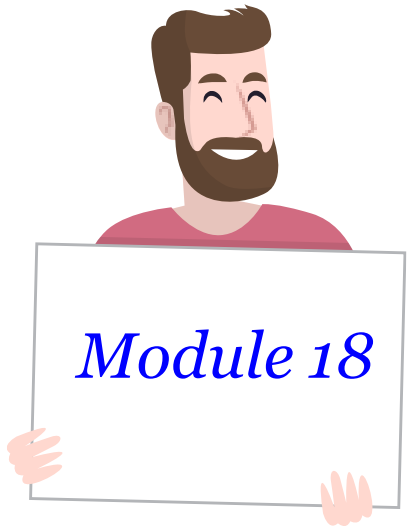
RNN神經網路介紹



designed by freepik

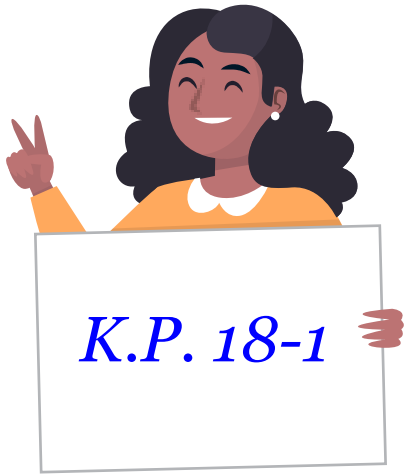
Estimated time:
45 min.

