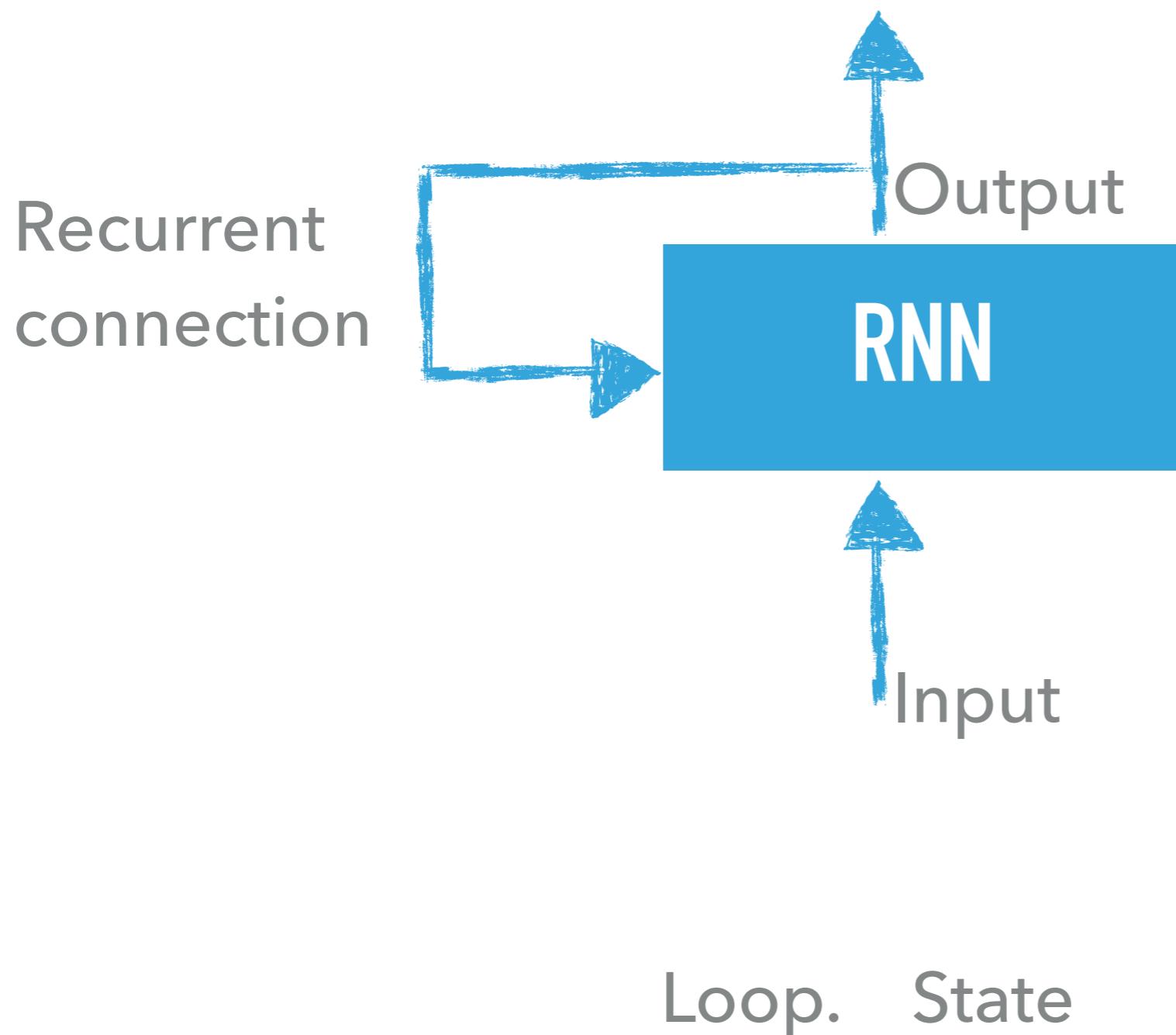


LONG SHORT TERM MEMORY

LSTM NETWORKS

Simple RNN - recurrent neural network



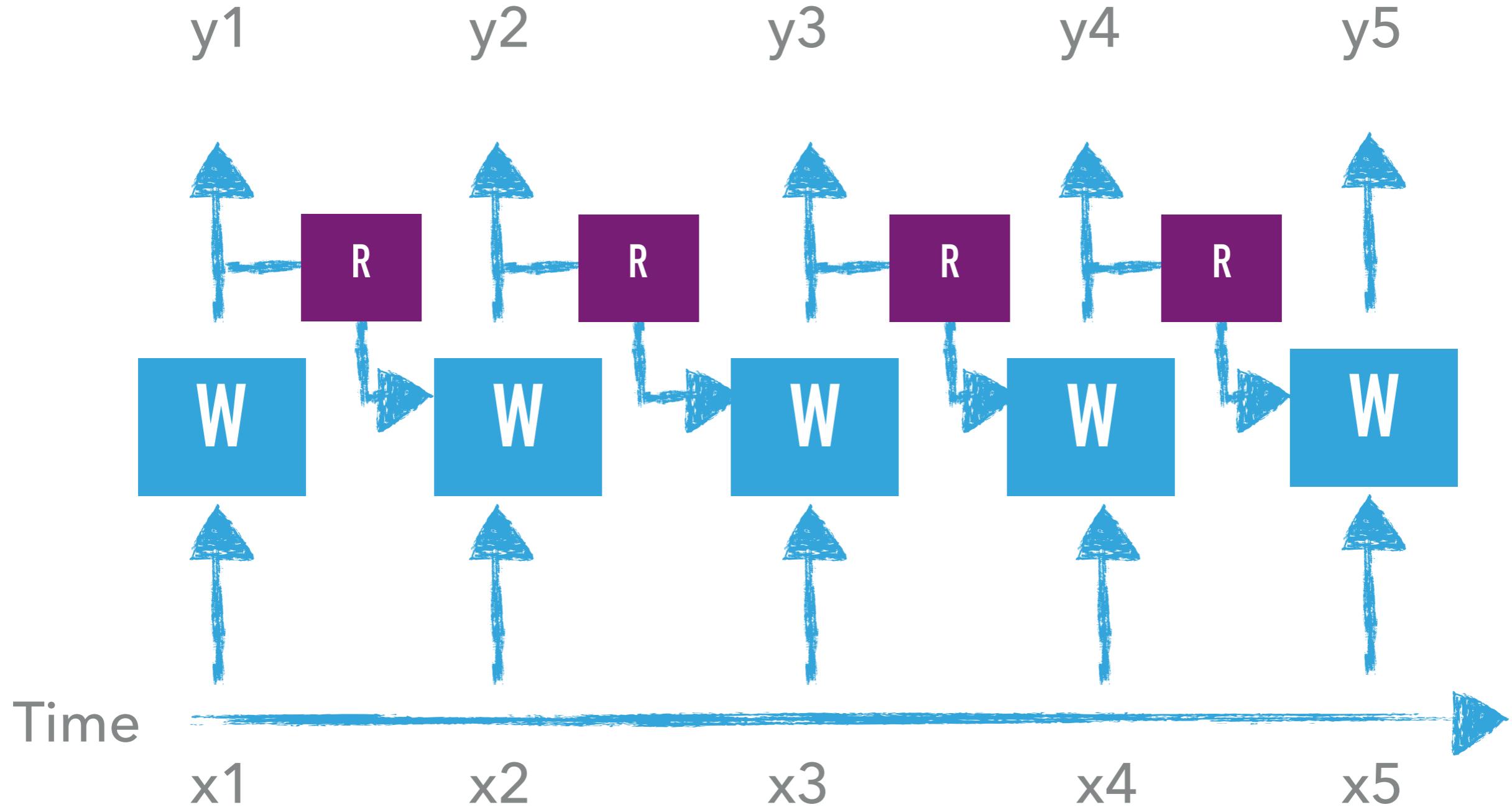
PLAIN OLD NN

```
activation(dot(W, input_t) + b)
```

RNN

```
activation(dot(W, input_t) +
           dot(U, state_t) + b)
```

RUN



MAIN PURPOSE

Remember what it has seen so far (state)

Capture Long Distance Dependencies

MAIN PURPOSE

Remember what it has seen so far (state)

Capture Long Distance Dependencies

Not so good

TURNS OUT ...

