專題

什麼是深度學習

機器學習的卷土重來

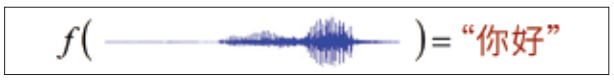
作者：**李宏毅**

李宏毅為臺灣大學電機工程系暨資訊網路與多媒體研究所助理教授，主要研究領域：機器學習、深度學習、語意理解、語音辨識。

自從 2016 年 3 月 AlphaGo 以四比一擊敗李世乭以後，AlphaGo 所使用的深度學習技術引起了各界的關注。事實上，早在 AlphaGo 問世之前，深度學習技術即已廣泛應用在各個領域。當我們對 iPhone 的語音助理軟體 Siri 說一句話，Siri 可以將聲音訊號辨識成文字，用的就是深度學習的技術；當我們上傳一張相片到 Facebook，Facebook 可以自動找出相片中的人臉，用的也是深度學習的技術。其實人們早已享受深度學習所帶來的便利很長一段時間了。

什麼是深度學習？

深度學習是機器學習的一種方法，「機器學習技術，就是讓機器可以自我學習的技術。」但實際上機器是如何學習的呢？一言以蔽之，機器學習就是讓機器根據一些訓練資料，自動找出有用的函數（function）。例如，如果將機器學習技術運用在語音辨識系統，就是要機器根據一堆聲音訊號和其對應的文字，找出如下的「語音辨識函數」：



輸入一段聲音訊號，輸出就是該聲音訊號所對應的文字。

如果機器學習技術應用在影像辨識系統，那就是要機器根據一堆圖片和圖片中物件名稱的標註，找出「影像辨識函數」：

（攝影：ilker）

輸入一張圖片，輸出是圖片中的物件名稱。

如果要機器下圍棋，就是讓機器根據一堆棋譜找出「下圍棋的函數」：

（圖片來自維基）

輸入是棋盤上所有黑子和白子的位置，輸出是下一步應該落子的位置。

以上要找的函數，共通點是它們都複雜到人類絕對沒有能力寫出它們的數學式，只有靠機器才有辦法找出來。那麼，機器要如何根據訓練資料找到函數呢？

一般機器學習方法要經過三個步驟：一、人類提供給機器一個由函數構成的集合（簡稱函數集）；二、人類根據訓練資料定義函數的優劣；三、機器自動從函數集內找出最佳的函數。深度學習也不例外。底下我們先介紹深度學習的基本架構，再分別就這三個步驟來介紹深度學習。

類神經網絡與神經元

「深度學習就是讓機器模擬人腦的運作方式，進而和人類一樣具備學習的能力。」這個科普的說

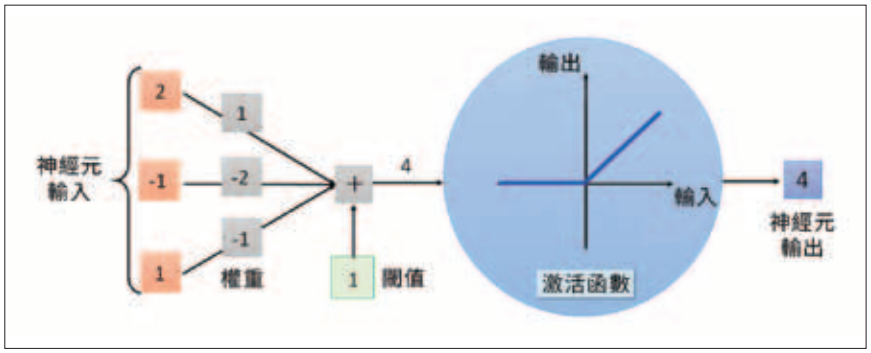


圖 1 類神經網絡中單一神經元及其運作方式。

法，相信大家都已耳熟能詳。會有這樣的說法，是因為深度學習中，人類提供的函數集是由類神經網絡（artificial neural network）的結構所定義。

類神經網絡和人腦確實有幾分相似之處，我們都知道人腦是由神經元（neuron）所構成，類神經網絡也是由「神經元」連接而成。類神經網絡中的神經元構造及其運作方式如圖 1 所示。每個神經元都是一個簡單的函數，這些函數的輸入是一組數值（也就是一個向量），輸出是一個數值。以圖 1 的神經元為例，該神經元的輸入為左側橘色框內的 2、-1、1 三個數值，輸出為右側藍色框內的數值 4。

