

Neural Architecture Search on Mutated Genome Space

Himanshu Kumar Anand

Data and Computational Science
Indian Institute of Technology, Jodhpur
Jodhpur, India
anand.9@iitj.ac.in

Puja Gupta

Data and Computational Science
Indian Institute of Technology, Jodhpur
Jodhpur, India
gupta.92@iitj.ac.in

Shaonli Pal

Data and Computational Science
Indian Institute of Technology, Jodhpur
Jodhpur, India
pal.14@iitj.ac.in

Abstract—We have implemented a search algorithm to search in a Neural Architecture Search(NAS) space for the best performing Convolutional Neural Network (CNN) architecture on the fashion-mnist data-set.

Index Terms—CNN, NAS Space, Genetic Algorithm, Hill Climbing Search, Mutated Search Space

I. INTRODUCTION

Neural Architecture Search(NAS) consists of a search space, search algorithm and a evolutionary strategy to find the best performing CNN architecture keeping a trade off between accuracy and trainable parameters[1].

The two major advantages of NAS are – Rather than experimenting with large search space, NAS takes small search space and improve, which usually works well with deep learning. The limited search space helps us to get a better model in a less time[3].

II. METHOD

We have created a main population of 20 manually selected genome strings which defines our initial NAS space. Each genome string should have 2 RC layers, one FL layer and any numbers of NC Layers. From our main population we have selected the two best performing genome strings.

TABLE I
TWO BEST PERFORMING CNN ARCHITECTURES

Genome String	Accuracy	Parameters
RC 128 3 tanh;NC 64 3 tanh; RC 64 3 tanh;FL sigmoid;	83.78%	116810
RC 128 3 tanh;NC 128 3 tanh; RC 64 3 tanh;FL tanh;	81.56%	227466

We have incorporated the idea of genetic algorithm to create a new mutated search space of 10 genome strings using the two parent genomes by randomly selecting NC, RC and FL layers of parent genomes with a constraint of maximum 8 neural layers.

Further Hill Climbing approach has been used to find the best performing genome on the mutated genome search space. If the accuracy of k+1 th genome is higher than the accuracy of kth genome , the process will continue. But if the accuracy of k+1 th genome is less than the accuracy of kth genome , process

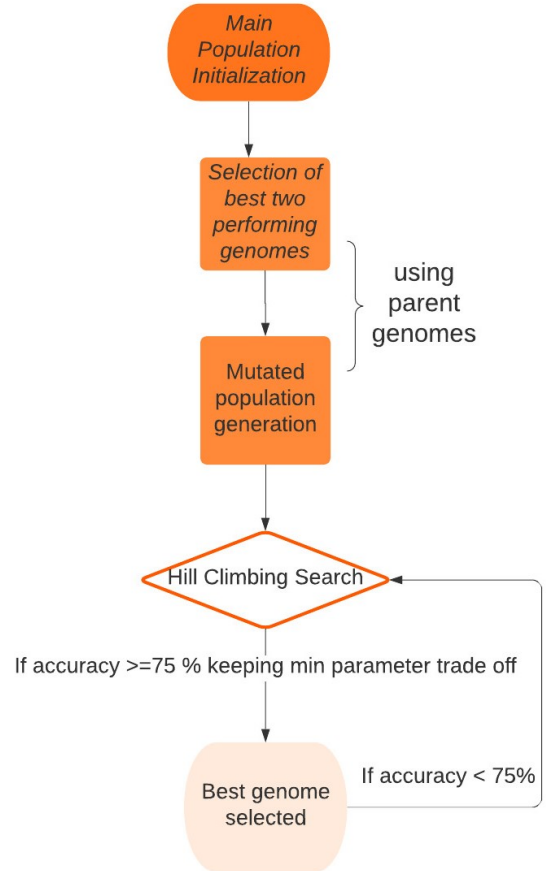


Fig. 1. Workflow of our algorithm

TABLE II
MUTATED GENOME STRINGS

Genome String
NC 64 3 tanh;RC 128 3 tanh;RC 128 3 tanh;NC 128 3 tanh;FL sigmoid;
RC 128 3 tanh;NC 128 3 tanh;RC 128 3 tanh;FL tanh;
NC 64 3 tanh;RC 64 3 tanh;NC 64 3 tanh;RC 64 3 tanh;
NC 128 3 tanh;NC 64 3 tanh;FL tanh;
RC 64 3 tanh;NC 128 3 tanh;RC 64 3 tanh;FL tanh;
NC 64 3 tanh;RC 64 3 tanh;RC 128 3 tanh;NC 128 3 tanh;
NC 128 3 tanh;NC 128 3 tanh;NC 64 3 tanh;FL sigmoid;

will stop and kth genome will be the best selected genome. To maintain the trade off between accuracy and number of trainable parameters, we also have checked that if the increase in number of trainable parameters is more than the accuracy increase in between (k+1) th and kth level , we will stop and kth level genome will be the best selected genome.

The minimum criteria for the selection of best performing CNN architecture is that accuracy should be greater than 75% with minimum parameter count.

III. RESULT

The final best selected CNN Architecture using our search technique over the NAS Space is: NC 64 3 tanh;RC 64 3 tanh;RC 128 3 tanh;NC 128 3 tanh;NC 128 3 tanh;NC 128 3 tanh;NC 64 3 tanh;FL sigmoid;

TABLE III
RESULT OF BEST SELECTED CNN ARCHITECTURE

Accuracy	Parameter Count
90.53%	632778

IV. CONCLUSION

The two best CNN architectures when selected from the main population search space was giving an accuracy of 83.78% and 81.56% respectively. After creating mutated population using the parent genomes and applying hill climbing search algorithm, we observed that with each iteration the number of parameter count has decreased with an increase in the accuracy. The highest accuracy we were able to attain was for genome "NC 64 3 tanh;RC 64 3 tanh;RC 128 3 tanh;NC 128 3 tanh;NC 128 3 tanh;NC 128 3 tanh;NC 64 3 tanh;FL sigmoid;" with an accuracy of 90.53%.

ACKNOWLEDGMENT

With a deep sense of gratitude, we wish to express sincere thanks to our guide, Dr. Deepak Mishra for giving us the opportunity to work on this project. We truly appreciate and value his esteemed guidance and encouragement from the beginning to the end of the project.

REFERENCES

- [1] <https://lilianweng.github.io/lil-log/2020/08/06/neura-architecture-search.html>
- [2] <https://towardsdatascience.com/artificial-neural-networks-optimization-using-genetic-algrithm-with-python-1fe8ed17733e>
- [3] <https://analyticsindiamag.com/neural-architecture-search-an-overview/>
- [4] <https://medium.com/@bhavek.mahyavanshi50/introduction-to-hill-climbing-artificial-intelligence-a3714ed2d8d8>