Simple RNNs are too simple to deal with long-distance dependencies

A RNN variant, LSTM work much better.

ALMOST ALL THE APPLICATIONS OF RNN USE LSTM

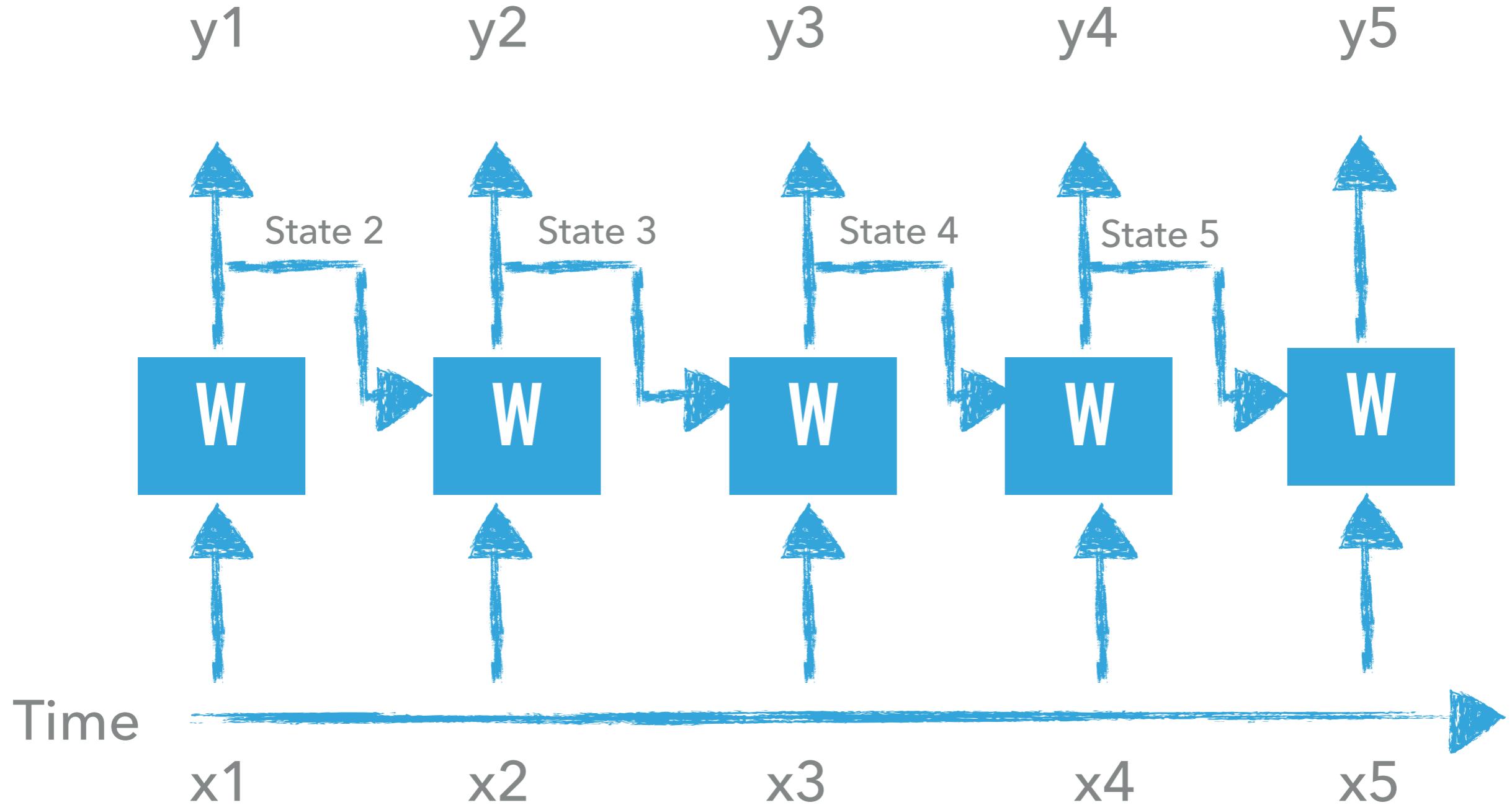
- ▶ Speech recognition
- ▶ Machine translation
- ▶ Image captioning
- ▶ ...

AS ALWAYS

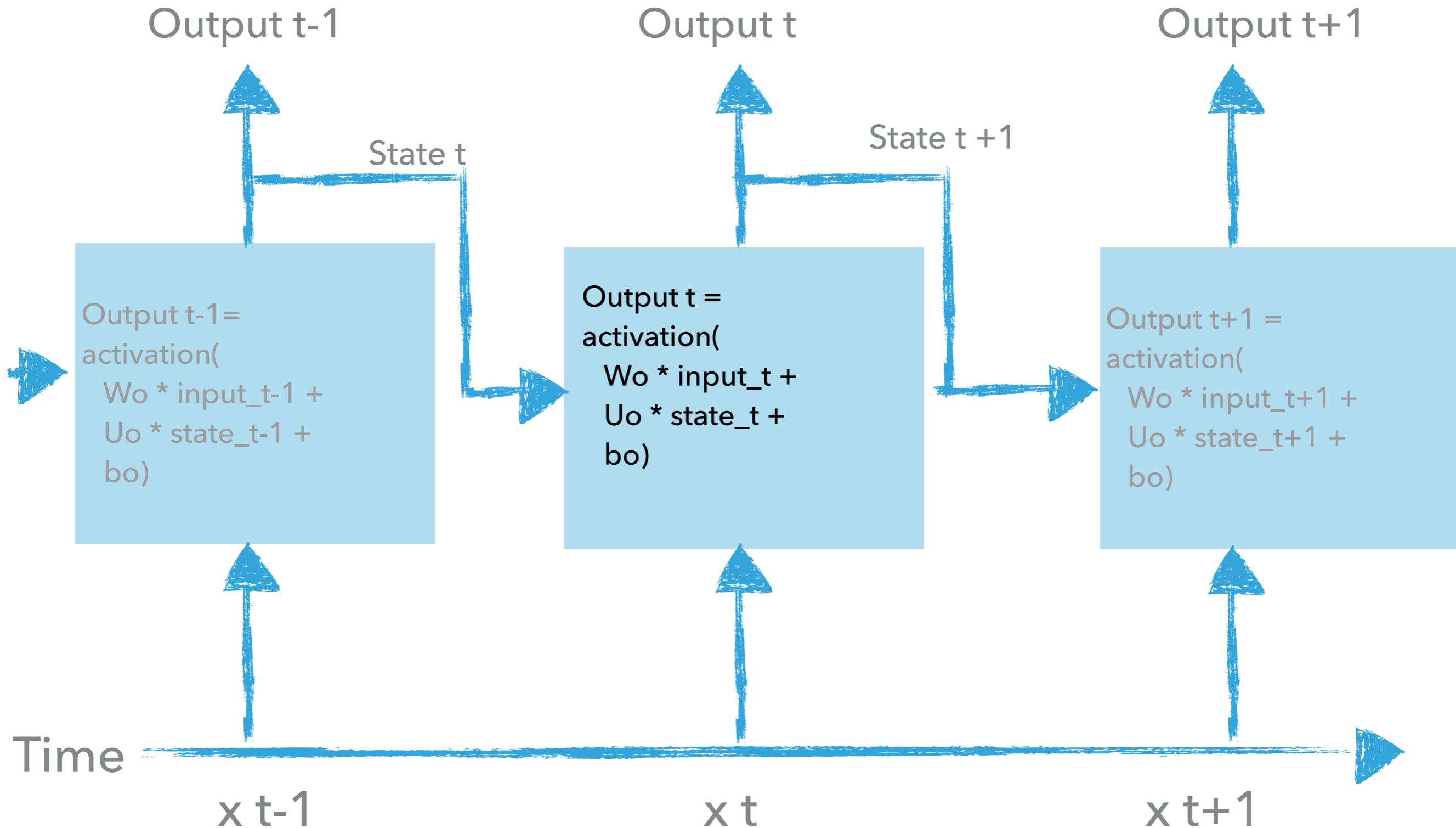
May look complicated

“Not so complicated—merely complex.” - Chollet

CHOLLET'S EXPLANATION



Plain Vanilla RNN



LSTM

Output t-1

Output t

Output t+1

c_t

c_{t+1}

State t

State t +1

Output t-1 =
activation(
 $W_o * \text{input}_{t-1} +$
 $U_o * \text{state}_{t-1} +$
 $V_o * c_{t-1}$
 b_o)

