AI.FINTECH
AI 金融科技中心

RNN: LSTM, Bi-LSTM, GRU

國立高雄科技大學金融資訊系教授兼AI金融科技中心主任
林萍珍

Hongyi Zhu and Hsinchun Chen
Artificial Intelligence Lab, University of Arizona
Updated April 2020

致謝

- **部分圖片和內容來自以下人士和機構:**

Andrew Ng, Stanford University, Chris Olah, Google Brain, Fei-Fei Li, Stanford University, Geoffrey Hinton, Google & University of Toronto, Hung-yi Lee, National Taiwan University, Ian Goodfellow, Google Brain, Yann LeCun, New York University, Yoshua Bengio, Universite de Montreal

Hongyi Zhu and Hsinchun Chen, Artificial Intelligence Lab, University of Arizona

- Colah's Blog, "Understanding LSTM Networks," 2015
- I. Witten, et al., "Data Mining," 2017

大綱

- 循環神經網路(Recurrent Neural Network, RNN)
  - RNN的限制: 梯度爆炸與梯度消失
- 記憶型網路
  - 長短時記憶網路 (LSTM)
  - 門控循環單元 (GRU)
- LSTM股價預測實例程式碼解說
  - 資料前處理: 正規化、訓練、驗證、測試資料分割
  - 將X、Y的以rolling window的形式，建構三維的numpy資料
  - 迴圈建構隱藏層與神經元
  - 預測結果與實際結果比對引用metrics套件計算準確度
- 建議提升準確度的策略
  - 參數(hidden layer, neurons, windows, epochs, batch size)校調
  - Input layer variable search
  - Input layer and output layer 日期移動參數校調

統計與AI有什麼不同?

- **以預測股票價格為例**

- 線性的統計方法, 屬模型導向(Model driven)

預測困難在於數學模型推導時
(1)許多假設(hypothesis)條件，而且(2)資料是不恒定(nonstationarity)

- 機器學習方法(AI的子集), 屬資料導向(Data driven)

以非線性模形大量資料的樣版(Pattern)預測股價走勢

- **機器學習的演算法很多種，相關究研很多，每一種都不斷在提出更新的演算法。**
- **多數文獻研究結果顯示非線性AI模型預測效能優於線性的統計模型。**
- **輸人變數多使以技術指標與經濟指數為主。**

循環神經網路

- **人類大腦處理信息流，大多數數據都是按順序獲取、處理和生成的。**
  - 例如，聽力：聲波 → 詞匯/句子
  - 例如，動作：大腦信號/指令 → 順序肌肉運動
- 人的思維具有持久性；人類不會每秒都從頭開始思考。
  - 當您閱讀這句話時，您會根據您之前的知識來理解每個詞。
- **標準人工神經網路（以及卷積網路）的應用受到限制，原因如下：**
  - 它們只接受固定大小的向量作為輸入（例如，圖像），並生成固定大小的向量作為輸出（例如，不同類別的機率）。
  - 這些模型使用了固定數量的計算步驟（例如，模型中的層數）。
- **循環神經網路(RNN)是一類神經網路，學習時間序列數據(Time series data)。**
  - 靈感來自於依賴時間和持久性的人類思維，可用在股市時間序列預測。

現實生活中的序列學習應用

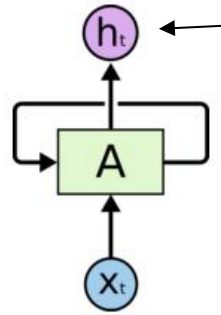
RNN （Recurrent neural network）**以學習時間模式應用於各種類型的序列數據。**

- 時間序列數據（例如，股票價格）→ 預測，均值回歸（Mean Reversion）
- 原始傳感器數據（例如，信號、語音、手寫）→ 標簽或文本序列
- 文本→ 標籤（例如，情感）或文本序列（例如，翻譯、摘要、答案）
- 圖像和視頻→ 文本描述（例如，字幕、場景解釋）

Task	Input	Output
Activity Recognition (Zhu et al. 2018)	Sensor Signals	Activity Labels
Machine translation (Sutskever et al. 2014)	English text	French text
Question answering (Bordes et al. 2014)	Question	Answer
Speech recognition (Graves et al. 2013)	Voice	Text
Handwriting prediction (Graves 2013)	Handwriting	Text
Opinion mining (Irsoy et al. 2014)	Text	Opinion expression

循環神經網路

- 循環神經網路是具有循環的網路，允許信息持久存在。



Recurrent Neural Networks have loops.

