- 決策樹

  Gain(A) = $Info$ (D) - $Info_A$(D)

age	income	student	credit_rating	buys_computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
31…40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31…40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31…40	medium	no	excellent	yes
31…40	high	yes	fair	yes
>40	medium	no	excellent	no

$Info$ (D) = I(9,5) = $\frac{9}{14}\ log_2\frac{9}{14} + \frac{5}{14}\ log_2\frac{5}{14}$ = 0.940

$Info_{age}$(D) = $\frac{5}{14}$ I(2,3) + $\frac{4}{14}$ I(4,0) + $\frac{5}{14}$ I(3,2)

= $\frac{5}{14}$ x -($\frac{2}{5}$ log $\frac{2}{5}$ + $\frac{3}{5}$log $\frac{3}{5}$) = $\frac{5}{14}$ x (0.529 + 0.442) = $\frac{5}{14}$ x 0.971 +

$\frac{4}{14}$ * -(0) +

$\frac{5}{14}$ x -($\frac{2}{5}$ log $\frac{2}{5}$ + $\frac{3}{5}$log $\frac{3}{5}$) = $\frac{5}{14}$ x (0.529 + 0.442) = $\frac{5}{14}$ x 0.971

= 0.694

Gain(age) = $Info$ (D) – $Info_{age}$(D) = 0.940 – 0.694 = 0.246 →最大選這個

Gain(income) = 0.029

Gain(student) = 0.151

Gain(credit_rating) = 0.048

- Continuous-Value Attributes 數字類型的屬性

  ✓ Best split point 最佳切入點 → $\frac{(a_i+a_{i+1})}{2}$

- CAPT 演算法 → 決策樹 → 快、簡單易懂、準確性

- Bayesian classification 貝式(貝葉斯)分類器

ear interval	eye interval	mouth width	class
25	10	8	handsome
30	15	15	ugly
27	12	10	handsome
25	10	8	handsome
30	12	7	ugly
20	8	7	handsome
27	10	8	ugly

總共 ：7；Handsome : 4 ；ugly : 3 →算機率

(1) X1：{ear interval = 27, eye interval = 10, mouth width = 8}

➔ Handsome : $\frac{1}{4}$ x $\frac{2}{4}$ x $\frac{2}{4}$ x $\frac{4}{7}$ = 0.036 (v)

➔ Ugly : $\frac{1}{3}$ x $\frac{1}{3}$ x $\frac{1}{3}$ x $\frac{3}{7}$ = 0.016

(2) X2：{ear interval = 27, eye interval = 12, mouth width = 8}

➔ Handsome : $\frac{1}{4}$ x $\frac{1}{4}$ x $\frac{2}{4}$ x $\frac{4}{7}$ = 0.018 (v)

➔ Ugly : $\frac{1}{3}$ x $\frac{1}{3}$ x $\frac{1}{3}$ x $\frac{3}{7}$ = 0.016

- Bayesian theorem 貝葉斯定理

→ 缺點:需要很多概率的初始知識，計算成本高

訓練資料 : X ；

假設 H 的後驗概率=P(H｜X)

$$P(H|\mathbf{X}) = \frac{P(\mathbf{X}|H)P(H)}{P(\mathbf{X})}$$

→ 後驗 = $\frac{\text{似然}\times\text{先驗}}{\text{證據}}$

- Naïve Bayesian Classifier 單純貝氏分類器

➔ 令 D 為元組及其關聯類標籤的訓練集，每個元組由 n-D（維度）

➔ 屬性向量 X = (x_1, x_2, ⋯, x_n) 表示

➔ 假設有 m 個類 C_1，C_2，...，C_m。

➔ 分類是求最大後驗，即最大 P(Ci|X)

➔ 這可以從貝葉斯定理推導出來

$$P(C_i|\mathbf{X}) = \frac{P(\mathbf{X}|C_i)P(C_i)}{P(\mathbf{X})}$$

➔ 由於 P(X) 對於所有類別都是常數，因此只有

$$P(C_i|\mathbf{X}) = P(\mathbf{X}|C_i)P(C_i)$$ →C_i判斷 X 屬於哪個類別

需要最大化

➔ Naïve Bayesian Classifier: Training Dataset 單純貝氏分類器:測試資料集

age	income	student	credit_rating	buys_computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
31…40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31…40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31…40	medium	no	excellent	yes
31…40	high	yes	fair	yes
>40	medium	no	excellent	no

