

# HM-NAS: Efficient Neural Architecture Search via Hierarchical Masking

Shen Yan<sup>†</sup>, Biyi Fang<sup>†</sup>, Faen Zhang<sup>†</sup>, Yu Zheng<sup>†</sup>, Xiao Zeng<sup>†</sup>, Hui Xu<sup>‡</sup>, Mi Zhang<sup>†</sup>

<sup>†</sup>Michigan State University, <sup>‡</sup>AlInnovation

{yanshen6, fangbiyi, zhengy30, zengxi a6, mi zhang}@msu.edu,

{zhangfaen, xuhui}@aiinnovation.com

## Abstract

The use of automatic methods, often referred to as Neural Architecture Search (NAS), in designing neural network architectures has recently drawn considerable attention. In this work, we present an efficient NAS approach, named HM-NAS, that generalizes existing weight sharing based NAS approaches. Existing weight sharing based NAS approaches still adopt hand designed heuristics to generate architecture candidates. As a consequence, the space of architecture candidates is constrained in a subset of all possible architectures, making the architecture search results sub-optimal. HM-NAS addresses this limitation via two innovations. First, HM-NAS incorporates a multi-level architecture encoding scheme to enable searching for more flexible network architectures. Second, it discards the hand designed heuristics and incorporates a hierarchical masking scheme that automatically learns and determines the optimal architecture. Compared to state-of-the-art weight sharing based approaches, HM-NAS is able to achieve better architecture search performance and competitive model evaluation accuracy. Without the constraint imposed by the hand designed heuristics, our searched networks contain more flexible and meaningful architectures that existing weight sharing based NAS approaches are not able to discover.

## 1. Introduction

Neural architecture search (NAS) has recently attracted significant interests due to its capability of automating neural network architecture design and its success in outperforming hand-crafted architectures in many important tasks such as image classification [1], object detection [2], and semantic segmentation [3]. In early NAS approaches, architecture candidates are first sampled from the search space; the weights of each candidate are learned independently and are discarded if the performance of the architecture candidate is not competitive [4, 1, 5, 6]. Despite their remarkable performance, since each architecture candidate requires a full

training, these approaches are computationally expensive, consuming hundreds or even thousands of GPU days in order to find high-quality architectures.

To overcome this bottleneck, a majority of recent efforts focuses on improving the computation efficiency of NAS using the weight sharing strategy [4, 7, 8, 9, 10]. Specifically, rather than training each architecture candidate independently, the architecture search space is encoded within a single over-parameterized supernet which includes all the possible connections (i.e., wiring patterns) and operations (e.g., convolution, pooling, identity). The supernet is trained only once. All the architecture candidates inherit their weights directly from the supernet without training from scratch. By doing this, the computation cost of NAS is significantly reduced.

Unfortunately, although the supernet subsumes all the possible architecture candidates, existing weight sharing based NAS approaches still adopt hand designed heuristics to extract architecture candidates from the supernet. As an example, in many existing weight sharing based NAS approaches such as DARTS [7], the supernet is organized as stacked cells and each cell contains multiple nodes connected with edges. However, when extracting architecture candidates from the supernet, each candidate is hard coded to have exactly two input edges for each node with equal importance and to associate each edge with exactly one operation. As such, the space of architecture candidates is constrained in a subset of all possible architectures, making the architecture search results sub-optimal.

Given the constraint of existing weight sharing approaches, it is natural to ask the question: will we be able to improve architecture search performance if we loosen this constraint? To this end, we present HM-NAS, an efficient neural architecture search approach that effectively addresses such limitation of existing weight sharing based NAS approaches to achieve better architecture search performance and competitive model evaluation accuracy. As illustrated in Figure 1, to loosen the constraint, HM-NAS incorporates a multi-level architecture encoding scheme which enables an architecture candidate extracted from the supernet to have

