那麼，神經元是如何運作的呢？每個輸入都有一個對應的權重（weight），圖 1 中每個輸入對應的權重，分別為灰色框內的 1、-2、-1 三個數值。先將每個輸入數值和其對應權重相乘後加總，再加上綠色框內的閾值（bias）後，其總和便成為神經元中激活函數（activation function）的輸入。圖 1 中激活函數的輸入是 4 ，也就是

$$2 \times 1 + (-1) \times (-2) + 1 \times (-1) + 1 = 4$$

激活函數是由人類事先定義好的非線性函數，其輸入和輸出都是一個數值，而其輸出就是神經元的輸出。圖 1 中的激活函數，其輸入和輸出的關係如藍色圓域所示（橫軸代表輸入、縱軸代表輸出），其中當輸入小於 0 時，輸出為 0；當輸入大於 0 時，輸出等於輸入。這種激活函數稱為整流線性單元（rectified linear unit，ReLU），是目前常用的激活函數。圖 1 中的激活函數輸入為 4，因為大於 0，故神經元的輸出就是 4。神經元中的權重和閾值都稱為參數（parameter），它們決定了神經元的運作方式。

步驟一：類神經網絡就是函數集

了解神經元後，接著來看類神經網絡。類神經網絡由很多神經元連接而成，人類只需要決定類神經網絡的連結方式，機器可以自己根據訓練資料找出每個神經元的參數。圖 2 是一個類神經網絡的例子，上方的類神經網絡共有六個神經元，分別排成三排，橘色方塊代表外界的輸入，外界的輸入可以是圖片、聲音訊號或棋盤上棋子的位置等等，只要能以向量（一組數字）表示即可。以圖片為例，一張 28×28 大小的黑白圖片，可以視為一個 784（=28×28）維的向量，每一分量對應到圖片中的一個像素，該分量的值表示像素顏色的深淺，接近黑色其值就接近 1，反之就接近 0。

來自外界的資訊被輸入給第一排的藍色神經元，藍色神經元的輸出是第二排黃色神經元的輸入，黃色神經元的輸出則是下一排綠色神經元的輸入，因為綠色神經元的輸出沒有再轉給其他神經元，故其

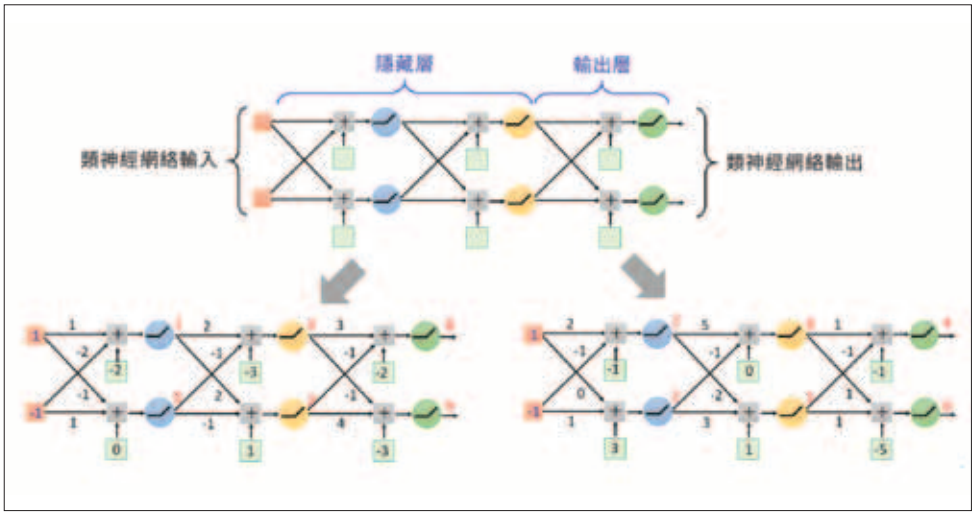


圖 2 圖上方為一完全連接前饋式網絡結構，下方為兩組不同的參數示例，分別代表兩個不同的函數。輸入同樣的數值，左下和右下的神經網絡會有不同的輸出。

輸出就是整個類神經網絡的輸出。

圖 2 這種類神經網絡結構，稱為完全連接前饋式網絡（fully connected feedforward network）。在這種架構中，神經元排成一排一排，每排稱為一「層」（layer），每層神經元的輸出為下一層各神經元的輸入，最後一層稱為「輸出層」（output layer），其他層則稱之為「隱藏層」（hidden layer）。所謂深度學習的「深」，意味著有很多的隱藏層（至於要多少隱藏層才足以稱之為「深」就見仁見智了）。