- Class: 總共 14 項資料

  C1:buys_computer = 'yes' →有 9 個 yes

  C2:buys_computer = 'no' →有 5 個 no

- Data sample :

  X = (age <=30, Income = medium, Student = yes, Credit_rating = Fair)

○ $P(C_i)$: P(buys_computer = "yes") = $\frac{9}{14}$ = 0.643

P(buys_computer = "no") = $\frac{5}{14}$ = 0.357

○ Compute $P(X|C_i)$ for each class

P(age = "<=30" | buys_computer = "yes") = $\frac{2}{9}$ = 0.222

P(age = "<= 30" | buys_computer = "no") = $\frac{3}{5}$ = 0.6

P(income = "medium" | buys_computer = "yes") = $\frac{4}{9}$ = 0.444

P(income = "medium" | buys_computer = "no") = $\frac{2}{5}$ = 0.4

P(student = "yes" | buys_computer = "yes) = $\frac{6}{9}$ = 0.667

P(student = "yes" | buys_computer = "no") = $\frac{1}{5}$ = 0.2

P(credit_rating = "fair" | buys_computer = "yes") = $\frac{6}{9}$ = 0.667

P(credit_rating = "fair" | buys_computer = "no") = $\frac{2}{5}$ = 0.4

- **X = (age <= 30 , income = medium, student = yes, credit_rating = fair)**

  (1)$P(X|C_i)$:

  P(X|buys_computer = “yes”) = 0.222 × 0.444 × 0.667 × 0.667 = 0.044

  P(X|buys_computer = “no”) = 0.6 × 0.4 × 0.2 × 0.4 = 0.019

  (2)$P(X|C_i)*P(C_i)$:

  P(X|buys_computer = “yes”) × P(buys_computer = “yes”) = 0.028

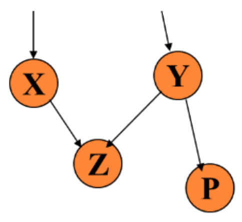
  P(X|buys_computer = “no”) × P(buys_computer = “no”) = 0.007

  Therefore, X belongs to class (“buys_computer = yes”)

- 避免 0 概率問題
  - ✓ 單純貝氏分類器需要每個條件概率，為『非零』。
    否則，預測的概率，將為零
  - ✓ 假設一個有 1000 個元組的數據集，
    income=low（0），income=medium（990），income=high（10），
    使用拉普拉斯校正（或拉普拉斯估計器）
    每個案例加 1
    概率（收入 = 低）= 1/1003
    概率（收入=中等）= 991/1003
    概率（收入 = 高）= 11/1003
    “更正”的概率。
    估計值接近於“未修正”的對應值

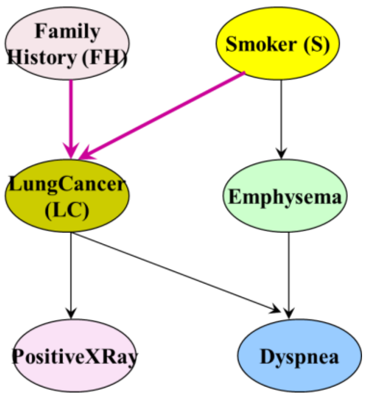
- Naïve Bayesian Classifier 單純貝氏分類器:
  優點 : 易於操作、大多數情況可以有良好效果
  缺點 : 假設條件獨立，損失準確性

- Bayesian Belief Network 貝式信念網路 :
  →表示 Naïve Bayesian Classifier 單純貝氏分類器的依賴關係
  →允許變量子集之間的類條件獨立性
  →因果關係的（有向非循環）圖形模型:
  1. 表示變量之間的依賴關係
  2. 給出聯合概率分佈的規範



- ✓ 節點：隨機變量
- ✓ 鏈接：依賴
- ✓ X 和 Y 是 Z 的父母，Y 是 P 的父母
- ✓ Z 和 P 之間沒有依賴關係
- ✓ 沒有循環/循環

→例子:



