Deep Learning Framework for Material Design Space Exploration using Active Transfer Learning and Data Augmentation

Yongtae Kim[1†], Youngsoo Kim[2†], Charles Yang[3], Kundo Park[1], Grace X. Gu[3], Seunghwa Ryu[1*]

Affiliations

[1]Department of Mechanical Engineering & KI for the NanoCentury, Korea Advanced Institute of Science and Technology (KAIST), 291 Daehak-ro, Yuseong-gu, Daejeon 34141, Republic of Korea

[2]Department of Nature-Inspired System and Application, Korea Institute of Machinery and Materials, Daejeon, (34103) 156, Republic of Korea

[3]Department of Mechanical Engineering, University of California, Berkeley, CA 94720, USA

[†]These authors contributed equally to this work.

*corresponding author: ryush@kaist.ac.kr

Keywords: Deep Learning; Active-Transfer learning; Data augmentation; Regression; Material design;

Abstract

Neural network-based generative models have been actively investigated as an inverse design method for finding novel materials in a vast design space. However, the applicability of conventional generative models is limited because they cannot access data outside the range of training sets. Advanced generative models that were devised to overcome the limitation also suffer from the weak predictive power on the unseen domain. In this study, we propose a deep neural network-based forward design approach that enables an efficient search for the superior materials far beyond the domain of the initial training set. This approach compensates for the weak predictive power of neural networks on an unseen domain through gradual updates of the neural network with active-transfer learning and data augmentation methods. We demonstrate the potential of our framework with a grid composite optimization problem that has an astronomical number of possible design configurations. Results show that our proposed framework can provide excellent designs close to the global optima, even with the addition of very small dataset corresponding to less than 0.5% of the initial training dataset size.

1. Introduction.

In order to discover or design novel materials having outstanding properties, significant effort has been paid to devise various material design approaches such as biomimicry, design of experiment methods, and other conventional optimization methods. [1-13]. However, these approaches often require in-depth physics-based analysis on the relationship between materials descriptors and properties. Hence, fundamental understanding on the underlying physical mechanisms determining the material properties is a primer for the material design. Machine learning models are alternative promising tools for materials design, because they enable design space exploration only with database representing the relationship between the descriptors of material (inputs) and the properties (outputs). Trained machine learning models can infer the relationship with several orders of magnitude speed up compared to actual data generation from experiments or physics-based simulations tools [14-23]. In many applications, the machine learning models, such as Gaussian process regression, radial basis function network, support vector machine, and deep neural network (DNN), are adopted as surrogate forward models, which predict the outputs from the corresponding inputs [24-26]. However, it requires a lot of effort to find desired materials in a vast design space with a forward design approach, because a large number of candidates must be tested to search for the optimal material due to the absence of the gradient of predicted output with respect to input features [19, 20, 27-29].

In this regard, inverse design methods, which adapt machine learning models as a designer directly suggesting promising candidate materials based on target properties, are being intensively studied to avoid the aforementioned arduous design space exploration process [18, 30-35]. Autoencoder (AE), variational autoencoder (VAE), and generative adversarial network (GAN) are three of the most commonly used generative models [18, 33, 35]. However, these

models cannot generate data in the unseen domain, i.e., the domain outside the ranges of the training dataset defined by the input feature space and output values. Hence, their applicability is limited, because it is computationally infeasible to generate training data large enough to cover the entire high-dimensional design space of most material design problems. Recently, Chen and Gu proposed generative inverse design network capable of suggesting promising candidates beyond the domain of training data based on backpropagation of DNN [32]. However, DNN is likely to have poor predictive performance on unseen domain unless the DNN learns a governing equation representing the relationship between the inputs and the outputs which requires domain expertise and some heuristics to create an appropriate mathematical formulation [36-42]. Thus, in most cases, it is inevitable to rely on the poor predictive power of DNN and repeat the validation of many suggested promising candidates through laborious simulations or experiments until the material with the desired property is obtained [32].