當類神經網絡中每個神經元參數都確定時，該類神經網絡就是一個非常複雜的函數。同樣的網絡結構，若參數不同將形成不同的函數。例如：圖 2 左下圖和右下圖分別具備兩組不同的參數，當參數如左下圖，若輸入 1 和 -1，這個類神經網絡所定義的函數輸出為 0 和 9；同理，以右下圖的參數可得到另一個函數，如果輸入一樣的 1 和 -1，這個函數的輸出則為 4 和 6 ❶。因為一組參數等於一個函數，如果先把神經元連接確定，再讓機器根據訓練資料自己找出參數時，就相當於先提供一個函數集，再讓機器從函數集自行選出有用的函數。

類神經網絡的結構，目前仍需要人類在機器學習前事先決定。如果結構設定不當，由此結構所定義的函數集根本找不到好的函數，接下來再怎麼努力都是徒勞。這就好比想要在大海撈針，結果針根本不在海裡。

不同任務適用的類神經網絡結構並不相同，例如捲積式類神經網絡（convolutional neural network, CNN）這種特殊結構，特別適合做影像處理（有關捲積式類神經網絡，請見本期〈關於深度學習網絡的兩個問題〉）。捲積式類神經網絡也是 AlphaGo 的架構，但 AlphaGo 中的捲積式類神經網絡和前者略有不同，沒有最大池化（max pooling）這個影像處理的常用結構，這應該是考慮圍棋特性而刻意設計的。

目前，雖然有些技術已能讓類神經網絡自動決定結構，但這些技術尚未有太多成功的應用。想要決定網絡結構，通常還是必須仰賴深度學習技術使用者的經驗、直覺，以及嘗試錯誤，其中的「運用之妙，存乎一心」，這裡無法詳談。

步驟二：定義函數的優劣

有了類神經網絡結構作為函數集後，接下來就要定義函數的優劣，以便在下一步驟讓機器挑選出最佳函數，但什麼叫作好的函數呢？

讓我們舉個具體的例子來說明。假設任務是手寫數字辨識，也就是讓機器檢視圖片，每張圖中有手寫數字，機器必須辨認出這個數字 ❷。執行手寫數字辨識的類神經網絡，其輸出層有 10 個神經元，分別對應到從 0 到 9 的數碼，而每個神經元的輸出值則分別表示對應數碼的信心分數，機器會把信心分數最高的數碼當作最終的輸出。

在手寫數字辨識中，訓練資料是一堆圖片和每張圖片中的數碼（數碼需由人工標註）。這些訓練資料告訴機器，當訓練資料中某張圖片輸入時，要辨識出哪個數碼才正確。如果將訓練資料中的圖片一張一張輸入到某些函數來辨識，越能夠正確辨識結果（亦即在輸出層中圖片數碼的信心分數最高）的函數，可能就越優秀。當然也可能發生某函數能準確辨識訓練資料，可是輸入新圖片時卻發生辨識錯誤的情況 ❸。因此函數優劣的定義本身是很大的學問，只考慮訓練資料的正確率是不夠的，此處不深入細談。

步驟三：找出最佳函數

根據訓練資料可以決定一個函數的優劣，接下來就要從類神經網絡所定義的函數集中，找到最佳的函數，也就是最佳的參數組合。我們可以把找出最佳函數的過程，想像成是機器在「學習」。

如何找出最佳的類神經網絡參數呢？最無腦的方法就是暴力窮舉類神經網絡中的所有可能參數值。然而，為了讓機器完成複雜的任務，現在的類神經網絡往往非常龐大，其中可能包括數十個隱藏層，每個隱藏層有上千個神經元，因此參數數目可能動輒千萬以上。要窮舉這上千萬的參數組合，再根據訓練資料計算每個參數組合（函數）的優劣來找出最佳函數，就算是電腦也辦不到，因此暴力窮舉法是不切實際的。