輸出是預測向量h_t，其中$output\ y_t=\varphi(h_t)$在某些時間點（t）上

在上圖中，神經網路的一部分，A = f_W，處理一些輸入X_t並輸出一個值h_t。允許信息從循環網路的一步傳遞到下一步。

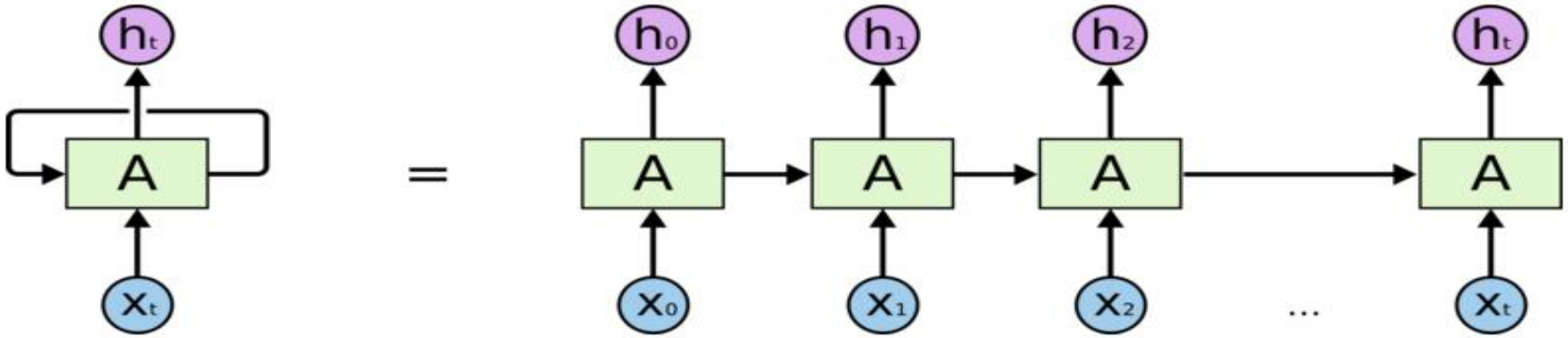
new state　　old state

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

function with parameter W　　Input vector at some time step

展開循環神經網路

- 展開*RNN*



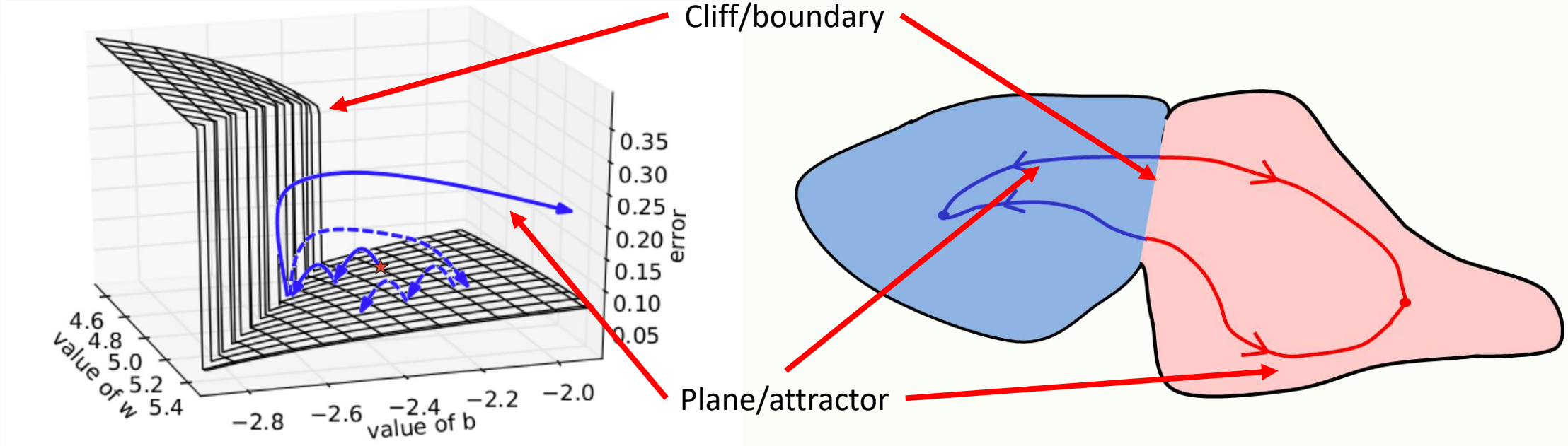
An unrolled recurrent neural network.

上圖顯示了如果我們展開**循環神經網路**會發生什麼。
可以將循環神經網路看作是同一網路的多個副本，每個副本將消息傳遞給後繼網路。

循環神經網路

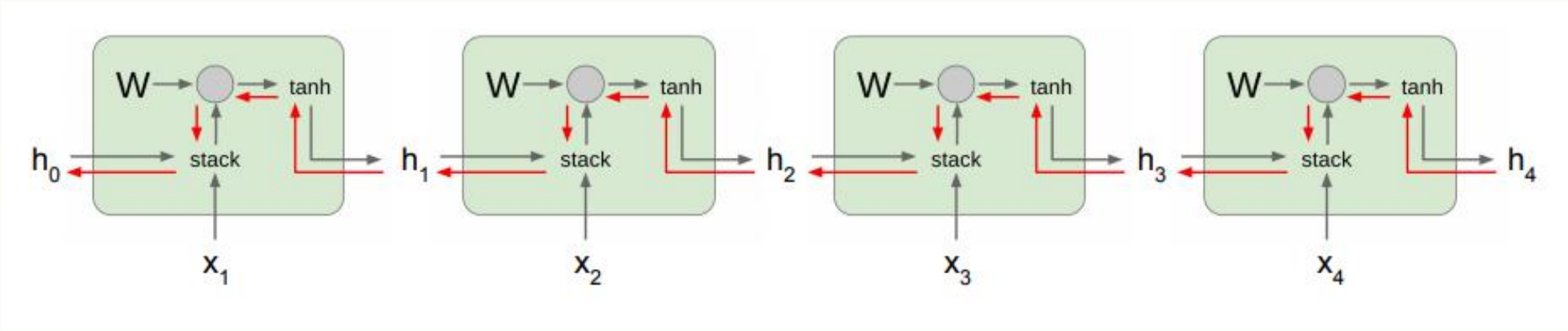
- RNN的循環結構使其具有以下特點:
  - 專門用於處理一序列的值 $x^{(1)}, \ldots, x^{(\tau)}$
    - 每個值 $x^{(i)}$ 都使用**相同的網路A進行處理，該網路保留了過去的信息**
  - 可以擴展到比沒有循環結構的網路**更長的序列**
    - 重覆使用網路A可以減少網路中所需的參數量
  - 可以處理**可變長度的序列**
    - 當輸入長度變化時，網路覆雜性不會發生變化
  - 然而，原型的RNN**由於梯度爆炸(Gradient Explored) 和梯度消失(Gradient Vanishing) 面臨訓練困難的問題。**

梯度爆炸和梯度消失



- **梯度爆炸**：如上右圖，幾乎正處在邊界（懸崖）上，微小變化可能會導致巨大差異。
- **梯度消失**：如上左圖，在一個平面或平坦表面內的軌跡，那麼我們從哪里開始的微小變化對我們最終停留的位置沒有什麼影響。
- 這兩種情況都會妨礙學習過程。

梯度爆炸和梯度消失



在原型的循環神經網路（RNN）中，計算梯度涉及到多個W_{hh}（以及重覆的tanh）。分解梯度乘法矩陣的奇異值(singular value)，

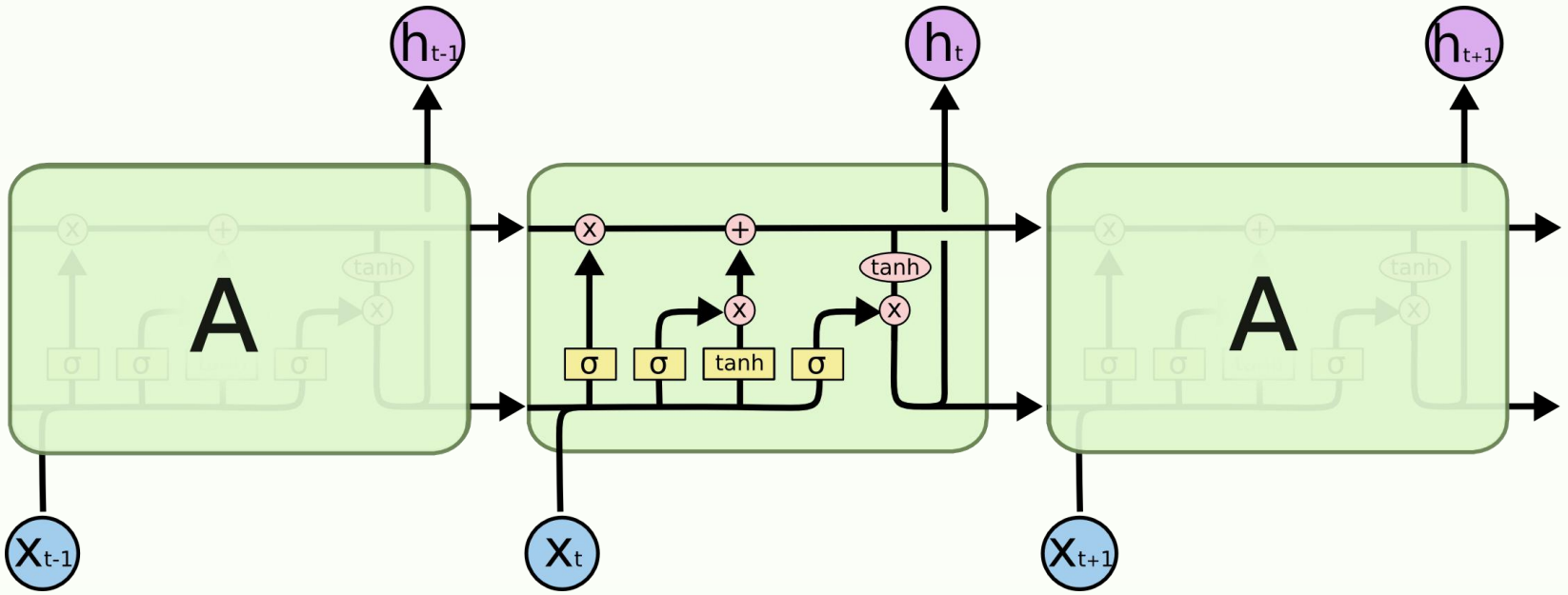
- 最大奇異值 > 1 → **梯度爆炸**
  - 在後期時間步驟中的輕微誤差會導致在早期時間步驟中進行急劇的更新 → **不穩定的學習**
- 最大奇異值 < 1 → **梯度消失**
  - 傳遞到前期時間步驟的梯度接近於0。 →**資訊貧乏的修正**