In this study, we propose a systematic neural network-based forward design approach to efficiently search for the desired materials outside the training data domain which overcomes the aforementioned limitations of the existing methods. It is known that DNN trained by gradient descent algorithms with unbounded activation function is capable of making reliable prediction on the unseen domain close to training dataset with linear approximation [37, 43, 44]. Hence, our framework gradually expands the reliable prediction domain of DNN toward the region of desired properties by updating DNN via active-transfer learning. Relatively sparse and small additional dataset including materials with incrementally superior properties is iteratively added to the training set based on a data augmentation technique to increase generalization of DNN, i.e., ability to make accurate and stable prediction on unseen data [45]. The limitation of a forward design approach is mitigated by using a hyper-heuristic genetic algorithm on top of the updated DNN. Our study demonstrates that materials with desired

properties can be designed out of inferior original training dataset with small dataset augmentation and validation.

2. Results

2.1. Schematic of the deep neural network (DNN) based forward design framework.

The schematic of our framework is depicted in Figure 1. DNN trained with the initial training dataset is capable of making reliable prediction on the design space slightly larger than the training data domain, as represented in bluish region. To find the materials with desired properties, which are positioned outside the domain of initial training data, DNN should be able to make a reliable prediction on the domain containing the desired design. In this regard, using the trained DNN to predict the properties of new material designs proposed by the genetic algorithm, a relatively small set of materials superior to those in the existing dataset is suggested. Since the newly proposed materials are outside of the current training set and the DNN predictions on them are not accurate, their properties are evaluated again with accurate physics-based simulations (if high throughput experimental facility is available, one can use experiments). Those data are integrated to the training data set with a data augmentation technique, and the DNN is updated based on the new training data with active-transfer learning as represented with black arrow. This process is repeated until the DNN is able to make a reliable prediction in the domain close to the optimal point represented as the large red point. The DNN after the last update is used to find the optimal design. The detailed explanation of our framework is provided in the following sections.

2.2. Architectures of the deep neural network (DNN) leveraging prediction upon unseen domain.

To leverage the prediction of DNN on unseen domain, the DNN architecture consists of unbounded activation function, i.e., leaky rectified linear unit activation function (leaky ReLU) with coefficient 0.1 [39, 44] (See the comparison with other activation functions in Figure S1). The architectures are constructed based on residual network (Resnet) with full pre-activation, which is known for good generalization performance with sufficient number of learnable parameters, with batch normalization layer as regularization methods [46, 47] (See the details of DNN architecture in Supplementary Information.). We check the predictive performance of the DNN in seen and unseen domain by setting the randomly chosen 10% of data as training sets, and the dataset with the highest 10% output values as validation sets, respectively. In the optimization procedure, one has to explore the dataset having output values higher than those of initial training set. Hence, validation on the dataset with highest 10% output values for the DNN trained on the dataset with lowest 90% output values would represent the predictive performance of the DNN in unseen domain during the optimization procedure. The flowchart representing the process for constructing the DNN architecture is depicted in Figure S3.

2.3. Prediction results upon seen domain and unseen domain.

In this study, we demonstrate the applicability of our framework by solving a representative problem with a large design space - the design of composite microstructures with superior mechanical properties, i.e., stiffness, strength, and toughness, which are close to the global optimum located far beyond the domain of initial training data. The details of data generations are presented in the methods section and Figure 2. The training results of DNN

regarding stiffness and strength upon seen and unseen domain are represented in Figure 3. The prediction accuracy gradually decrease as data is located further away from the training data, as reported in the literature [37, 44]. The training results on seen domain generally show better results compared to the results on unseen domain, as expected. Still, despite the mismatch in the absolute values, the DNN network is able to distinguish relative magnitudes to some extent, as shown in Figure 3 (c)-(d). However, the training results for toughness show that it is infeasible to make prediction on unseen output value range (Figure S4). We suspect that the poor predictive performance on toughness originates from the complexity in determining toughness from the entire stress-strain curve encompassing the full failure process involving complex crack propagation and branching process. Hence, we leave the optimization of toughness as a future work. In the stiffness and strength training results in unseen domain, the DNN predictions gradually deviate more as the data positioned further away from the training dataset.

