Deep Learning with Edge Computing A Review

Introduction

為什麼需要邊緣計算的研究? edge computing 跟先前的 cloud computing 或 distributed computing 有什麼本質上的區別?

為了滿足深度學習的計算需求，一種常見的方法是利用 cloud computing。而若要使用雲端資源，數據必須從網路邊緣(network edge)的數據源位置(例如，從智慧型手機和物聯網傳感器)移動到雲端(centralized location in the cloud)，而這種將數據從 source 移到雲端的潛在解決方案帶來了一些挑戰及問題如下。

1.延遲(latency): 實時推理對許多應用程序至關重要。例如，來自自動駕駛汽車的攝像頭需要快速處理以檢測和避開障礙物，或者基於語音的輔助應用程序需要快速解析和理解用戶的要求(query)並回應(return a response)。但是，將數據發送到雲端進行推理或訓練可能會導致網路產生額外 queuing 和傳播延遲(propagation delay)，並且無法滿足實時交互式應用程序所需的嚴格的端到端低延遲要求；例如，實際實驗表明，將 1 幀(a camera frame)卸載(offloading)到 AWS 服務器並執行計算機視覺任務需要超過 200 毫秒的端到端時間。

2.可擴展性(scalability): 將數據從 source 發送到雲端可能會有擴展性問題，因為隨著連接設備數量的增加，對雲端網路訪問可能成為瓶頸。就網路資源利用而言，將所有數據上傳到雲端也是沒有效率的，特別是深度學習其實並非需要所有來源的所有數據。頻寬密集型(Bandwidth-intensive)數據源，例如影片(video stream)，特別令人擔憂。

3.隱私(privacy): 將數據發送到雲端可能會導致擁有數據的用戶或其行為被數據捕獲的用戶的隱私問題。用戶可能擔心將他們的敏感信息(例如，面部或語音)上傳到雲端，以及雲端或應用程序將如何使用這些數據。

邊緣計算是解決本節前面描述的延遲、可擴展性和隱私挑戰的可行解決方案。但在邊緣實現深度學習仍然存在幾個主要挑戰。
第一個主要挑戰是如何在功能較弱、資源較少的邊緣裝置上滿足深度學習的高資源要求，如算力、通訊、記憶體...等等。
第二個挑戰是了解邊緣裝置在異構處理能力(heterogeneous processing capabilities)和動態網路情況(dynamic network conditions)下應如何與其他邊緣裝置和雲端協調，以確保良好

的端到端(end-to-end)應用級性能(application-level performance)。
最後，隱私仍然是一個挑戰，即使邊緣計算通過將數據保留在網路邊緣本地來自然地提高隱私，因為一些數據仍然常常需要在邊緣設備甚至可能和雲端之間交換。

本文的目的是調查深度學習和邊緣計算兩大趨勢匯合處的作品，特別關注軟體方面及其獨特的挑戰

Vigil 由無線攝像頭網路組成，這些攝像頭在邊緣計算節點智慧地選擇影像(幀)進行分析處理
Vigil 邊緣計算的動機有兩個: 與將所有幀上傳到雲端進行分析相比，減少頻寬消耗，以及可隨攝像機數量的增加實現擴展性
while analyzing IoT sensor data in real time is not always a requirement, and communication bandwidth requirements from sensors are typically small (unless cameras are involved), privacy is a major concern that motivates IoT processing on the edge.
IoT 實時分析物聯網傳感器數據並不總是必要的，並且來自傳感器的通訊頻寬要求通常很小(除非涉及影像)，就目前來看隱私是推動物聯網邊緣處理的主要問題

討論圍繞三個主要架構的研究:
1)邊緣裝置(on-edge)上計算，其中 DNN 在終端設備上執行；
2)基於邊緣伺服器(edge-server based)的架構，來自終端設備的數據被發送到一個或多個邊緣伺服器進行計算；
3)邊緣裝置、邊緣服務器和雲端之間(joint computation among end devices, servers, and the cloud)的聯合計算。
此外還討論數據在邊緣裝置和雲端之間通訊時的隱私保護技術

Mobilenet: 16.9MB, squeezenet: 4.9MB, Tiny YOLOv3: 55.9MB
參數量化採用現有的 DNN，並通過從浮點數變為低位寬數(low-bit width numbers)來壓縮其參數，從而避免代價高昂的浮點乘法。
修剪也是採用現有的 DNN 去除最不重要的參數
知識蒸餾是創建一個較小的 DNN，以模仿更大、更強大的 DNN 的行為

當 DNN 在邊緣服務器上運行時，來自多個邊緣設備的 DNN 任務需要在共享計算資源上運行和有效管理
2) focusing on the tradeoffs between accuracy, latency, and number of requests served

討論四種卸載場景:

1) DNN 二元卸載，決定是否卸載整個 DNN；

2) DNN 部分卸載，其中決定是否應該卸載 DNN 計算的哪一部分；

3) 跨邊緣設備、邊緣伺服器和雲端的組合執行卸載的分層架構；

4) 分佈式計算方法，將 DNN 計算分佈在多個對等設備上

C. Computing Across Edge Devices

1) Multi-Column CNN, 這些決策基於對這些參數之間權衡的經驗，例如不同 DNN 模型的能耗、準確性、延遲和輸入資料大小

The catalog of different DNN models: existing popular models or new model variants constructed through knowledge distillation or by "mix-and-matching"

是否卸載的決定取決於數據的大小、硬件能力、要執行的 DNN 模型以及網路品質等因素

2)除了按層對 DNN 進行分區外，DNN 還可以按輸入維度進行分區

3)雖然單獨卸載到雲端可能會違反所考慮的深度學習應用程序的實時要求

4) MoDNN, DeepThings DNN 分區決策是基於邊緣設備的計算能力和/或記憶體。在運行時輸入數根據負載平衡原則分發 以考慮計算資源可用性或網路條件的動態變化

D. Private Inference

1) 保護 user data, guarantees a model does not remember details about any specific device's input data

2) 保護 user data 跟 server 確保邊緣設備在不了解 DNN 模型的情況下接收推理結果，並且邊緣服務器在不了解設備數據的情況下處理數據。利用 Low-degree 多項式來逼近 DNN 中使用的常見計算

安全多方計算側重於計算中間步驟的隱私，而差分隱私則側重於整體構建模型的隱私保證

Open Challenges

A. Migration 當用戶移動時 深度學習應用程式應該如何遷移

B. Tradeoff between latency, accuracy, battery 來自設備 A 的第 1 幀的請求是否應該比來自設備 B 的第 100 幀更優先? Priority?

D. 用戶集更小，深度學習模型更專業 more specialized deep learning models

本文回顧了深度學習在網路邊緣運行的當前技術水平 描述了跨邊緣設備 伺服器和雲端加速深度學習推理的方法