



A BAYESIAN APPROACH BASED ON ACQUISITION FUNCTION FOR OPTIMAL SELECTION OF DEEP LEARNING HYPERPARAMETERS: A CASE STUDY WITH ENERGY MANAGEMENT DATA

Muhammad Ali*

School of Engineering and Information Technology (SEIT)
University of New South Wales, Canberra
Australia
z5218023@ad.unsw.edu.au

Krishneel Prakash

School of Engineering and Information Technology (SEIT)
University of New South Wales, Canberra
Australia
krishneel.prakash@adfa.edu.au

Hemanshu Pota

School of Engineering and Information Technology (SEIT)
University of New South Wales, Canberra
Australia
Email z8904035@ad.unsw.edu.au

*Corresponding author's Email: <u>z5218023@ad.unsw.edu.au</u>.

Peer-review under responsibility of 4th Asia International Multidisciplinary Conference 2020 Scientific Committee http://connectingasia.org/scientific-committee/

© 2020 Published by Readers Insight Publisher,

lat 306 Savoy Residencia, Block 3 F11/1,44000 Islamabad. Pakistan,

editor@readersinsight.net

This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).





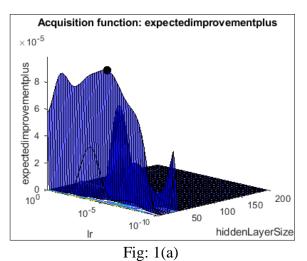


Abstract

With the recent rollout of smart meters, huge amount of data can be generated on hourly and daily basis. Researchers and industry persons can leverage from this big data to make intelligent decisions via deep learning (DL) algorithms. However, the performance of DL algorithms are heavily dependent on the proper selection of parameters. If the hyperparameters are poorly selected, they usually lead to suboptimal results. Traditional approaches include a manual setting of parameters by trial and error methods which is time consuming and difficult process. In this paper, a Bayesian approach based on acquisition is presented to automatic selection of optimal parameters based on provided data. The acquisition function was established to search for the best parameter from the input space and evaluate the next points based on past observations. The tuning process identifies the best model parameters by iterating the objective function and minimizing the loss for optimizable variables such as learning rate and Hidden layersize. To validate the presented approach, we conducted a case study on real-life energy management datasets while constructing a deep learning model on MATLAB platform. A performance comparison was drawn with random parameters and optimal parameters selected by presented approach. The comparison results illustrate that the presented approach is effective as it brings a notable improvement in the performance of learning algorithm.

Keywords: Optimal Parameter Selection, Energy Management Datasets, Automatic Tuning, Deep Learning Algorithms

Graphical Results



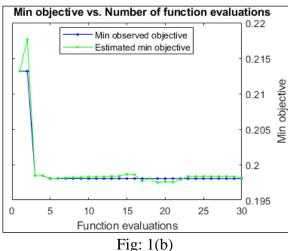


Fig: 1(a): Acquisition function plot with optimizable parameters such as hidden layer size and learning rate. The objective is to find the next search point in iterative space. One could





arbitrarily pick the points in optimization, but this is a complicated and time-consuming process.

Fig: 1(b): Objective function plot to determines the best feasible parameters. It indicates the measured and predicted function values against the number of function evaluations

Research Objectives

One of the key challenges in the implementation of deep learning (DL) algorithms is the correct settings of its parameters to achieve optimal results. The need for tuning DL parameters is ubiquitous in engineering industry as they heavily influence the performance of learning algorithm. In practice, the process of finding good parameters is computationally expensive and complex task. The existing approaches (1–3) include a manual search of parameters which is time-consuming and complex process. The methods like random search (4), grid search (5), evolutionary search (6) and guided search (7) are used heavily in the past. However, these methods are relatively inefficient for the automatic selection of parameters as they do not choose the next hyperparameters based on previous results. Additionally, these methods need a significant amount of time for calculation of parameters which is not desirable in practical applications. In this study, a practical approach based on Bayesian optimization and acquisition function was established for the automatic selection of DL parameters. The study aimed at identifying the best feasible parameters while conducting the experiments with real-life energy management datasets.

Methodology

This study begins with the construction of objective function in MATLAB environment for hyper tuning of parameters. Two key parameters i.e hidden layers size and the learning rate are considered during experiments as they mainly influence the performance of the algorithm. The tuning process follows the concept of Bayesian optimization based on acquisition function. The prior distribution of parameters was formed during the optimization process and acquisition function utilizes this distribution to systematically decide the new set of hyperparameter configurations. To determine the best parameters, K-fold cross-validation (8) was constructed to determine the loss at each iteration. The goal of hypertuning proocess is to obtain the set of hyperparameter values that can give the lowest Root Mean Square Error (RMSE) for the deep learning model. The parameters with the lowest cross-validation loss are considered as optimal parameters and they are selected to train the final model. To validate the presented approach, a typical deep learning model was constructed in a MATLAB environment





and performance is compared with random and tuned parameters. We theoretically analyze our method by using three real-world datasets.