學習目標

- 18-1: 為什麼需要RNN網路
- 18-2: RNN神經網路介紹
- 18-3: RNN文字產生器

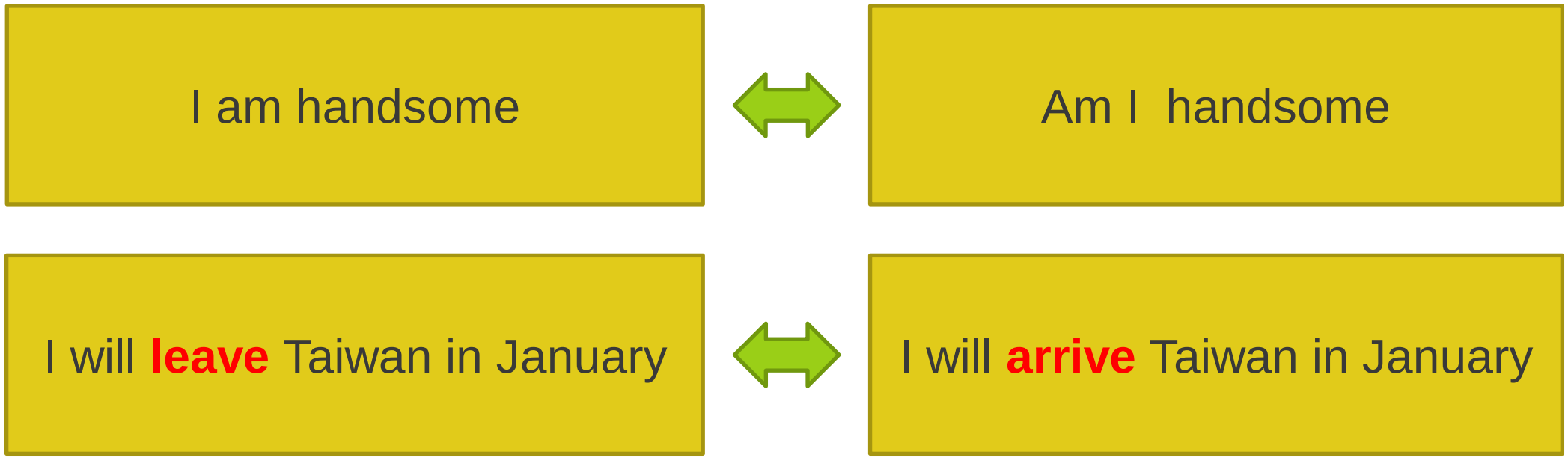
18-1: 為什麼需要RNN網路

- 為什麼需要RNN網路
- RNN初期建構的想法



為什麼需要RNN網路

- 文字的位置對於整句話的意思會有很大的影響
- 稍微更換一個字可能整個句子的意義就不一樣了

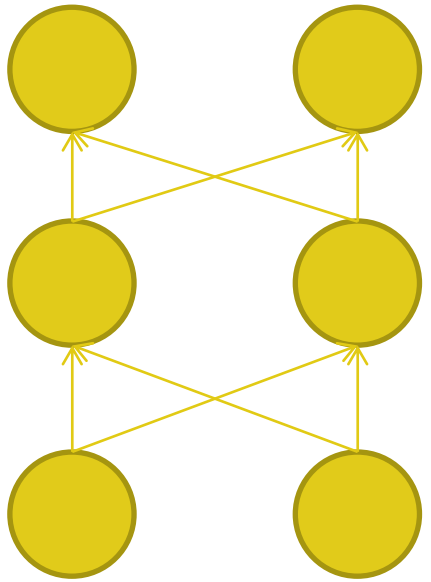


為什麼需要RNN網路

- 傳統神經網路的缺點
  - 缺乏序列的觀念，無法記憶住過去所發生的事情
  - 每次預測都與之前發生的事情獨立，例如當我們把Taiwan丟入網路內的時候，傳統神經網路並無法去記憶前面是leave還是arrive

