# AI 普及給嵌入式設計人員帶來新挑戰

本文探討了人工智慧 (AI) 的普及給嵌入式設計人員帶來的新挑戰。在創建“邊緣機器學習 (ML)”應用時，設計人員必須確保其能有效運行，同時最大限度地降低處理器和儲存開銷，以及物聯網 (IoT) 設備的功耗。

■作者：Yann LeFaou
Microchip 觸控和手勢業務部門副總監

從監控和存取控制到智能工廠和預測性維護，基於機器學習 (ML) 模型構建的人工智慧 (AI) 在工業物聯網邊緣處理應用中已變得無處不在。隨著這種普及，支援 AI 的解決方案的構建已經變得“大眾化”——從資料科學家的專業領域轉為嵌入式系統設計人員也需要了解的領域。這種大眾化帶來的挑戰在於，設計人員並不一定具備定義要解決的問題以及以最恰當方式擷取和組織資料的能力。此外，與消費性解決方案不同，工業 AI 實現的現有資料集很少，通常需要用戶從頭開始創建自己的資料集。

## 融入主流

AI 已經融入主流，深度學習和機器學習 (DL 和 ML) 是我們現在習以為常的許多應用的背後力量，這些應用包括自然語言處理、計算機視覺、預測性維護和資料挖掘。早期的 AI 實現是基於雲或伺服器的，需要大量的處理能力和儲存空間，以及 AI/ML 應用與邊緣 ( 終端 ) 之間的高頻寬連接。儘管生成式 AI 應用 ( 如 ChatGPT、DALL-E 和 Bard) 仍然需要此類設置，但近年來已經出現了邊緣處理的 AI，即在資料擷取點即時處理資料。邊緣處理極大減少了對雲的依賴，使整體系統 / 應用更快、需要更少的功耗並且成本更低。許多人認為安全性得到了提高，但更準確地說，主要的安全重點從保護雲與終端之間的通訊轉移到了使邊緣設備更安全。

邊緣的 AI/ML 可以在傳統的嵌入式系統上實現，這些系統的設計人員可以使用強大的微處理器、圖形處理單元和豐富的記憶體元件，即類似於 PC 的資源。然而，越來越多的商業和工業物聯網設備需要在邊緣具備 AI/ML 功能，這些設備通常硬體資源有限，而且在許多情況下由電池供電。

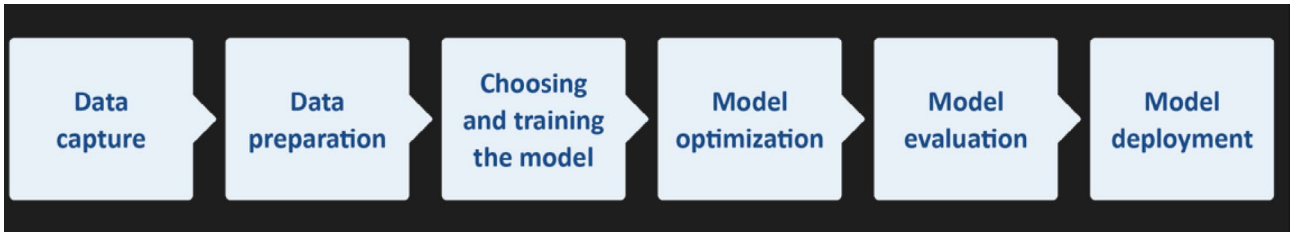
在資源和功耗受限的硬體上運行的邊緣 AI/ML 的潛力催生了“TinyML”這一術語。實際應用涵蓋工業 ( 如預測性維護 )、建築自動化 ( 環境監控 )、建築施工 ( 監督人員安全 ) 和安防等領域。

## 資料流

AI ( 及其子集 ML) 需要從資料擷取 / 收集到模型部署的工作流程 ( 見圖 1)。對於 TinyML 而言，由於嵌入式系統資源有限，因此每個工作流程階段的優化至關重要。