目前最常採用的「學習」方法稱為「梯度下降法」（gradient descent）。在此法中，機器先隨機指定第一組參數，再稍為調整第一組參數，找出比第一組參數略佳的第二組參數，接下來再稍為調整第二組參數，找出比第二組參數略佳的第三組參數，以此類推，讓這個步驟反覆進行，直到找不出更佳的參數時就停止。參數的調整次數可能多達上萬次，這就是經常聽說「深度學習需要耗用大量運算資源」的原因。

因為類神經網絡中有大量參數，所以還會使用名為「反向傳遞法」（backpropagation）的演算法，來提高參數調整的效率。梯度下降法和反向傳遞法沒有什麼高深的數學，理論上只要高三以上的理組學生就有能力理解這個方法，有興趣深入研究深度學習的讀者，可參考尼爾森（Michael Nielsen）線上教科書的第一、二章 [1]。

雖然用梯度下降法可以訓練出厲害的 AlphaGo，但這個方法本身其實不怎麼厲害。它無法保證一定能從函數集中挑出最佳函數，而僅能從類神經網絡定義的函數集中，找出局部最好的函數。更糟的是，這個演算法有隨機性，每次找出來的函數可能都不一樣，因此能找出多好的函數要靠點運氣。也因此，在深度學習的領域裡，充斥各種可以幫助梯度下降法找到較佳函數的「撇步」，但至今尚未有任何撇步可以保證能找到最佳函數。如何讓機器自動有效挑選出最佳函數，仍是個尚未克服的挑戰。

---

❶ 圖 2 左下圖和右下圖的紅色數值代表輸入 1 和 -1 時每個神經元的輸出值，你可以自己根據圖中給定的參數算算看，如果算出來跟圖 2 的結果一樣，那你就了解類神經網絡的運作方式了。

❷ 這個系統可用於辨認信封上的手寫郵遞區號，讓機器自動進行郵件分類。

❸ 這就像學生把課本習題解法死背硬記下來，考試時卻完全無法活用。

深度學習是新的突破嗎？

其實，上述的類神經網絡技術在 1980 年代前就已經發展成熟，只是當時並不用「深度學習」這個詞彙，而稱為「多層次感知器」（multi-layer perception, MLP）。許多人把深度學習近年的走紅，歸功於辛騰（Geoffrey Hinton）在 2006 年提出，以限制波茲曼機（Restricted Boltzmann Machine, RBM）初始化參數的方法 [2]。因此曾有一度，深度學習和多層次感知器的差異在於，隨機初始化參數的叫作多層次感知器；用限制波茲曼機初始化的稱為深度學習。不過實際上，只要給予適當的激活函數、足夠的訓練資料，限制波茲曼機法所帶來的幫助並不顯著 [3]，故此方法已不能算是深度學習的同義詞。

今日深度學習所應用的類神經網絡，和 1980 年代的多層次感知器雖然本質上非常相似，但還是有些不同。首先，80 年代的網絡通常不超過三層，但現在的網絡往往比三層深得多，例如語音辨識需要七、八層，影像辨識需要 20 餘層，微軟用來辨識影像的「深度殘留網絡」（deep residual network），甚至深達 152 層 [4]。另一個不同是，過去比較流行用 S 型函數（sigmoid function）作為激活函數，但 S 型函數在網絡很深時，難以訓練出好的結果，故近年較流行用前述整流線性單元 [5]，在訓練很深的網絡時，整流線性單元的效果遠優於 S 型函數。此外，現在還有技術可以讓機器自己決定激活函數 [6]。

另外，雖然訓練還是以梯度下降法為主，卻有一些新的訓練技巧，例如：Adam 演算法可以減少訓練時參數更新的次數，加速網絡提早完成訓練 [7]；Dropout 演算法在訓練時隨機丟掉一些神經元，可讓網絡在遇到沒看過的資料時表現得更好 [8]。

硬體的發展同樣也不容忽視，例如以圖形處理器（graphics processing unit, GPU）來加速矩陣運算，使得訓練時間大幅縮短，可以多次嘗試錯誤，縮短開發週期，也是加速深度學習這個領域發展的原因之一。

為什麼需要「深」？

為什麼類神經網絡需要很多層神經元？簡單的答案是，深層類神經網絡比淺層厲害。例如在實務上，較深的類神經網絡可得到較低的語音辨識錯誤率 [10]。但是這個論述的說服力夠嗎？