2.4. Material design process based on the deep neural network (DNN) prediction in unseen domain.

A flowchart of the proposed material design framework is depicted in Figure 4. DNN tested with randomly selected validation set are adopted to allow reliable prediction in a broader domain of output values. For the genetic algorithm, 30 microstructures having highest properties are selected for the mating pool as greedy sampling method to enable sufficient genomic variations at each generation. We note that selecting proper amount of data for the mating pool is important when considering the trade-off between the risk of being stuck in local minima and computational time. Additionally, we utilize the intuition from solid mechanics that the symmetrical microstructure is beneficial for the load bearing capacity and that soft

material at the crack tip is able to relieve the stress concentration at the crack tip [1, 4]. A hyper-heuristic genetic algorithm combining these domain knowledges is implemented by constraining the explored microstructure design to satisfy the prescribed conditions. The constraints accelerated the optimization process compared to conventional particle swarm optimization and genetic algorithm (See the details of the comparison for each optimization methods in the Supplementary Information). In genetic algorithm, the crossovers are implemented by selecting two microstructures from the mating pool as parents and randomly assigning stiff material to the area occupied by stiff materials in parent configurations. The mutations are applied by randomly switching the position of stiff material block and soft material block by keeping the ratio between the stiff and soft blocks. Approximately 4×10^4 unique candidate microstructures are generated from mating pool at each generation.

The output values of the candidate microstructures are predicted with the DNN. Because the microstructures proposed by genetic algorithm are located close to the microstructures from the previous generation in terms of feature space and output values, the DNN could make reliable predictions on the microstructures suggested by the genetic algorithm for certain generations. Based on those predictions, we select candidate microstructures for the mating pool, and continue this process until DNN-predicted output values do not show improvement, i.e., convergence to predictable limit of DNN. After the convergence, we calculate the true output values of the candidate microstructures in the mating pool with FEA simulations and integrate them to the training data with data augmentation techniques. The data augmentation is implemented by oversampling the new data by 50 times. By employing the data augmentation methods, emphasis is placed on the generalization on unseen domain, as suggested by previous study [45, 48, 49]. Without the data augmentation, the added data are diluted in the training data because it is relatively sparser and smaller compared to the existing training dataset. We note that different data augmentation scheme can

be adopted depending on the characteristics of design problem. The update of DNN is conducted by re-training the DNN based on the integrated training data with reduced learning rate (10^{-6}) and reduced training epoch (10) in transfer learning scheme. Because DNN requires relatively moderate update on learnable parameters to expand the reliable prediction domain, it takes a dramatically reduced time for DNN updates compared to the initial training.

2.5. Design of 11×11 grid composite microstructures with maximum mechanical properties.

The design process for maximizing stiffness and strength are represented in Figure 5. The data points at 0^{th} update represent the initial training dataset. In the design process for maximizing stiffness shown in Figure 5 (a), approximately 1.6×10^5 unique microstructures are investigated within 4 hours through DNN. This corresponds to the several orders of magnitude speed up compared to FEA simulation tools which would take more than 8 weeks for the generation of 10^5 data. In the design process for maximum stiffness, the significant improvement is observed with the initial training of DNN, because the stiffness prediction upon unseen output value range indicates that the DNN is able to distinguish the relatively magnitudes of stiffness despite some mismatch in absolute values. We hypothesize that the moderate level of extrapolation beyond the seen domain is made possible due to the piecewise linear unbound function from leaky ReLU used in the DNN. Indeed, our test on various activation functions shows that unbounded ReLU-based DNN has reasonable predictive power on unseen domain, while bounded activation functions is inefficient for the task (Figure S1.). The iterative update of DNN is also beneficial in making robust improvement on local minima explorable with the DNN and in escaping from local minima, as demonstrated by the gradual improvement of converged values shown in Figure 5 (a). Given that almost identical microstructure designs come out from the 3^{rd} and 4^{th} updated of DNN, we believe that the

design process is converged at the optimization process after 4th update. Identical process is also applied for finding the microstructure with the maximum strength, which is shown in Figure 5 (b). The maximum strength value increases until we update the DNN iteratively five times.