例如，TinyML 的資源需求被認為是 1 MHz 到 400 MHz 的處理速度、2 KB 到 512 KB 的 RAM 和 32 KB 到 2 MB 的儲存空間 ( 快閃記憶體 )。此外，150 µW 至 23.5 mW 的小功耗

圖 1：圖為簡化的 AI 工作流程。雖然圖中未顯示，但模型部署本身必須將資料反饋回流程中，甚至可能影響資料的收集。




預算也常常帶來挑戰。

此外，在將 AI 嵌入資源有限的嵌入式系統時，還有更重要的考慮因素或權衡。模型是系統行為的關鍵，但設計人員經常發現自己在模型品質 / 精確度 ( 影響系統可靠性 / 依賴性和效能，主要是運行速度和功耗 ) 之間做出妥協。

另一個關鍵因素是決定使用哪種類型的 AI/ML。通常有三種算法可供使用：監督學習、無監督學習和強化學習。

## 解決方案

即使是對 AI 和 ML 有良好理解的設計人員，可能也會在優化 AI/ML 工作流程的每個階段並在模型精確度與系統效能之間找到完美平衡方面遇到困難——那麼缺乏以往經驗的嵌入式設計人員如何應對這些挑戰呢？

首先，重要的是不要忽視一個事實：如果模型小且 AI 任務僅限於解決簡單問題，那麼部署在資源有限的物聯網設備上的模型將會更有效。

幸運的是，ML ( 特別是 TinyML) 進入嵌入式系統領域，帶來了新的 ( 或增強的 ) 整合開發環境 (IDE)、軟體工具、架構和模型——其中許多都是開源的。例如，TensorFlow Lite for Microcontrollers(TF Lite Micro) 是一個面向 ML 和 AI 的免費開源軟體庫，它專為在只有幾 KB 記憶體的元件上實現 ML 而設計。此外，程式可以用開源和免費的 Python 語言編寫。

關於 IDE，Microchip 的 MPLAB X 就是此類環境的一個範例。該 IDE 可與公司的 MPLAB ML 一起使用，MPLAB ML 是專門開發的 MPLAB X 插件，用於構建優化的 AI 物聯網感測器識別程式碼。MPLAB ML 由 AutoML 提供支援，可將 AI ML 工作流程的每一步完全自動化，無需重複、繁瑣和耗時的模型構建。特徵提取、訓練、驗證和測試確保滿足微控制器和微處理器記憶體限制的優化模型，使開發人員能夠快速在基於 Microchip Arm Cortex 的 32 位元 MCU 或 MPU 上創建和部署 ML 解決方案。

## 流程優化

工作流程優化任務可以透過使用現成的資料集和模型來簡化。例如，如果一個支援 ML 的物聯網設備需要圖像識別，從現有的標記靜態圖像和視頻片段資料集開始進行模型訓練 ( 測試和評估 ) 是合理的；需要注意的是，監督學習算法需要標記資料。

許多圖像資料集已經存在於計算機視覺應用中。然而，由於它們是為基於 PC、伺服器或雲的應用設計的，通常都很大。例如，ImageNet 包含超過 1400 萬張標註圖像。

根據 ML 應用的不同，可能只需要少量子集；例如，有很多人但只有少量靜物的圖像。例如，如果在建築工地使用支援 ML 的攝像頭，當有不戴安全帽的人進入其視野時，它們可以立即發出報警。ML 模型需要訓練，但可能只需要少量戴或不戴安全帽的人的圖像。然而，對於帽子類型，可能需要更大的資料集和足夠的資料集範圍，以考慮不同的光照條件等各種因素。

圖 1 中第 1 步到第 3 步的內容分別是獲得正確的即時 ( 資料 ) 輸入和資料集、準備資料和訓練模型。模型優化 ( 第 4 步 ) 通常是壓縮，這有助於減少記憶體需求 ( 處理期間的 RAM 和用於儲存的 NVM) 和處理延遲。

在處理方面，許多 AI 算法 ( 如卷積神經網路 (CNN)) 在處理複雜模型時會遇到困難。一種流行的壓縮技術是剪枝 ( 見圖 2)，剪枝有四種類型：權重剪枝、單元 / 神經元剪枝和迭代剪枝。

