Machine Learning

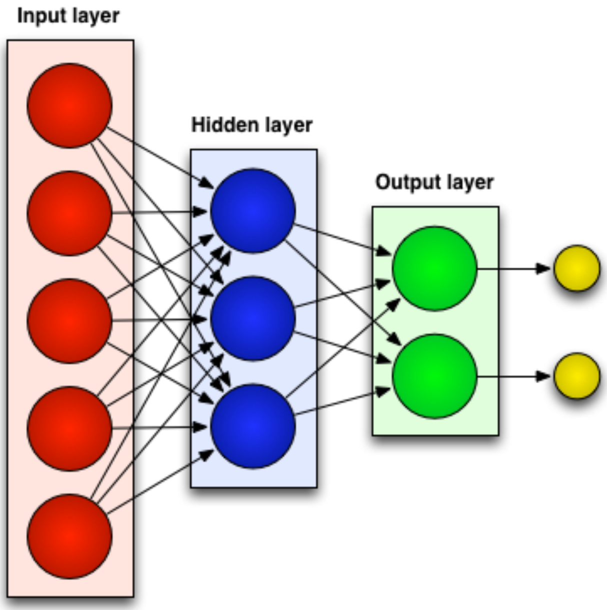
Lecture 11
Recurrent Neural Network (RNN) & Long Short-Term Memory

Chen-Kuo Chiang (江 振 國)
ckchiang@cs.ccu.edu.tw
中正大學 資訊工程學系

Feed-Forward Neural Networks

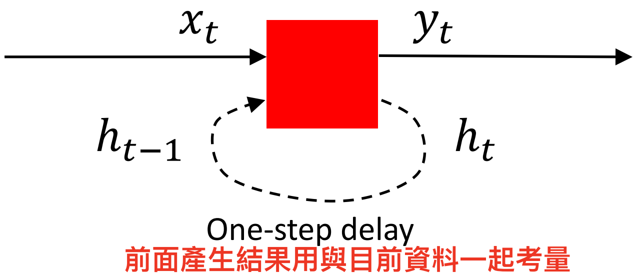
- Feedforward Neural Networks:
  - Connections between the units do not form a cycle.
  - The topological ordering is used for activation propagation, and for gradient back-propagation.

沒有cycle



Recurrent Neural Network (RNN)

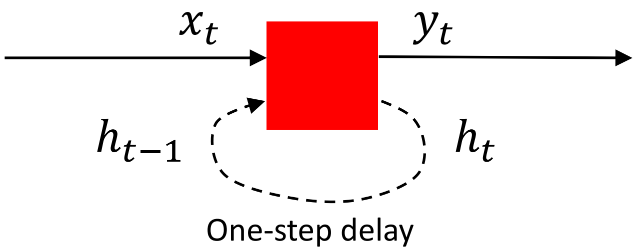
- We now will input one x_i at a time, and re-use the same edge weights.

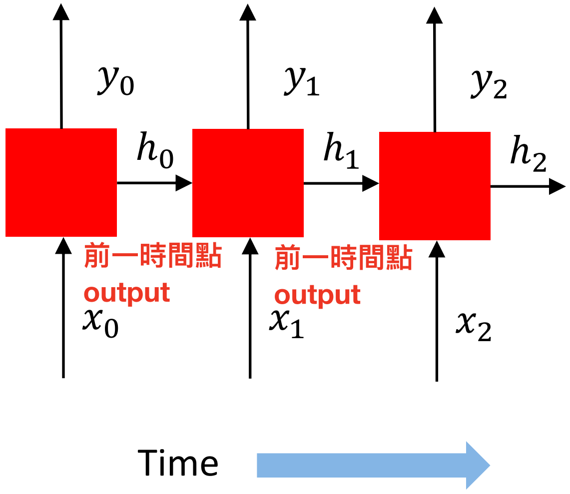


- Recurrent networks introduce cycles and a notion of time.
  - They are designed to process sequences of data $x_1, \dots, x_n$ and can produce sequences of outputs $y_1, \dots, y_m$.

Recurrent Neural Network (RNN)

- RNNs can be unrolled across multiple time steps.

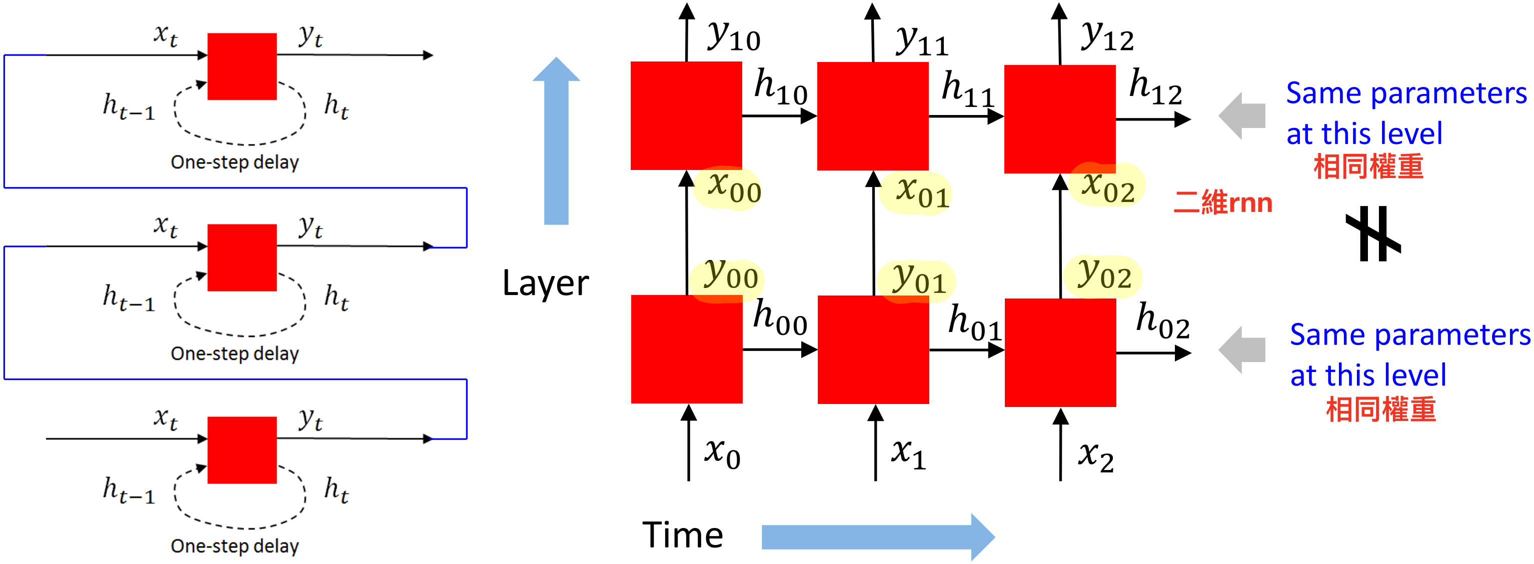




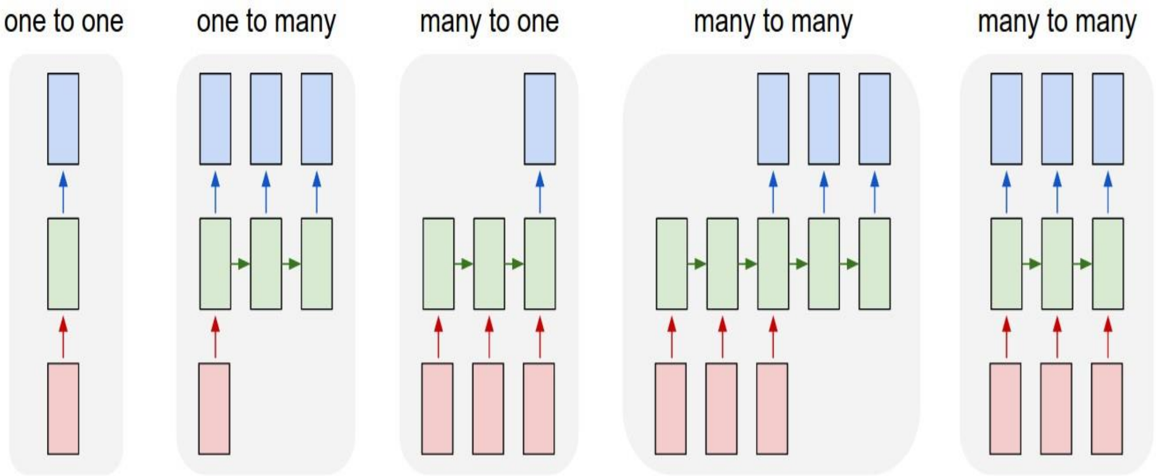
RNN Structure

- Layers can be stacked vertically (deep RNNs):