✓ CPT：可變肺癌的條件概率表

✓ 顯示其父項的每個可能組合的條件概率

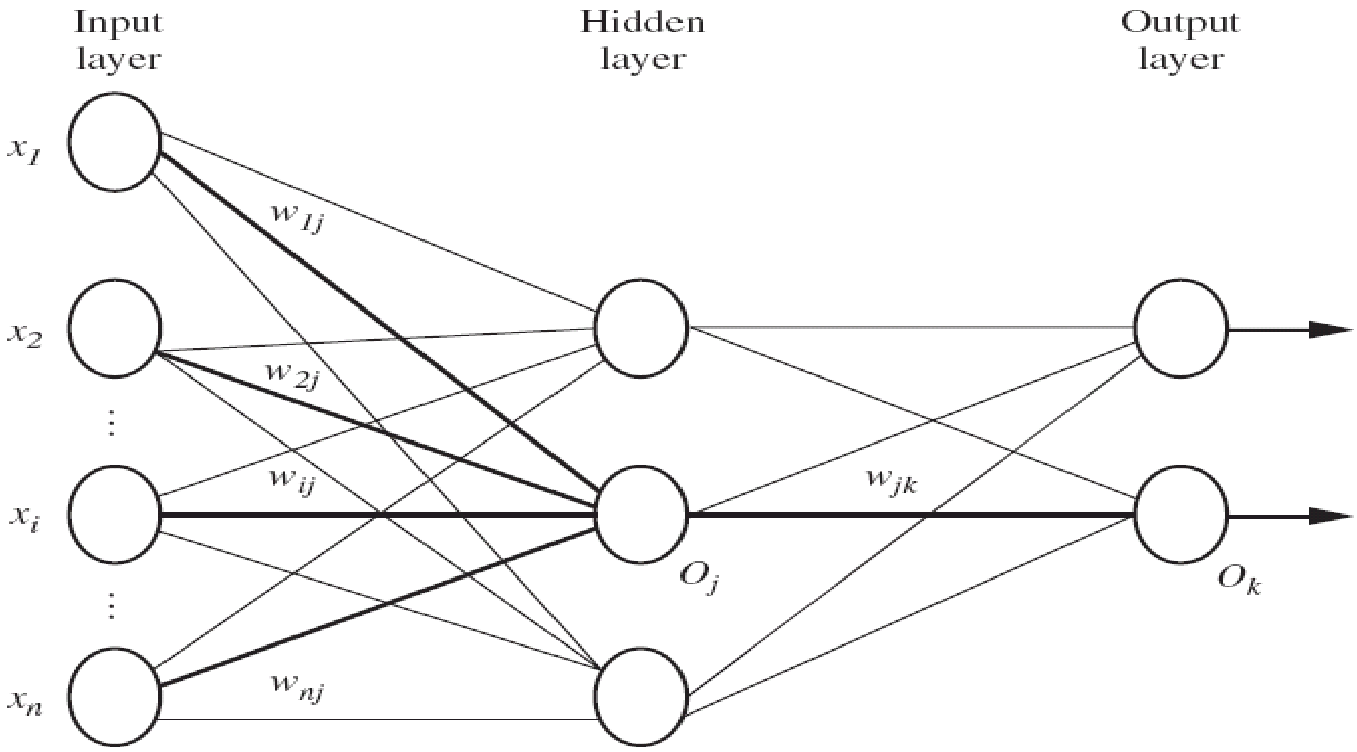
	(FH,S)	(FH,~S)	(~FH,S)	(~FH,~S)
LC	0.8	0.5	0.7	0.1
~LC	0.2	0..5	0.3	0.9

✓ 從 CPT 推導 X 值的特定組合的概率：

- Rule-based classification 基於規則的分類

Artificial neural network (ANN) 人工神經網絡

→是一種模仿生物神經網路的結構和功能的數學模型或計算模型。



→每個點跟點之間都會連線，也會有各自的權重(w)，輸出出現”類別的機率”

