

NetAdapt: Platform-Aware Neural Network Adaptation for Mobile Applications

Tien-Ju Yang¹ [0000–0003–4728–0321], Andrew Howard², Bo Chen²,
Xiao Zhang², Alec Go², Mark Sandler², Vivienne Sze¹, and Hartwig Adam²

¹ Massachusetts Institute of Technology

² Google Inc.

{tjy,sze}@mit.edu, {howarda,bochen,andypassion,ago,sandler,hadam}@google.com

Abstract. This work proposes an algorithm, called NetAdapt, that automatically adapts a pre-trained deep neural network to a mobile platform given a resource budget. While many existing algorithms simplify networks based on the number of MACs or weights, optimizing those indirect metrics may not necessarily reduce the direct metrics, such as latency and energy consumption. To solve this problem, NetAdapt incorporates direct metrics into its adaptation algorithm. These direct metrics are evaluated using empirical measurements, so that detailed knowledge of the platform and toolchain is not required. NetAdapt automatically and progressively simplifies a pre-trained network until the resource budget is met while maximizing the accuracy. Experiment results show that NetAdapt achieves better accuracy versus latency trade-offs on both mobile CPU and mobile GPU, compared with the state-of-the-art automated network simplification algorithms. For image classification on the ImageNet dataset, NetAdapt achieves up to a $1.7\times$ speedup in measured inference latency with equal or higher accuracy on MobileNets (V1&V2).

1 Introduction

Deep neural networks (DNNs or networks) have become an indispensable component of artificial intelligence, delivering near or super-human accuracy on common vision tasks such as image classification and object detection. However, DNN-based AI applications are typically too computationally intensive to be deployed on resource-constrained platforms, such as mobile phones. This hinders the enrichment of a large set of user experiences.

A significant amount of recent work on DNN design has focused on improving the efficiency of networks. However, the majority of works are based on optimizing the “indirect metrics”, such as the number of multiply-accumulate operations (MACs) or the number of weights, as proxies for the resource consumption of a network. Although these indirect metrics are convenient to compute and integrate into the optimization framework, they may not be good approximations to the “direct metrics” that matter for the real applications such as latency

This work was done while Tien-Ju Yang was an intern at Google.