將很多堆疊起來



Flexibility of Recurrent Networks



Vanilla Neural Networks 今年預測明年

Flexibility of Recurrent Networks

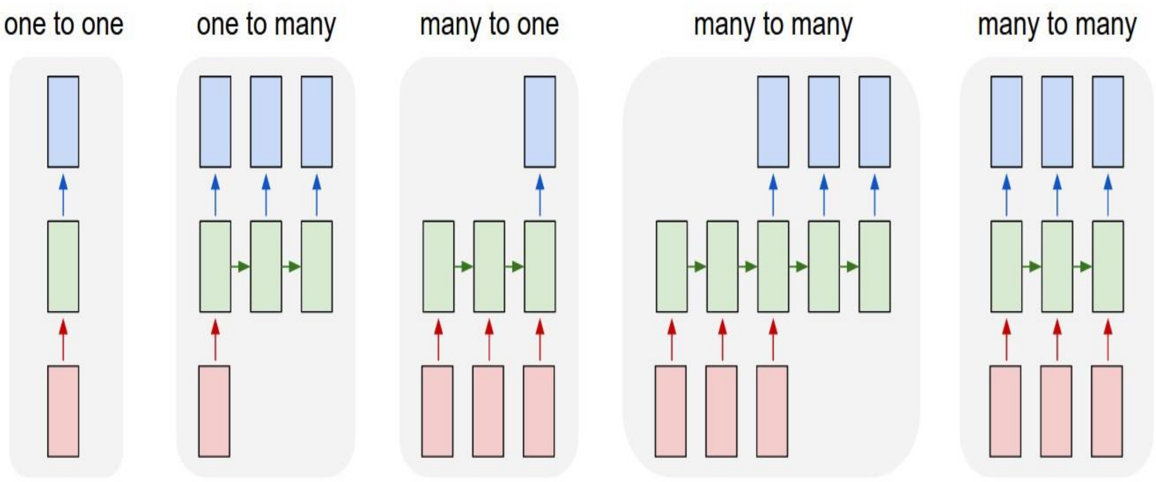
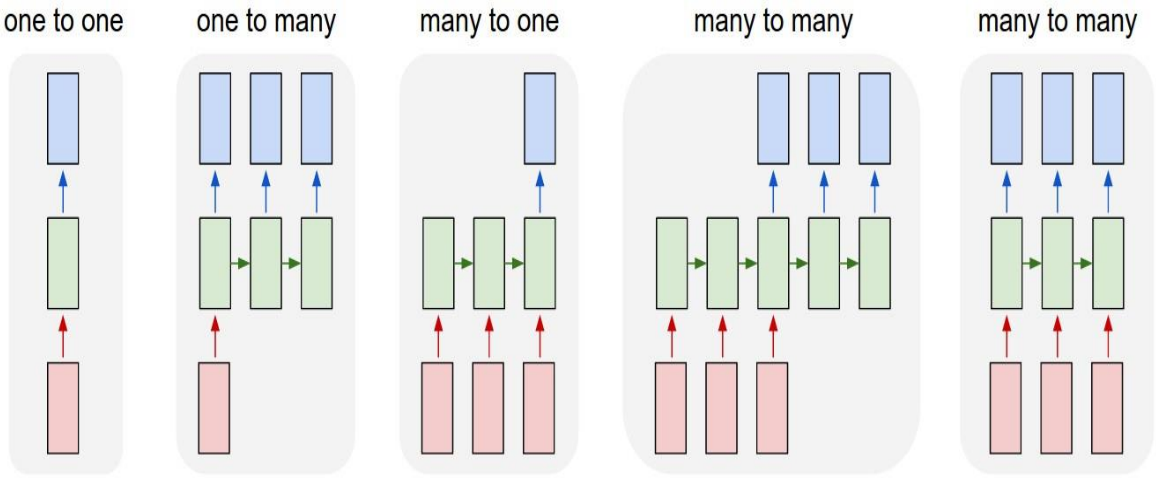


Image Captioning 圖片下標題
image -> sequence of words

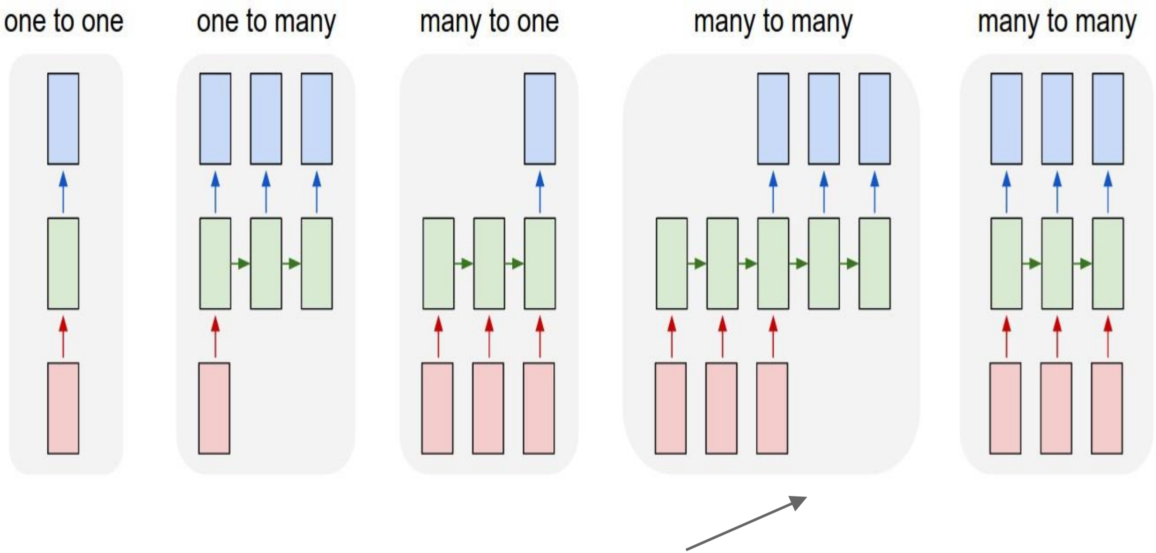
Flexibility of Recurrent Networks



Sentiment Classification 很多文字對應最後結果

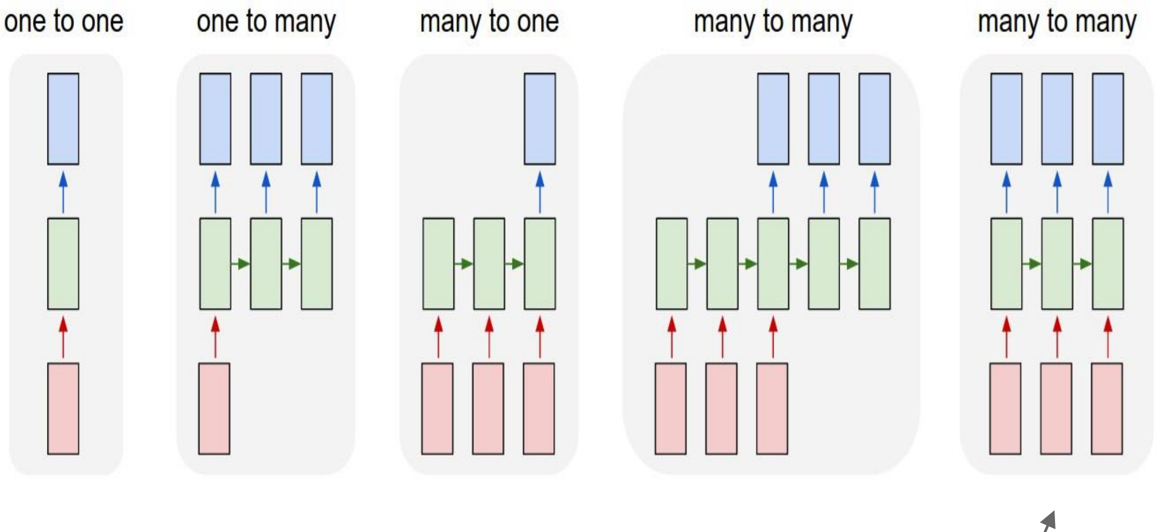
sequence of words -> sentiment

Flexibility of Recurrent Networks



e.g. **Machine Translation**
seq of words -> seq of words 文字翻譯

Flexibility of Recurrent Networks



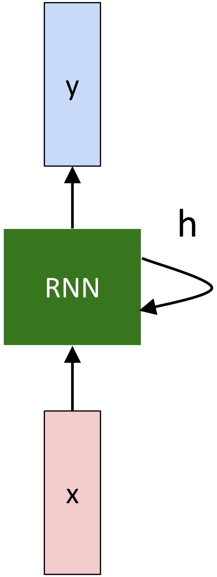
Video classification on frame level 影像內畫面分類