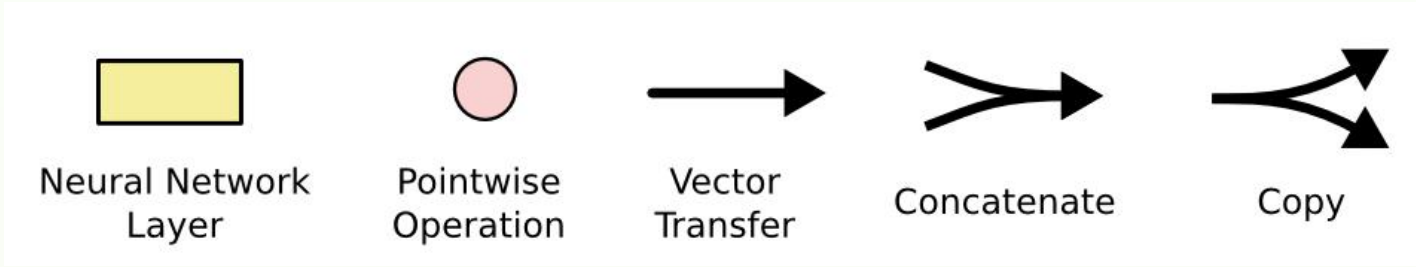
* 參考 Bengio 等人 (1994) 或 Goodfellow 等人 (2016)

Long Short Term Memory LSTM長短期記憶

- LSTM網絡在每個記憶單元中添加了額外的閘門單元。
  - 遺忘閘門 (forget gate)
  - 輸入閘門 (input gate)
  - 輸出閘門 (output gate)
- 防止梯度消失/爆炸問題，使網絡能夠在較長時間內保留狀態訊息。

LSTM網路架構





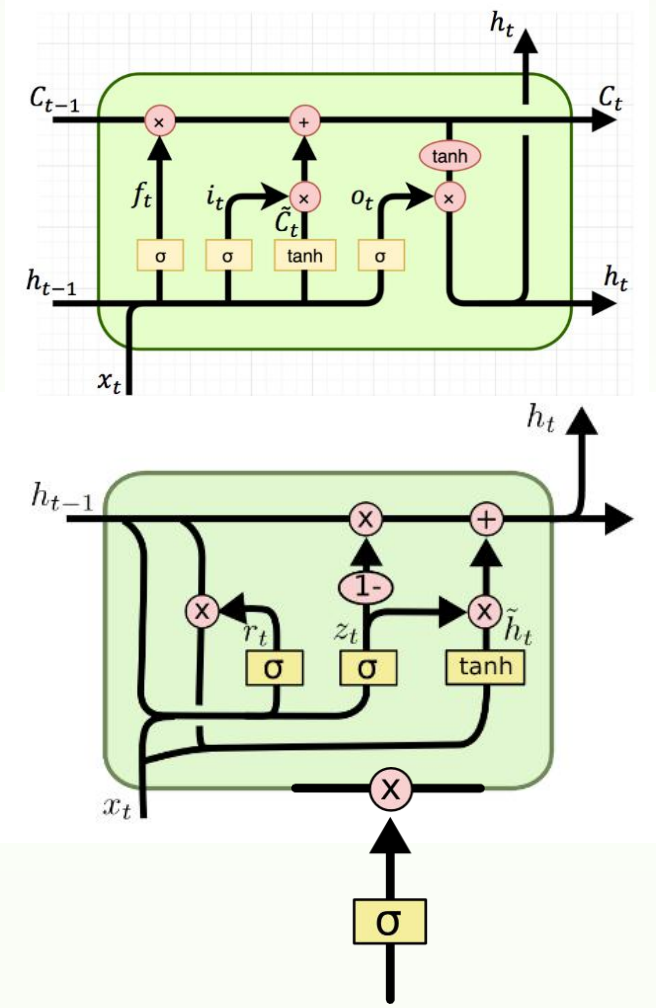
記憶性網路

- 原型RNN以“乘法”方式運作（重覆的tanh）。

- 有兩種循環單元設計被提出並廣泛採用：
  - **長短時記憶（LSTM）**（Hochreiter和Schmidhuber，1997）
  - 門控循環單元（GRU）（Cho等，2014）

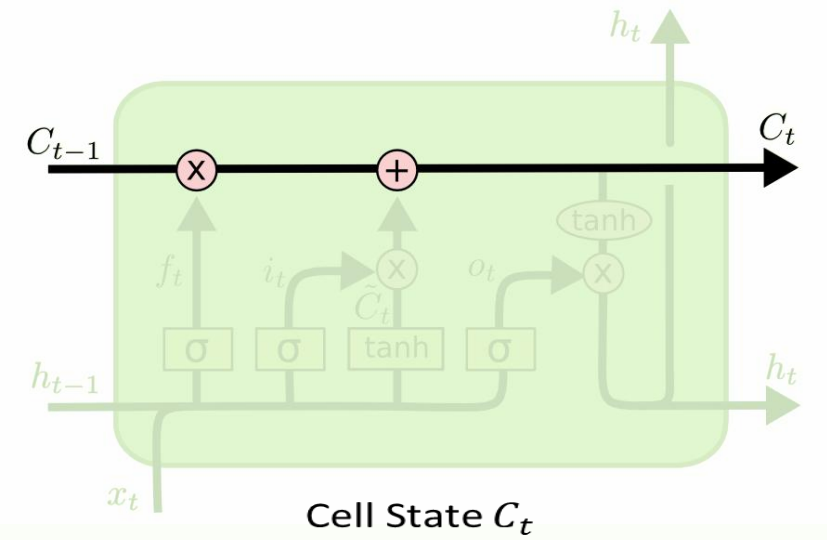
- 這兩種設計以“加法”方式處理信息，使用門控制信息流動。
  - **Sigmoid門輸出介於0和1之間的數字，描述多少元件需要通過轉換。**

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



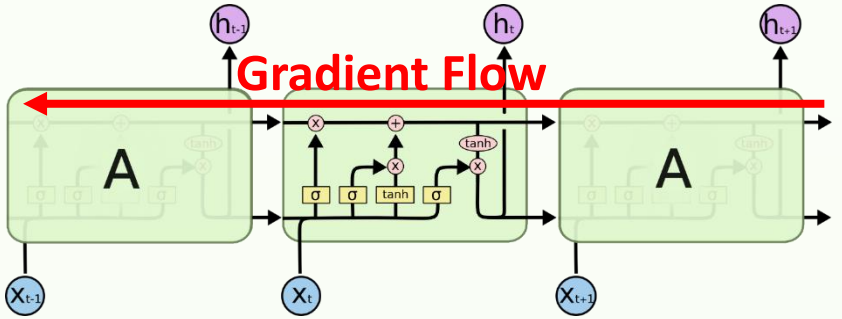
長短期記憶（LSTM）

- LSTM的關鍵是細胞狀態。
  - **存儲過去的信息➔長期記憶**
  - **通過輕微的線性交互作用傳遞時間步驟➔“加法”**
  - 導致不間斷的**梯度流➔**過去的誤差影響未來的學習