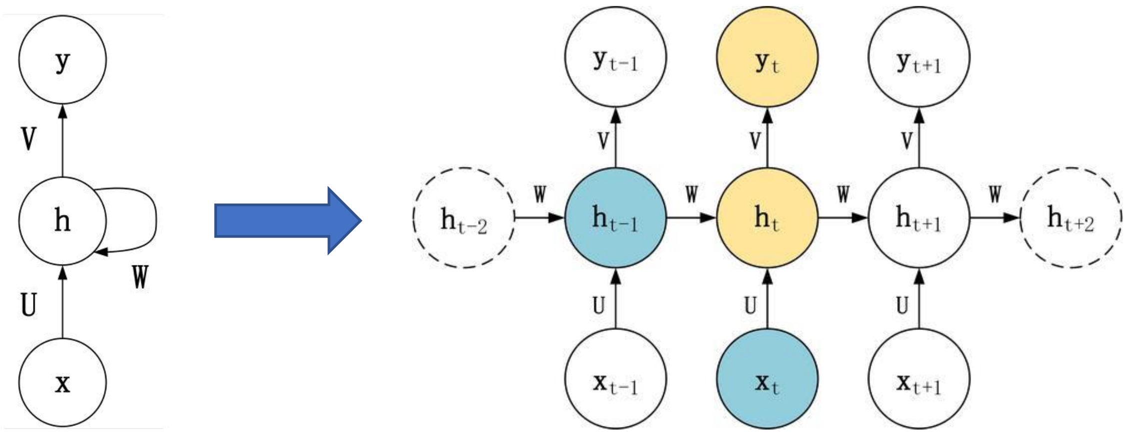
Recurrent Neural Network (RNN) 循環神經網路

→是神經網路的一種

→資料是有順序性的

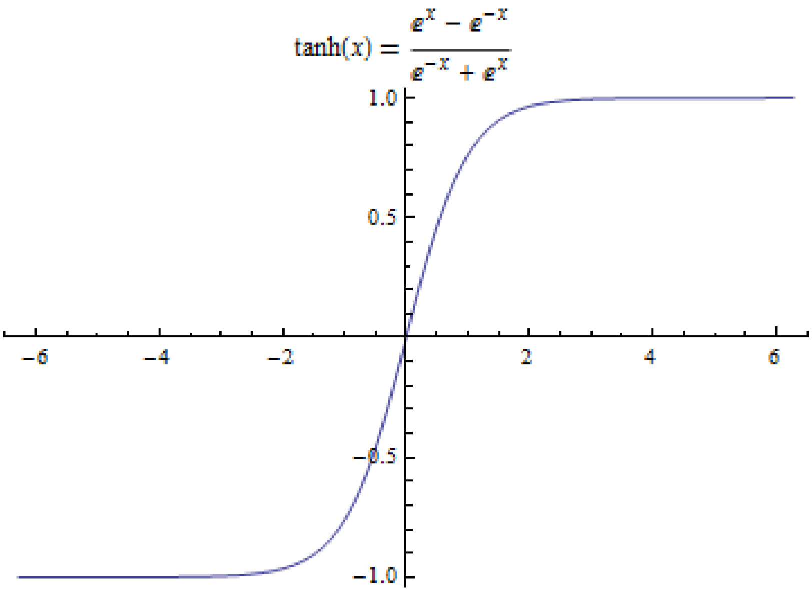
→單純的 RNN 因為無法處理隨著遞歸，權重指數級爆炸或梯度消失問題，難以捕捉長期時間關聯

