Missouri University of Science and Technology
Scholars' Mine

Electrical and Computer Engineering Faculty Research & Creative Works

Electrical and Computer Engineering

01 Jan 2022

A Dnn-Ensemble Method for Error Reduction and Training Data Selection in Dnn based Modeling

Ling Zhang

Da Li

Jiayi He

Bhyrav Mutnury

et. al. For a complete list of authors, see https://scholarsmine.mst.edu/ele_comeng_facwork/4679

Follow this and additional works at: https://scholarsmine.mst.edu/ele_comeng_facwork

Part of the Electrical and Computer Engineering Commons

Recommended Citation

L. Zhang and D. Li and J. He and B. Mutnury and B. Pu and X. D. Cai and C. Hwang and J. Fan and J. L. Drewniak and E. P. Li, "A Dnn-Ensemble Method for Error Reduction and Training Data Selection in Dnn based Modeling," *2022 IEEE International Symposium on Electromagnetic Compatibility and Signal/Power Integrity, EMCSI 2022*, pp. 175 - 180, Institute of Electrical and Electronics Engineers, Jan 2022.
The definitive version is available at https://doi.org/10.1109/EMCSI39492.2022.9889587

This Article - Conference proceedings is brought to you for free and open access by Scholars' Mine. It has been accepted for inclusion in Electrical and Computer Engineering Faculty Research & Creative Works by an authorized administrator of Scholars' Mine. This work is protected by U. S. Copyright Law. Unauthorized use including reproduction for redistribution requires the permission of the copyright holder. For more information, please contact scholarsmine@mst.edu.

2022 IEEE International Symposium on Electromagnetic Compatibility & Signal/Power Integrity (EMCSI) | 978-1-6654-0929-2/22/$31.00 ©2022 IEEE | DOI: 10.1109/EMCSI39492.2022.9889587

A DNN-Ensemble Method for Error Reduction and Training Data Selection in DNN Based Modeling

Ling Zhang[#1], Da Li[#2], Jiayi He[*3], Bhyrav Mutnury[§4], Bo Pu[*5], Xiao-Ding Cai[†6], Chulsoon Hwang[*7], Jun Fan[*8], James Drewniak[*9], Er-Ping Li[#10]

[#]College of Information Science and Electronic Engineering, Zhejiang University, Hangzhou, China
[*]EMC Laboratory, Missouri University of Science and Technology, Rolla, MO, USA
[§]Dell, Enterprise Product Group, Round Rock, TX, USA
[†]Cisco, San Jose, CA, USA

[1]lingzhang_zju, [2]li-da, [10]liep@zju.edu.cn
[3]hejiay, [7]hwangc, [8]jfan, [9]drewniak@mst.edu, [5]bobpu@ieee.org
[4]bhyrav.mutnury@dell.com
[6]kecai@cisco.com

***Abstract*—Deep neural networks (DNNs) have been widely adopted in modeling electromagnetic compatibility (EMC) problems, but the training data acquisition is usually time-consuming through various simulators. This paper presents a powerful approach using an ensemble of DNNs to effectively reduce the training data size in DNN-based modeling problems. A batch of training data with the largest uncertainties is selected using active learning through the variance among the ensemble of DNNs. Subsequently, a greedy sampling algorithm is applied to select a data subset using diversity. Thus, the proposed method can achieve both uncertainty and diversity in data selection. By averaging the outputs of the DNN ensemble, the prediction error can be further reduced. Simple mathematical functions are used to validate the proposed method, and a high-dimensional stripline modeling problem also demonstrates the effectiveness of this DNN-ensemble approach. The proposed method is task agnostic and can be used in other surrogate modeling problems with DNNs.**

Keywords—Active learning, stripline modeling, deep neural network, machine learning, ensemble-based, greedy sampling.

I. Introduction

Deep neural networks (DNN) have been broadly utilized in the surrogate modeling of electromagnetic compatibility (EMC) problems by fitting the complex relationships between different features ("input" and "output" features) [1]-[6]. However, generating large amounts of data through various simulators to achieve the desired training accuracy has always been problematic due to the considerable simulation time. Hence, developing a methodology that can effectively relieve the burden of overly large data generation is necessary to promote a more successful application of machine learning in simulation-based modeling.

In previous machine-learning-related research, simple sampling methods, such as the Latin Hypercube Sampling [3] or Halton Sequence Sampling [6], were adopted to generate training data samples instead of random sampling [1], [5]. However, the Latin Hypercube Sampling and Halton Sequence Sampling methods are simple space-filling methods in the input space, which cannot handle complex functional variations in the output space.