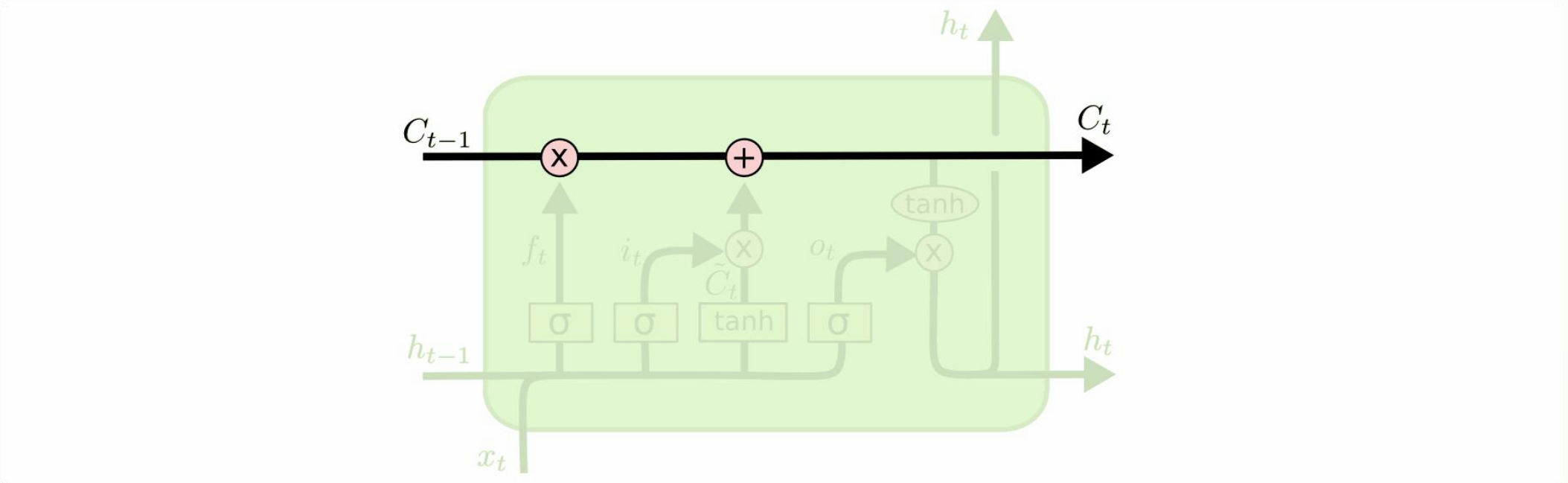
Cell State C_t

- LSTM單元使用三個門來操作輸入信息。
  - **輸入門➔**控制新信息的輸入
  - **遺忘門➔**確定要更新細胞狀態的哪部分
  - **輸出門➔**確定要輸出細胞狀態的哪部分



細胞狀態(Cell State)

- 維護一個與隱藏狀態C_t具有相同維度的向量h_t
- 可以通過遺忘門和輸入門向這個狀態向量添加或刪除訊息。

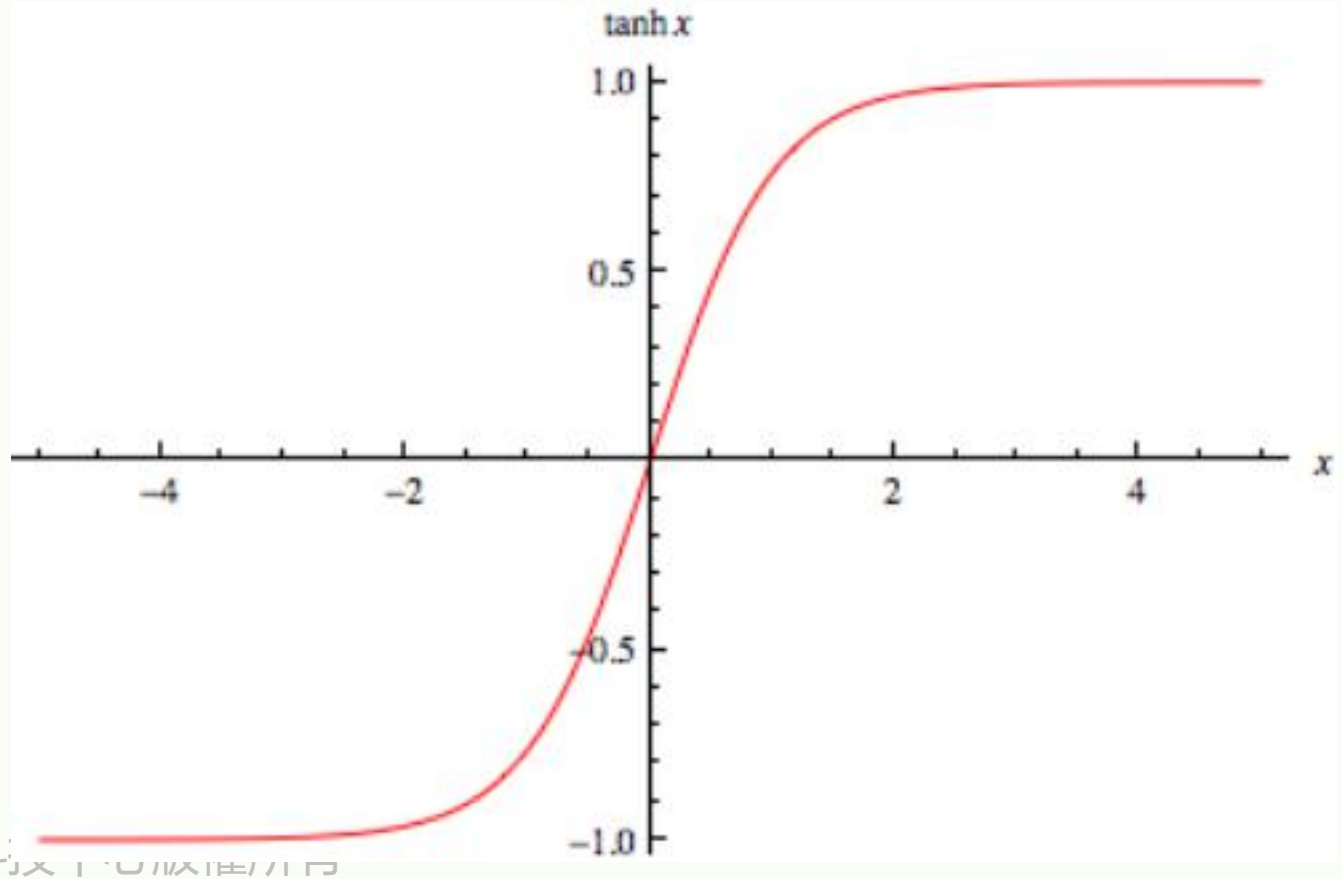


遺忘門

- 遺忘門使用邏輯Sigmoid輸出函數從輸入X_t和當前隱藏狀態h_t計算一個0 到 1的值:
- 與細胞狀態相乘，當門輸出值接近0時，“遺忘”訊息。

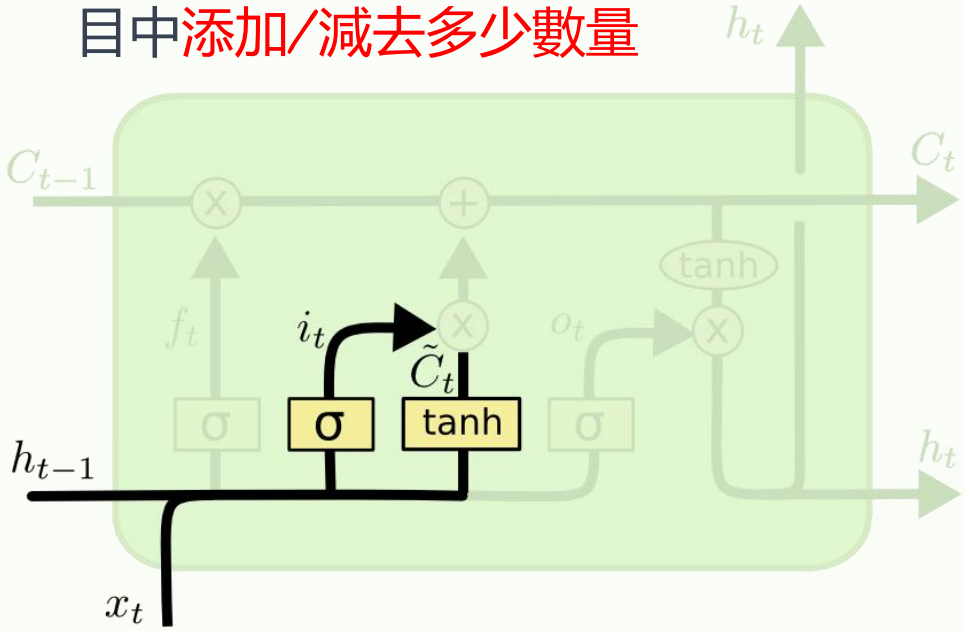
雙曲正切單元(Hyperbolic Tangent Units)