Property of Recurrent Neural Network

- RNNs are a neural network with memory. 有記憶功能
- Recurrent since they receive inputs, update the hidden states depending on the previous computations, and make predictions for every element of a sequence. 考慮前一時間點結果
- RNNs are very powerful for sequence tasks, such as speech recognition, since they maintain a state vector that implicitly contains information about the history of all the past elements of a sequence. 更新hidden layer，包含前一次結果。

Formulation of Recurrent Neural Network

- We can process a sequence of vectors x by applying a recurrence formula at every time step:



$$\boxed{h_t} = \boxed{f_W}(\boxed{h_{t-1}}, \boxed{x_t})$$

new state

old state

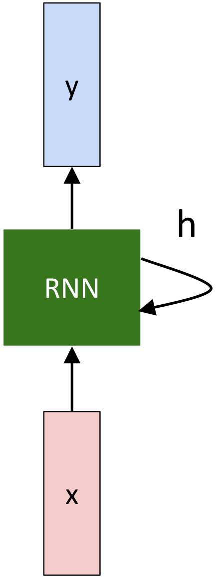
input vector at some time step

some function with parameters W

Notice: the same function and the same set of parameters are used at every time step.

Formulation of Recurrent Neural Network

- The state consists of a single “hidden” vector h:



$$y_t = W_{hy} h_t$$

第t時間點　　　乘上對應權重

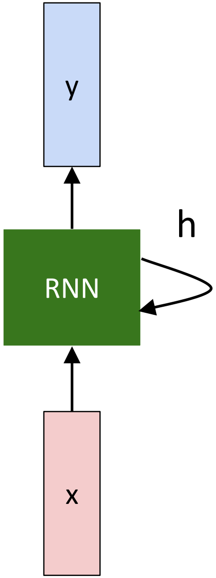
$$h_t = \tanh(W_{hh} h_{t-1} + W_{xh} x_t)$$

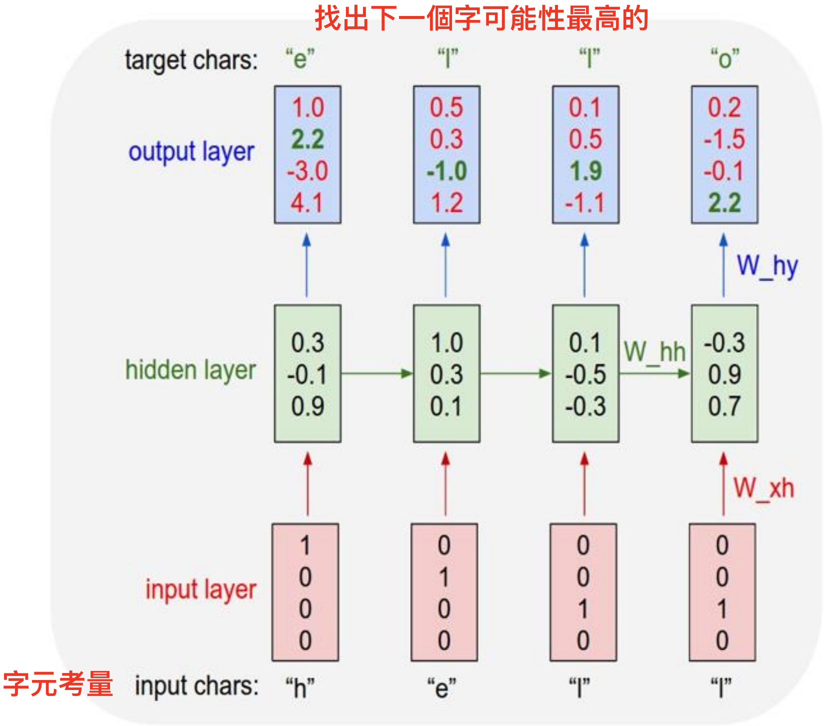
$$h_t = f_W(h_{t-1}, x_t)$$

在加前一個

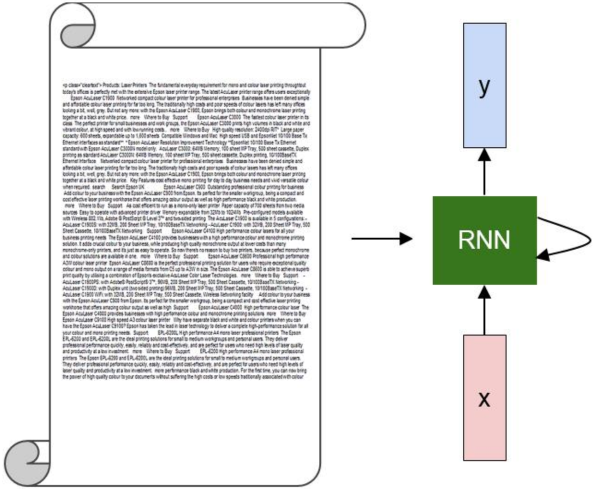
Applications : Character-level Language Model

- Use Vocabulary [h,e,l,o] to train sequence “hello”





Applications : Text Generation Model



at first:

```
tyntd-iafhatawiaoihrdemot  lytdws  e ,tfti, astai f ogoh eoase rrranbyne 'nhthnee e
plia tklrgd t o idoe ns,smtt   h ne etie h,hregtrs nigtike,aoaenns lng
```

train more

```
"Tmont thithey" fomesscerliund
Keushey. Thom here
sheulke, anmerenith ol sivh I lalterthend Bleipile shuwy fil on aseterlome
coaniogennc Phe lism thond hon at. MeiDimorotion in ther thize."
```

train more

```
Aftair fall unsuch that the hall for Prince Velzonski's that me of
her hearly, and behs to so arwage fiving were to it beloge, pavu say falling misfort
how, and Gogition is so overelical and ofter.
```

train more

```
"Why do what that day," replied Natasha, and wishing to himself the fact the
princess, Princess Mary was easier, fed in had oftened him.
Pierre aking his soul came to the packs and drove up his father-in-law women.
```

Applications : Image Captioning 輸入照片給說明

"man in black shirt is playing guitar."

"construction worker in orange safety vest is working on road."

"two young girls are playing with lego toy."

"boy is doing backflip on wakeboard."

"a young boy is holding a baseball bat."

"a cat is sitting on a couch with a remote control."

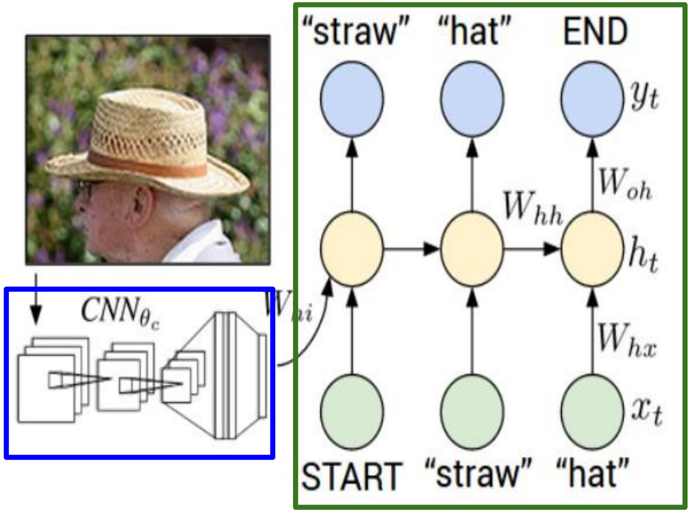
"a woman holding a teddy bear in front of a mirror."

"a horse is standing in the middle of a road."

Applications : Image Captioning

- Explain Images with Multimodal Recurrent Neural Networks, Mao et al.
- Deep Visual-Semantic Alignments for Generating Image Descriptions, Karpathy and Fei-Fei Li.
- Show and Tell: A Neural Image Caption Generator, Vinyals et al.
- Long-term Recurrent Convolutional Networks for Visual Recognition and Description, Donahue et al.
- Learning a Recurrent Visual Representation for Image Caption Generation, Chen and Zitnick.