Output t =
activation(
 $W_o * \text{input}_t +$
 $U_o * \text{state}_t +$
 $V_o * c_t$
 b_o)

Output t+1 =
activation(
 $W_o * \text{input}_{t+1} +$
 $U_o * \text{state}_{t+1} +$
 $V_o * c_{t+1}$
 b_o)



Time

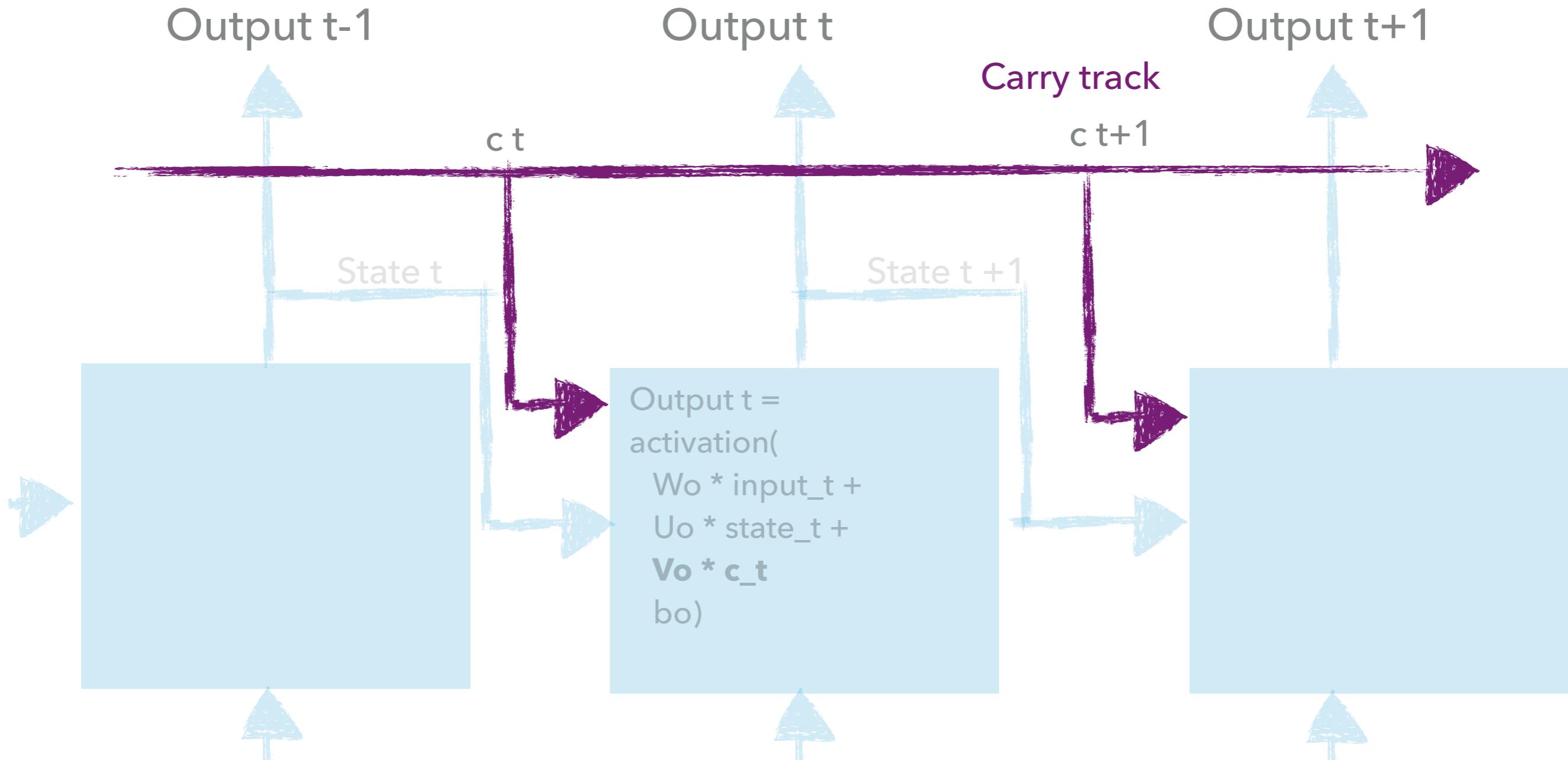
x_{t-1}

x_t