- Tanh可以用作替代非線性函數，用於互補Sigmoid邏輯（0-1）輸出函數。
- 用於生成在-1和1之間的閥值輸出。



輸入門

- 首先，通過計算0-1的Sigmoid函數輸出，來確定要更新的細胞狀態中的哪些項目。
- 然後，通過計算輸入和隱藏狀態的tanh輸出（值為-1到1）來確定要從這些項目中添加/減去多少數量

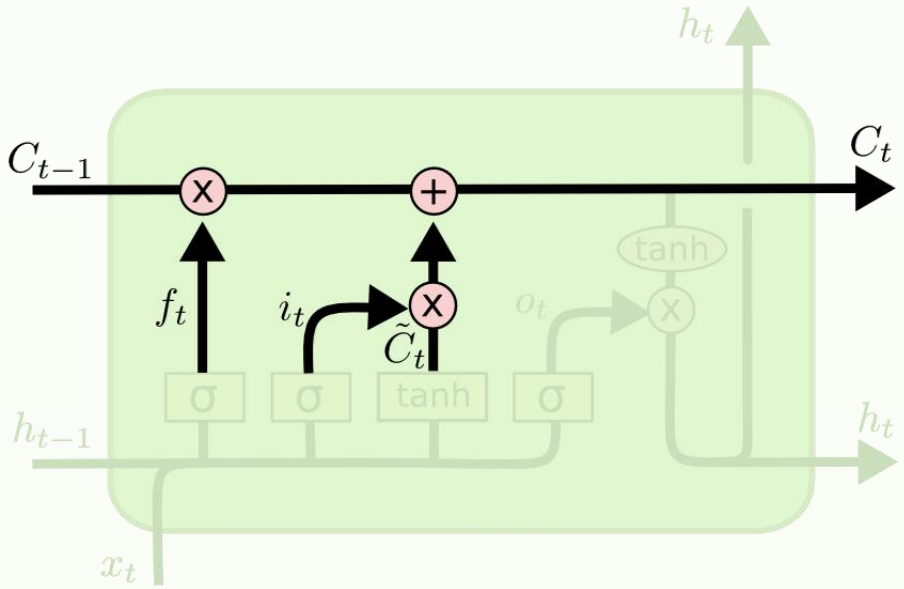


$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \; + \; b_i\right)$$

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

更新細胞狀態

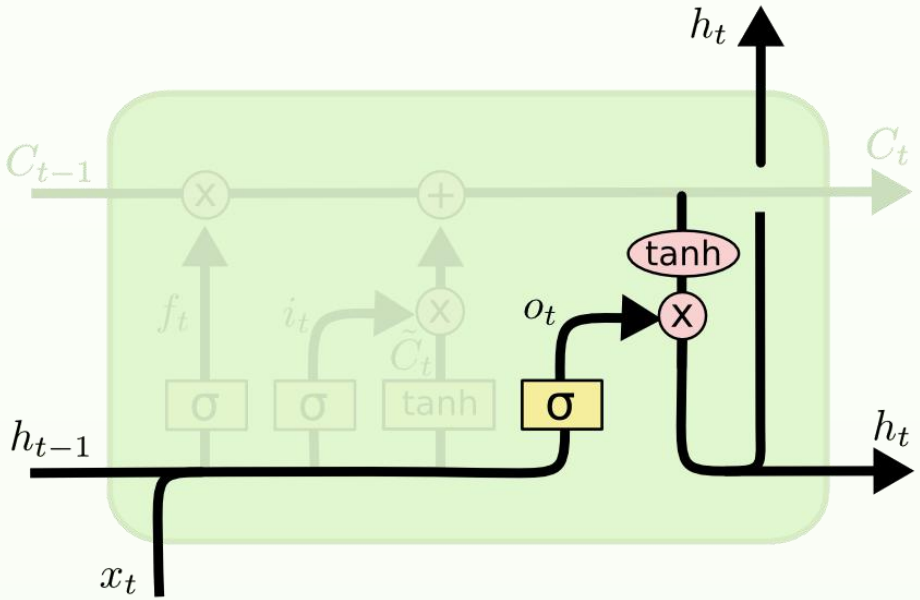
- 細胞狀態通過使用分量方式的向量乘法來“遺忘”，用向量加法來將“輸入”新訊息進行更新。



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

輸出門

- 隱藏狀態是基於細胞狀態的“過濾”版本進行更新的，通過*tanh*縮放到-1到1。
- 輸出門計算輸入和當前隱藏狀態的*Sigmoid*函數，以確定細胞狀態的哪些元素要“輸出”。



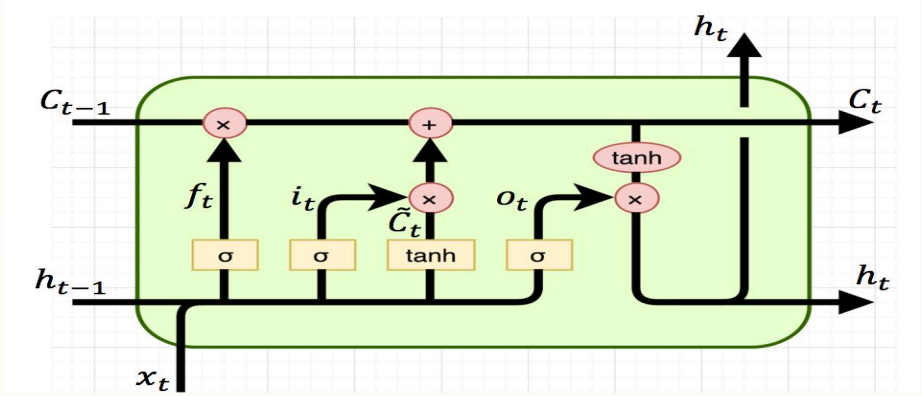
$$o_t = \sigma\left(W_o\left[h_{t-1}, x_t\right] + b_o\right)$$

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

LSTM：組成和流程

- LSTM單元的輸出
- **輸出門**單元
- 轉換後的記憶細胞內容
- 門控制記憶細胞單元的更新

- **遺忘門**單元
- **輸入門**單元
- 記憶細胞的潛在輸入



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

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

$$\tanh(C_t)$$

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

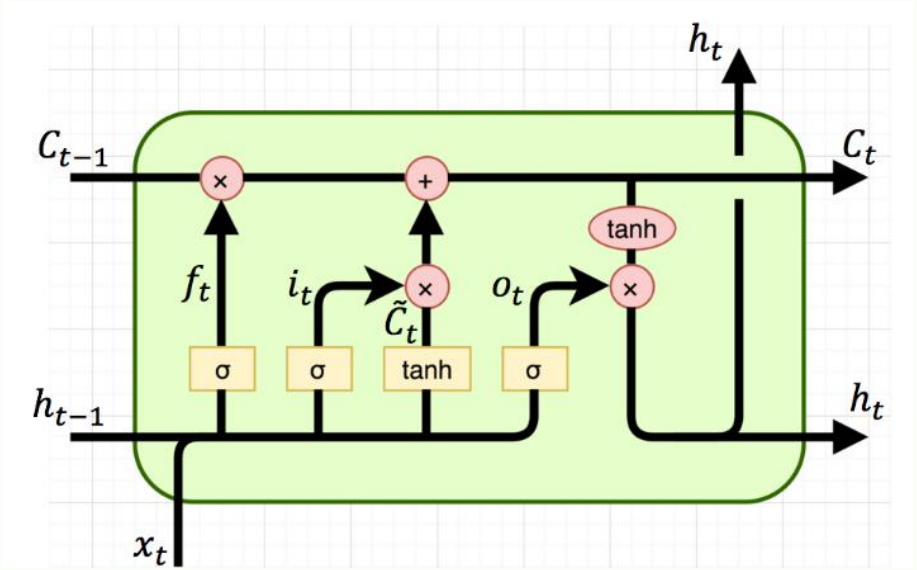
$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$