Recurrent Neural Network

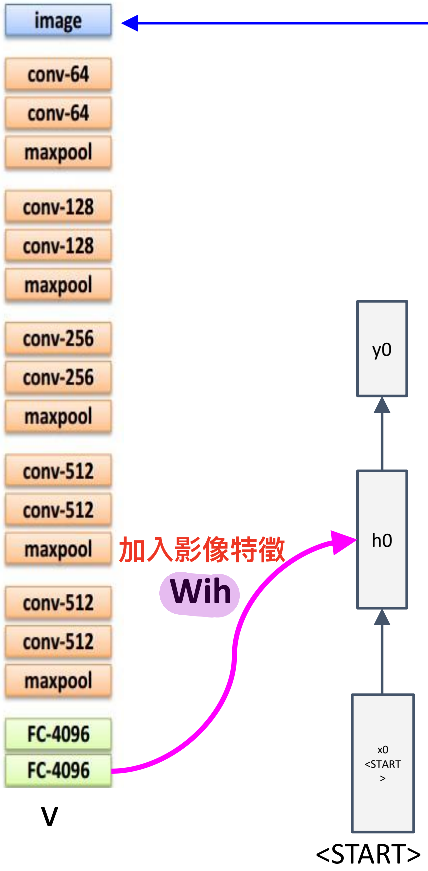


Convolutional Neural Network

VGG分類

Training Image

取出影像特徵



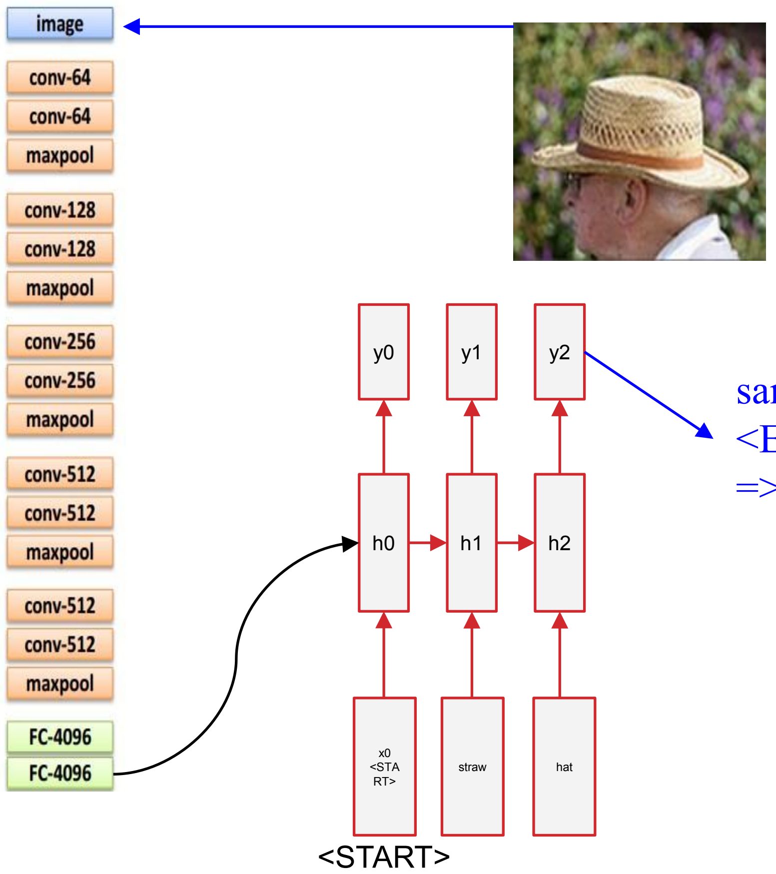
Training Image

before: 文字

$$h = \tanh(W_{xh} * x + W_{hh} * h)$$

now:

$$h = \tanh(W_{xh} * x + W_{hh} * h + W_{ih} * v)$$



Training Image

sample
<END> token
=> finish.

Problems with RNN Model

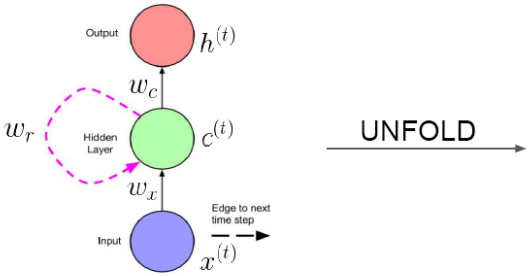
有記憶功能模型

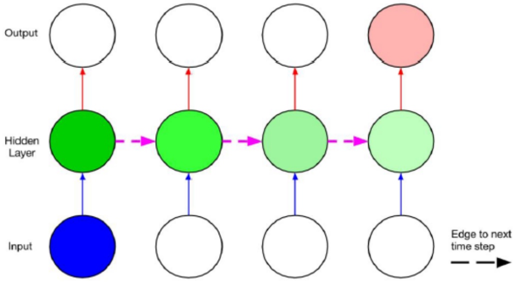
太老的資料記不住

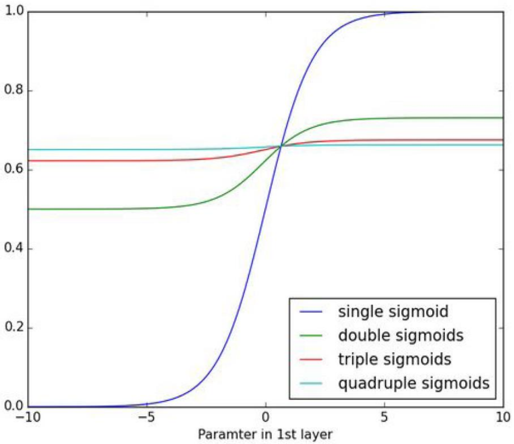
- When dealing with a time series, it tends to **forget old information**.
  - In practice, the range of contextual information that standard RNNs can access are limited to approximately 10 time steps between the relevant input and target events. 包含前面很多記憶 大概記憶前10筆資料

- **Vanishing gradient problem**. Gradient會越來越小
  - The influence of a given input on the hidden layer, and therefore on the network output, either decays or grows exponentially as it propagates through an RNN.

Problems with RNN Model







$$h^{(t)} = \sigma(w_c \cdot c^{(t)})$$ 第t個時間點

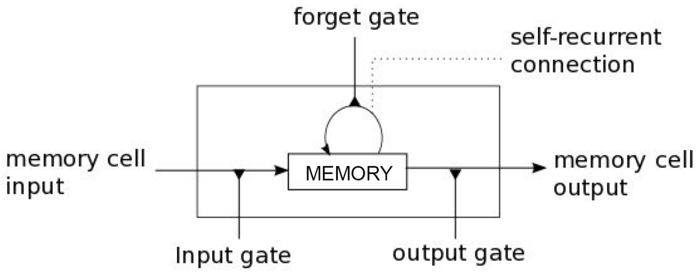
$$c^{(t)} = \sigma(w_r \cdot c^{(t-1)} + w_x \cdot x^{(t)})$$

$$h^{(3)} = \sigma(w_c \cdot c^{(3)})$$ 之前資料

$$\begin{aligned}
&= \sigma(w_c \cdot \sigma(w_x \cdot x^{(3)} + w_r \cdot c^{(2)})) \\
&= \sigma(w_c \cdot \sigma(w_x \cdot x^{(3)} + w_r \cdot \sigma(w_x \cdot x^{(2)} + w_r \cdot c^{(1)})))))) \\
&= \sigma(w_c \cdot \sigma(w_x \cdot x^{(3)} + w_r \cdot \sigma(w_x \cdot x^{(2)} + w_r \cdot \sigma(w_x \cdot x^{(1)} + w_r \cdot c^{(0)}))))))
\end{aligned}$$

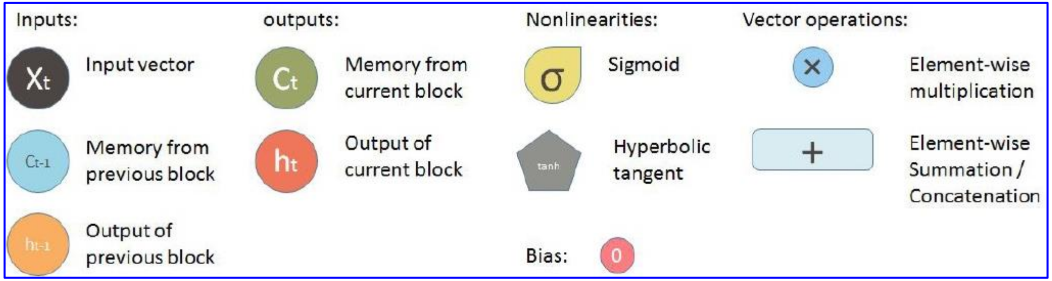
Solution : Long Short-Term Memory

- When there is a distant relationship of unknown length, we wish to have a “memory” to it.
- **Idea**: Design a memory cell which can maintain its state over time, consisting of an explicit memory (i.e the cell state vector) and gating units which regulate the information flow into and out of the memory.