如果深層類神經網絡較淺層效能好的原因，來自神經元數量的增加，而跟多層次的結構沒有關係，那麼，把所有神經元放在同一層，會不會效果跟深層網絡一樣好？甚至已有理論保證，單一隱藏層的淺層類神經網絡，只要神經元夠多，就可以描述任何函數（有興趣的讀者可參考 [1] 第四章深入淺出的說明）。於是把神經元排成多層的理由似乎更削弱了，甚至有人懷疑把神經元排成多層的深度學習只是噱頭。

有趣的是，雖然淺層類神經網絡可以表示和深層類神經網絡一樣的函數，但淺層網絡卻需要更多神經元才能描述一樣的函數。為什麼會這樣呢？打個比方，如果把網絡的輸入比擬為原料、輸出比擬為產品，那麼類神經網絡就是一條生產線，每個神經元是一位工作人員，把來自前一站的半成品加工，再送往下一站。假設現在輸入的是圖片，輸出是圖

片中的物件，第一層神經元的工作可能是偵測圖片中直線、橫線的位置 ❹，將結果交給第二層，第二層神經元根據第一層的輸出，判斷圖片中有沒有出現圓形、正方形等幾何圖案，再將結果交給第三層……直到最後一層的神經元輸出結果。眾所周知，生產線可以提高生產效率，因此很深的類神經網絡好比一條分工很多的生產線，可以比淺層網絡更有效率 ❺。

同樣的函數，淺層網絡需要較多神經元，也意味著需要較多的訓練資料。這表示淺層網絡想達成深層網絡的效能，需要更多的訓練資料。也就是說，如果固定訓練資料量，深層網絡將有淺層網絡難以企及的效能。

是的，你沒看錯，要達成同樣的任務，**深層網絡需要的訓練資料比淺層網絡少**。一般人聽到「深」這個字眼，往往聯想到需要很「多」資料，但事實剛好相反。我們常聽到「人工智慧」=「大數據」+「深度學習」的說法，但這並不表示「因為大數據，所以深度學習才比其他機器學習法成功」。事實是大數據可以讓所有機器學習方法都獲得提升，不獨厚深度學習。相反的，正是因為數據永遠都嫌少，所以才需要深度學習。

類神經網絡需要記憶力

閱讀到這裡，讀者還記得前面圖 2 介紹的前饋式類神經網絡嗎？想必你還記得，因為人腦有記憶的能力，但前饋式類神經網絡卻沒有。對它而言，每次輸入都是獨立的，它並不記得曾經有過什麼樣的輸入。這樣的類神經網絡無法像人類一樣處理非常複雜的問題，例如語音辨識和語言理解。

假設你想訓練一個智慧型訂票系統，讓使用者以自然語言（以英文為例）作為輸入訂票。當使用者 A 輸入「from Taipei to Kaohsiung」時，機器要自動標出每個字詞的類別，例如：Taipei 是「出發地」，Kaohsiung 是「目的地」，而 from 和 to 是「其他」等，以進行後續訂票。

前饋式類神經網絡可以用來處理這個問題，只要這個網絡做到輸入 Taipei ❻ 就輸出「出發地」❼，輸入 Kaohsiung 就輸出「目的地」，好像就解決了上述問題。但事實不然。假設另一名使用者 B 輸入「from Kaohsiung to Taipei」，人類可輕易根據文句判斷，Taipei 是 A 的「出發地」，卻是 B 的「目的地」。但對這個前饋式網絡，輸入一樣，輸出就一樣，它無法判斷這兩個 Taipei 的不同。要讓神經網絡可以根據文句脈絡判斷文義的不同，需要有記憶的神經網絡。

遞迴式類神經網絡（recurrent neural network, RNN）是一種有記憶力的類神經網絡，其架構與運作方式如圖 3 所示。在遞迴式類神經網絡中有一

❹ 實際上每個神經元會做的事，取決於其參數，是在上述第三步驟透過學習自動習得的。

❺ 有電機背景的讀者可從「邏輯電路」來思考。在邏輯電路中，兩層邏輯閘可以描述任何布林函數，理論上可以只用兩層邏輯製造電腦。但這實際上不可行，因為這樣的電路需要太多邏輯閘，因此複雜電路需要使用多層邏輯閘。