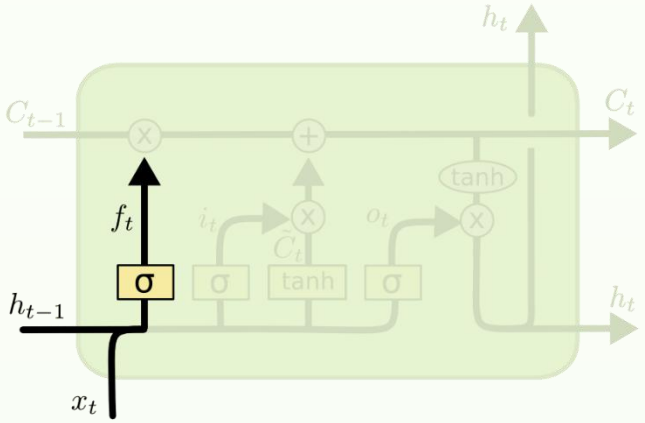
$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$

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

逐步 LSTM 解析

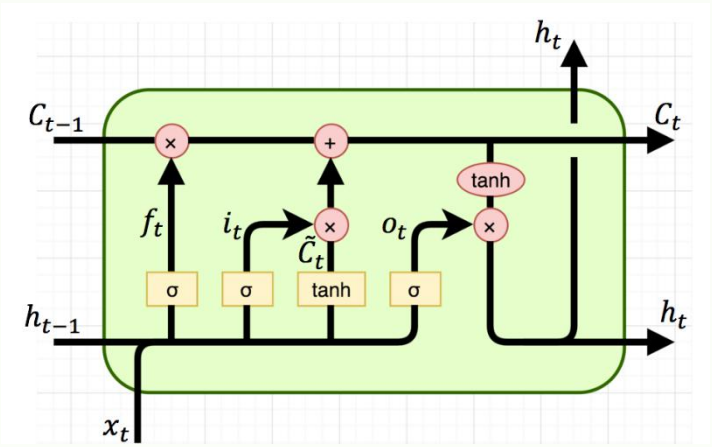
- 步驟1：決定從細胞狀態（記憶）中丟棄(遺忘)哪些信 $f_t * C_{t-1}$
  - 前一個狀態 h_{t-1} 的輸出和新信息 x_t共同決定要忘記什麼
    - h_{t-1}包含了從記憶 C_{t-1}中選擇的特徵
  - 遺忘門 f_t取值範圍在 [0, 1] 之間



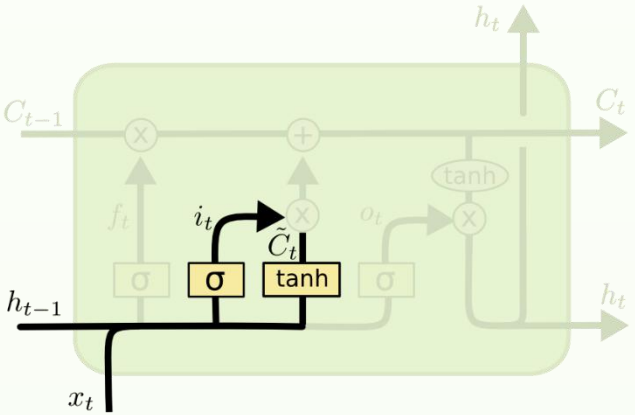


$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \; + \; b_f\right)$$

逐步 LSTM 解析



- 步驟2： 為細胞狀態準備來自輸入的更新 $i_t * \tilde{C}_t$
  - 通過以h_{t-1}的指導，從新訊息 x_t創建一個備選細胞狀態$\tilde{c}_t$ 。
  - 輸入門 i_t 的取值範圍在 [0, 1] 之間。

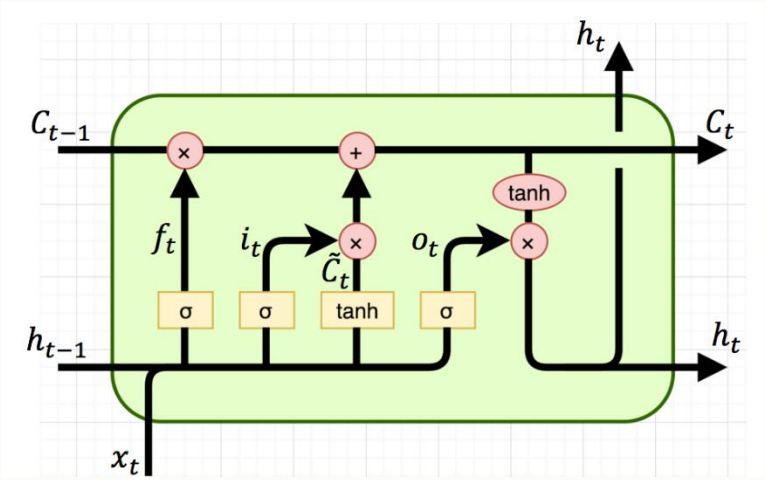


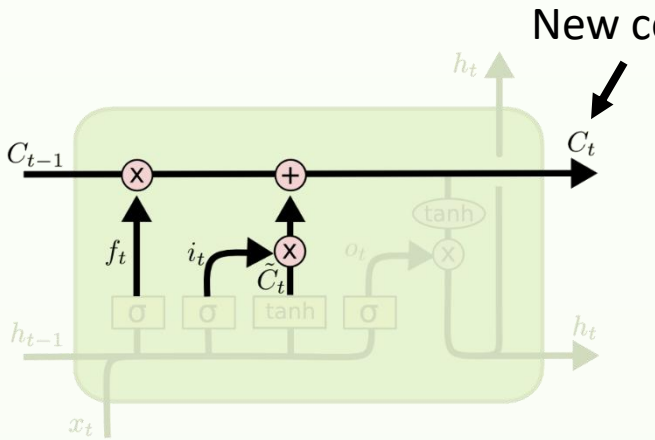
$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \; + \; b_i\right)$$

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

逐步 LSTM 解析

- 步驟3： 更新細胞狀態➔ $f_t * C_{t-1} + i_t * \tilde{C}_t$
  - 新的細胞狀態C_t包括來自過去的信息 $f_t * C_{t-1}$和有價值的新信息 $i_t * \tilde{c}_t$
  - $*$ 表示逐元素相乘。