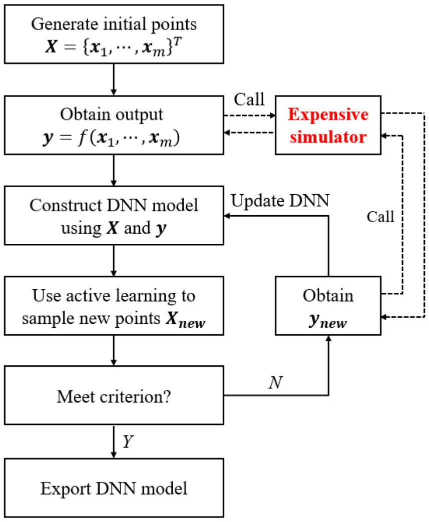

Generate initial points
$X = \{x_1, \cdots, x_m\}^T$
Obtain output
$y = f(x_1, \cdots, x_m)$
Call
Expensive simulator
Construct DNN model using X and y
Update DNN
Call
Use active learning to sample new points X_{new}
Obtain y_{new}
Meet criterion?
N
Y
Export DNN model

Fig. 1. Illustration of the active learning procedure in simulation-based surrogate modeling problems.

There are also some adaptive sampling methods to tackle functional variations in the output space, such as the variance-based method [7], the cross-validation-based method [7], and the gradient-based methods [7]. But these approaches are either model-dependent or cannot handle high-dimensional problems using DNN.

The active learning (AL) technique has attracted increasing interest in computer vision to reduce image labeling costs [8]-[11]. A deep Bayesian AL method was proposed to pick new data samples through uncertainty estimation using dropout layers [8], [9]. But dropout layers, used to prevent overfitting, will deteriorate the training accuracy of simulation-based regression modeling problems in many cases.

Further, a core-set approach was developed for image classification to select new data by evaluating the geometric distances and increasing the diversity among training samples [10]. Also, a greedy sampling (GS) method was proposed for regression problems, which pursues diversity in the input and output space [12]. The greedy sampling algorithm in the input

978-1-6654-0929-2/22/$31.00 ©2022 IEEE

Authorized licensed use limited to: Missouri University of Science and Technology. Downloaded on February 17,2023 at 19:39:10 UTC from IEEE Xplore. Restrictions apply.

space (*GSx* algorithm) [12] is similar to the core-set approach that seeks diversity in the training data. However, whether this greedy sampling approach performs well on complex high-dimensional regression problems using DNN remains a question. Some researchers have also attempted to balance uncertainty and diversity [13][14]. Even though AL has been helpful in classification problems, little investigation has been performed on the training data selection of high-dimensional regression problems in DNN-based modeling.

This paper adopts an ensemble-based AL method to tackle DNN-based regression problems. This ensemble-based AL method is similar to the query-by-committee method [7], [15] and has shown promise in image classification by estimating uncertainty from the outputs of an ensemble of DNNs [11]. As illustrated in Fig. 1, a training strategy is proposed to iteratively train the DNN model and query new samples with high uncertainty from an expensive simulator to minimize the required data size. To prevent the clustering phenomenon by selecting data with large uncertainties only, the GS method [12] is further applied to select a subset of data with enough diversity. The proposed method is referred to as the ALGS (AL + GS) method in this paper. Moreover, it is found that the prediction error can be greatly reduced by averaging the output from the ensemble of DNNs. The power and advantage of the DNN-ensemble approach are validated using simple mathematical functions and a high-dimensional stripline modeling problem.

The rest of this paper is organized as follows. The proposed ALGS method is elaborated in Section II. Section III shows the verification result using two simple mathematical functions, and Section IV discusses the validation using a high-dimensional stripline modeling problem. Finally, the conclusion is drawn in Section IV.

II. Proposed Method

A. Ensemble-Based Active Learning

The ensemble-based AL method [11] is similar to the deep Bayesian AL method [8], [9]. They both train DNNs using existing data and evaluate the uncertainty of unlabeled data. The unlabeled data with the most considerable uncertainties, together with the existing samples, are expected to represent the entire search space more effectively and bring more significant changes to the DNN parameters.

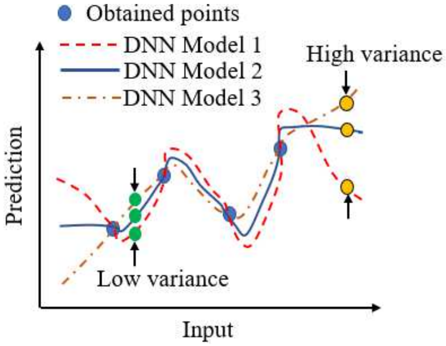


Fig. 2. Illustration of the ensemble-based active learning algorithm.