Results

The results obtained from the presented approach are tabulated in Table 1. A deep learning model is trained with random and tuned parameters and performance is measured in terms of RMSE. For a fair comparison, we used 4 K-fold cross-validation and iterated the results for 4 times. The average of these results concluded that the proposed tuning method achieved better results compared to manual tuning. The RMSE is lower on all three datasets. Additionally, we noticed a reasonable average percentage improvement for Dataset 1 (Home A), Dataset 2 (Home B) and Dataset 3 (Home C) with RMSE of 32.34 %, 47.55 %, 79.69 % respectively.

Table 1: Comparison of performance with random parameters and presented approach

Dataset	RMSE without hyper tuning of parameters		RMSE with hyper tuning of parameters		*RMSE Improvement (%)
Home A	Iteration	RMSE	Iteration	RMSE	
	Iteration:1	35.66	Iteration:1	32.28	34.75
	Iteration:2	32.43	Iteration:2	32.27	31.43
	Iteration:3	32.50	Iteration:3	32.30	31.51
	Iteration:4	32.65	Iteration:4	32.29	31.66
		Avg:		Avg:	Avg:
		33.31		32.28	32.34
Home B	Iteration:1	46.92	Iteration:1	46.81	45.92
	Iteration:2	48.03	Iteration:2	47.10	47.05
	Iteration:3	52.06	Iteration:3	47.56	51.16
	Iteration:4	47.05	Iteration:4	46.76	46.06
		Avg:		Avg:	Avg:
		48.51		47.06	47.55
Home C	Iteration:1	79.29	Iteration:1	74.81	78.35
	Iteration:2	84.07	Iteration:2	72.01	83.21
	Iteration:3	77.90	Iteration:3	77.41	76.91
	Iteration:4	81.27	Iteration:4	79.10	80.30





Avg:	Avg:	Avg:	
80.63	75.82	79.69	

*RMSE: Root Mean Square Error

*RMSE improvement= $\frac{\text{RMSE without Hypertuning} - \text{RMSE Proposed Method}}{\text{RMSE without Hypertuning}} \times 100$

Findings

The results of three real-world datasets validate that the presented scheme is statistically significant compared to the manual tuning of parameters. The investigation has lead to several benefits. For instance, it can:

- reduce the human intervention required for the implementation of DL algorithms. In power engineering applications, this is particularly crucial.
- enhance the accuracy of DL algorithms by iteratively finding the globally optimal parameters.
- offers the opportunity to automate the deep learning parameters in real-time systems as it achieves a good set of parameters in a small number of experiments.

Acknowledgement

The authors would like to thank University of New South Wales (UNSW) Canberra, Australia for sponsoring the research.

REFERENCES

- [1]. Lorenzo PR, Nalepa J, Kawulok M, Ramos LS, Pastor JR. Particle swarm optimization for hyper-parameter selection in deep neural networks. In: Proceedings of the Genetic and Evolutionary Computation Conference [Internet]. Berlin, Germany: Association for Computing Machinery; 2017 [cited 2020 Mar 31]. p. 481–488. (GECCO '17). Available from: https://doi.org/10.1145/3071178.3071208
- [2]. Maclaurin D, Duvenaud D, Adams RP. Gradient-based Hyperparameter Optimization through Reversible Learning. :10.
- [3]. Feurer M, Hutter F. Hyperparameter Optimization. In: Hutter F, Kotthoff L, Vanschoren J, editors. Automated Machine Learning: Methods, Systems, Challenges [Internet]. Cham: Springer International Publishing; 2019 [cited 2020 Mar 31]. p. 3–33. (The

^{*}Avg: Average





Springer Series on Challenges in Machine Learning). Available from: https://doi.org/10.1007/978-3-030-05318-5_1

- [4]. Jahedbozorgan M, Amjadifard R. Sunshine: A novel random search for continuous global optimization. In: 2016 1st Conference on Swarm Intelligence and Evolutionary Computation (CSIEC). 2016. p. 12–7.
- [5]. Taijia Xiao, Dong Ren, Shuanghui Lei, Junqiao Zhang, Xiaobo Liu. Based on grid-search and PSO parameter optimization for Support Vector Machine. In: Proceeding of the 11th World Congress on Intelligent Control and Automation. 2014. p. 1529–33.
- [6]. Young SR, Rose DC, Karnowski TP, Lim S-H, Patton RM. Optimizing deep learning hyper-parameters through an evolutionary algorithm. In: Proceedings of the Workshop on Machine Learning in High-Performance Computing Environments [Internet]. Austin, Texas: Association for Computing Machinery; 2015 [cited 2020 Mar 31]. p. 1–5. (MLHPC '15). Available from: https://doi.org/10.1145/2834892.2834896
- [7]. Klau GW, Lesh N, Marks J, Mitzenmacher M. Human-guided search. J Heuristics. 2010 Jun 1;16(3):289–310.
- [8]. Yadav S, Shukla S. Analysis of k-Fold Cross-Validation over Hold-Out Validation on Colossal Datasets for Quality Classification. In: 2016 IEEE 6th International Conference on Advanced Computing (IACC). 2016. p. 78–83.