I will **leave** Taiwan on January

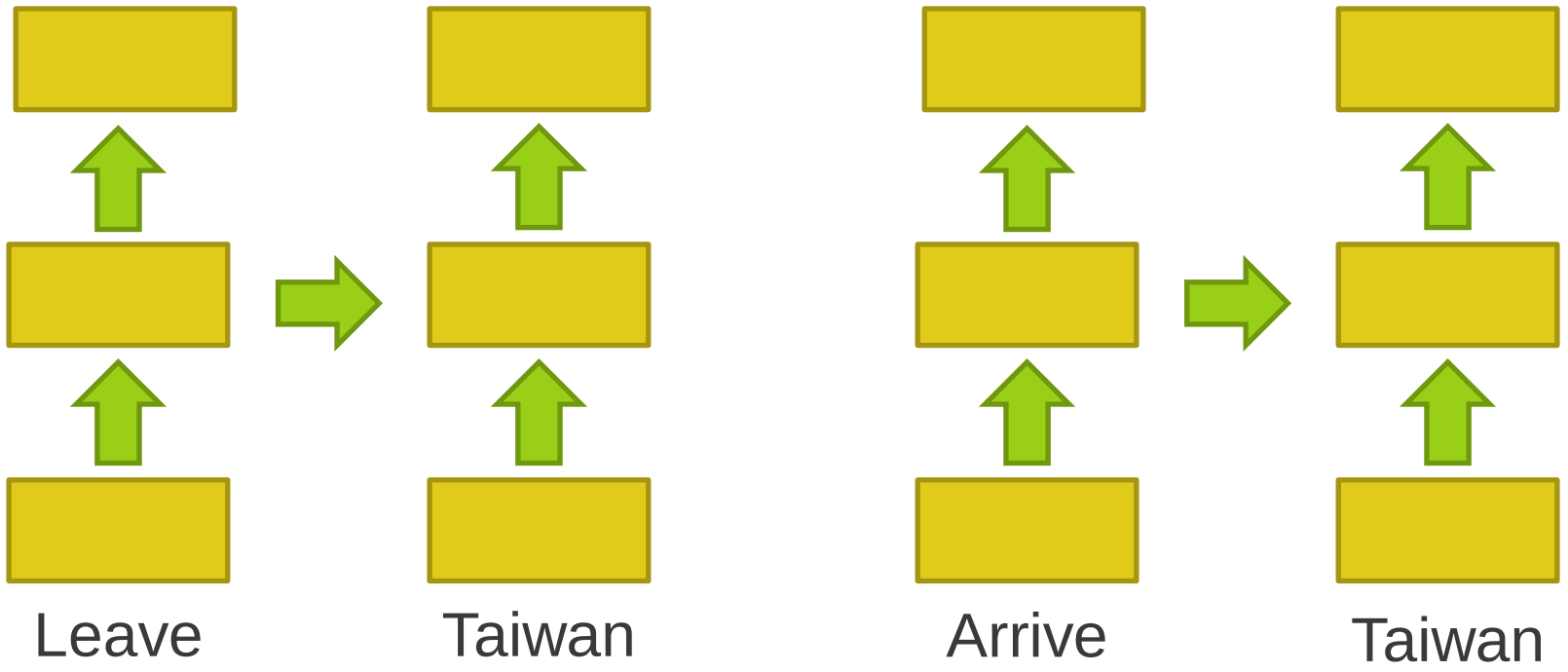
I will **arrive** Taiwan on January



Taiwan

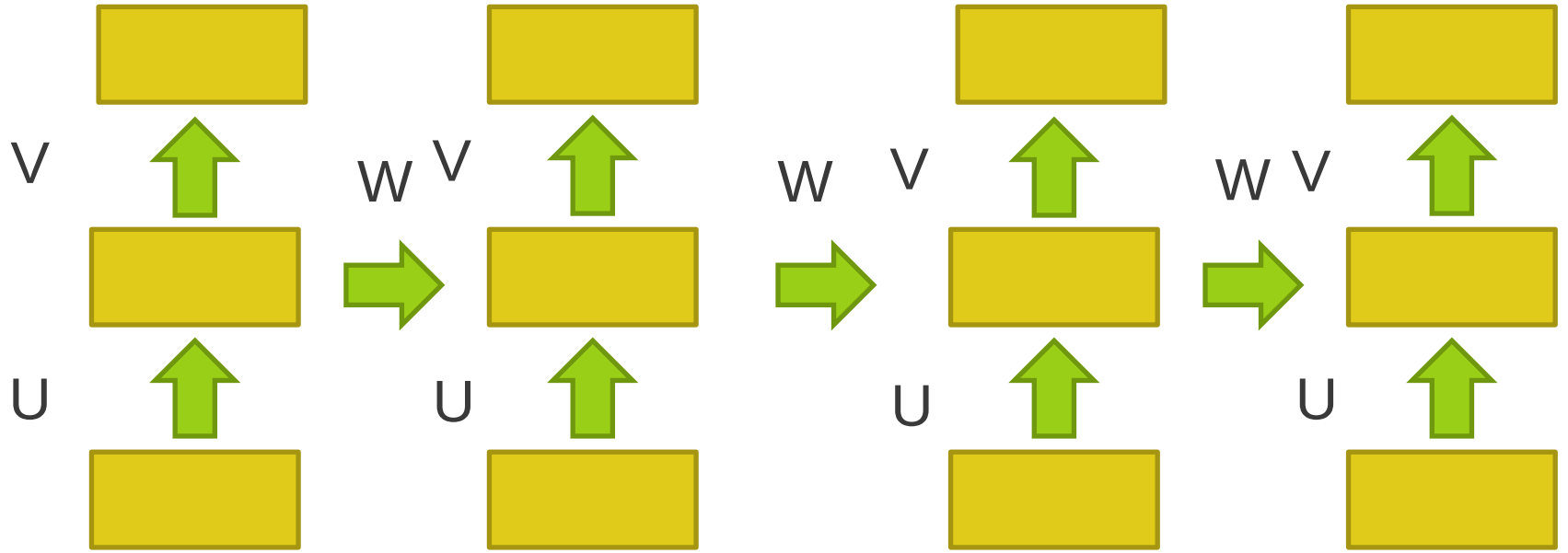
RNN初期建構的想法

- 於是有人提出了或許我們應該把神經網路建構如下
  - 一個神經網路應該有多個分支，這樣把”leave Taiwan”與”arrive Taiwan”丟入網路，網路就能辨識出其意思上的差異



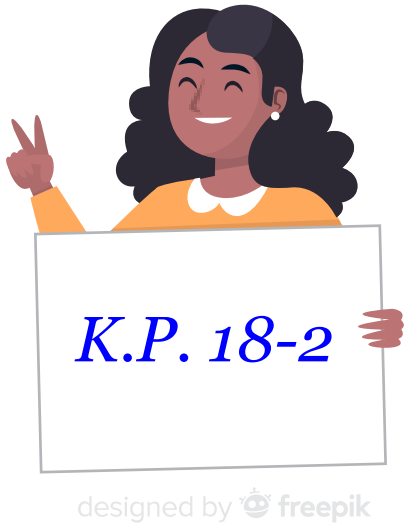
RNN初期建構的想法

- 但是把RNN在時間軸上面串接非常長，參數的增長量會非常快
  - 有人提出我們應該讓位置相對應的邊共享一樣的參數
  - 雖然在時間軸上有不同分支，但總共只有U、V、W三組參數