Fig. 2 illustrates how the ensemble-based AL method estimates uncertainties through an ensemble of DNN models. A popular way of generating these DNN models is to use the same network architecture but different random weight initializations [11]. In Fig. 2, a one-dimensional example and three DNN models are shown for simplification. The DNN models are trained with the same points, and their disagreement, i.e., variance on new samples, is used to evaluate uncertainty. The regions with larger variance are prioritized to be sampled. In contrast, the new samples with lower discrepancies among the DNN models are less valuable for improving the generalization performance of the DNN models.

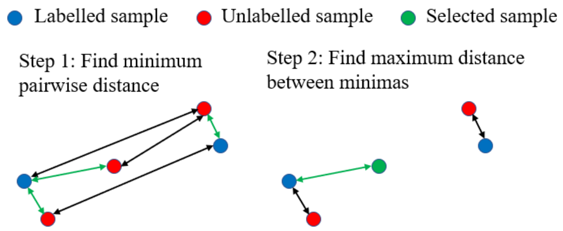
B. Greedy Sampling (GS) Algorithm

In real applications, selecting a data batch rather than a single sample is desired, rendering the data generation (in this way, parallel data generation is feasible) and DNN training (the number of times to train the DNNs can be reduced) more efficient. However, multiple data samples chosen at one time can be located close to each other and cause the clustering phenomenon. Therefore, this paper further adopts the GS algorithm [10], [12] to select a data subset with the most diversity.

Assume there are k labelled samples $\{\mathbf{X}_i\}_{i=1}^{k}$ and N unlabelled samples $\{\boldsymbol{X}_j\}_{j=1}^{N}$. The process of the GS method can be summarized as

$$\text{Selected sample} = \arg\max_{j}\left(\min_{i}\left\|\mathbf{X}_j - \mathbf{X}_i\right\|\right) . \qquad (1)$$
$$i = 1,\cdots,k;\ j = 1,\cdots,N$$

The GS process can be decomposed into two steps, as illustrated in Fig. 3. In each iteration, one unlabelled sample is selected with the longest distance to the labelled samples. Selecting multiple unlabelled samples can be accomplished by choosing and labelling each unlabelled sample sequentially.



● Labelled sample
● Unlabelled sample
● Selected sample
Step 1: Find minimum pairwise distance
Step 2: Find maximum distance between minimas

Fig. 3. Illustration of the GS algorithm [10][12]. Step 1: For each unlabelled sample, find its minimum distance to the labelled samples. Step 2: Select the unlabelled sample with the maximum distances calculated in Step 1.

C. Proposed Method

Fig. 4 depicts a flow chart explaining the procedure of the proposed ALGS (AL + GS) method that seeks both uncertainty and diversity. In this paper, the input variables are assumed to have continuous variation space. Firstly, a certain number of sparse initial points $\boldsymbol{X}$ are generated using random sampling or space-filling methods such as the Latin Hypercube Sampling (LHS) and Halton Sequence Sampling methods. In this paper, the initial data points are generated using the LHS sampling method. A simulator is employed to obtain the corresponding output values $\boldsymbol{y}$. Afterward, an ensemble of DNNs with the same architecture but different random weight initializations

Authorized licensed use limited to: Missouri University of Science and Technology. Downloaded on February 17,2023 at 19:39:10 UTC from IEEE Xplore. Restrictions apply.

are generated and trained with the existing data $\boldsymbol{X}$ and $\boldsymbol{y}$. Since the input variables have continuous variation space with numerous possible combinations, a data pool with N_p samples are generated using the LHS sampling. Subsequently, N_u samples are selected from this data pool with the largest variance by evaluating the disagreement of the DNN ensemble. Afterward, N_d samples that have the greatest diversity compared with the existing training data are chosen from the N_u samples. And this new set of data is added to the existing dataset, which can be further used to train the ensemble of DNN models. This iterative process is repeated until the testing accuracy of the trained DNNs satisfies the desired criterion.