The material designs having six highest stiffness and strength at the last DNN update are depicted in Figure 6. The best microstructures for stiffness and strength are positioned at the upper left corners of Figure 6 (a) and Figure 6 (b), respectively. Microstructures ranked at 2nd to 6th are visualized at the bottom. The theoretical upper bound of elastic modulus (E_{max}) of composite having volume fraction of stiff materials (f) can be achieved with following equation: $E_{max} = E_{stiff}f + E_{stiff}(1-f)$. For our design problem with $f = 71/121$, E_{max} is 1223.7 MPa if it were not for the pre-existing crack. The stiffness of optimized microstructures (1144.7 MPa) is close to the theoretical upper bound even though a pre-crack exists that inherently reduces the stiffness of structure. Therefore, we can infer that the maximum stiffness design found from our framework is close to the global optimum. For the composite with maximum strength, the strength is significantly improved compared to the best design in the training set. In those designs, a majority of stiff blocks are located in the region far from the pre-crack (right side), because such unbalanced material distribution leads to the increased stiffness while reducing the negative effect from the pre-crack. At the same time, since the crack initiation may occur any point with high stress concentration other than the pre-existing crack tip, it is important to reduce the stress concentration at the vertices made by different material blocks. Figure 6 (b) shows that the stress concentration at the crack tip is relived owing to the optimized structure compared to some randomly chosen composite structures in the initial training data (Figure S8). As a result, we could obtain the design that is more than twice as strong as the best design in original data set, (i.e., the maximum strength

increases from 0.076165 MPa to 0.16276 MPa throughout the optimization process). The proposed designs also outperform the composite structures which are manually designed based on the physical intuitions (See the details in the Supplementary information). The results for maximum stiffness and strength are obtained only by augmenting 366 and 424 microstructures, respectively. The augmented dataset size is about 0.4% of the initial training dataset size. The histograms for the additional data are depicted in Figure 7.

3. Discussion

In this study, we propose a systematic forward material design framework to obtain superior design far beyond the domain of initial training dataset. The framework is applied to a composite microstructure design problem for obtaining the maximum mechanical properties of 11×11 grid composite, which has an astronomically high number of possible configurations. Because this type of design process inherently requires the efficient search in unseen domain, a forward design approach is adopted by gradually expanding the reliable prediction domain of DNN with active-transfer learning and data augmentation. The better design candidates are firstly proposed by the genetic algorithm based on DNN predictions. The properties of the candidate are obtained with FEA calculations before they are augmented to the training data set, in order to secure the predictive performance of the DNN for newly added dataset. The limitations in forward design, such as stuck in local minima, are mitigated by updating DNN and controlling the mutation methods in a hyper-heuristic genetic algorithm. As a result, our framework enables the design of superior composite microstructure with very small additional data generation within a vast design space. We expect that the proposed framework can be applied to various optimization problems with astronomical number of design space.

4. Methods

4.1. Data generation based on finite element analysis (FEA)

As a training data set, the mechanical properties, i.e., stiffness, strength, and toughness, of two-dimensional 11×11 grid composites are obtained from finite element analysis (FEA) under plane stress condition. The composites consist of perfectly bonded 70 stiff material blocks and 51 soft material blocks containing a pre-existing crack. The input features are formatted as one-hot binary encoding representing the position of stiff and soft materials, and the outputs are defined as corresponding mechanical properties. The 100,000 unique microstructures are constructed, and the corresponding stress-strain curves are obtained by applying uniaxial tension with quasi static infinitesimal strain increment of 0.0000227 until the complete fracture occurs. 100,000 data correspond to the fraction of ~10^{-29} out of the total number of available combinations, $\mathrm{C}(120,70) \sim 10^{34}$. The stiffness, strength, and toughness are measured by the initial gradient of a stress-strain curve, the maximum stress point, and the total area under the stress-strain curve, respectively. The materials are assumed as linear elastic materials with the following elastic modulus (E), Poisson's ratio (v) and critical strain energy release rate (G_c), respectively, for stiff and soft materials: $E^{stiff} = 2100\ MPa, v^{stiff} = 0.3, G_c^{stiff} = 50\ J/m^2, E^{soft} = 21\ MPa, v^{soft} = 0.3,\ G_c^{soft} = 50 J/m^2$. For the simulation of the crack growth, the hybrid crack phase field (CPF) model are adopted in commercial FEA software ABAQUS with user-element subroutine [50]. The hybrid CPF model enables the modeling of complex crack nucleation and propagation by describing the failure of material with continuous scalar field called crack phase field $d(x)$. The material completely loses its load bearing capacity when $d(x)$ reach to 1, whereas $d(x) = 0$ indicates the absence of any

damage. The detailed explanation and implementation of the model can be found in our previous study [50].

Data availability

The data obtained in this study are available from the authors upon reasonable request.

Code availability

The codes developed in this study are available from the authors upon reasonable request.

