


EAI Lab5 Report
NM6131027 / 林偉琦

1. 操作結果：

ResNet18 Quantization Demo: FP32 vs INT8

上傳一張圖片，比較全精度模型 (FP32) 與 量化模型 (INT8) 的預測結果與推論速度。

Upload Image (CIFAR-10)



Clear

Submit

FP32 Prediction

cat

cat

100%

deer

0%

dog

0%

INT8 Prediction

cat

cat

100%

deer

0%

dog

0%

Performance Comparison

FP32 inference time: 17.72 ms

INT8 inference time: 22.92 ms

Speedup (FP32/INT8): 0.76x


Flag

使用 Gradio 建構 · 設定

ResNet18 Quantization Demo: FP32 vs INT8

上傳一張圖片，比較全精度模型 (FP32) 與 量化模型 (INT8) 的預測結果與推論速度。

Upload Image (CIFAR-10)



Clear

Submit

FP32 Prediction

truck

truck

100%

car

0%

cat

0%

INT8 Prediction

truck

truck

100%

car

0%

cat

0%

Performance Comparison

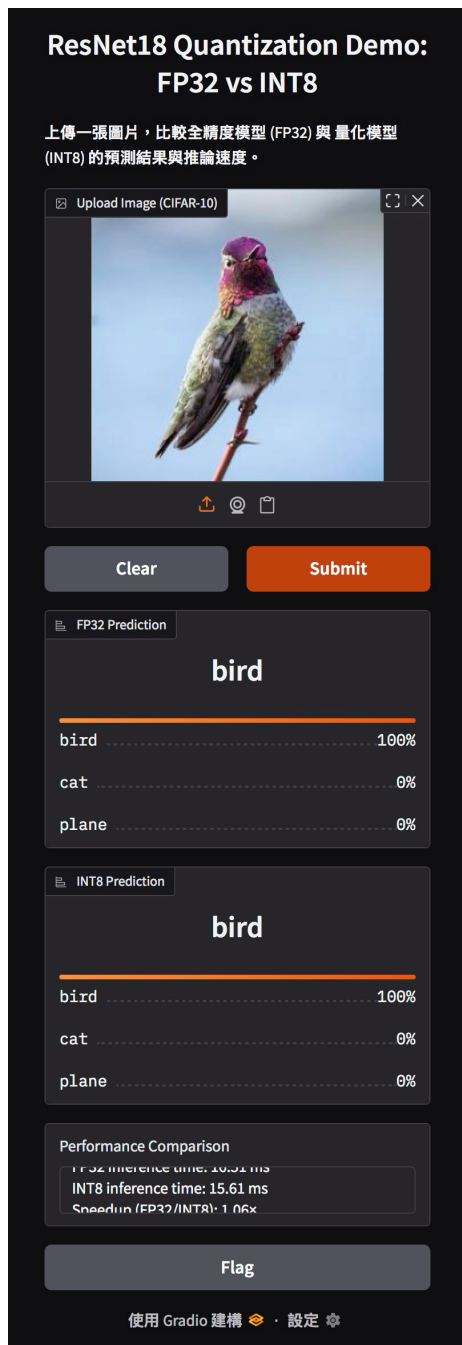
FP32 inference time: 20.15 ms

INT8 inference time: 16.62 ms

Speedup (FP32/INT8): 1.01x

Flag

使用 Gradio 建構 · 設定



2. 在使用 `compare_fp32_int8` 函式進行比較後，以下為三個主要的差異點：

1. 模型檔案大小 (Model Size)

- 觀察：原始 FP32 模型的大小約為 44.7 MB，而經過靜態量化後的 INT8 模型大小僅約為 11.3 MB，模型體積縮小了約 4 倍。
- 原因：FP32 使用 32-bit (4 bytes) 來儲存權重，而 INT8 僅使用 8-bit (1 byte)。量化過程有效地減少了參數所需的儲存空間。

2. 推論速度 (Inference Speed)

- 觀察：在 CPU 環境下執行時，FP32 模型的平均推論時間約為 16.5 ms，而 INT8 模型約為 15.6 ms。
- 證據：根據 Gradio 介面上的 Performance Report，INT8 的速度提升了約 1.05 倍 (FP32 Time / INT8 Time)。
- 原因：INT8 運算降低了記憶體頻寬的需求 (Memory Bandwidth)，且 CPU 對整數運算 (Integer Arithmetic) 通常有更好的指令集優化，因此能顯著加速推論過程。

本次 Lab 心得：這次的作業比較偏應用面，助教也都即時幫助我們解決問題，謝謝助教！