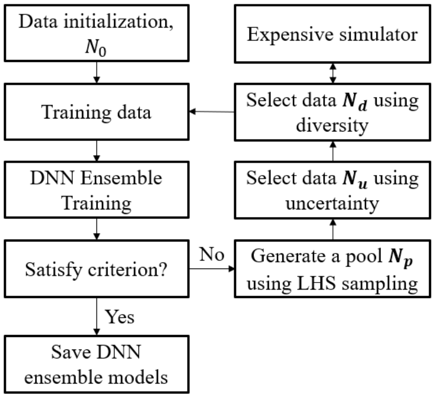

Data initialization, N_0
Expensive simulator
Training data
Select data $\boldsymbol{N_d}$ using diversity
DNN Ensemble Training
Select data $\boldsymbol{N_u}$ using uncertainty
Satisfy criterion?
No
Generate a pool $\boldsymbol{N_p}$ using LHS sampling
Yes
Save DNN ensemble models

Fig. 4. Procedure of the proposed ALGS algorithm

Denote D_{pool} as the data pool in each selection iteration, N as the total number of DNN models (N = 5 in this paper), f_i as the i-th DNN model, and x as the input data samples. In each iteration, new data points are selected by choosing the samples with the largest variance among the DNN ensemble [11] using

$$x_{new} = \underset{x \subset D_{pool}}{\arg\max} \frac{1}{N} \sum_{i=1}^{N} \left[f_i(x) - \mu \right]^2, \tag{2}$$

where μ is calculated as

$$\mu = \frac{1}{N} \sum_{i=1}^{N} f_i(x). \tag{3}$$

A data batch is selected in each iteration by identifying the data samples with the highest variances calculated using (2) and 3).

Moreover, the output values of the DNN ensemble are averaged to further reduce the prediction error. The effect of the averaging operation will be shown in later sections. Therefore, the ensemble of DNNs is used not only to select new data samples from the discrepancy among the DNNs, but also to improve the prediction accuracy by taking their average values.

In this paper, five DNNs with the same architectures but different random weight initializations (which can be realized in Pytorch [16]) are utilized as the ensemble of DNNs. This number of DNNs is a tradeoff between the training accuracy and training time. The more DNNs are used, the higher accuracy in the variance estimation and output averaging can be achieved, but more training time is required.

III. Validation Using Simple Functions

This section uses two simple two-dimensional functions to validate the proposed method and visualize the sample distribution. The first function is

$$f_1(x,y) = \frac{x\sin(14x)}{2+\cos(13x)} + \frac{y\sin(56y)}{2+\cos(55y)},\ x,y \in [0,1] \tag{4}$$

The function $f_1(x,y)$ and its gradient distribution are plotted in Fig. 5. It can be observed that this function has many local maximum and minimum regions, and the function gradient also changes dramatically.

Four methods are compared to select sampling points, including random sampling, greedy sampling [10][12], active learning, and the proposed ALGS method. The active learning method means selecting data according to uncertainties only, while the greedy sampling method only uses the diversity information.

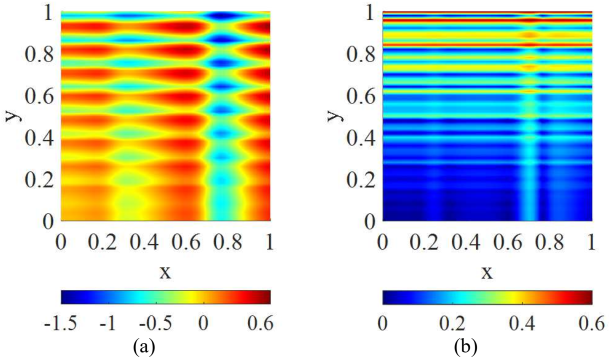


Fig. 5. (a) Function $f_1(x,y)$. (b) Gradient of function $f_1(x,y)$.

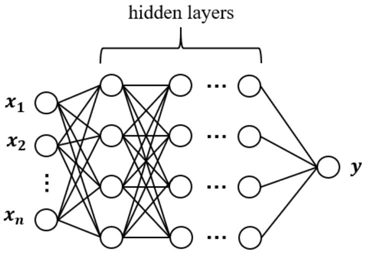


Fig. 6. DNN structure for the ensemble of DNNs. The activation function after each hidden layer (except for the last hidden layer) is LeakyReLU [17].

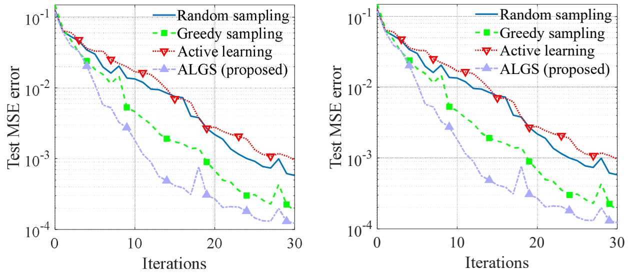
The DNN architecture used for the DNN ensemble is shown in Fig. 6. The number of input variables n = 2. There are four hidden layers with 256, 128, 64, and 32 neurons. The LeakyReLU [17] function is used as the activation function after the first three hidden layers, and the last layer is a linear fully-connected layer without any activation functions. For a fair comparison, all the methods use an ensemble of five DNNs with different weight initializations. The random sampling and greedy sampling methods only use the DNNs to obtain their average results. The active learning and the proposed ALGS

Authorized licensed use limited to: Missouri University of Science and Technology. Downloaded on February 17,2023 at 19:39:10 UTC from IEEE Xplore. Restrictions apply.

method use the DNNs to calculate the uncertainties of unlabelled samples and also acquire the average output values.