→而結合不同的 LSTM 可以很好解決這個問題。

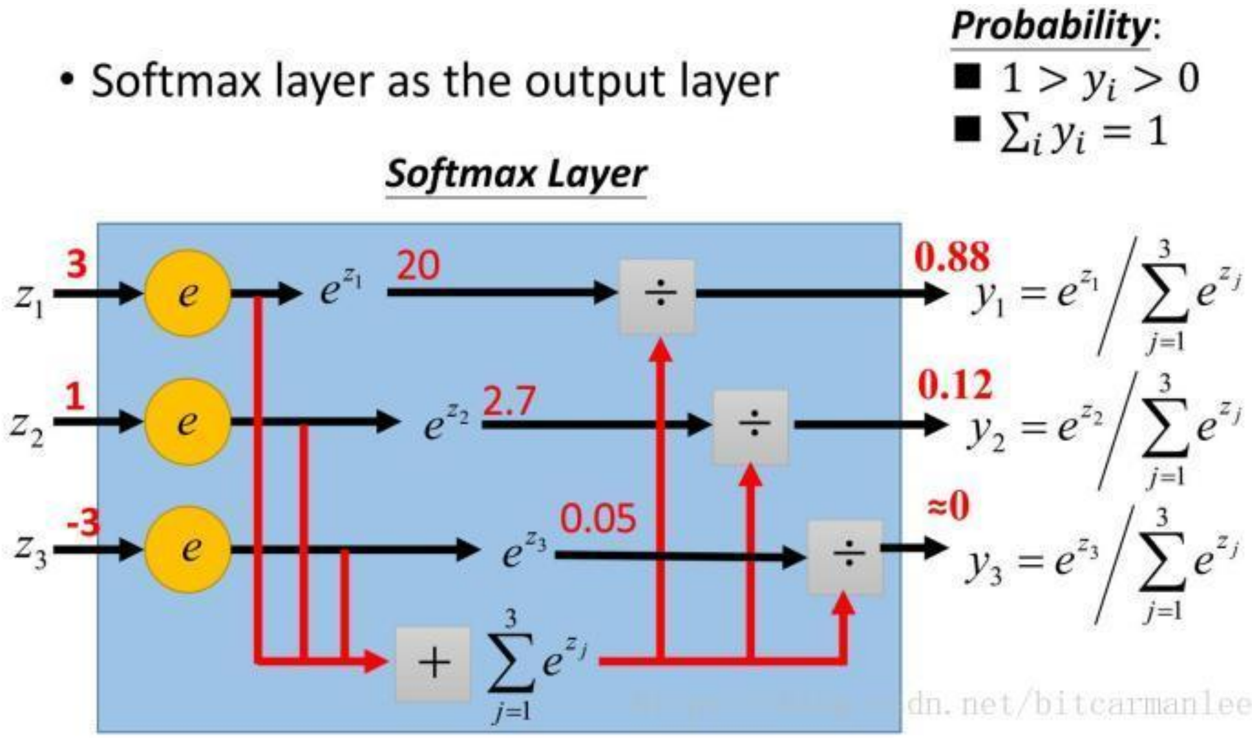


$$h_t = \tanh(Ux_t + Wh_{t-1} + b)$$

$$y_t = \mathrm{softmax}(Vh_t + c)$$



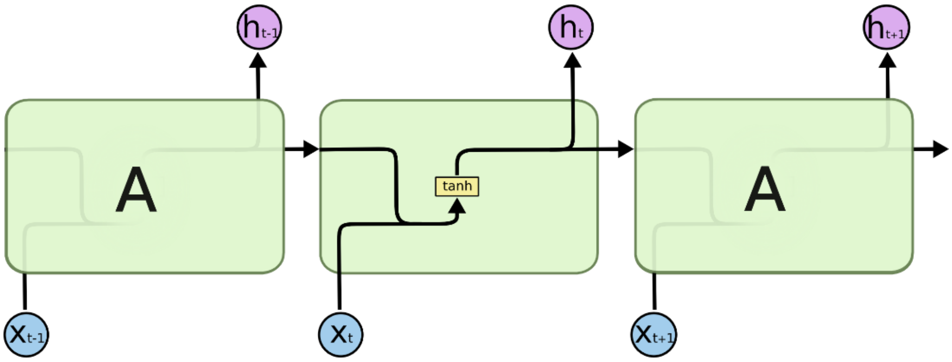
$\tanh(x) = \frac{e^x - e^{-x}}{e^{-x} + e^x}$
1.0
0.5
−0.5
−1.0
−6
−4
−2
2
4
6

