Plan

Term Paper Plan

Josias Moukpe†  
 Computer Engineering and Sciences  
 Florida Institute of Technology  
 Melbourne Florida USA  
 jmoukpe2016@my.fit.edu

1 Introduction

Today, Artificial Neural Networks (ANN) have proven themselves to be very powerful Machine Learning (ML) model, capable of learning and approximating any arbitrary function, given enough learnable parameters.

However, they aren’t currently without flaws. Anyone who has developed an ANN to fit a particular application dataset would have undoubtedly have to toil with the frustration of choosing and tuning the numerous hyperparameters of the model. This is because, ANN require the adequation combination of hyperparameters values to perform best, and those hyperparameters are not learned by the model, leaving this arduous task of hyperparameter tuning to the developer. To make things worse, those hyperparameters range in infinity. The list of potential hyperparameters to tune include but is not limited to:

1. Numerical hyperparameters

* Number of hidden layers
* Number of hidden units within each hidden layer
* Learning rate
* Number of training epochs
* Momentum rate
* Batch size (if input batches are being used)
* Dropout rate
* Weight Initialization tensor
* Weight decay rate
* Number of folds (if K-fold cross validation is applied)

1. Non-numerical hyperparameters

* Activation function (possibly at each unit)
* Optimization algorithm
* Loss function
* Regularization techniques

Unlike other machine learning approaches such as Decision Trees, ANN have to many hyperparameters to tune, resulting in a lot of development time spend fine-tuning them and retraining each time, before optimally training for the best model for the chosen application. Deep Learning (DL), the branch of machine learning currently dealing with ANN, currently is more art than science and that unfortunately slows the progress that can be made in not only DL, but also ML, and Computer Science as a whole. It’s imperative to solve the problem of hyperparameter initialization and tuning by providing an automated or learned way to reliably set all the hyperparameters to the optimal value for any given task. Furthermore, some hyperparameters can even be adaptively adjusted for performance gains (e.g., adaptive learning rate to reduce training time). This problem is also called Auto-ML or meta-machine learning [6].

2 Related Work

Our problem being hyperparameter search, we are aiming at providing a method that perform better while being more reliable and efficient at hyperparameter initialization and tuning than the simple manual search. Other attempts at hyperparameter search or optimization are random search, grid search, automated hyperparameter tuning using Bayesian optimization or Genetic Algorithm, and Artificial Neural Network tuning using deep reinforcement learning [3]. Before exploring our approach, let’s first look at the state-of-the-art in this domain and some of the most influential work related to this problem include:

1. Random Search for Hyperparameter Optimization [4]

After manual search, the next technique that comes to mind is grid search, where every hyperparameter and its values are arranged on a grid and exhaustively searched to find the best combination of hyperparameter values. Unfortunately, Grid search is only applicable at low dimension (2-4 hyperparameters to search). Random search improves on that but at the cost of guaranteed optimality. Random search is application at larger search spaces and provides better results in less iteration compared to Grid search. Random search can also be run in parallel. However, like Grid Search, Random search doesn’t leverage the information gain from previous iteration, each new guess being independent from the previous one.

1. Regularized Evolution for Image Classifier Architecture Search [5]
2. Population based training of neural networks [7]

3 Approach

The updated template, user manuals, samples, and required fonts,

4 Evaluation

The updated template, user manuals, samples, and required fonts,

4.1 Dataset 1

In the below paragraph, it is explained how alt-txt value is placed

4.2 Dataset 2

In the below paragraph, it is explained how alt-txt value is placed

REFERENCES

[1] Koutsoukas, Alexios, et al. “Deep-Learning: Investigating Deep Neural Networks Hyper-Parameters and Comparison of Performance to Shallow Methods for Modeling Bioactivity Data.” Journal of Cheminformatics, vol. 9, no. 1, 2017, <https://doi.org/10.1186/s13321-017-0226-y>. Conference Name:ACM Woodstock conference

[2] Mitchell, Tom Michael. Machine Learning. McGraw-Hill, 1997. Conference Name:ACM Woodstock conference

[3] Ippolito, Pier Paolo. “Hyperparameters Optimization.” Medium, Towards Data Science, 26 Sept. 2019, <https://towardsdatascience.com/hyperparameters-optimization-526348bb8e2d>.

[4] Bergstra, James, and Yoshua Bengio. “Random Search for Hyper-Parameter Optimization.” Journal of Machine Learning Research 13, (2012) 281-305, 2 Dec. 2012. <https://www.jmlr.org/papers/volume13/bergstra12a/bergstra12a.pdf>

[5] Real, Esteban, et al. “Regularized Evolution for Image Classifier Architecture Search.” Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, 2019, pp. 4780–4789., <https://doi.org/10.1609/aaai.v33i01.33014780>.

[6] Gozzoli, Alessio. “Practical Guide to Hyperparameters Optimization for Deep Learning Models.” FloydHub Blog, FloydHub Blog, 1 July 2020, <https://blog.floydhub.com/guide-to-hyperparameters-search-for-deep-learning-models/>.

[7] Jaderberg, Max, et al. "Population based training of neural networks." arXiv preprint arXiv:1711.09846 (2017). <https://arxiv.org/abs/1711.09846>

Conference Short Name:WOODSTOCK’18

Conference Location:El Paso, Texas USA

ISBN:978-1-4503-0000-0/18/06

Year:2018

Date:June

Copyright Year:2018

Copyright Statement:rightsretained

DOI:10.1145/1234567890

RRH: F. Surname et al.

Price:$15.00