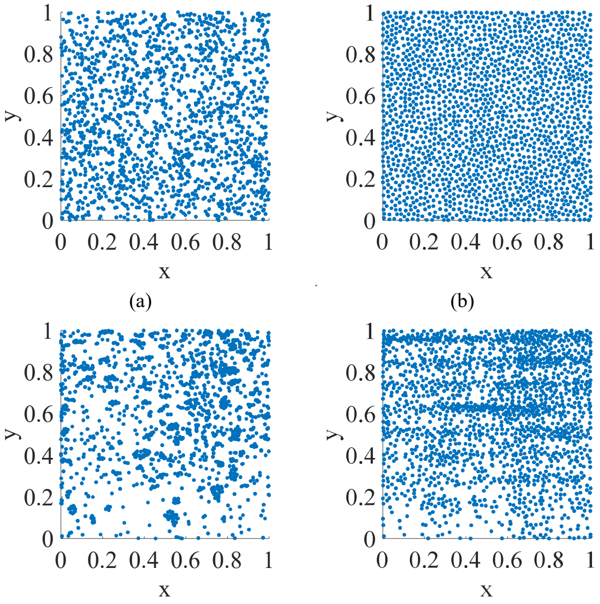
Other hyper-parameters of the data selection process include: N_0= 200, N_p = 10000, N_u = 200, N_d = 50 (namely, 50 new samples are added in each iteration), the learning rate is 0.001, the training batch size is 512, the test data size (randomly generated) is 200, and the Adam optimizer is used. Moreover, after new data is obtained in each iteration, the DNNs are not trained from scratch but started from the states saved in the previous iteration to save the training time.

Fig. 7 compares the testing mean square error (MSE) of the four methods as the number of iterations increases. Fig. 7 (a) shows the testing MSE by averaging the outputs of the DNN ensemble, while Fig. 7 (b) shows the maximum testing MSE of the DNN ensemble. It can be observed that the proposed ALGS method has the lowest testing MSE. Also, comparing Fig. 7(a) and (b), it can be found that averaging the output of the DNN ensemble further reduces the testing MSE.



(a) (b)

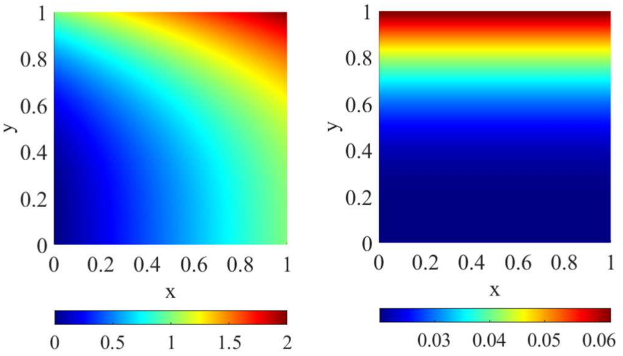
Fig. 7. (a) Testing error comparison using the averaging values of the DNN ensemble for function $f_1(x,y)$. (b) Testing error comparison using the maximum prediction error from the DNN ensemble for function $f_1(x,y)$.



(a) (b) (c) (d)

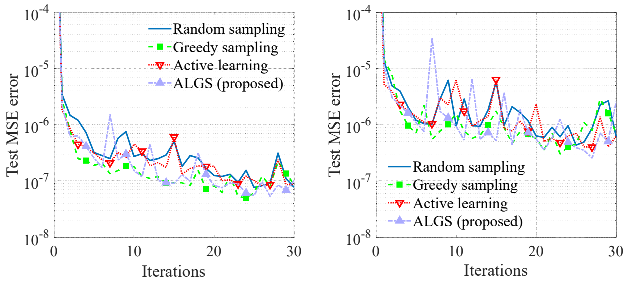
Fig. 8. Sample distribution comparison of the four methods. (a) Random sampling. (b) Greedy sampling. (c) Active learning. (d) ALGS.