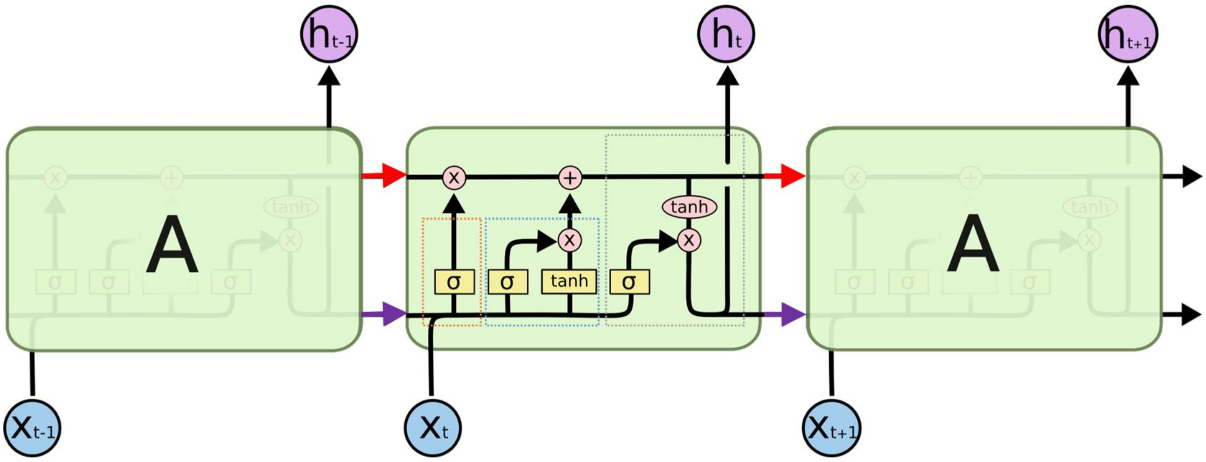


• Softmax layer as the output layer
Probability:
$1 > y_i > 0$
$\sum_i y_i = 1$
Softmax Layer
z_1
3
z_2
1
z_3
-3
e
e^{z_1}
20
e^{z_2}
2.7
e^{z_3}
0.05
÷
+
$\sum_{j=1}^{3} e^{z_j}$
0.88
$y_1 = e^{z_1} \Big/ \sum_{j=1}^{3} e^{z_j}$
0.12
$y_2 = e^{z_2} \Big/ \sum_{j=1}^{3} e^{z_j}$
≈0
$y_3 = e^{z_3} \Big/ \sum_{j=1}^{3} e^{z_j}$
dn.net/bitcarmanlee

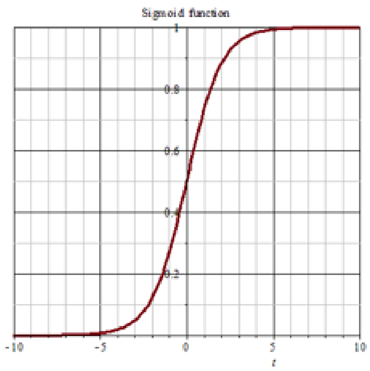
Long Short-Term Memory (LSTM)