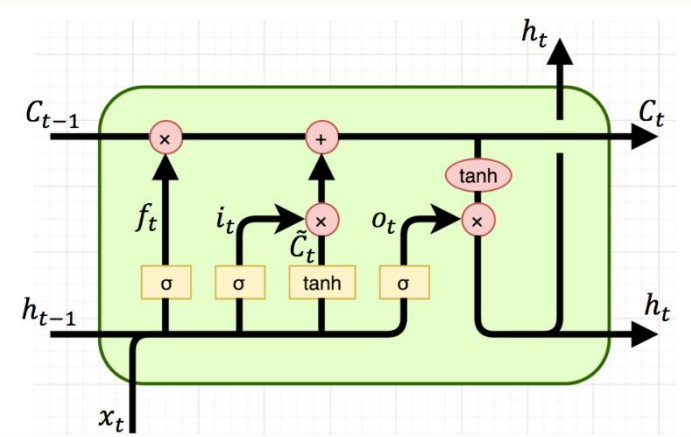


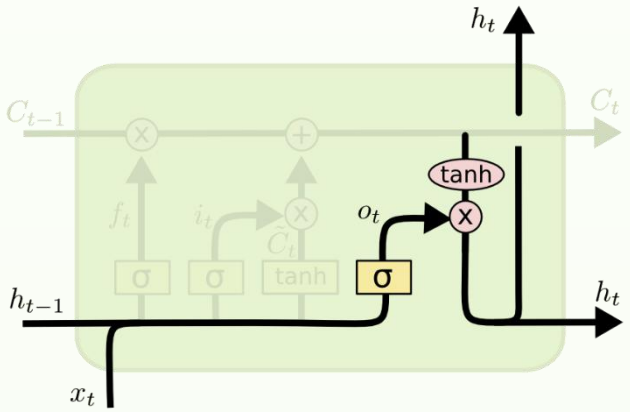


$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

逐步 LSTM 解析

- 步驟4：決定來自新細胞狀態的篩選輸出➔ $o_t * \tanh(C_t)$
  - tanh函數對新細胞狀態進行過濾以判別要儲存或丟棄的訊息
    - C_t中的重要的息 ➔ ±1
    - 均值➔ 0
  - 輸出門 O_t的範圍在 [0, 1] 之間
  - h_t作為下一個時間步的控制信號。



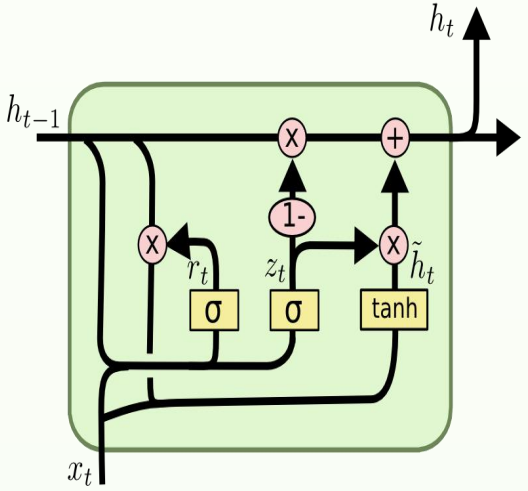


$$o_t = \sigma\left(W_o\left[h_{t-1}, x_t\right] + b_o\right)$$

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

門控循環單元(GRU)

- GRU是LSTM的一種變化，也採用了門控設計。
- 不同之處：
  - GRU使用**更新門** z 來替代輸入門和遺忘門 i_t和 f_t
  - 將在LSTM中的細胞狀態 C_t和隱藏狀態 h_t合併為一個單一的細胞狀態 h_t
  - GRU在參數更少和更快的收斂速度下表現出與LSTM相似的性能（Cho等人，2014）。



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

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

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

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

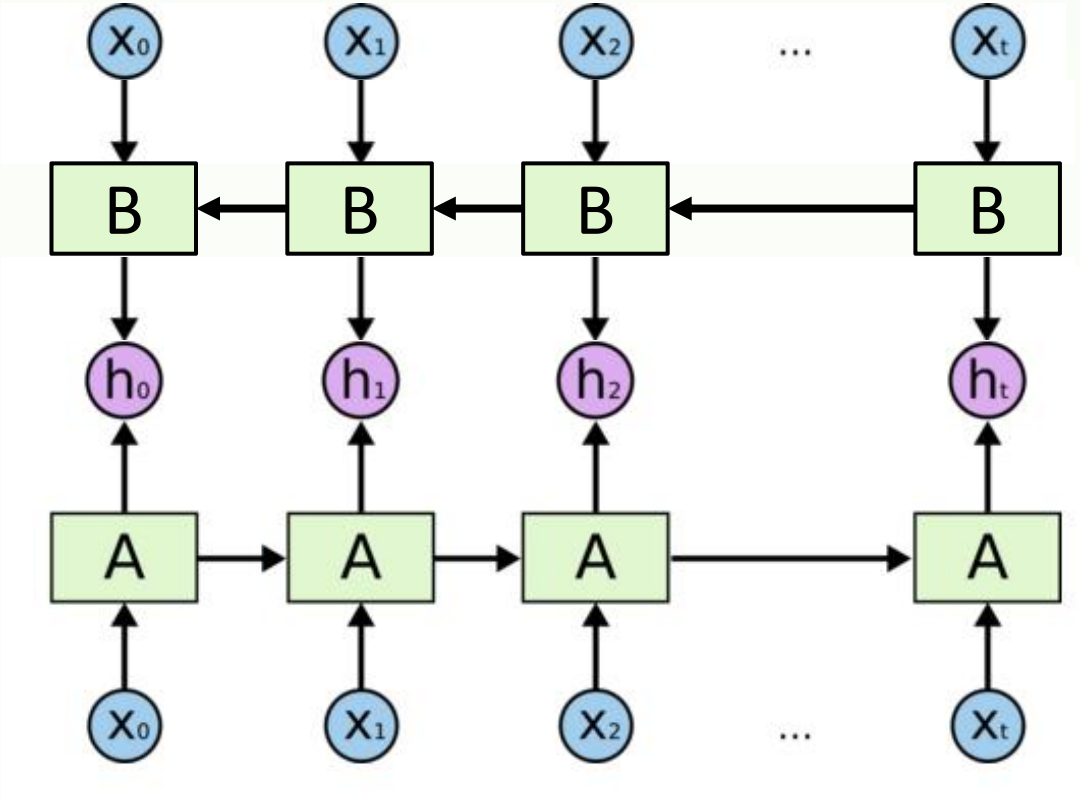
序列學習體系結構

- 當梯度消失/爆炸問題得到解決時，基於*RNN*的學習更加健壯。
  - 現在，*RNN*可以應用於不同的序列學習任務。

- 循環神經網路架構可以靈活地處理各種向量序列。
  - 輸入、輸出或在最一般的情況下都可以是序列
  - 具有一個或多個*RNN*層的體系結構

雙向RNN的序列學習 bi-LSTM

- 雙向迴圈神經網路
  - 將兩個方向相反的迴圈單元（同步的多對多模型）連接到同一輸出。
  - 從輸入序列中捕獲向前和向後的資訊
  - 適用於在給定未來資訊（例如，x_1 、x_2,…, x_t ）時可以更好地確定當前狀態（例如，h_0 ）的數據。

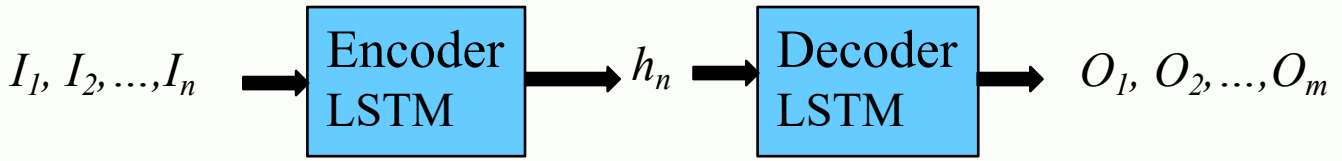


LSTM訓練

- 可以使用反向傳播導數進行訓練，例如：
  - 隨機梯度下降（在每個時期中隨機排列示例的順序），帶有動量（偏差權重的變化方向與上次更新一致）。
  - ➢ 採用ADAM優化器效果佳（Kingma&Ma，2015）
- 每個單元都有許多參數（Wf、Wi、WC、Wo）
  - 通常需要大量的訓練數據。
  - 需要大量的計算時間，可以充分利用GPU集群。