The sample distributions of the four methods are plotted in Fig. 8. It can be observed that the greedy sampling method tends to sample the data uniformly on the input space. In the active learning method, some of the samples are clustered together as expected since diversity is not considered. Comparing Fig. 8 (d) and Fig. 5, it can be noticed that the proposed ALGS method samples denser points over the regions with greater gradient values. This is reasonable because, intuitively, the areas with more function variations should be sampled more densely to improve the fitting accuracy.



(a) (b)

Fig. 9. (a) Function $f_2(x,y)$. (b) Gradient of function $f_2(x,y)$.



(a) (b)

Fig. 10. (a) Testing error comparison using the averaging values of the DNN ensemble for function $f_2(x,y)$. (b) Testing error comparison using the maximum prediction error from the DNN ensemble for function $f_2(x,y)$.

Another simple function $f_2(x,y) = x + y^3$ with weaker functional variations is used to validate the proposed approach. The function and its gradient distribution are shown in Fig. 9. The hyper-parameters are the same as the verification using function $f_1(x,y)$. The testing MSE comparison is shown in Fig. 10. Different from the function $f_1(x,y)$, the proposed ALGS approach does not have advantages over other methods for the function $f_2(x,y)$. This indicates that the proposed ALGS method is only advantageous for problems with large functional variations. But still, averaging the output results from the DNN ensemble can effectively reduce the prediction error, as shown in the comparison between Fig. 10 (a) and (b).

IV. Validation Using Stripline Modeling

A high-dimensional stripline modeling problem is introduced in this section to demonstrate the reduction in the training data for the proposed ensemble-based ALGS method. The input and output parameters are explained, and the DNN training process is clarified.

Authorized licensed use limited to: Missouri University of Science and Technology. Downloaded on February 17,2023 at 19:39:10 UTC from IEEE Xplore. Restrictions apply.

A. Stripline Model

Even though modeling differential striplines using DNN has been successful [1], [2], little investigation has been performed on intelligently selecting suitable training sets and improving prediction accuracy. To validate the effectiveness of the proposed method, a differential stripline modeling scenario is chosen with 14 input variables. The stripline cross-section structure is shown in Fig. 11, and the physical meaning and variation range of each input variable are listed in Table I.

The output variable for the DNN models to learn is the differential impedance at 1 GHz, which ranges from about 20 Ohms to 180 Ohms. Even though some input variables in Table I are not closely related to the differential impedance, they are still considered in the input parameters to validate the capability of the proposed method to handle high-dimensional modeling tasks.

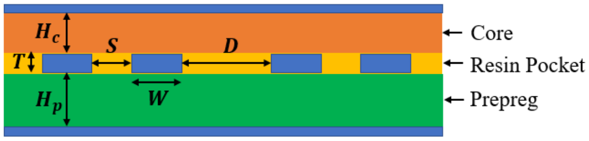


Fig. 11. Illustration of the differential stripline structure being modeled.

TABLE I. Input and output parameter definition

Parameter	Meaning	Range
x_1	Trace width (W)	2-16 mils
x_2	Trace spacing (S)	2-16 mils
x_3	Core height (H_c)	2-10 mils
x_4	Core DK (DK_c)	2.5-4.5
x_5	Core DF (DF_c)	0.001-0.03
x_6	Trace height (T)	0.4-2.6 mils
x_7	Resin Pocket DK (DK_t)	2.5-4.5
x_8	Resin Pocket DF (DF_t)	0.001-0.03
x_9	Prepreg height (H_p)	2-30 mils
x_{10}	Prepreg DK (DK_p)	2.5-4.5
x_{11}	Prepreg DF (DF_p)	0.001-0.3
x_{12}	Roughness level (R)	-10-10
x_{13}	Pair-to-pair distance (D)	6-40 mils
x_{14}	Etch factor (E)	0.1-1
y	Differential impedance (Z)	(From simulator)

A commercial 2D cross-section simulation tool is employed to compute the differential impedance given the 14 input parameters. Each simulation consumes 1 minute on average. To generate a large number of simulation cases, 5000, for example, approximately 84 hours in total are required. The active learning technique can effectively relieve this high burden of time and computational resource consumption, as will be shown later.

B. DNN Training

The DNN structure for the ensemble of DNNs is illustrated in Fig. 6. The input layer has 14 neurons, and the output layer has one neuron. Seven hidden layers with 512, 1024, 256, 128, 64, 32, and 16 neurons are embedded between the input and output layers.

As shown in Table I, different input variables have different variation ranges. To equalize the weighted contribution from each variable and accelerate the training convergence, each variable is linearly normalized between 0 and 1.

