深度學習在行動裝置上的應用
Deep Learning for Mobile Application

組別：B08　　指導教授：孫民　　組員：董若蘋

Introduction

隨著深度學習(Deep Learning)的發展，機器學習和人工智慧被廣泛的使用，這個技術透過巨量資料去訓練出模型，因此需要龐大的計算資源；然而隨著應用的範圍漸廣，出現需要將這種技術移植到行動裝置上的需求和趨勢。要將這樣的技術轉移到行動裝置上運作，最大的困難是行動裝置上的硬體資源相對非常有限，因此要實現這樣的系統，必須除了追求辨識的準確率之外，同時盡可能地縮減模型的大小和佔用的資源，因此我們對各種模型壓縮(model compression)的方法進行研究。

我們實作的目標是為一個以食物為主題的社群軟體來訓練一個可以在行動裝置上自動辨識餐點及餐廳的影像分類器(image classifier)，以往的做法是讓使用者將照片上傳之後，在伺服器上做辨識並將結果顯示在app的介面上；然而使用者為隱私考量，未必會將照片主動上傳，因此希望能在行動裝置端上就可以做簡單的初步辨識，並透過結果來提示使用者相關的照片可上傳分享以吸引使用者使用。由於手機資源的耗用會影響使用意願，因此運行時佔用的記憶體應盡量縮減，也節省電量使用。

實作的結果，我們成功地將這個模型在手機上運作，並且將原本的模型大小縮減了75%，在手機當中佔用的資源僅有30 MB，同時運行的速度也達到可以做影像的即時同步辨識。

Compression Algorithm

- **XNOR-Net [1]**

這個方法將數值用二進位(binary)來表達，並以二進位的形式作運算，好處是CPU在計算二進位比計算浮點數 (floating point) 來的快許多，而行動裝置上不一定配有GPU，通常是以CPU在做運算，因此可以有效加速。

- **Deep Compression [2]**

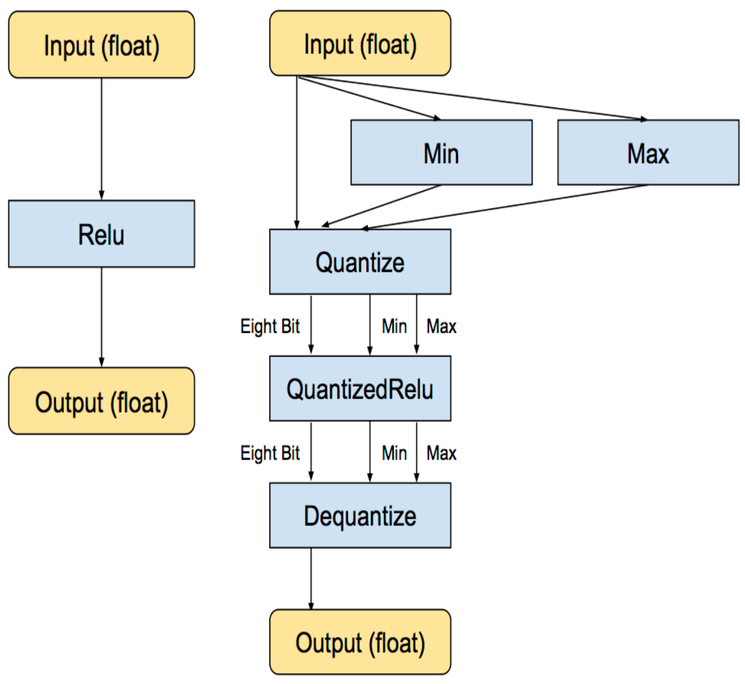
Deep compression指的是三階段的model compression: pruning、quantization、和Huffman coding 。