❻ 為了容許語句成為類神經網絡的輸入，需要想辦法將其以向量表示。此處是將語句依序分拆成字詞輸入，再用下文提到的詞向量，將每個詞表示為向量，細節在此不詳述。

❼ 類神經網絡輸出層的每個神經元分別對應到一個類別，如果對應「出發地」的神經元信心分數最高，輸出就是「出發地」。

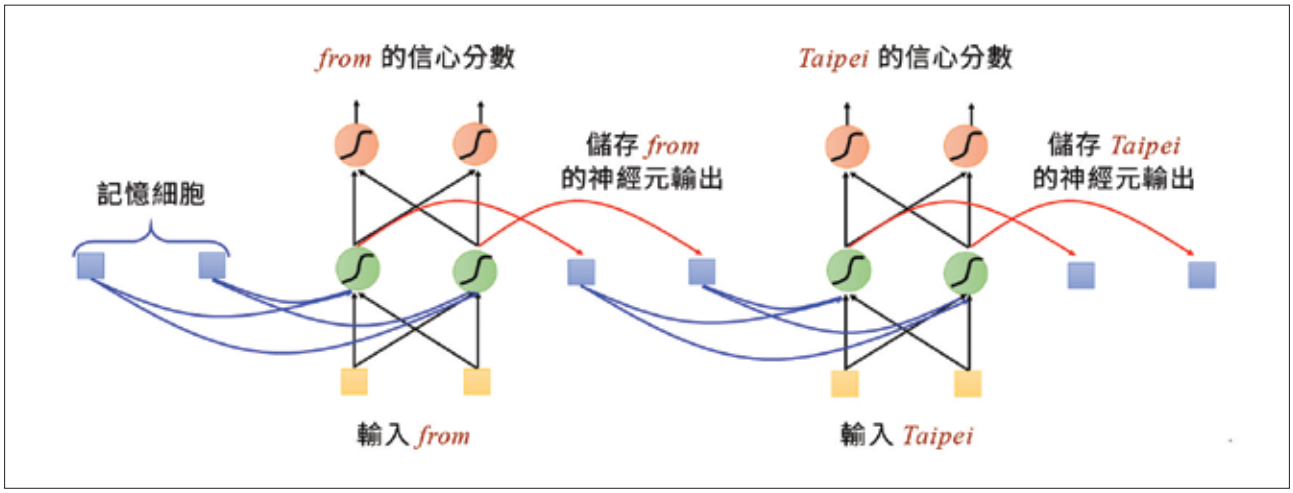


圖 3 遞迴式類神經網絡的結構與運作方式，輸入為「from Taipei……」。

組「記憶細胞」（圖 3 中的藍色方塊），每個細胞中存有一個數值。神經元會將記憶細胞中的數值作為輸入，而神經元的輸出則被儲存到記憶細胞中。在圖 3 的例子中，假設使用者輸入的文句為「from Taipei to Kaohsiung」，遞迴式類神經網絡會從句首依序閱讀這個句子，每次輸入一個字詞，第一個輸入的字詞是 from，機器會根據輸入字詞和記憶細胞儲存的值 ❽，判斷 from 在每個類別的信心分數。而在判斷過程中，隱藏層神經元（綠色神經元）輸出會被存入記憶細胞內（圖 3 中央的紅色箭頭），接下來輸入 Taipei，如此反覆進行，直到整個句子處理完畢。對遞迴式類神經網絡而言，在看到 Taipei 這個字詞前，已輸入的是 from 或是 to，將使得記憶細胞所儲存的內容不同，因此輸入同樣的字詞，可得到不同的輸出。

接下來，我們來談談記憶細胞。目前最常見的記憶細胞是長短期記憶細胞（long short-term memory cell）[11] ❾，這種記憶細胞有三個閘門：

○**輸入閘**（input gate）：當輸入閘開啟時，神經元的輸出才會寫入細胞中，否則細胞會無視神經元寫入的訊息。

○**遺忘閘**（forget gate）：當遺忘閘關閉時，記憶細胞會清除之前儲存的內容。

○**輸出閘**（output gate）：當輸出閘開啟時，神經元才能從記憶細胞中讀取資料，否則讀到的數值為 0。

這些閘門在什麼情況下會開啟或關閉，是根據訓練資料透過「學習」而得的。

另一種目前廣泛使用的記憶細胞為門閘遞迴單元（gated recurrent unit, GRU）[12]，它是長短期記憶細胞的簡化版，其精神是「舊的不去，新的不來」。在門閘遞迴單元中，輸入閘和遺忘閘是連動的，只有當遺忘閘關閉清除儲存內容時，輸入閘才會同步開啟儲存新的資料。