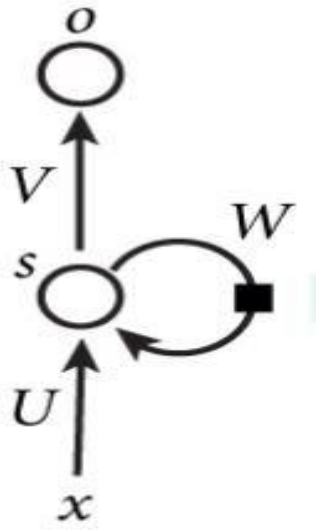
18-2: RNN神經網路介紹

- **RNN神經網路建構**



RNN神經網路建構

- RNN神經網路是一種遞迴式神經網路
  - 可以讓神經網路擁有記憶
  - 在RNN裡面，多了一個叫做狀態state的概念，可以把它想像成神經網路的記憶體

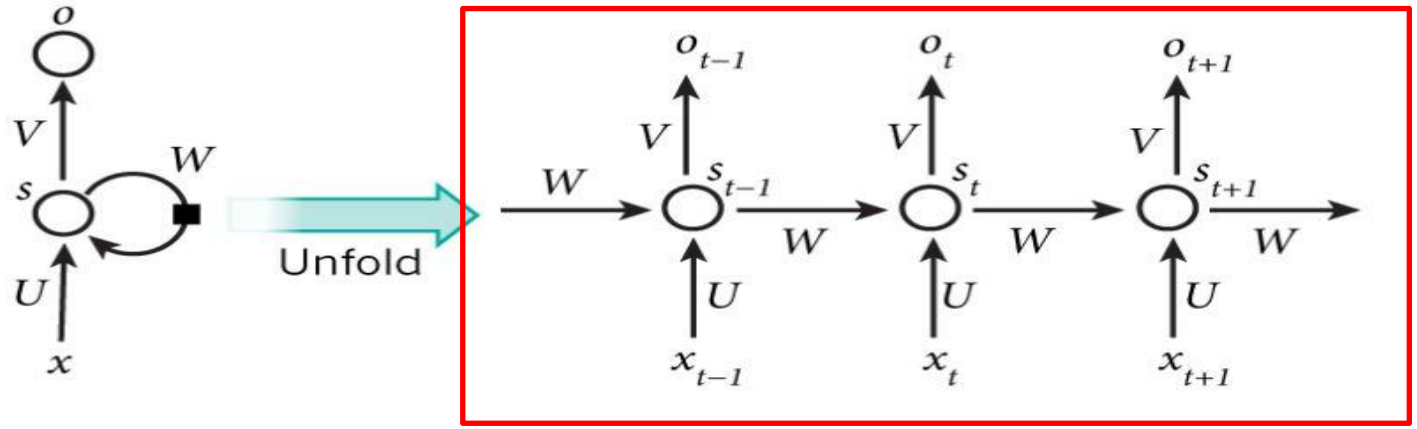


RNN神經網路建構

- RNN神經網路的運算方法如左下，可以發現輸入xt、狀態st-1與st成一個遞迴的式子
  - RNN很常也被表示成展開的樣子，方便我們去理解，如右下圖
  - 特別注意U、V、W是同一組參數