The hyper-parameters of the training process include: N_0= 100, N_p = 20000, N_u = 400, N_d = 100 (100 new samples are added in each iteration), the learning rate is 0.0005, the training batch size is 256, the test data size (randomly generated) is 500, and the Adam optimizer is used. The DNN training is performed on an NVIDIA Tesla K80 GPU.

C. Validation Result

Similar to the verification using the simple functions, the four methods are compared in the stripline modeling problem. The loss convergence comparison between the four methods is shown in Fig. 12. As observed in Fig. 12, the proposed ALGS method does not have significant advantages over the other three methods. The possible reason is that the stripline modeling problem does not have enough complexity and function variations to demonstrate the advantages of our proposed method. The relationship between the impedance value and each relevant input parameter by fixing the other input parameters can be considered monotonic. Even so, by averaging the results of the DNN ensemble, the prediction error can be greatly reduced, as shown in Fig. 12. With the same amount of training data, the prediction error using the DNN ensemble is lower than the prediction error using a single DNN. In other words, to achieve the same desired prediction accuracy, the DNN ensemble can significantly reduce the required data size.

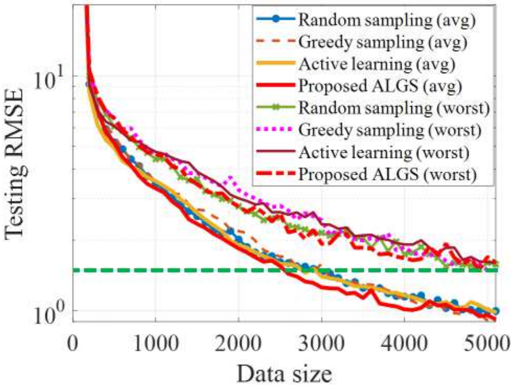


Fig. 12. Loss convergence of the four methods: random sampling, greedy sampling [12], [10], active learning, and the proposed ALGS method.

The training time for the five DNN models in each iteration is approximately 2 minutes in the active learning algorithm, which is negligible compared to the consumed time for simulating 100 new samples. Even though training the ensemble of five DNNs takes some extra time, the simulation time of generating data is still dominant. Therefore, the ensembled-based AL method can effectively reduce the required data size and the corresponding data acquisition time. For example, assume the dashed horizontal line in Fig. 12 is the desired testing accuracy. In that case, the proposed ALGS method by averaging the results of the DNN ensemble only

Authorized licensed use limited to: Missouri University of Science and Technology. Downloaded on February 17,2023 at 19:39:10 UTC from IEEE Xplore. Restrictions apply.

needs approximately 2500 groups of data (about 42 hours of simulation time) to achieve this accuracy, while the other methods using a single DNN require about 5000 groups of data (about 84 hours of simulation time).

The histogram of the impedance prediction errors for the 500 testing cases is plotted in Fig. 13. It is shown that the impedance prediction errors can be significantly reduced by averaging the prediction results of the five DNNs. In this stripline modeling problem, the advantage of the proposed method is dominated by the averaging effect of the DNN ensemble.

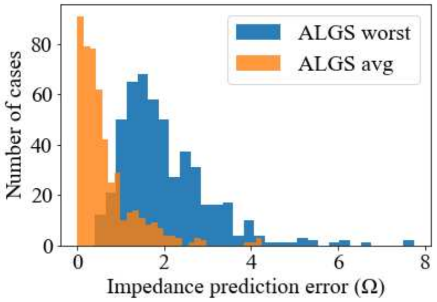


Fig. 13. Histogram of the impedance prediction errors for the 500 testing cases. “ALGS worst” means the worst prediction error of the five DNNs using the proposed ALGS method; “ALGS avg” means the prediction error by averaging the predictions of the five DNNs using the proposed ALGS method.

V. Conclusion

This paper presents an effective DNN-ensemble-based approach to reducing the prediction error and selecting valuable training data for DNN-based surrogate modeling. The DNN ensemble is utilized to estimate the uncertainties of unlabelled data samples. Also, by averaging the prediction results of the DNN ensemble, the prediction error can be further reduced. To prevent the samples with high uncertainties from being clustered together, a greedy sampling method is incorporated to select a data subset with diversity. The proposed method is validated by two simple functions. It is found that the proposed method shows more advantages in training data selection for the function with larger functional variations. For the function with small function variations, the proposed method does not exhibit enough advantages in the data selection. However, averaging the prediction results of the DNN ensemble is always effective in improving the prediction accuracy. A high-dimensional differential-stripline modeling problem is further used to validate the effectiveness of this approach. The proposed approach is not advantageous over other sampling methods in the data selection, which may be due to the simple function variations in the stripline modeling problem. But by averaging the output results of the DNN ensemble, the impedance prediction error can still be effectively decreased. The proposed DNN-ensemble-based method is robust and task agnostic, which can be applied in various EMC modeling problems using DNN that require a long simulation time for data generation.