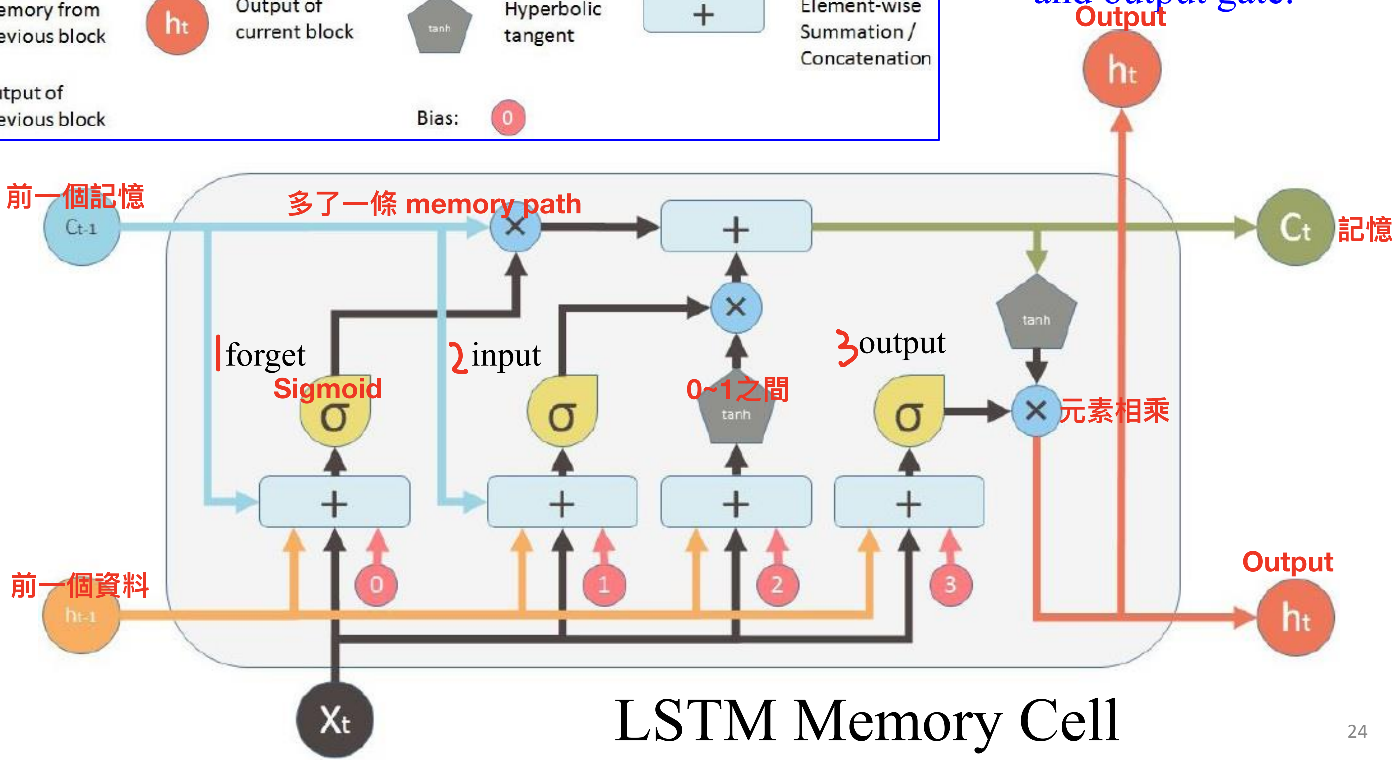


Memory與閘門
讓記憶可以更好

LSTM Memory Cell



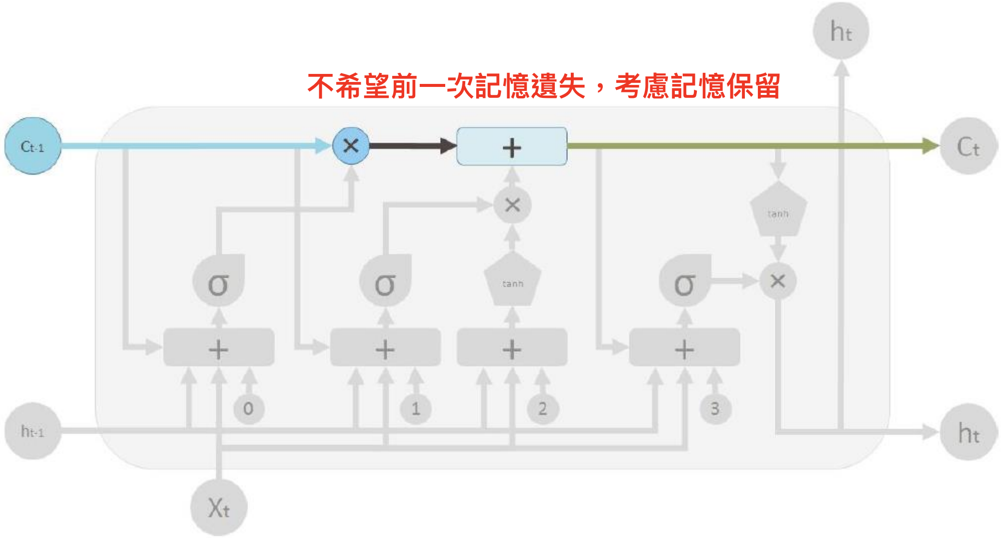
- **Three components:** forget gate, input gate and output gate.



LSTM Memory Cell

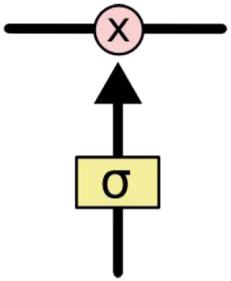
Memory - Cell State Vector

- New concept to RNN model, representing the memory of the LSTM.
- Undergoes changes via forgetting of old memory (forget gate) and addition of new memory (input gate) Cell



Gates

- Gate: sigmoid neural net layer followed by pointwise multiplication operator.
  - Recall sigmoid outputs values from 0 to 1.
  - Values are discarded if 0 is used for pointwise multiplication .
- Gates control the flow of information to/from the memory
- Gates are controlled by a concatenation of the output from the previous time step and the current input and optionally the cell state vector.

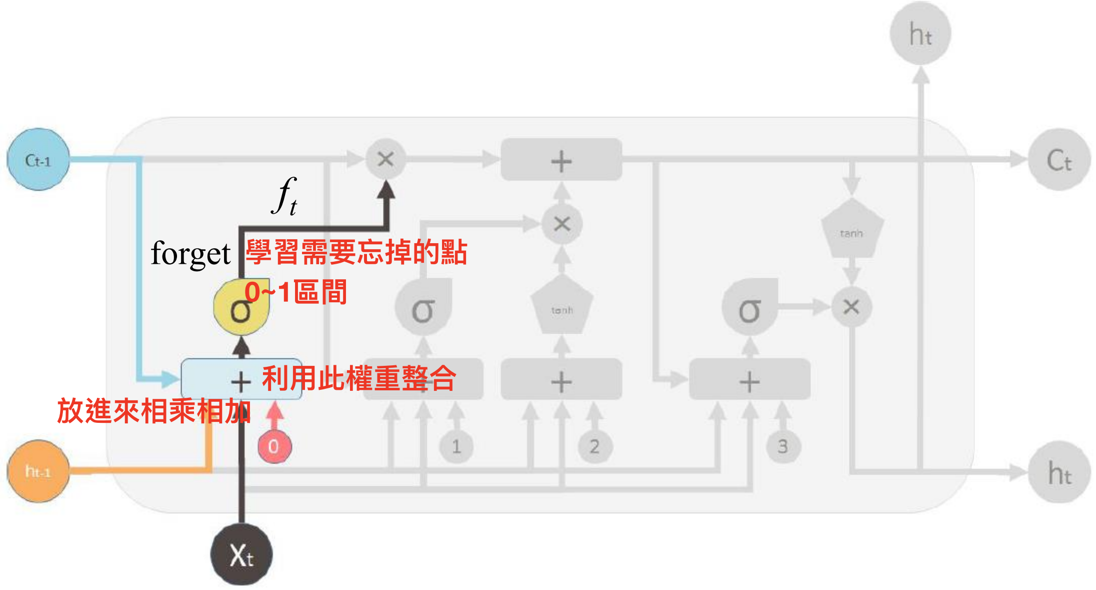


當input vector，每個vector可以乘上一個值。
成為資料輸出的閘門。

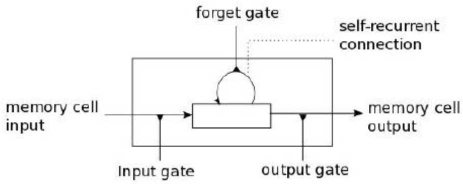
Gate (sigmoid layer followed by pointwise multiplication)

小考 Forget Gate

- Controls what information to throw away from memory.