$$s_t = f(Ux_t + Ws_{t-1})$$

$$o_t = \text{softmax}(Vs_t)$$



RNN神經網路建構

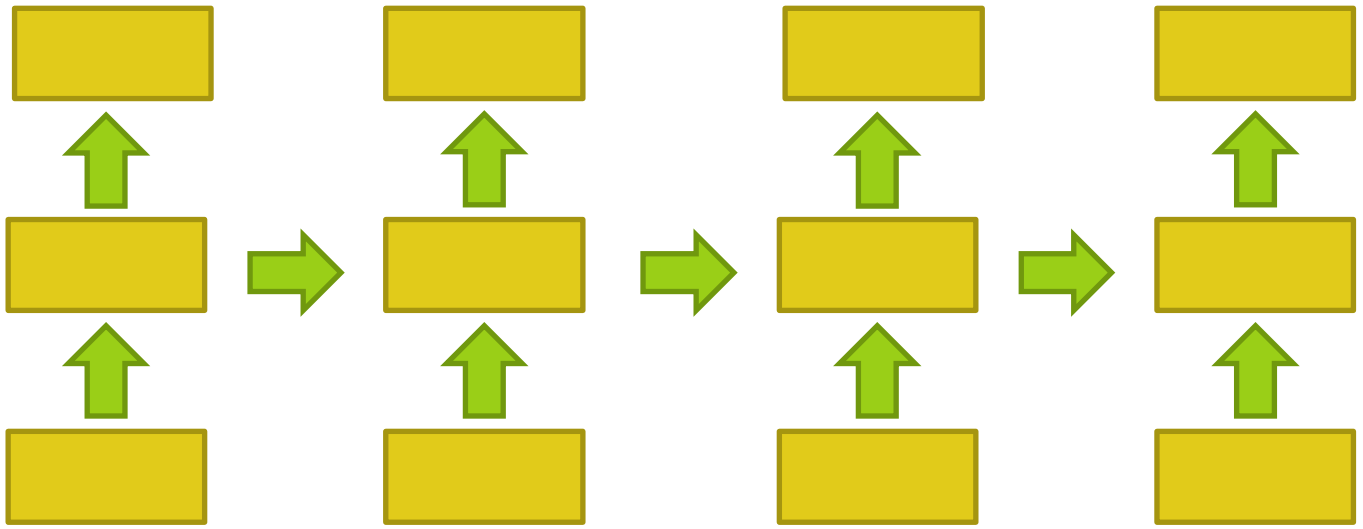
- RNN裡面的s代表是狀態
  - 在RNN裡面，所謂的state size指的是s這個向量的長度
  - 此向量如果越長，代表能記憶住越多東西

$$s_t = f(U x_t + W s_{t-1})$$

$$o_t = \mathrm{softmax}(V s_t)$$

RNN神經網路建構

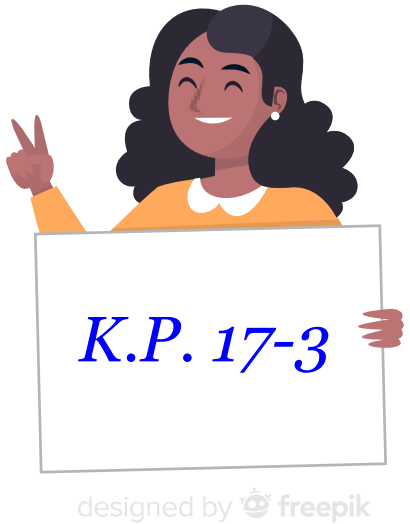
- 如果一個RNN的模型跟四個時間點的輸入有關係
  - 我們稱其為time step為4



Time step = 4

18-3: RNN文字產生器

- **RNN文字產生器**



RNN文字產生器

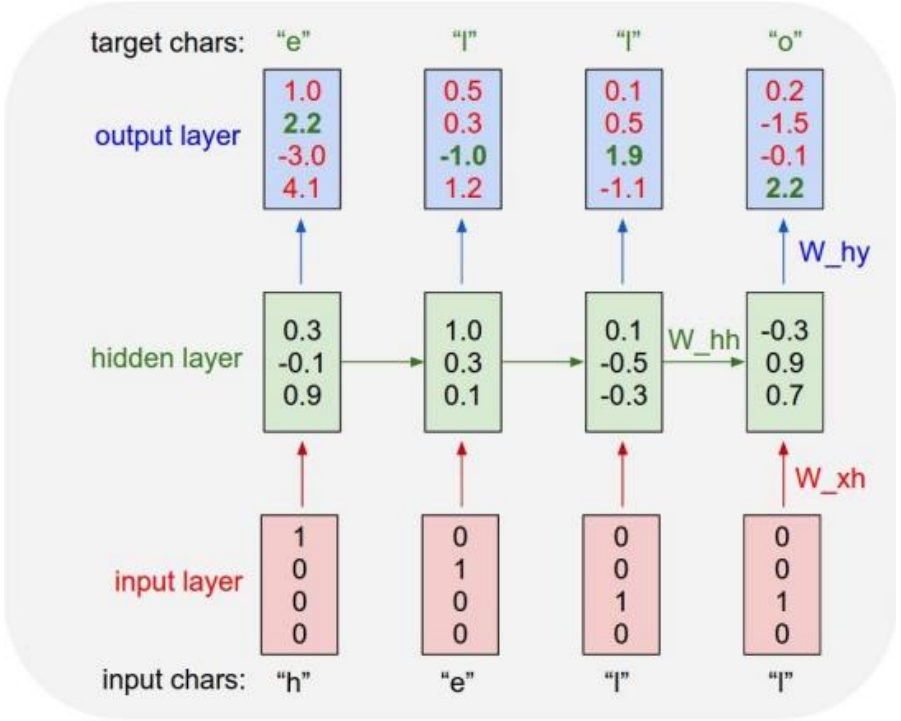
- 假設我們想要讓一個RNN神經網路去學寫小說
  - 一個常見的做法是一個字元一個字元將小說輸入到網路裡面去學習字源之間的機率分布
  - 此種模型叫做字元模型