遞迴式類神經網絡特別擅長處理成串有序的輸入資料，例如語音和語言。傳統的語音辨識系統往往由多個模組構成，有的負責把聲音訊號轉為音素，

有的負責將一串音素轉為對應的字詞。這個多模組的結構，目前已可以全部用一個複雜多層的遞迴式類神經網絡來取代 [13]。在語言翻譯上，傳統的翻譯系統也是由很多模組所構成，例如：文法剖析模組、語意剖析模組等等，目前語言翻譯也可以直接用一個遞迴式類神經網絡來完成 [14]。

機器也要學會專注

讀者閱讀到這裡，你的視線會專注在這段文字，聽覺則會忽略旁人的對話。人的五感隨時隨地都有大量的資訊輸入，但是人腦知道如何忽視和當前任務無關的輸入，只會選擇專注在和當前需求相關的資訊上。和人類一樣，機器也需要具備專注的能力。近年已經出現一些模擬專注力的神經網絡架構，其中具代表性的例子有記憶網絡（memory network）[15]、類神經涂林機（neural Turing machine）[16]、動態記憶網絡（dynamic memory network）[17] 等，這類模型統稱專注式模型（attention-based model）。

專注式模型適合用在問答（Question Answering, QA）系統上 [15]，讓機器可以針對使用者提出的問題，回答正確的答案。當使用者輸入問題時，專注式模型可以幫助機器從資料庫中選擇相關資料，進而只處理相關資料，得到問題的答案。近年，問答系統的研究已經從文字擴展到多媒體，機器可以看圖片回答相關問題（例如：圖中的人穿什麼顏色的衣服）[18]。在臺灣大學語音處理實驗室，我們嘗試讓機器做托福聽力測驗的問題。跟人一樣，機器聽一段聽力測驗的文章，將文章內容記下並加以理解後，根據問題從四個選項中挑選正確的選項。機器目前已能做到兩題答對一題的程度 [19][20]。此外，專注式模型也被應用在自動為新聞做摘要 [21]、自動產生圖片／影像說明 [22]、語音辨識 [23] 與機器自動翻譯等任務上 [24]。

機器能不能無師自通

在前文中，我們假設機器要找尋的函數，其輸入與輸出都已包含在訓練資料中，例如手寫數字辨識，訓練資料必須包含影像及其對應的數碼，這樣的學習情境叫做**督導式學習**（supervised learning）。在督導式學習中，機器就像是身邊有老師手把手教學的學生，每個問題都有老師提供正確的答案。近來，深度學習的研究逐漸往**強化式學習**（reinforcement learning）發展，在強化學習中，機器做出好的行為就會得到正面回饋，做錯則得到負面回饋，但卻沒有老師告訴機器什麼是正確的行為，機器必須自己找出得到正面回饋的方式。（關於強化學習，請見本期〈打開 AlphaGo！〉）更進一步，機器能不能做到**非督導式學習**（unsupervised learning），也就是無師自通呢？

機器能不能聽有聲書學會人類語言？ [25] 在臺大語音實驗室，我們嘗試讓機器聽有聲書來學習人類語言。機器就像嬰兒一樣，對人類語言沒有任何先備知識，但在聽了數小時的有聲書後，機器可以自動判別哪些聲音訊號屬於同樣的字詞，如此一來

❽ 在未有任何輸入前，記憶細胞會有初始值，可根據訓練資料讓機器自動找出適當的初始值。

❾ 其實這種記憶細胞的概念，早在 1997 年即已出現，只是近年隨著深度學習的發展才逐漸廣泛應用。

便可協助處理聲音訊號的搜尋，也就是當使用者以語音輸入欲查詢的關鍵字詞後，機器可自動從語音資料庫中找出同樣的字詞。下一步我們的目標是讓機器不只可以知道聲音訊號背後對應的字詞，也能知道其背後所隱含的語意。