The proposed DNN-ensemble-based method will be applied to other high-dimensional EMC modeling problems with complex functional variations in the near future. Further, how to appropriately select the hyper-parameter values to achieve an optimum balance between uncertainty and diversity will be investigated.

Acknowledgment

This paper is supported in part by the Zhejiang Provincial Natural Science Foundation of China (ZPNSFC), Grant No. LD21F010002, and Natural Science Foundation of China (NSFC), Grant No. 62071424 and 62027805.

References

[1] H. Kim, C. Sui, K. Cai, B. Sen, and J. Fan, “Fast and Precise High-Speed Channel Modeling and Optimization Technique Based on Machine Learning,” *IEEE Trans. Electromagn. Compat.*, vol. 60, no. 6, pp. 2049-2052, Dec. 2018.

[2] S. Penugonda, S. Yong, A. Gao, K. Cai, B. Sen, and J. Fan, “Generic Modeling of Differential Striplines Using Machine Learning Based Regression Analysis,” *Proc. Int. Symp. IEEE Electromagn. Compat.*, pp. 226-230, 2020.

[3] H. M. Torun and M. Swaminathan, “High-Dimensional Global Optimization Method for High-Frequency Electronic Design,” *IEEE Trans. Microw. Theory Tech.*, vol. 67, no. 6, pp. 2128-2142, June 2019.

[4] T. Lu, J. Sun, K. Wu, and Z. Yang, “High-Speed Channel Modeling With Machine Learning Methods for Signal Integrity Analysis,” *IEEE Trans. Electromagn. Compat.*, vol. 60, no. 6, pp. 1957-1964, Dec. 2018.

[5] L. Zhang, J. Juang, Z. Kiguradze, et al., “Fast impedance prediction for power distribution network using deep learning,” *Int J Numer Model*, e2956, 2021.

[6] H. Jin, Z. Gu, T. Tao, and E. Li, “Hierarchical Attention-Based Machine Learning Model for Radiation Prediction of WB-BGA Package,” *IEEE Trans. Electromagn. Compat.*, vol. 63, no. 6, pp. 1972-1980, Dec. 2021.

[7] H. Liu, Y.S. Ong, and J. Cai, “A survey of adaptive sampling for global metamodeling in support of simulation-based complex engineering design,” S*truct. Multidiscipl. Optim.*, 57(1), pp. 393-416, 2018.

[8] Y. Gal, R. Islam, and Z. Ghahramani, “Deep Bayesian active learning with image data,” *International Conference on Machine Learning (ICML)*, 2017.

[9] R. Caramalau, B. Binod, and T. K. Kim, “Active learning for bayesian 3d hand pose estimation,” In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 3419-3428. 2021.

[10] O. Sener, and S. Silvio, “Active learning for convolutional neural networks: A core-set approach,” *arXiv preprint arXiv*, 1708.00489, 2017.

[11] W. H. Beluch, T. Genewein, A. Nürnberger, and J. M. Köhler, “The power of ensembles for active learning in image classification,” In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 9368-9377, 2018.

[12] D. Wu, C. Lin, and J. Huang, “Active learning for regression using greedy sampling,” *Information Sciences*, 474, pp. 90-105, 2019.

[13] Y. Yang, Z. Ma, F. Nie, X. Chang, and A. G. Hauptmann, “Multi-class active learning by uncertainty sampling with diversity maximization.” *International Journal of Computer Vision* 113, no. 2: 113-127, 2015.

[14] A. Kirsch, J. Van Amersfoort, and Y. Gal, 2019. “Batchbald: Efficient and diverse batch acquisition for deep bayesian active learning,” *Advances in neural information processing systems*, 32, 2019.

[15] R. Burbidge, J. J. Rowland, and R. D. King, “Active learning for regression based on query by committee,” In *International conference on intelligent data engineering and automated learning*, Springer, Berlin, Heidelberg, pp. 209-218. 2007.

[16] https://pytorch.org/.

[17] B. Xu, N. Wan, T. Chen, and M. Li, “Empirical evaluation of rectified activations in convolutional network,” *arXiv Preprint arXiv*, 1505.00853, 2015.

Authorized licensed use limited to: Missouri University of Science and Technology. Downloaded on February 17,2023 at 19:39:10 UTC from IEEE Xplore. Restrictions apply.