hello, I am Andy

‘h’ ‘e’ ‘l’ ‘l’ ‘o’ ‘,’ ‘ ’ ‘I’ ‘ ‘ ‘a’ ‘m’ ‘ ‘ ‘A’ ‘n’ ‘d’ ‘y’

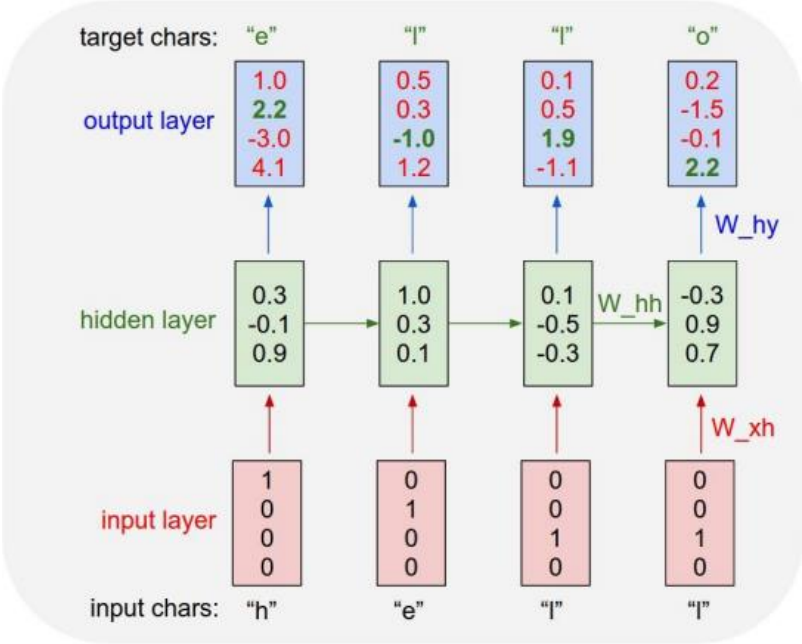
RNN文字產生器

- 我們可以將RNN網路建構如下，並將”hello, I am Isaac”前幾個字元輸入



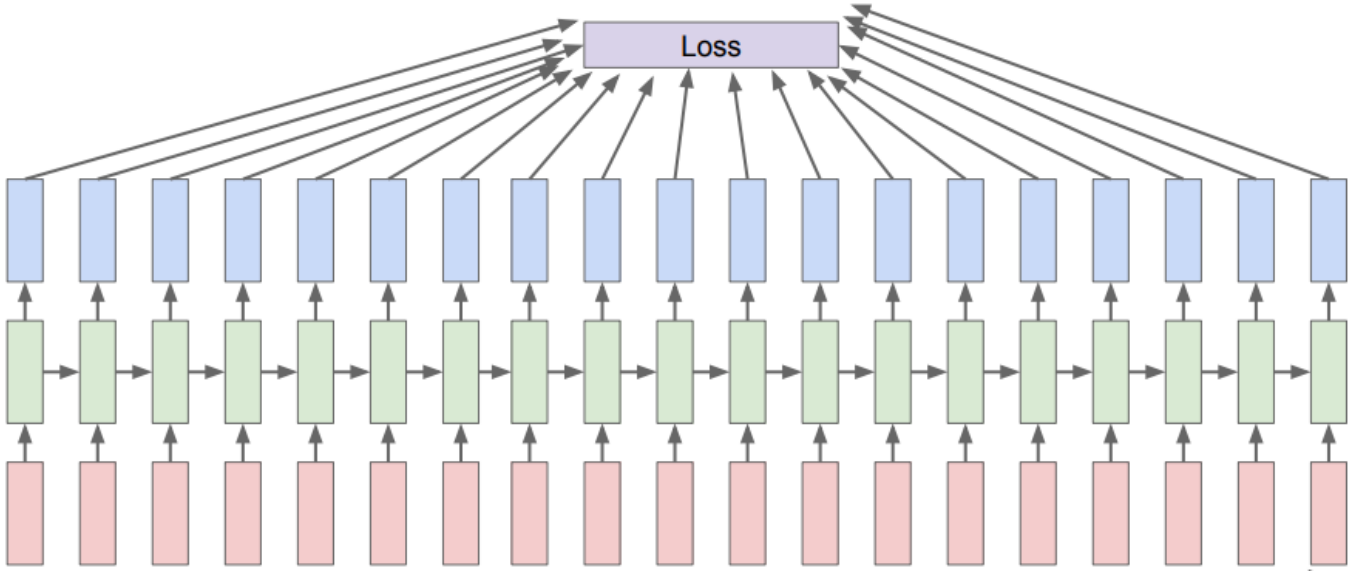
RNN文字產生器