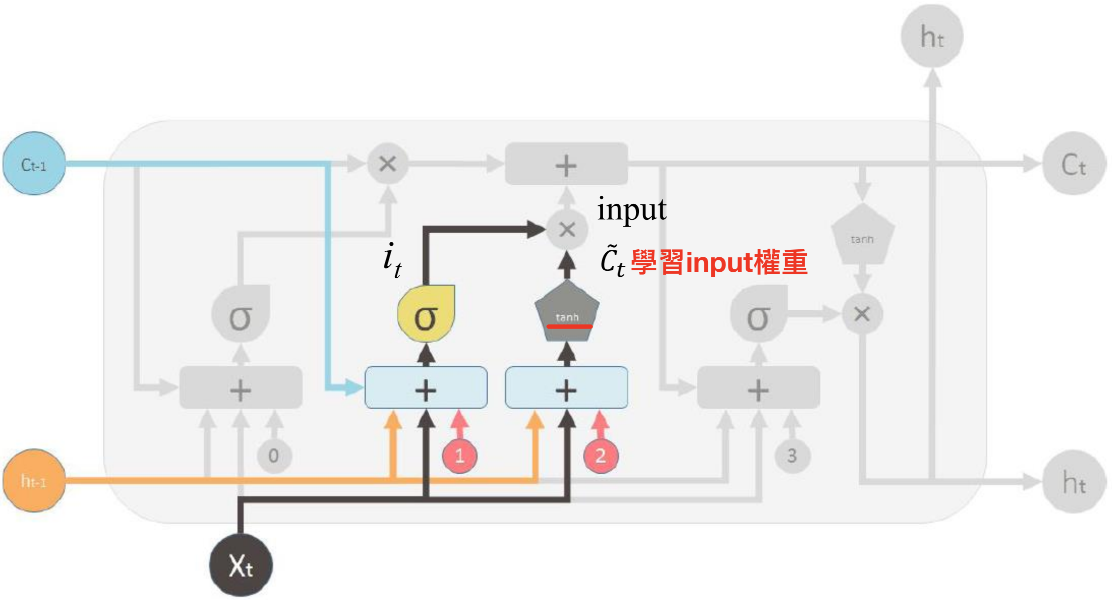


$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \ + \ b_f\right)$$



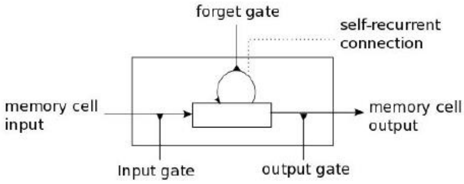
Input Gate

- Controls what new information is added to cell state from current input.



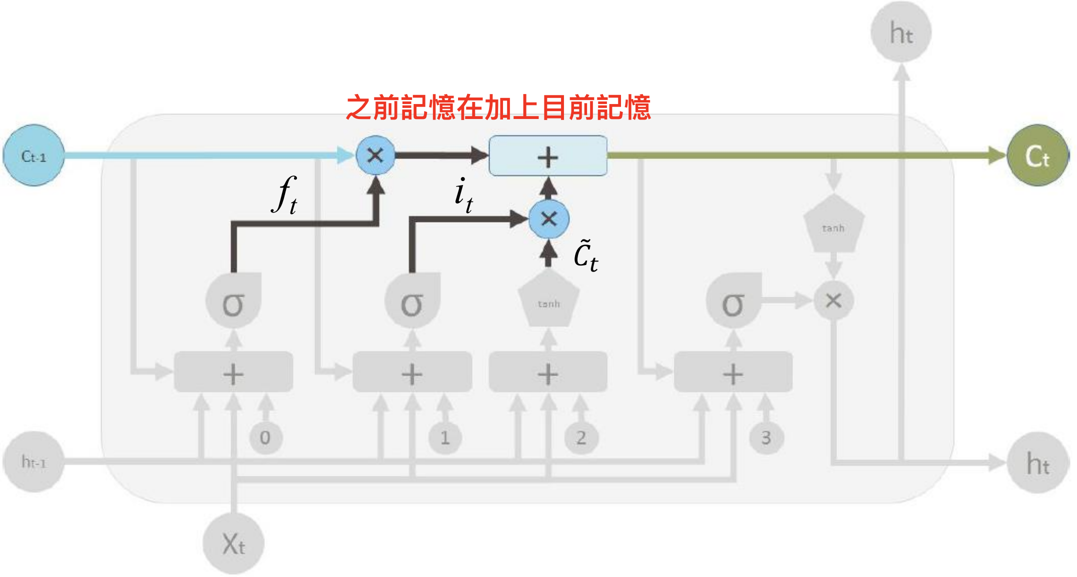
$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \ + \ b_i\right)$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] \ + \ b_C)$$



Memory Update

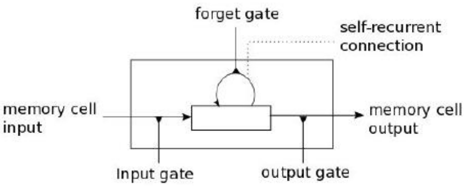
- The cell state vector aggregates the two components (old memory via the forget gate and new memory via the input gate).



忘掉權重

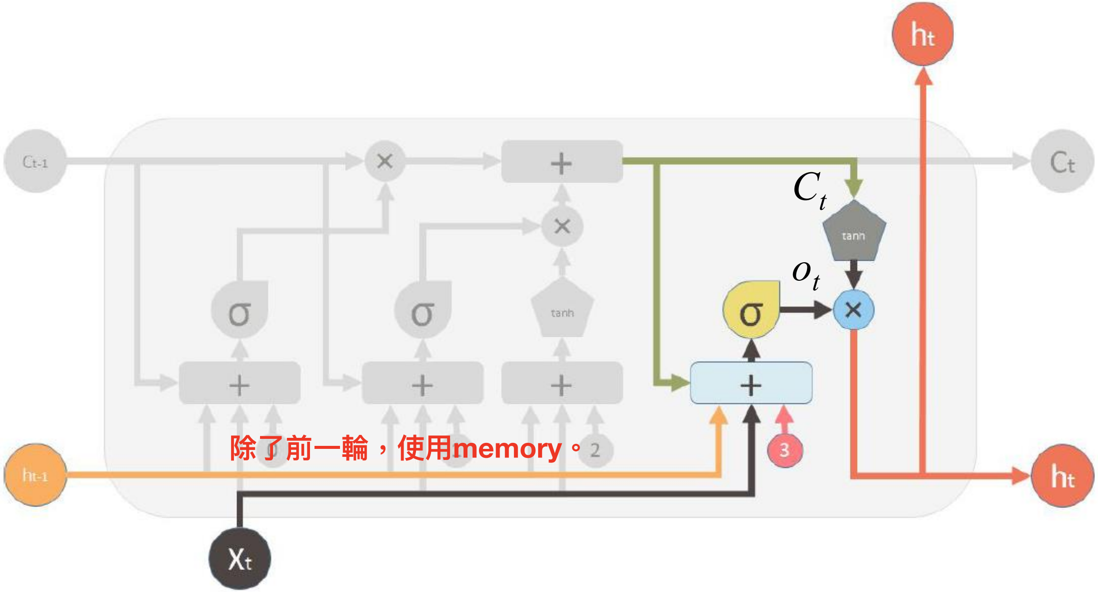
$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

此輪資料與需要忘掉權重



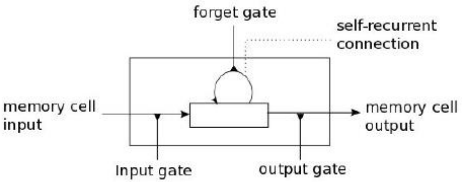
Output Gate

- Conditionally decides what to output from the memory.