- RNN revisited



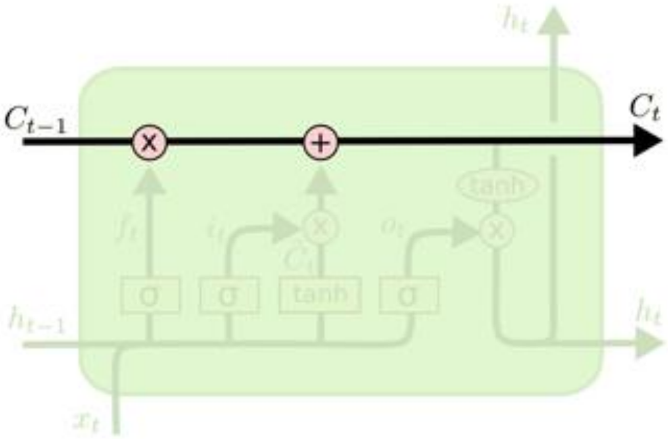


Sigmoid function

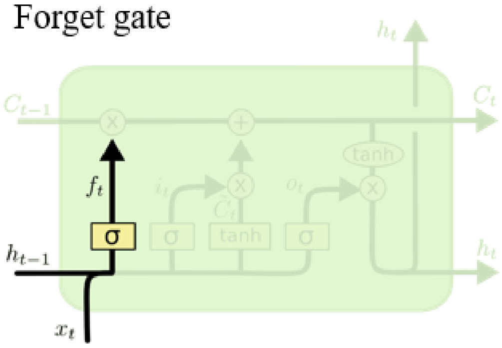


$$S(t) = \frac{1}{1 + e^{-t}}$$

Cell state

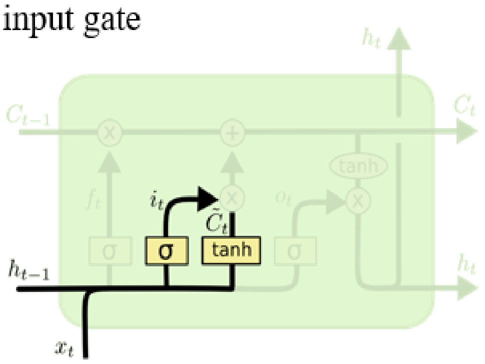


Forget gate



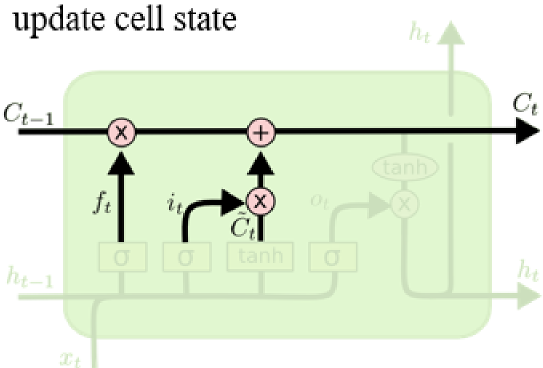
$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \ + \ b_f\right)$$

input gate



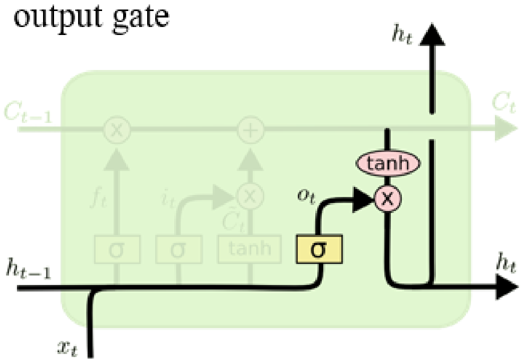
$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \ + \ b_i\right)$$
$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] \ + \ b_C)$$

update cell state



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

output gate



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

Word2vec →word embedding 詞嵌入

- Word2vec 是一群用來產生詞向量的相關模型。
- skip-grams 或連續詞袋（CBOW）來建立神經詞嵌入

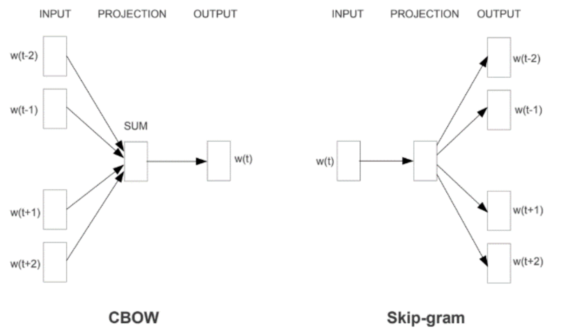
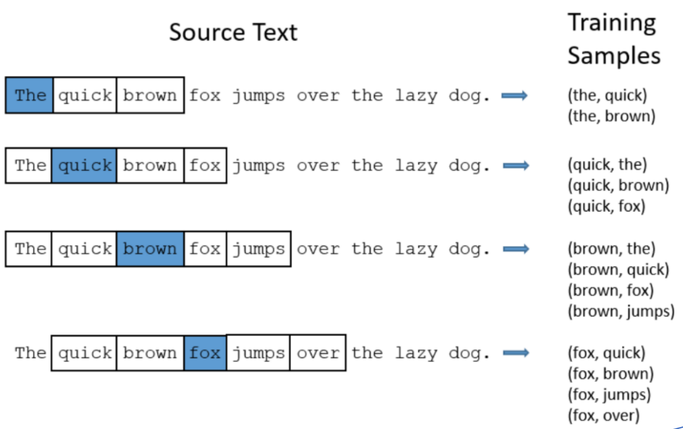
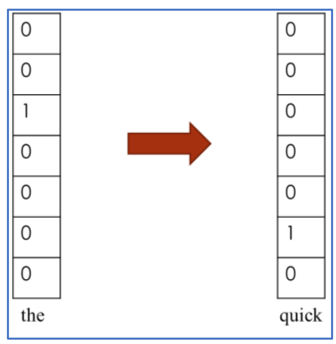


Figure 1: New model architectures. The CBOW architecture predicts the current word based on the context, and the Skip-gram predicts surrounding words given the current word.