x_{t+1}

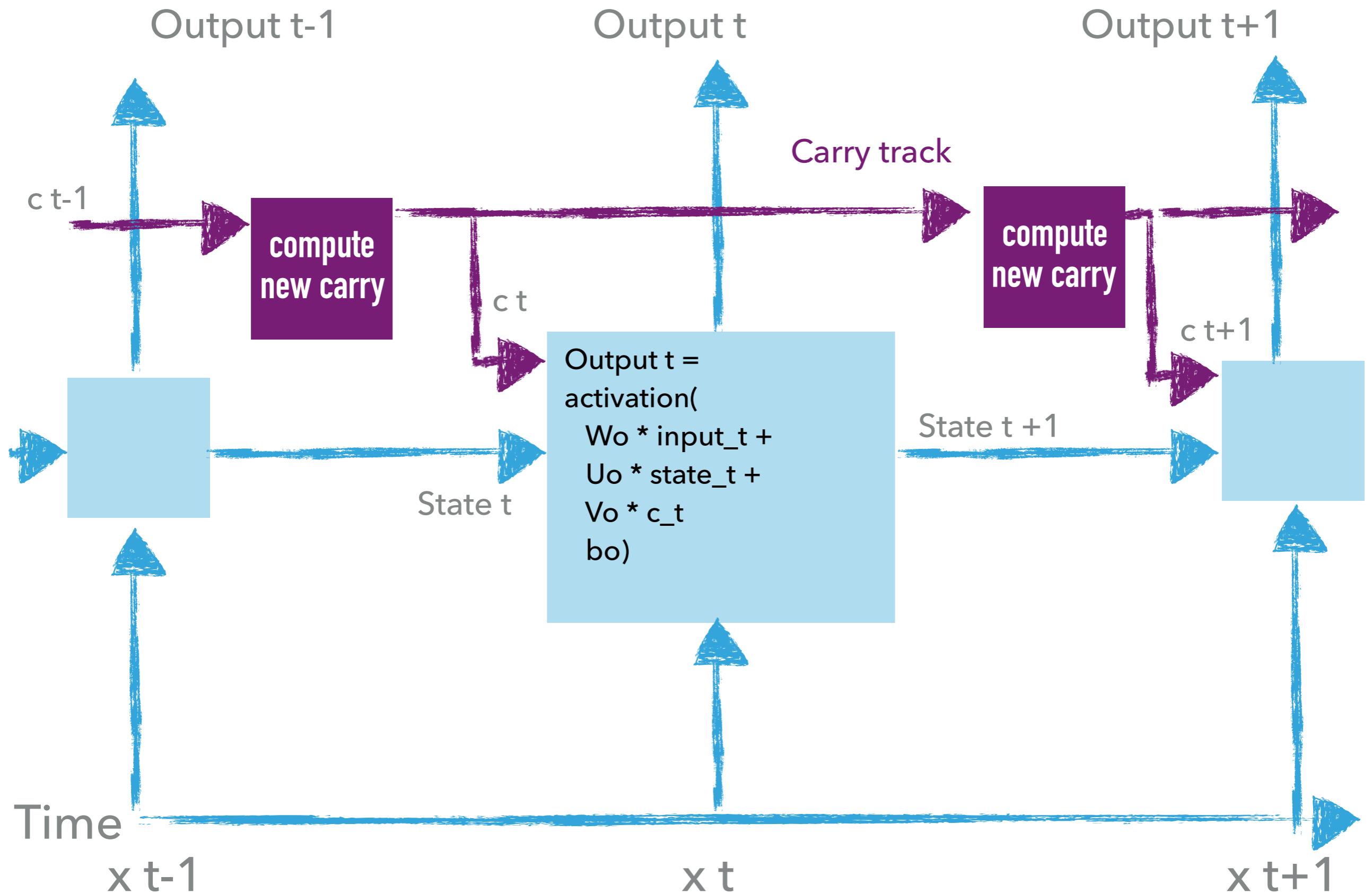
Carry track

LSTM

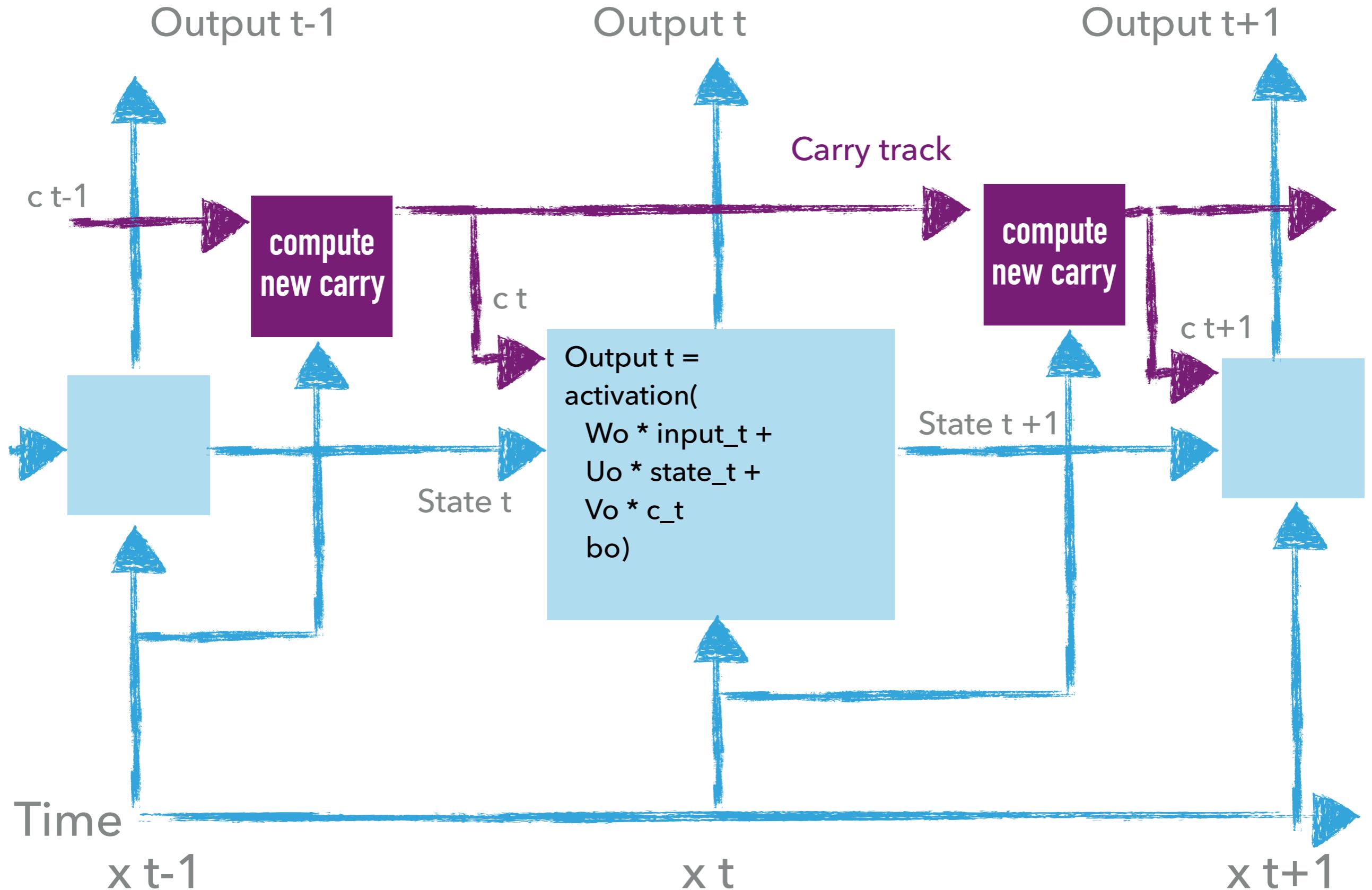


CARRY TRACK: information flows -> add new info -> forget selected info

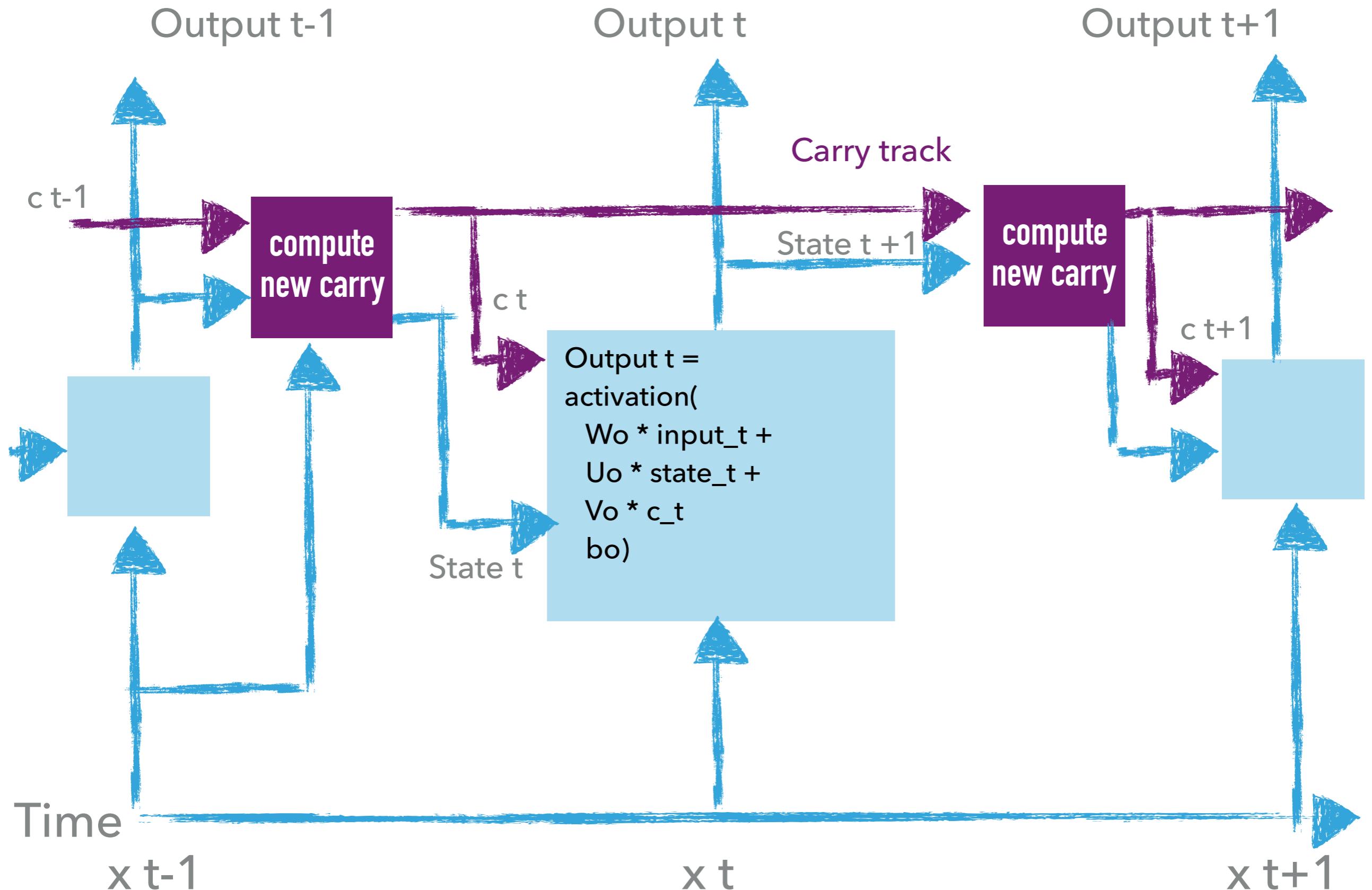
Compute new carry based on previous carry