- 我們的目標是讓RNN可以學習到字元之間的機率分布，因此，每個時間點都會有自己輸入的字元以及每個時間點期望的字元
  - 例如第一個時間點輸入以及期望輸出為(“h”, “e”)、第二個時間點輸入以及期望輸出為(“e”, “l”)以此類推



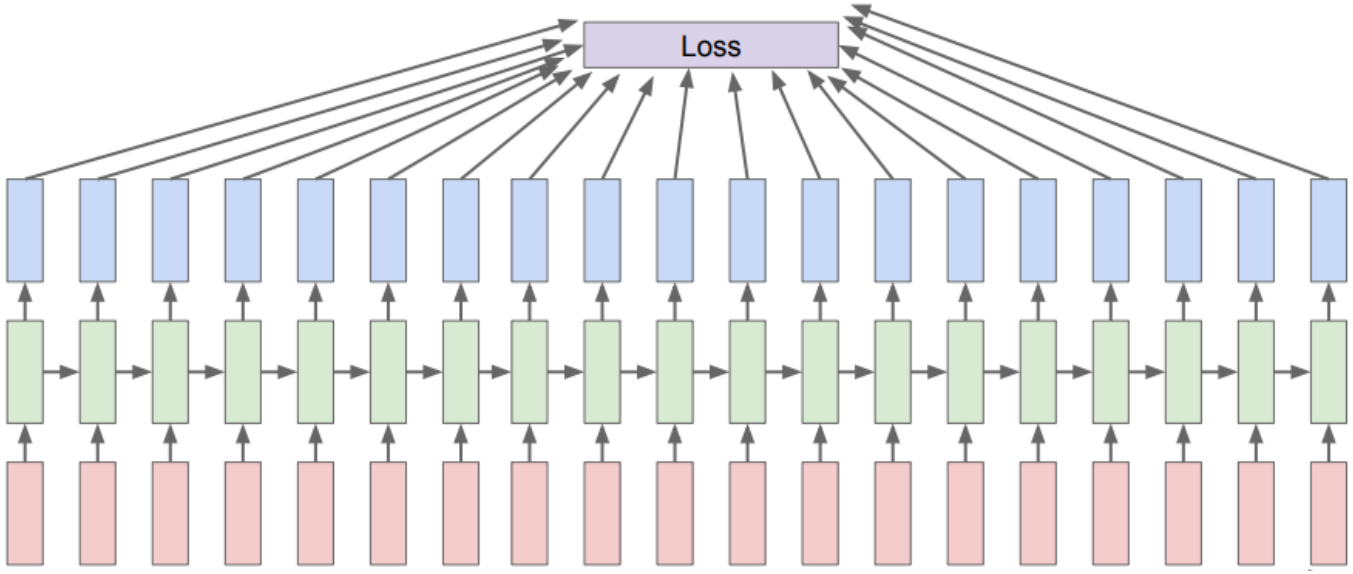
RNN文字產生器

- 每個時間點我們會預測出一個預測向量，同時每個時間點也會有對應的期望向量，我們可以把不同時間點這兩個向量的差值做總和，來當作總體損失函數
  - 這個觀念跟DNN、CNN一樣，只是在RNN裡面有多個時間點