量化是另一種流行的壓縮技術。量化是將高精確度格式 ( 如 32 位元元浮點 (FP32)) 的資料轉換為低精確度格式 ( 如 8 位元元整數 (INT8)) 的過程。量化模型 ( 見圖 3) 的使用可以透過以下兩種方式之一納入機器訓練。

- 訓練後量化涉及使用 FP32 格式的模型，當訓練完成後，再進行量化以便部署。例如，可以使用標準 TensorFlow 在 PC 上進行初始模型訓練和優化。然後模型可以進行量化，並透過 TensorFlow Lite 嵌入到物聯網設備中。
- 量化感知訓練可類比推斷時量化，創建一個模型供下游工具用於生成量化模型。

雖然量化很有用，但不應過度使用，因為它類似於透過使用較少的位元表示顏色和 / 或使用較少的像素來壓縮數位圖像——亦即，會存在一個圖像變得難以解釋的點。

圖 2：剪枝減少了神經網路的密度。上圖中，某些神經元之間的連接權重被設為零。但有時神經元也可以被剪掉 ( 圖中未顯示 )。

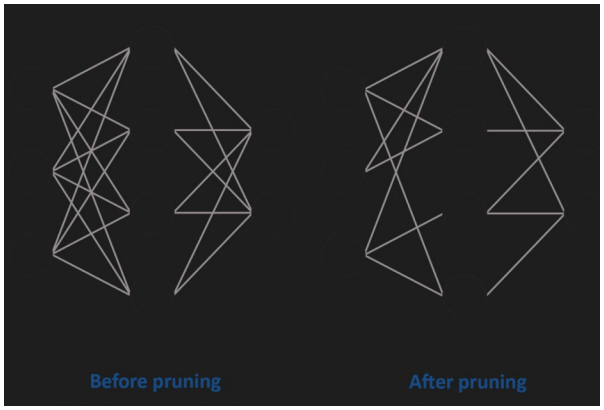
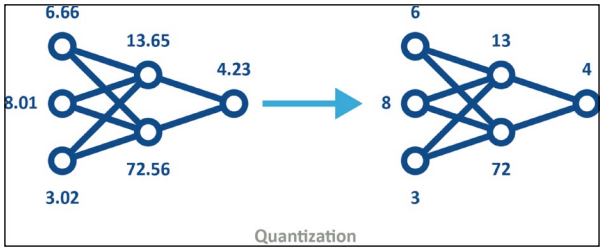



圖 3：量化模型使用低精確度，進而減少記憶體和儲存需求並提高能源效率，同時仍保留相同的形狀。




## 總結

正如我們在開頭所提到的，AI 現在已經深深融入嵌入式系統領域。然而，這種大眾化意味著以前不需要了解 AI 和 ML 的設計工程師正面臨將 AI 解決方案實現到其設計中的挑戰。

儘管創建 ML 應用並充分利用有限硬體資源的挑戰可能令人望而卻步，但這對經驗豐富的嵌入式系統設計人員來說並不是一個新挑戰。好消息是，工程社群內有豐富的訊息 ( 和課程 )，以及像 MPLAB X 這樣的 IDE、MPLAB ML 這樣的模型構建工具以及各種開源資料集和模型。這種生態系統可幫助不同理解水平的工程師快速完成現在可以在 16 位元元甚至 8 位元元微控制器上實現的 AL 和 ML 解決方案。

關於作者：

Yann LeFaou 是 Microchip 觸控和手勢業務部門的副總監。在這個職位中，LeFaou 領導一個團隊開發電容式觸控技術，並推動公司在微控制器和微處理器上的機器學習 (ML) 計劃。他在 Microchip 擔任過一系列連續的技術和市場職位，包括領導公司在電容式觸控、人機界面和家用電器技術方面的全球市場活動。LeFaou 擁有法國電力機械專業學院 (ESME Sudria) 的學位。CTA