As well as current input



And the current state



THAT'S THE

Mid-level description

A CLOSER LOOK

Pseudocode

PSEUDOCODE

```
output_t = activation(dot(state_t, Uo) +  
                      dot(input_t, Wo) +  
                      dot(c_t, Vo) + bo)
```

PSEUDOCODE

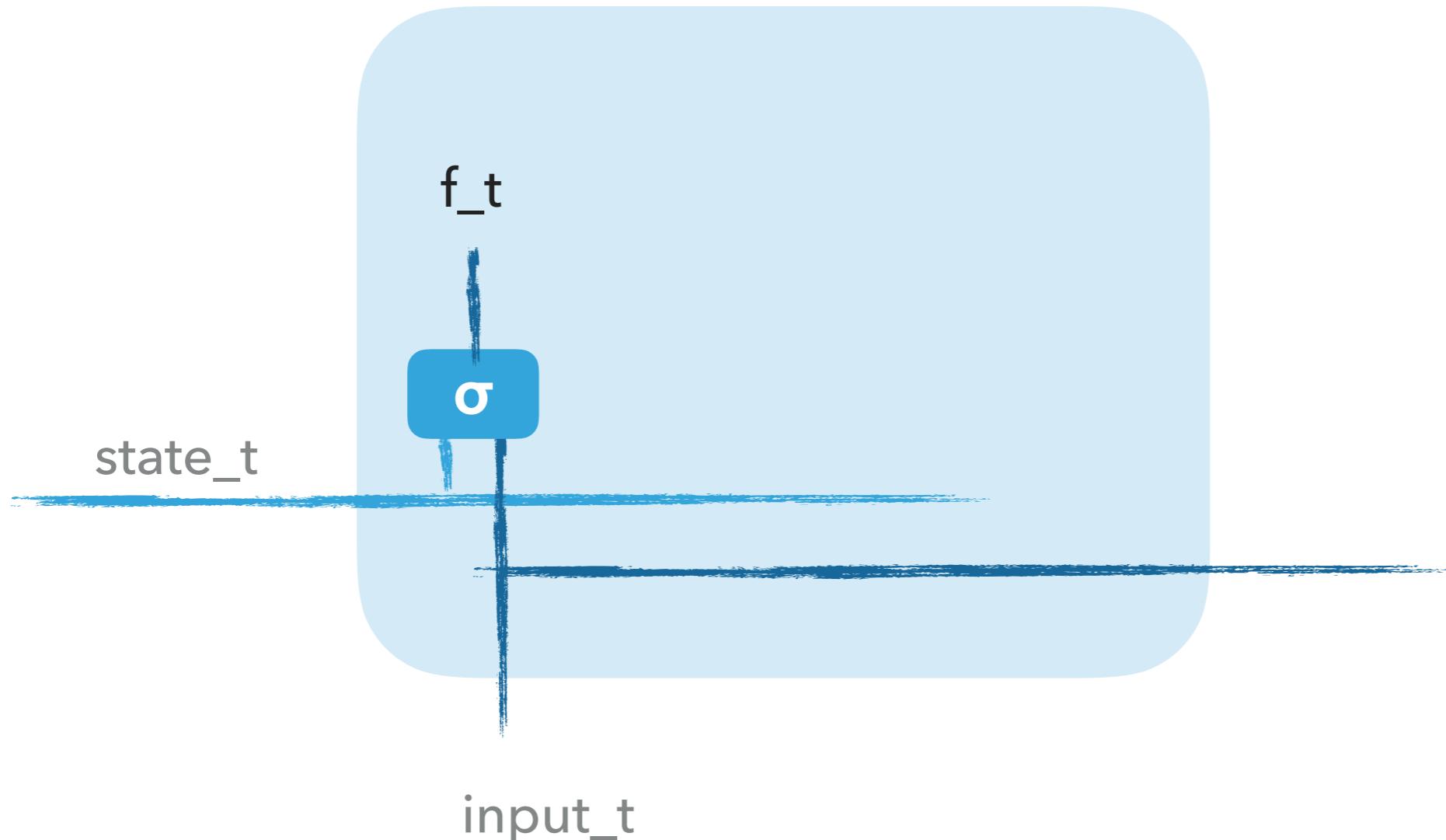
```
output_t = activation(dot(state_t, Uo) +  
                      dot(input_t, Wo) +  
                      dot(c_t, Vo) + bo)
```

```
dot([1, 2, 3, 4, 5], [10, 20, 30, 40, 50])  
= 1 * 10 + 2 * 20 + 3 * 30 + 4 * 40 + 5 * 50  
= 10 + 40 + 90 + 160 + 250  
= 590
```

PSEUDOCODE

Forget gate layer -> what information are we going to throw out

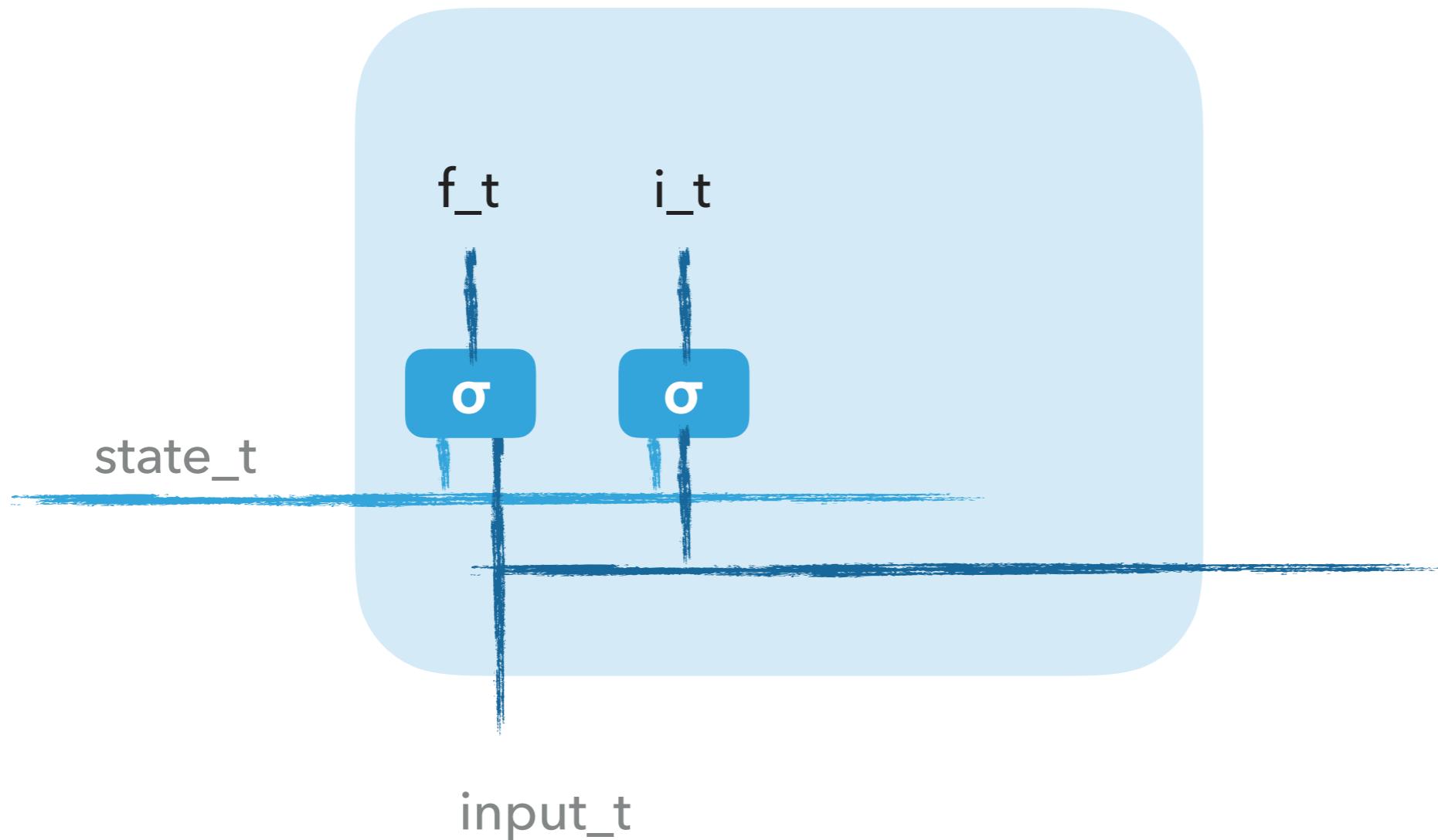
```
f_t = activation(dot(state_t, Uf) + dot(input_t, Wf) + bf)
```



