CLAIMS

1. A method performed on a computing device, the method comprising:

performing two or more iterations of an iterative model-growing process, the iterative model-growing process comprising:

selecting a particular parent model from a parent model pool of one or more parent models;

generating a plurality of candidate layers and initializing the plurality of candidate layers while reusing learned parameters of the particular parent model, including parameters learned when initializing the candidate layers;

selecting particular candidate layers to include in child models for training, respective child models including the particular parent model and one or more of the particular candidate layers;

training the plurality of child models to obtain trained child models;

evaluating the trained child models using one or more criteria; and

based at least on the evaluating, designating an individual trained child model as a new parent model and adding the new parent model to the parent model pool; and

after the two or more iterations, selecting at least one trained child model as a final model and outputting the final model.

2. The method of claim 1, further comprising:

determining computational costs of training or testing the child models; and

using the computational costs as a first criterion when evaluating the trained child models.

3. The method of claim 2, further comprising:

determining losses associated with the trained child models; and

using the losses as a second criterion when evaluating the trained child models.

4. The method of claim 3, further comprising:

plotting the child models on a graph having a first axis reflecting the computational costs and a second axis reflecting the losses; and

selecting the new parent model based at least on a corresponding location of the new

parent model on the graph.

5. The method of claim 4, further comprising:
determining at least one of a lower convex hull or a Pareto frontier on the graph; and
selecting the new parent model based at least on proximity of the new parent model to the lower convex hull or the Pareto frontier.

6. The method of claim 5, wherein the selecting comprises:
identifying a subset of the trained child models that are within a predetermined vicinity of the lower convex hull or the Pareto frontier;
determining respective probabilities for the subset of the trained child models; and
selecting the new parent model based at least on the respective probabilities.

7. The method of claim 1, wherein generating an individual candidate layer comprises:
selecting a target layer from the particular parent model to receive outputs of the individual candidate layer;
selecting one or more input layers from the particular parent model to provide inputs to the individual candidate layer; and
selecting a particular operation to be performed by the individual candidate layer on the inputs.

8. The method of claim 7, the selecting the particular operation comprising:
defining a group of operations; and
randomly selecting the particular operation from the group of operations.

9. The method of claim 7, further comprising:
selecting the target layer and at least one input layer randomly from the particular parent model.

10. The method of claim 1, the final model being a neural network.

11. A system comprising:
a hardware processing unit; and
a storage resource storing computer-readable instructions which, when executed by the

hardware processing unit, cause the hardware processing unit to:

perform an iterative model-growing process that involves modifying parent models to obtain child models, the iterative model-growing process comprising selecting candidate layers to include in the child models based at least on weights learned in an initialization process of the candidate layers; and

output a final model, the final model being selected from the child models.

12. The system of claim 11, wherein the computer-readable instructions, when executed by the hardware processing unit, cause the hardware processing unit to:

generate different candidate layers that share connectivity to the parent models and perform different operations; and

initialize the different candidate layers together to obtain different weights for the different candidate layers.

13. The system of claim 12, wherein the computer-readable instructions, when executed by the hardware processing unit, cause the hardware processing unit to:

based at least on the different weights of the different candidate layers, select individual candidate layers for inclusion in the child models.

14. The system of claim 11, wherein the computer-readable instructions, when executed by the hardware processing unit, cause the hardware processing unit to:

train the final model using training data for at least one classification, machine translation, or pattern recognition task; and

provide the final model for execution, the final model being adapted to perform the at least one classification, machine translation, or pattern recognition task.

15. The system of claim 11, wherein the computer-readable instructions, when executed by the hardware processing unit, cause the hardware processing unit to:

execute the final model to perform the at least one classification, machine translation, or pattern recognition task.