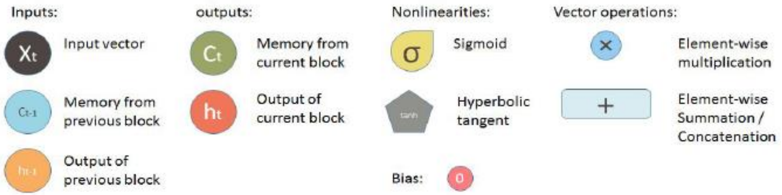


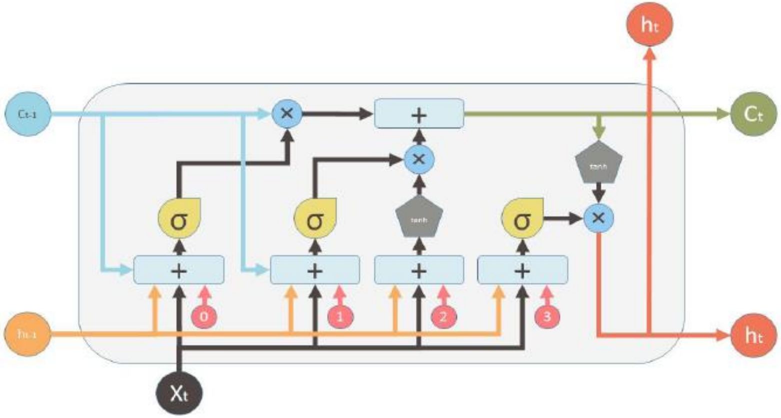
$$o_t = \sigma\left(W_o\left[h_{t-1}, x_t\right] + b_o\right)$$

$$h_t = o_t * \tanh\left(C_t\right)$$



LSTM Memory Cell Summary





需學習的權重

$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \ + \ b_f\right)$$

$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \ + \ b_i\right)$$

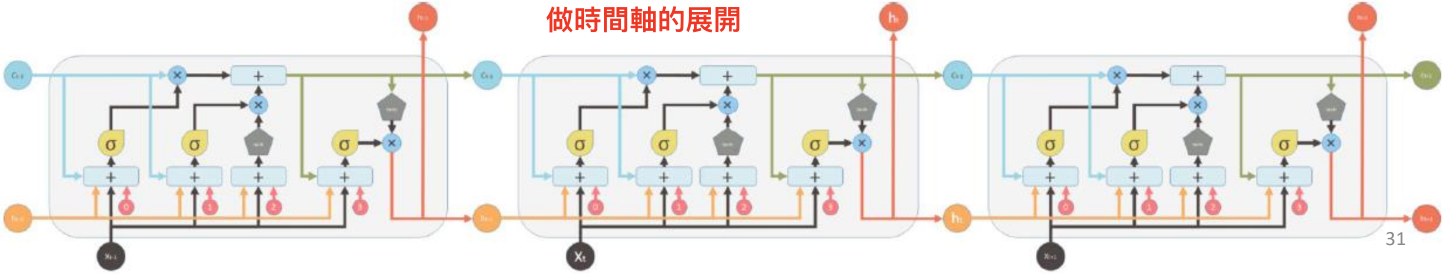
$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] \ + \ b_C)$$

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

$$o_t = \sigma\left(W_o \ [h_{t-1}, x_t] \ + \ b_o\right)$$

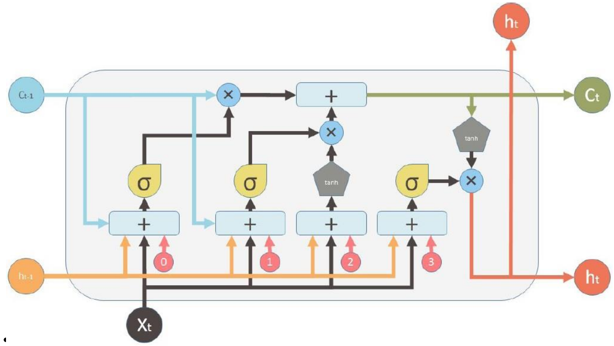
$$h_t = o_t * \tanh\left(C_t\right)$$

做時間軸的展開



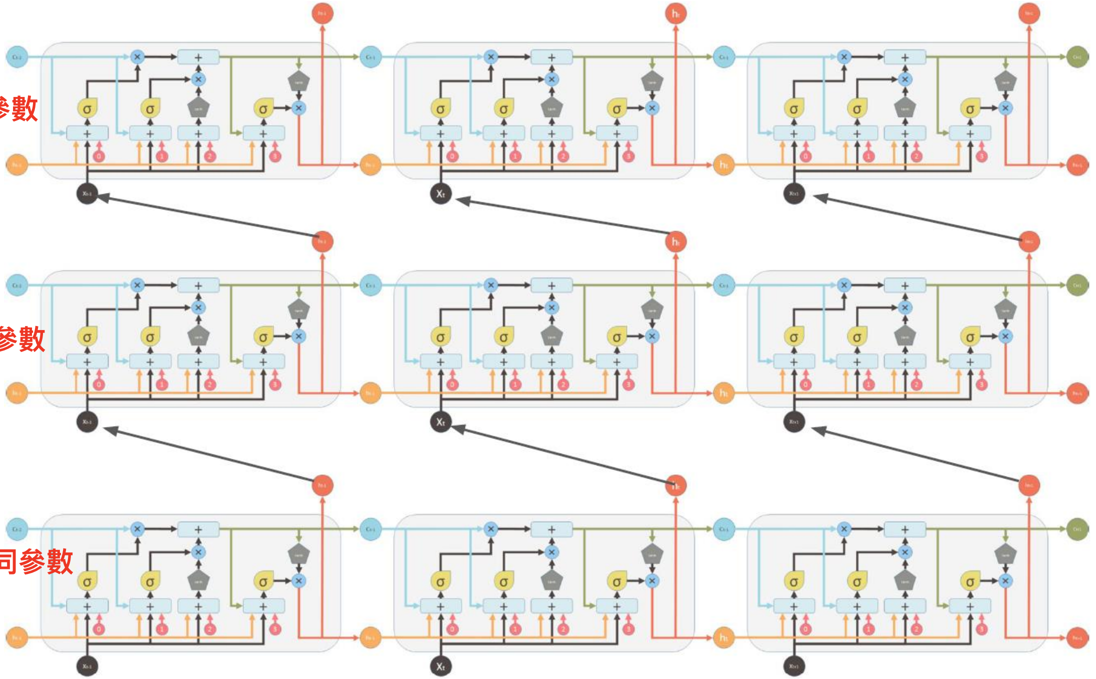
LSTM Training

- Backpropagation Through Time (BPTT) is most common.決定backprop
- What weights are learned?學習
  - Gates (input/output/forget)
  - Input tanh layer
- Outputs depend on the task: 解決問題
  - Single output prediction for the whole sequence.
  - One output at each time step (sequence labeling).

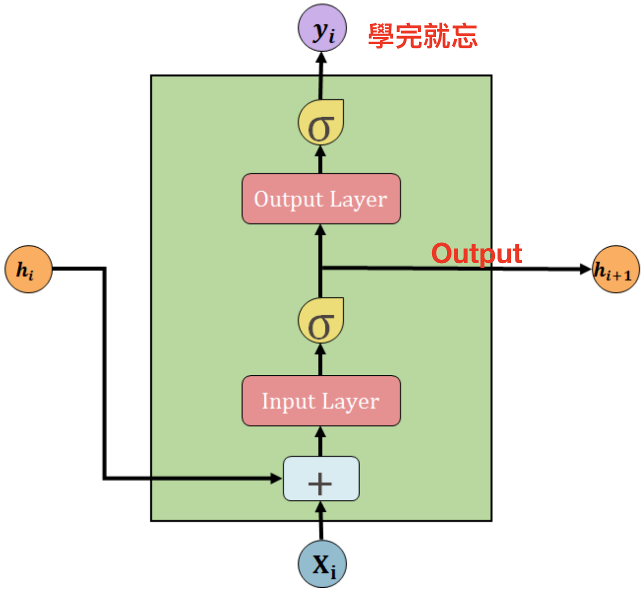


Deep LSTMs 垂直堆疊

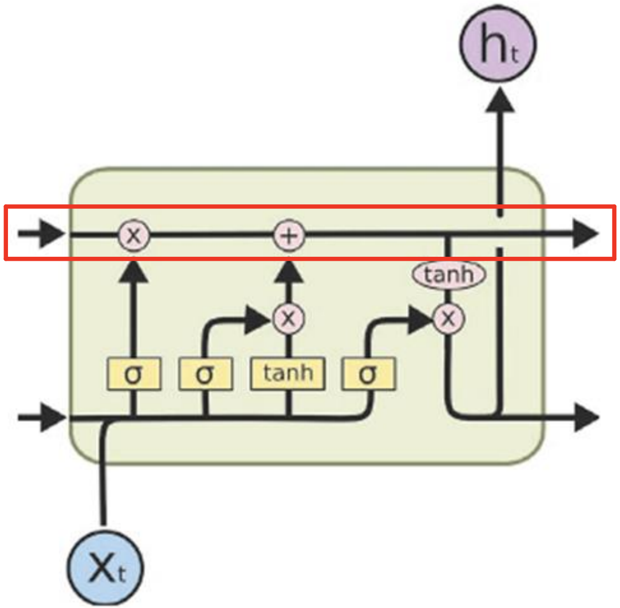
- Deep LSTMs can be created by stacking 相同參數 multiple LSTM layers vertically, with the output sequence of one layer forming the input sequence of the next. 相同參數
- Increases the number of parameters - but given sufficient data, performs significantly 相同參數 better than single-layer LSTMs.