PSEUDOCODE

Input gate layer -> contributes to the information are we going to keep

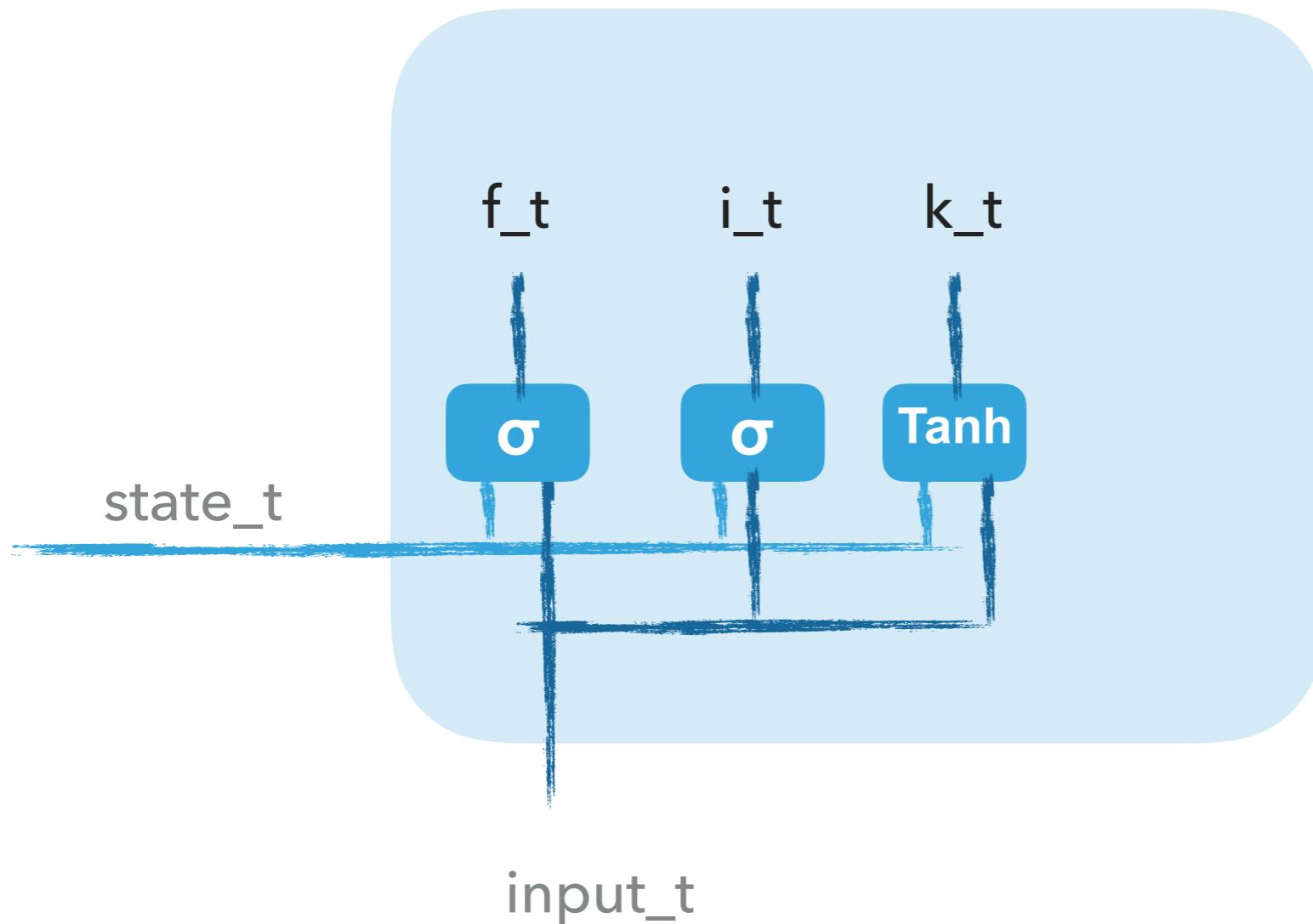
```
i_t = activation(dot(state_t, Ui) + dot(input_t, Wi) + bi)
```



PSEUDOCODE

K-gate. - Keep gate

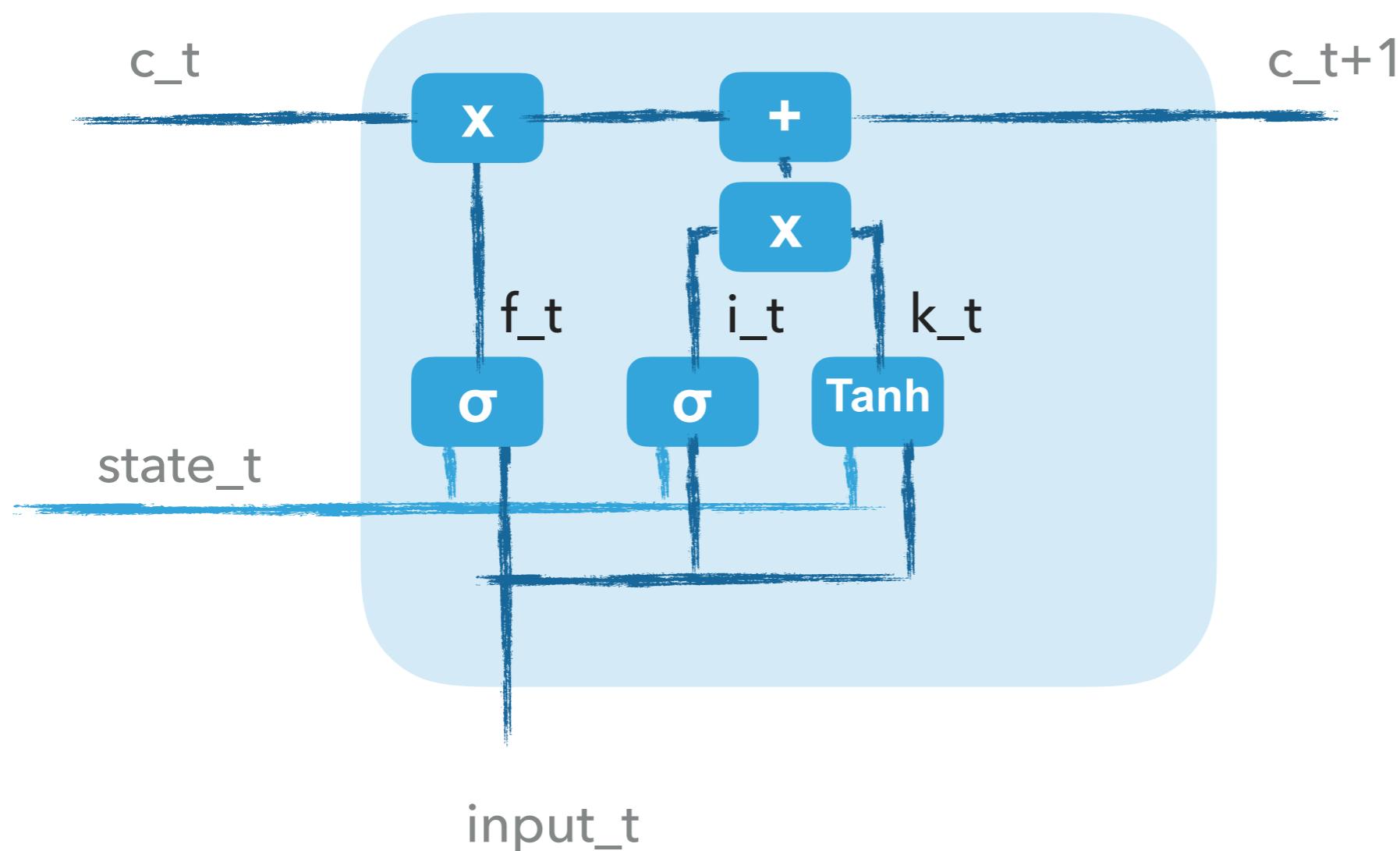
```
k_t = activation(dot(state_t, Uk) + dot(input_t, Wk) + bk)
```