序列對序列轉換（映射）

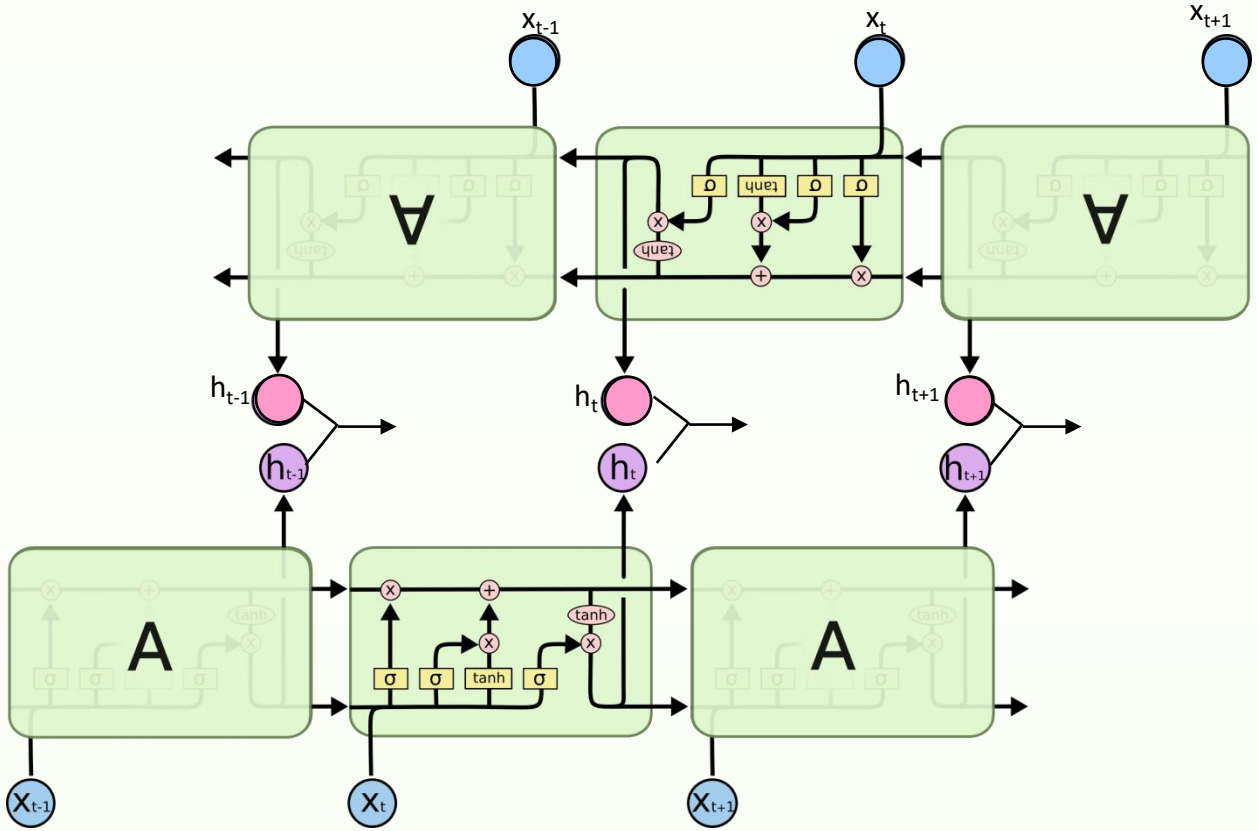
- 編碼器/解碼器框架將一個序列映射到一個“深度向量”，然後另一個LSTM將這個向量映射到輸出序列。



- 在序列對的輸入/輸出上“點對點”訓練模型。

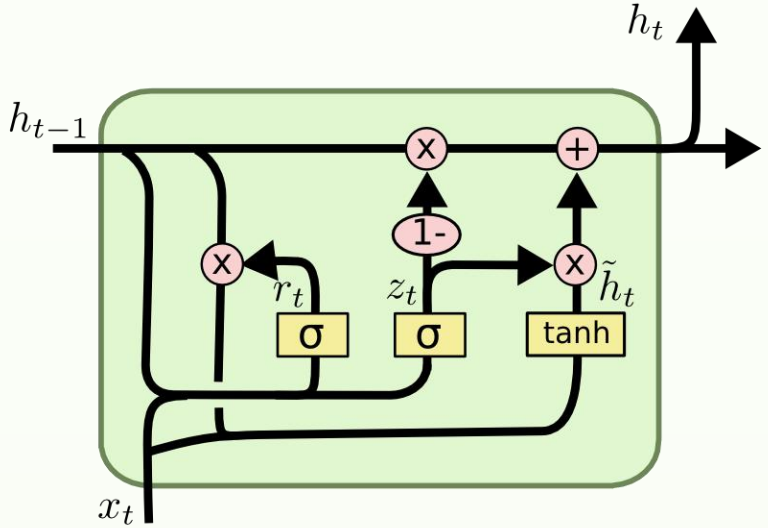
雙向LSTM（Bi-LSTM）

- 分別處理正向和反向的序列，每個時間步長的隱藏層連接起來形成細胞輸出。



門控循環單元（GRU）

- GRU是LSTM的替代循環神經網絡，使用較少的門（Cho等人，2014年）。
  - 將遺忘門和輸入門合併為“更新”門。
  - 不包括細胞狀態向量。



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

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

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

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

GRU與LSTM的比較

- GRU具有更少的參數，訓練速度更快。
- 對比這兩者的實驗結果仍然沒有定論，它們在許多問題上表現相同，但在某些問題上各自有更好的表現。

總結

- 通過在循環神經網絡中添加“門”，可以防止梯度消失/爆炸的問題。
- 經過訓練的LSTM/GRU可以更長時間地保留狀態訊息並處理長距離依賴性。
- 在一系列具有挑戰性的自然語言處理問題上取得了令人印象深刻的最新結果。
- LSTM和GRU是RNN，它們保留過去的信息並通過門控設計進行更新。
  - “加法”結構避免了梯度消失問題。
- RNN 允許靈活的架構設計，以適應不同的序列學習要求。
- RNN 具有廣泛的現實應用。
  - 文本處理、機器翻譯、信號提取/識別、圖像字幕
  - 移動健康分析、日常生活活動、老年護理

*: p-value<0.05, **: p-value<0.01, ***: p-value<0.001

重要參考資料

- Bengio, Y., Simard, P., & Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. IEEE transactions on neural networks, 5(2), 157-166.
- Goodfellow, I., Bengio, Y., Courville, A., & Bengio, Y. (2016). Deep learning (Vol. 1). Cambridge: MIT press.
- Graves, A. (2012). Supervised sequence labelling with recurrent neural networks. https://www.cs.toronto.edu/~graves/preprint.pdf
- Jozefowicz, R., Zaremba, W., & Sutskever, I. (2015, June). An empirical exploration of recurrent network architectures. In International Conference on Machine Learning (pp. 2342-2350).
- Lipton, Z. C., Berkowitz, J., & Elkan, C. (2015). A critical review of recurrent neural networks for sequence learning. arXiv preprint arXiv:1506.00019.
- Salehinejad, H., Baarbe, J., Sankar, S., Barfett, J., Colak, E., & Valaee, S. (2017). Recent Advances in Recurrent Neural Networks. arXiv preprint arXiv:1801.01078.

章節到此結束，有任何問題歡迎提出來討論！