- ➔ 分別爲輸入層（input），映射層（projection）和輸出層（output）
- ➔ 輸入層爲詞 W(t)周圍的 n-1 個單詞的詞向量，如果 n 取 5，則詞 W(t)的前兩個詞爲 W(t-2)，W(t-1)，後兩個詞爲 W(t+1)，W(t+2)
- ➔ 它們對應的向量記爲 V(W(t-2))，V(W(t-1))，V(W(t+1))，V(W(t+2))，從輸入層到映射層即將 4 個詞的向量形式相加
- ➔ 而從映射層到輸出層需構造 Huffman 樹，從根節點開始，映射層的值沿着 Huffman 樹進行 logistic 分類，並不斷修正各中間向量與詞向量，得到詞 W(t)所對應的詞向量 V(W(t))。

- Skip-gram example 跳過語法
- ➔ “The quick brown fox jumps over the lazy dog.”
- ➔ Set window size to 2.

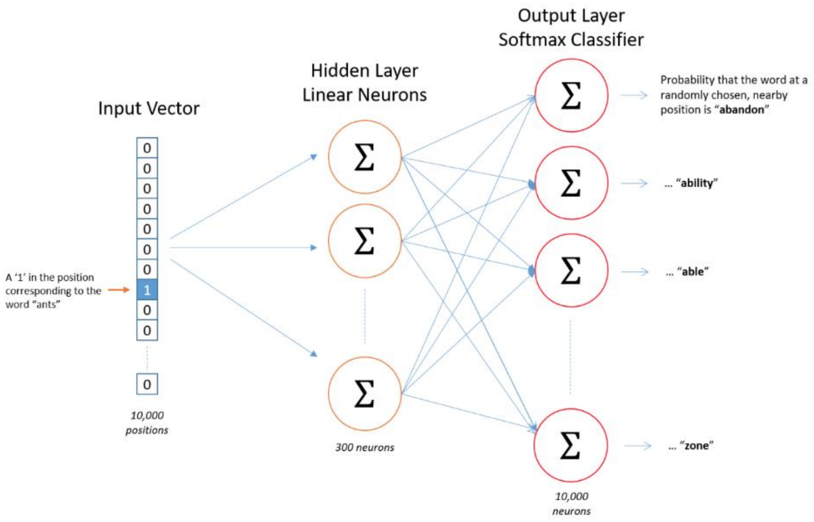




- ➔ One-hot encoding 一次性編碼:

1. 為了改良數字大小沒有意義的問題，將不同的類別分別獨立為一欄
2. 缺點是需要較大的記憶空間與計算時間，且類別數量越多時越嚴重

➔ The training model 訓練模型



➔ Vector 向量

$$
\begin{array}{ccccc} ① & ② & ③ & ④ & ⑤ \\ [0 & 0 & 0 & \boxed{1} & 0] \\ & & & \text{有值} & \end{array}
\times
\begin{array}{c c} & \begin{array}{ccc} ① & ② & ③ \end{array} \\ \begin{array}{c} ① \\ ② \\ ③ \\ ④ \\ ⑤ \end{array} & \begin{bmatrix} 17 & 24 & 1 \\ 23 & 5 & 7 \\ 4 & 6 & 13 \\ \boxed{10} & \boxed{12} & \boxed{19} \\ 11 & 18 & 25 \end{bmatrix} \end{array}
= [10 \quad 12 \quad 19]
$$

➔ 其他:

- GloVe : 基於全局詞頻統計的詞表徵工具，是一種非監督學習算法，用來獲取詞向量表示，對語料進行詞與詞的共現統計聚類，生成一個詞向量空間的線性子結構。
- ELMo : 嵌入式語言模型
- GPT : 由 OpenAI 提出的非常強大的預訓練語言模型
- BERT : 基於變換器的雙向編碼器表示技術是用於自然語言處理的預訓練技術
- XLNet :為 BERT 的升級模型