PSEUDOCODE

Compute new carry

$$c_{t+1} = i_t * k_t + c_t * f_t$$



In 2009, deep multidimensional LSTM networks demonstrated the power of deep learning with many nonlinear layers, by winning three ICDAR 2009 competitions in connected handwriting recognition, without any prior knowledge about the three different languages to be learned

A Novel Connectionist System for Improved
Unconstrained Handwriting Recognition

The fire brigade has arrived.
Adenauer is in a tough spot. Waiting.

bring support and comfort to
Commonwealth countries do

(a)

Socrates was a Classical Greek philosopher credited as one of the founders of Western philosophy; he is an enigmatic figure known only through the classical accounts of his students. Plato's dialogues are the most comprehensive accounts of Socrates to survive from antiquity. Forming an accurate picture of the historical Socrates and his philosophical viewpoints is problematic at best. This issue is known as the Socratic problem. The knowledge of the man, his life, and his philosophy is based on writings by the students and contemporaries. Foremost among them is Plato; however, works by Xenophon, Aristotle, and Aristophanes also provide important insights. The difficulty of finding the real Socrates arises because these works are often philosophical or dramatic texts rather than straightforward histories. Aside from Thucydides who makes no mention of Socrates or philosophers in general, there is in fact no such thing as a straightforward history contemporary with Socrates that dealt with his own time and place.

(b)

the real Socrates arises because these works are often philosophical or dramatic texts rather than straightforward histories. Aside from Thucydides who makes no mention of Socrates or philosophers in general, there is in fact no such thing as a straightforward history contemporary with Socrates that dealt with his own time and place.

(c)

contemporaries. Foremost among the works by Xenophon, Aristotle, provide important insights. The real Socrates arises because these works are often philosophical or dramatic straightforward histories. A straightforward history contemporary with Socrates who makes no mention of

LSTM

LAB

SEQ2SEQ

CANONICAL SEQ2SEQ LEARNING PHASE

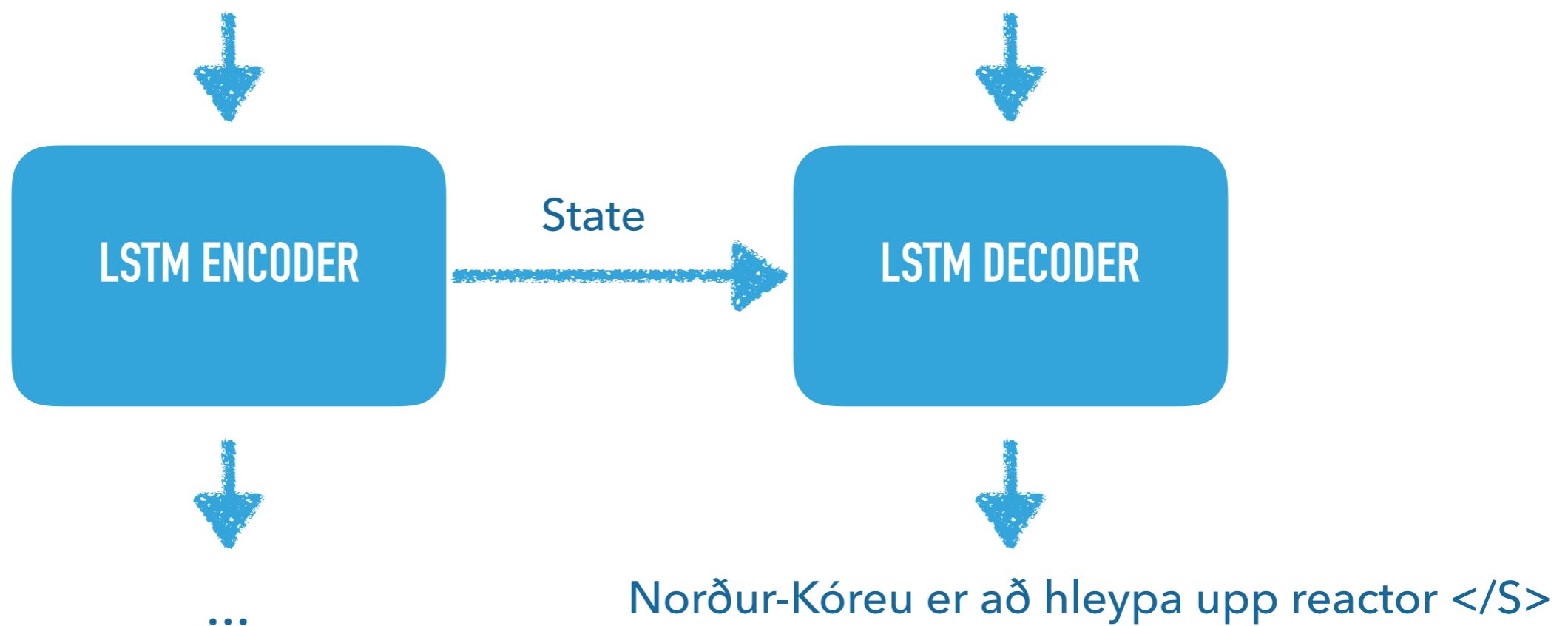
- ▶ Encoder RNN layer(s)
 - ▶ processes the input sequence and returns its own internal state.
 - ▶ We discard the outputs of the encoder RNN, only recovering the state.
- ▶ Decoder RNN
 - ▶ trained to predict the next characters of the target sequence, given previous characters of the target sequence.
 - ▶ uses as initial state the state vectors from the encoder, which is how the decoder obtains information about what it is supposed to generate.

INFERENCE PHASE

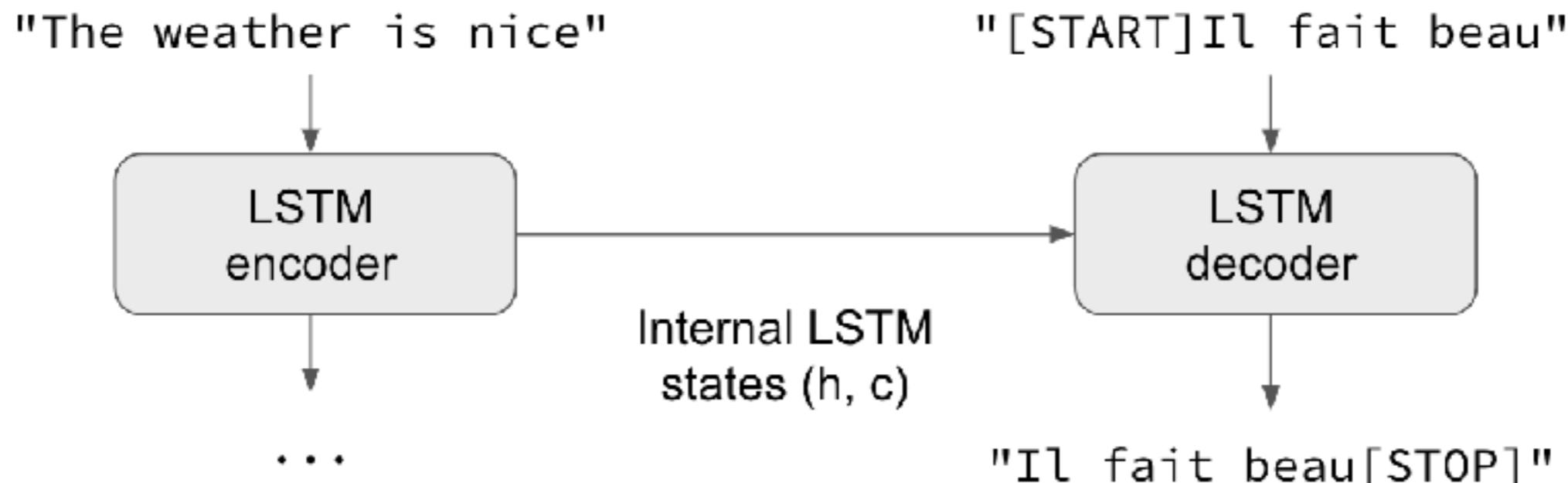
- ▶ Run source language sentence through encoder to get state
- ▶ Feed state and start symbol (for ex., $<S>$) to decoder to get first character/word.
- ▶ Repeat until you generate $</S>$ (end of sequence symbol)