機器如果上 PTT，能不能成為鄉民？學校老師沒有教過我們每個字詞的意思，但透過大量閱讀，人類可以學會很多詞彙。機器可否跟人一樣，透過大量閱讀，知道人類字詞背後的語意呢？這是有可能的，因為機器可以根據上下文的一致性判斷字詞的意思。例如：機器讀入很多新聞，發現有一篇 2012 年的新聞提到「馬英九 520 宣示就職」、另一篇 2016 年的新聞提到「蔡英文 520 宣示就職」，雖然機器不知道「馬英九」和「蔡英文」是誰，但從上下文可以判斷出「馬英九」和「蔡英文」有著非常類似的身分。機器閱讀的文章越多，判斷就越精準。如果要求機器把每個字詞用一個向量來表示，會發現語意越相近的字詞，其對應向量的距離就越接近，這個技術叫做詞向量（word embedding） [26]。根據這些向量中的數值，機器甚至可以做出簡單的推論、類比，目前已廣泛應用在自然語言處理上。在臺大語音實驗室，我們運用這項技術，嘗試讓機器閱讀大量 PTT 的文章。就算沒人教，機器可以學到部分鄉民用語的意思，例如機器學到：「魯蛇」之於「loser」，等於「溫拿」之於「winner」。

結語

隨著深度學習如今蓬勃發展，人工智慧的能力已越來越強，但我相信這仍只是個起步。雖然機器號稱在電玩和圍棋上可以痛宰人類 [27][28]，或者在辨識影像中，辨識物件的正確率已經高過人類 [29]，但如果追問機器根據圖片回答問題的能力，可能還不如四歲小孩 [18]，離科幻電影中人類想像的人工智慧仍有很大的差距。目前我們看到的深度學習技術就好比寒武紀的生物大爆發。寒武紀之後，生物還要經歷多次演化才成為今日的樣貌。現今的深度學習技術可能不是人工智慧的最終型態，在前面所述深度學習的三個步驟中——如何選擇函數集、如何定義優劣、如何挑選函數，每一步都有很多尚未解決的問題，等著人們來挑戰。

本文參考資料請見〈數理人文資料網頁〉http://yaucenter.nctu.edu.tw/periodical.php

延伸閱讀

▶ Nielsen, Michael A. "Neural Networks and Deep Learning"（2015）Determination Press。這是作者多次推薦的線上教科書：

http://neuralnetworksanddeeplearning.com/

▶李宏毅〈一天搞懂深度學習〉，作者 2016 年 9 月 24 日在「2016 臺灣資料科學愛好者年會」給了一天四場的課程。底下是其演講幻燈片。對於本文有興趣的讀者可以參考：

http://www.slideshare.net/tw_dsconf/ss-62245351

▶ Deep Learning CONCEPTS，YouTube 視頻的系列介紹課程。

https://goo.gl/BD5djv

深度學習模仿大師風格

2015 年，L. Getys 等人發表一篇論文（見下方正式版本資訊），以捲積式類神經網路提取畫家的風格特徵，套用到一般照片（如 A），並得到仿畫家風格的「畫」。A 為德國杜賓根 Neckarfront 小鎮的風景照（Andreas Praefcke 攝）；B 左下原圖為英國浪漫主義風景畫家透納（J.M.W. Turner）作品《海難》；C 左下原圖為荷蘭後印象派畫家梵谷（Vincent Van Gogh）作品《星夜》；D 左下原圖為挪威畫家孟克（Edvard Munch）的表現主義風格作品《吶喊》；E 左下原圖為西班牙畫家畢卡索的立體派風格作品《坐著的裸女》；F 左下原圖為俄國抽象藝術先驅康丁斯基作品《構成第七號》。

在後來的論文中（見下方），作者們發展新方法，可將畫家的筆觸風格與畫作顏色分離，在風格套用到照片時，能保持「正確的」顏色。（左上）原圖是紐約夜景圖（中上）畢卡索《坐著的裸女》（右上）用原來的演算法得出之圖像。（左下）利用顏色轉換把原圖的顏色和畫家風格結合（右下）配合正確的原圖亮度來做顏色與風格的結合。

本刊感謝底下團隊，同意使用論文中的圖片。

○ Gatys, L. A., Ecker, A. S., and Bethge, M. "Image Style Transfer Using Convolutional Neural Networks" (2016) , *Proc. CVPR* .
○ Gatys, L. A., Ecker, A. S., Bethge, M., Hertzmann, Aaron, and Shechtman, Eli "Preserving Color in Neural Artistic Style Transfer" (2016), Arxiv 預印稿。