- **Model Distilling [3]**

將一個訓練好的大模型裡面經訓練的過程學到的資訊「萃取」出來，並將這些資訊教給一個較小的模型，可以讓小模型的表現比自己訓練出來的效果還好。

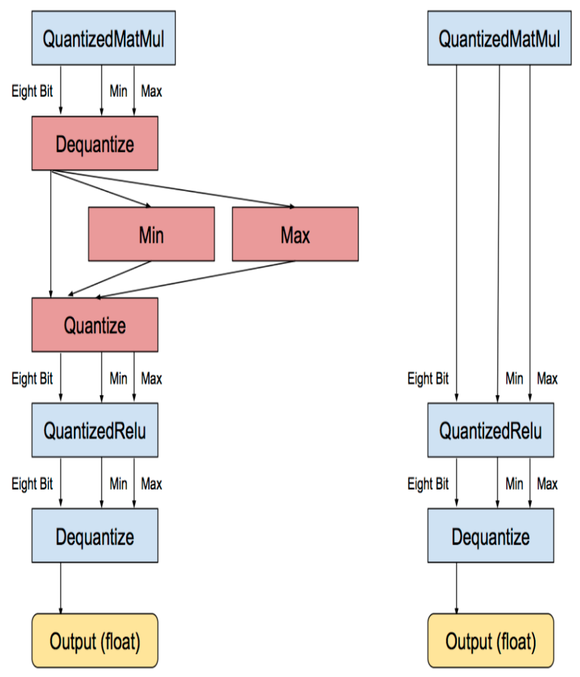
Implementation

我們實作了8 bit quantization的方式來進行model compression 。這個方式不需要重新訓練模型，對模型架構的影響也較小。首先先將一些常用的運算子 (convolution、matrix multiplication、concatenation……等) 定義其8 bit quantization的運算方式，在計算時直接使用這些定義好的函式。[6]



上方左圖為一般運算，右圖為quantize的運算。首先將輸入的浮點數進行quantize，依照指定的數值範圍(Max/Min) 對應到quantize的數值，接著使用定義好的8 bit quantization運算方式計算，最後將數值進行dequantize轉換回浮點數。

下方圖為當數值做一連串的運算，在每個運算之間數值都必須重複的進行多餘的quantize和dequantize的動作，因此合併這些quantize的動作，減少不必要的重複轉換。



Experiment Results

- Dataset

原始數據包含六大類的圖片，包括甜點(dessert)、主菜(dish)、飲料(drink)、餐廳環境(look)、菜單(menu)、人物(people)，每個類別約有2000張圖片，共約12000張圖片，以8:2的比例隨機分為training和testing兩組資料作為訓練和測試用。

- Model

Google inception network v1 [4] 及 Google inception network v3 [5]

- Results

使用Google inception networks v1 訓練出的模型以及經過model compression之後的表現

	Accuracy	Model Size
Before Compression	83.9%	19.2 MB
After Compression	83.8%	4.9 MB

實作結果成功將模型壓縮至僅約原本的0.25倍，但是準確率幾乎不變，在實務上此模型已可以作為這個食物社群app於真實運作上實際運用。

References

[1] Rastegari, Mohammad, Ordonez, Vicente, Redmon, Joseph, and Farhadi, Ali, “Xnor-net: Imagenet classification using binary convolutional neural networks. “AarXiv preprint arXiv:1603.05279, 2016.
[2] S. Han, H. Mao, and W. J. Dally, “Deep Compression: Compressing Deep Neural Networks with Pruning, Trained Quantization and Huffman Coding,” ICRL, 2016.
[3] Hinton, G. Vinyals, O. and Dean, J. “Distilling knowledge in a neural network,” Deep Learning and Representation Learning Workshop, NIPS, 2014.
[4] C. Szegedy et al, “Going Deeper with Convolutions,” CVPR, 2015.
[5] C. Szegedy et al, “Rethinking the Inception Architecture for Computer Vision,” ArXiv e-prints, 2015.
[6] 圖片來源：https://www.tensorflow.org/how_tos/quantization/