RNN vs LSTM



(a) RNN



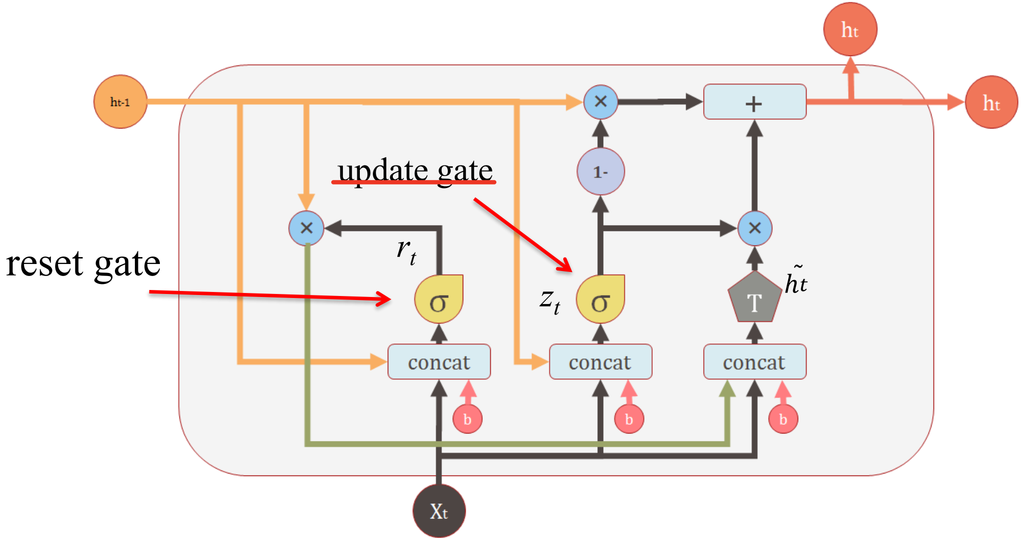
(b) LSTM

GRU – Gated Recurrent Unit 變形

- It also merges the $\boxed{\text{cell state}}$ and $\boxed{\text{hidden state}}$.
- It combines the $\boxed{\text{forget and input}}$ into a single update gate.

結合

- Simpler and more compressed than LSTM.



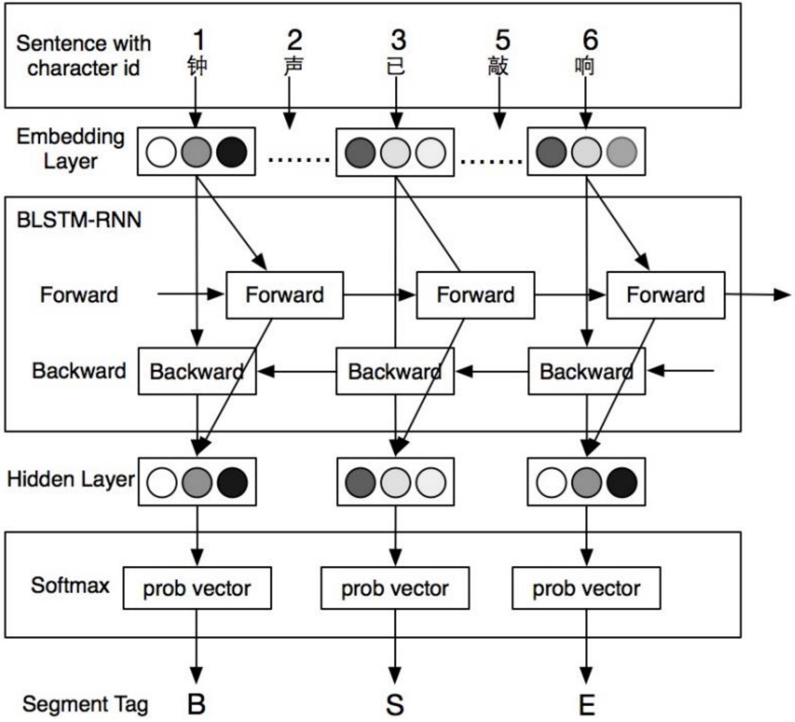
$$z_t = \sigma\left(W_z \cdot [h_{t-1}, x_t]\right)$$

$$r_t = \sigma\left(W_r \cdot [h_{t-1}, x_t]\right)$$

$$\tilde{h}_t = \tanh\left(W \cdot [r_t * h_{t-1}, x_t]\right)$$

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

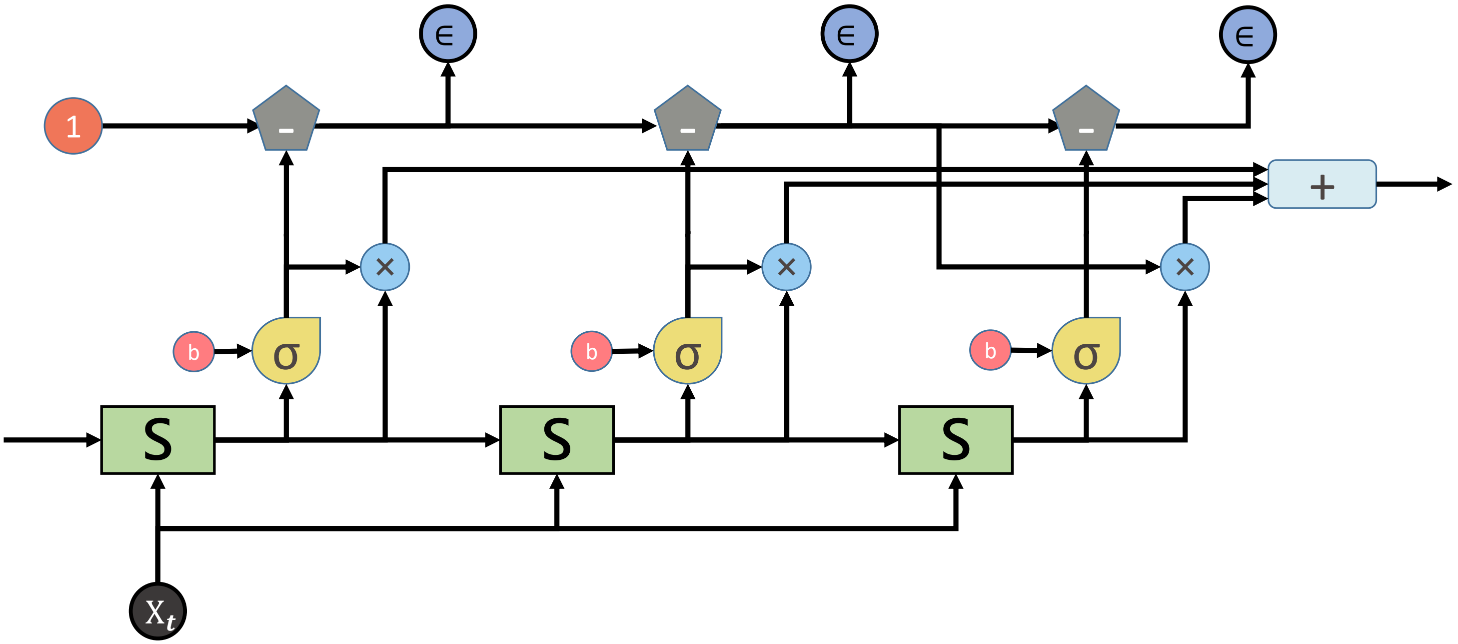
Bidirectional LSTM (Bi-LSTM)



推測前後出現字

Google DeepMind, A Graves – Bidirectional LSTM Networks for Improved Phoneme Classification and Recognition, 2005.

Adaptive computation Time RNN (ACT RNN)



Google DeepMind, A Graves – Adaptive Computation Time for Recurrent Neural Networks, 2016.

Reference

- Based on notes from Andrej Karpathy, Fei-Fei Li, Justin Johnson
- Slides of 'Recurrent Neural Network Introduction', Yun-Zhing Lu
- Slides of 'Long Short-Term Memory', Akshay Sood