Evolving Neural Network Code

Team 3
강성민
김대영
김민
이형진

Motivation

"Make pristine-performance neural networks automatically at the code level, so that they are more accessible"

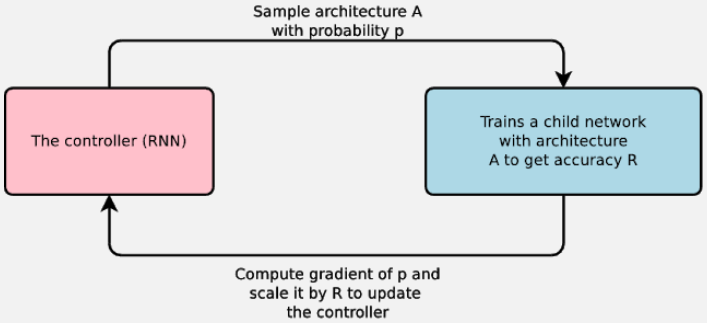
- ***S.M.Kang***

What is the problem?

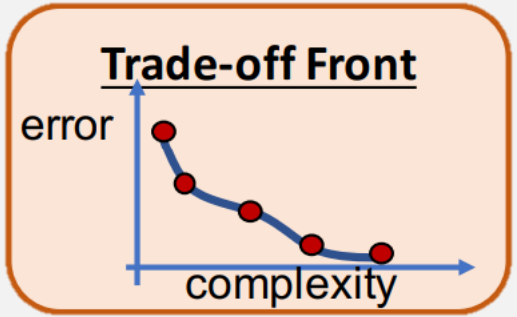
Given a dataset that is subject to a certain learning task, create well-performing neural network ***source code*** based on existing open-source neural network code.

Input : Dataset
Output : Optimized Neural Network Code

Related Work



AutoML



NSGA-Net

Potential Benefits

Unlike them, we directly manipulate code. Why?

- Can apply to training code too, not just the model
- Code sampled from DB is already optimized
- More effectively utilize functions of deep learning library

Methodology

Line of Attack:
Genetic Improvement(GI)

1. Encoding

2. Search Procedure
    a. Initialization
    b. Cross-over
    c. Mutation

3. Evaluation / Fitness Function

Encoding

We do not require a special encoding technique as we use the code itself as our encoding.

Search Procedure

Step 1 : Initialization

Use a simple network that can be extended upon (e.g. ConvNet, FCN, of varying sizes)

Search Procedure

Step 2 : Exploration

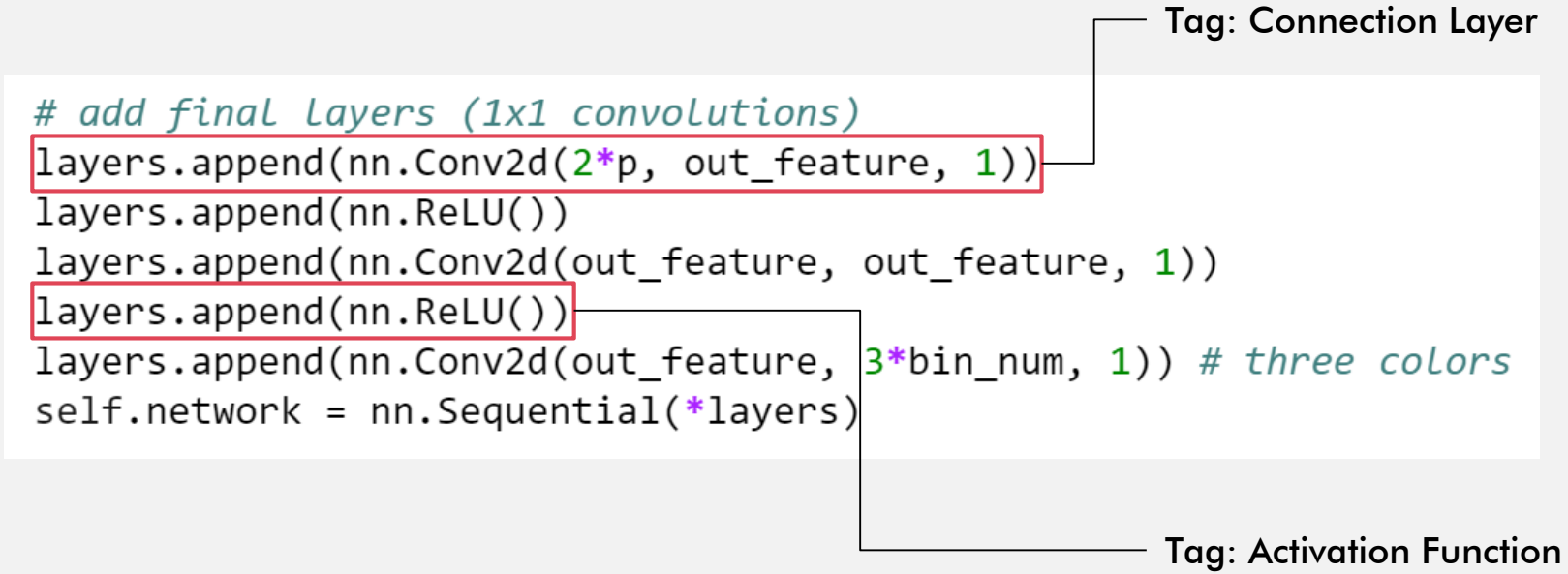
Cross-over

- Read code line by line, assign type*
- Perform crossover on lines of same type.

*Types

- Connective layers, activation functions

Type Assignment



Example Cross-over

```
slayers.append(nn.Linear(2*p, 100))
slayers.append(nn.Tanh())
slayers.append(nn.Linear(100, 10))
slayers.append(nn.Softmax()) # for classification
```

```
# add final layers (1x1 convolutions)
layers.append(nn.Conv2d(2*p, out_feature, 1))
layers.append(nn.ReLU())
layers.append(nn.Conv2d(out_feature, out_feature, 1))
layers.append(nn.ReLU())
layers.append(nn.Conv2d(out_feature, 3*bin_num, 1)) # three colors
self.network = nn.Sequential(*layers)
```

```
# add final layers (1x1 convolutions)
layers.append(nn.Conv2d(2*p, out_feature, 1))
layers.append(nn.ReLU())
slayers.append(nn.Linear(100, 10))
slayers.append(nn.Softmax()) # for classification
```

Step 2 : Exploration

Mutation – Randomly do the following

- Insertion
- Deletion

Evaluation

Using well-known datasets in image classification (CIFAR-10, MNIST), we will obtain following attributes

- Execution Time
- Number of Parameters
- Data Set Accuracy

Then perform a multi-objective optimization on above variables.

Summary

Objective : Evolving neural network code

Methodology : Genetic improvement based on code as genotype

Evaluation : Evaluating existing dataset on quantitative attributes

Q&A