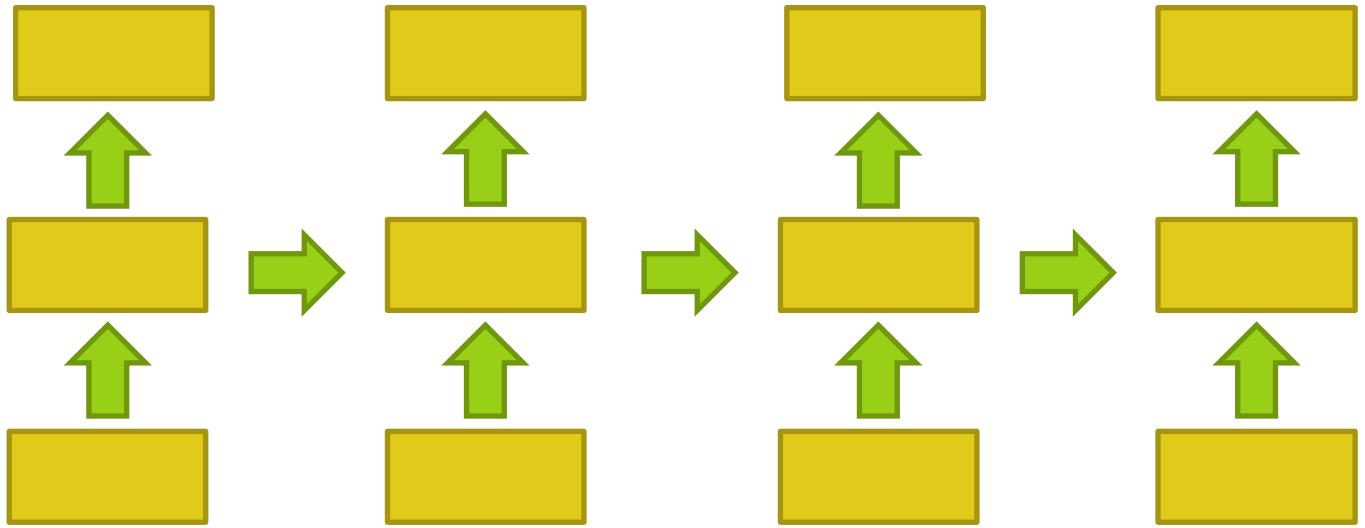
RNN文字產生器

- 當定義好總損失函數後，我們就可以使用優化去去調整RNN網路裡面的參數(U、V、W)
  - 直到每個時間點預測向量跟期望向量很接近



RNN文字產生器

- 當優化完RNN後，我們變得到字元之間的機率分布，假設我們想要使用這個網路產生一個以英文字母”P”開頭的文章，我們可以在第一個時間軸輸入”P”，讓RNN不斷去預測下個時間點的字元



“P”

RNN文字產生器

- 這個是讓RNN去閱讀莎士比亞小說並使用RNN產生的一個結果

```
PANDARUS:
Alas, I think he shall be come approached and the day
When little srain would be attain'd into being never fed,
And who is but a chain and subjects of his death,
I should not sleep.

Second Senator:
They are away this miseries, produced upon my soul,
Breaking and strongly should be buried, when I perish
The earth and thoughts of many states.

DUKE VINCENTIO:
Well, your wit is in the care of side and that.

Second Lord:
They would be ruled after this chamber, and
my fair nues begun out of the fact, to be conveyed,
Whose noble souls I'll have the heart of the wars.

Clown:
Come, sir, I will make did behold your worship.

VIOLA:
I'll drink it.
```

Demo 18-3

- TensorFlow unstack使用
- TensorFlow stack使用
- RNN實作MNIST分類

線上Corelab

- 題目1： unstack數據組
  - 給予一個數據組，針對軸0取unstack
- 題目2： stack數據組
  - 給予多個數據組，針對軸0以及軸1分別取stack
- 題目3：RNN實作MNIST分類
  - 完成RNN cell程式碼做MNIST分類

本章重點精華回顧

- 為什麼需要RNN
- RNN神經網路
- 用RNN產生文章

Lab:RNN MNIST分類

- **Lab01: TensorFlow unstack使用**
- **Lab02: TensorFlow stack使用**
- **Lab03: RNN實作MNIST分類**

Estimated time:
20 minutes


THANK YOU