INFERENCE PHASE

North Korea is firing up reactor



INFERENCE PHASE

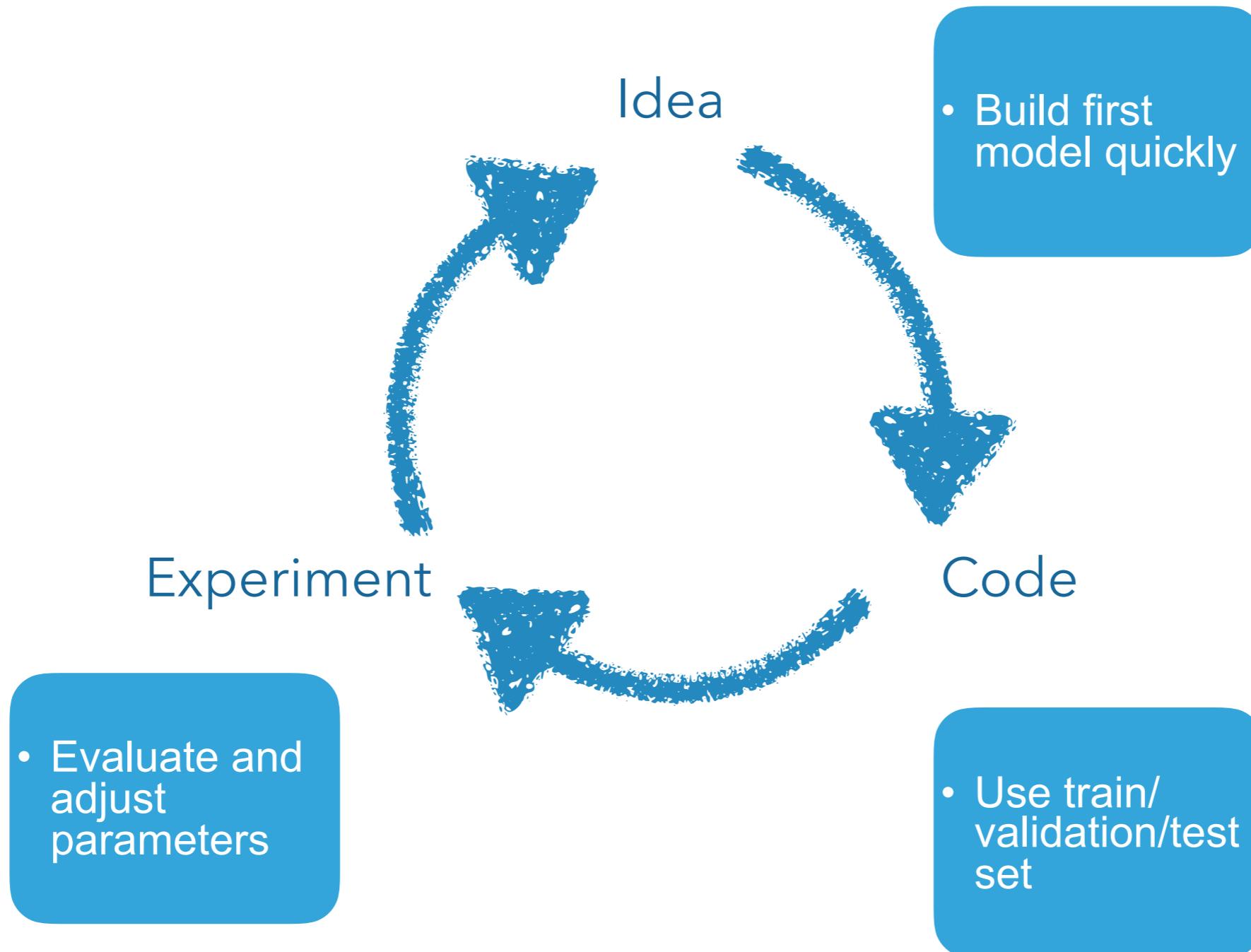


PSEUDOCODE

```
output_t = activation(dot(state_t, Uo) +
                      dot(input_t, Wo) +
                      dot(c_t, Vo) + bo)

state_t+1 = dot(output_t, activation(c_t))
```

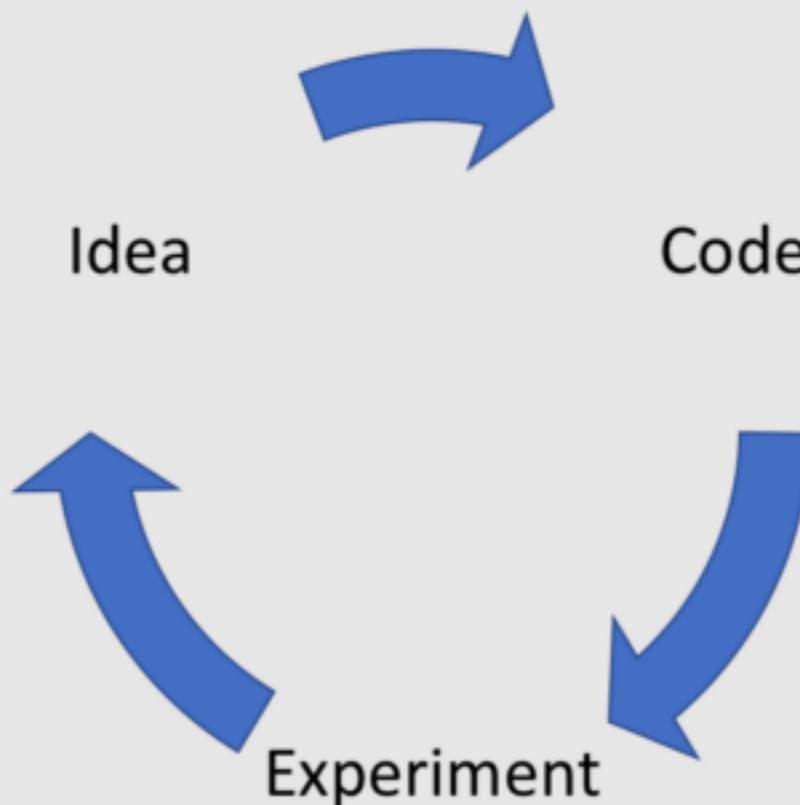
APPLIED DEEP LEARNING IS A VERY EMPIRICAL PROCESS



From Andrew Ng

Quickly iterate over your ML project following these strategic guidelines

- Build your first model quickly
- Define a single optimizing metric
- Define constraints and satisfying metrics



- Use a 98/1/1% train/dev/test set distribution
- Dev/test sets have the same data distribution
- Dev/test sets contain only examples to optimize for

- Evaluate errors to estimate bias, variance and data mismatch
- Manually analyze and label 100 misclassified examples
- Consider end-to-end, transfer or multitask learning

PSEUDOCODE

```
f_t = activation(dot(state_t, Uf) + dot(input_t, Wf) + bf)
i_t = activation(dot(state_t,Ui) + dot(input_t,Wi) + bi)
k_t = activation(dot(state_t,Uk) + dot(input_t,Wk) + bk)

c+t+1 = i_t * k_t + c_t * f_t
```

PSEUDOCODE

```
output_t = activation(dot(state_t, Uo) +  
                      dot(input_t, Wo) +  
                      dot(c_t, Vo) + bo)
```

```
dot([1, 2, 3, 4, 5], [10, 20, 30, 40, 50])  
= 1 * 10 + 2 * 20 + 3 * 30 + 4 * 40 + 5 * 50  
= 10 + 40 + 90 + 160 + 250  
= 590
```