Exploring Randomly Wired Neural Networks for Image Recognition (Supplementary Material)

Saining Xie Alexander Kirillov Ross Girshick Kaiming He

Facebook AI Research (FAIR)

A. Appendix

A.1. Mapping a NAS cell to a graph.

If one maps a combining op (*e.g.*, addition, concatenation) to a node, and a unary transformation (*e.g.*, 3×3 conv, 5×5 conv, identity) to an edge (Figure 1, right), then all cells in the NAS search space share this property: *internal nodes all have precisely input degree 2 and output degree 1*. This is an implicit prior induced by the design.

The mapping from the NAS cell to a graph is not unique. One may map both combining and unary transformations to nodes, and data flow to edges (Figure 1, left). The above property on the NAS search space can be instead described as: internal *merging* nodes all have precisely input degree 2 and output degree 1.

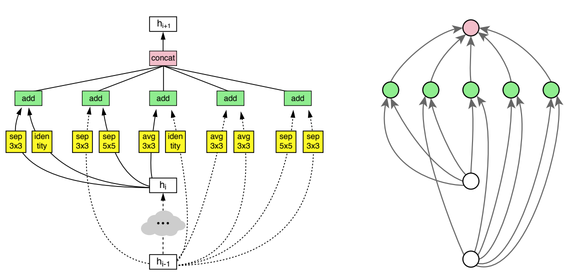

Figure 1. Mapping a NAS cell (left, credit: [1]) to a graph (right).

A.2. Converting undirected graphs into DAGs.

ER, BA, and WS models generate random undirected graphs. We convert them to DAGs using a simple heuristic: we assign indices to all nodes in a graph, and set the direction of every edge as pointing from the smaller-index node to the larger-index one. This heuristic ensures that there is no cycle in the resulted directed graph. The node indexing strategies for the models are — ER: indices are assigned in a random order; BA: the initial M nodes are assigned indices 1 to M, and all other nodes are indexed following their order of adding to the graph; WS: indices are assigned sequentially in the clockwise order.

References

[1] B. Zoph, V. Vasudevan, J. Shlens, and Q. V. Le. Learning transferable architectures for scalable image recognition. In *CVPR*, 2018. 1