Corresponding authors

Correspondence to Seunghwa Ryu

Acknowledgements

This research was supported by Basic Science Research Program (2019R1A2C4070690) and Creative Materials Discovery Program (2016M3D1A1900038) through the National Research Foundation of Korea (NRF) funded by the Ministry of Science and ICT (MSIT) of the Republic of Korea, as well as the KAIST-funded Global Singularity Research Program for 2019 (N11190118). Additionally, the authors acknowledge funding support from 3M and HP, Inc.

Author contributions

Yongtae Kim and Youngsoo Kim conducted the research under the guidance of Seunghwa Ryu. Yongtae Kim and Seunghwa Ryu wrote this manuscript. Charles Yang and Kundo Park contributed to the results analysis. Grace X. Gu revised the paper. All authors revised and approved the final manuscript.

Competing interests

The authors declare no competing interests.

Reference

1. Libonati, F., et al., *Bone-inspired materials by design: toughness amplification observed using 3D printing and testing.* Advanced Engineering Materials, 2016. **18**(8): p. 1354-1363.
2. Sigmund, O. and S. Torquato, *Design of materials with extreme thermal expansion using a three-phase topology optimization method.* Journal of the Mechanics and Physics of Solids, 1997. **45**(6): p. 1037-1067.
3. Sigmund, O. and S. Torquato, *Design of smart composite materials using topology optimization.* Smart Materials and Structures, 1999. **8**(3): p. 365.
4. Kim, Y., et al., *Designing tough isotropic structural composite using computation, 3D printing and testing.* Composites Part B: Engineering, 2019. **167**: p. 736-745.
5. Guan, A.-Y., et al., *Discovery of pyridine-based agrochemicals by using intermediate derivatization methods.* Bioorganic & medicinal chemistry, 2016. **24**(3): p. 342-353.
6. Zhu, H., et al., *Discovery of TaFeSb-based half-Heuslers with high thermoelectric performance.* Nature communications, 2019. **10**(1): p. 1-8.
7. Carraturo, M., et al., *Graded-material design based on phase-field and topology optimization.* Computational Mechanics, 2019. **64**(6): p. 1589-1600.
8. Chen, C.-T., D.C. Chrzan, and G.X. Gu, *Nano-topology optimization for materials design with atom-by-atom control.* Nature communications, 2020. **11**(1): p. 1-9.
9. Holder, A.M., et al., *Novel phase diagram behavior and materials design in heterostructural semiconductor alloys.* Science advances, 2017. **3**(6): p. e1700270.
10. Kim, Y., et al., *A three-dimensional fracture pattern diagram of staggered platelet structures.* Composite Structures, 2019. **220**: p. 769-775.
11. Radman, A., X. Huang, and Y. Xie, *Topological design of microstructures of multi-phase materials for maximum stiffness or thermal conductivity.* Computational materials science, 2014. **91**: p. 266-273.
12. de Kruijf, N., et al., *Topological design of structures and composite materials with multiobjectives.* International Journal of Solids and Structures, 2007. **44**(22-23): p. 7092-7109.
13. Osanov, M. and J.K. Guest, *Topology optimization for architected materials design.* Annual Review of Materials Research, 2016. **46**: p. 211-233.
14. Pruksawan, S., et al., *Prediction and optimization of epoxy adhesive strength from a small dataset through active learning.* Science and technology of advanced materials, 2019. **20**(1): p. 1010-1021.
15. Lavecchia, A., *Machine-learning approaches in drug discovery: methods and applications.* Drug discovery today, 2015. **20**(3): p. 318-331.
16. Butler, K.T., et al., *Machine learning for molecular and materials science.* Nature, 2018. **559**(7715): p. 547-555.
17. Liu, X., et al., *A machine learning approach to fracture mechanics problems.* Acta Materialia, 2020.

18. Hanakata, P.Z., et al., *Forward and inverse design of kirigami via supervised autoencoder.* Physical Review Research, 2020. **2**(4): p. 042006.
19. Kim, Y., et al., *Designing adhesive pillar shape with deep learning-based optimization.* ACS Applied Materials & Interfaces, 2020.
20. Pilania, G., et al., *Accelerating materials property predictions using machine learning.* Scientific reports, 2013. **3**(1): p. 1-6.
21. Ramprasad, R., et al., *Machine learning in materials informatics: recent applications and prospects.* npj Computational Materials, 2017. **3**(1): p. 1-13.
22. Chen, C.T. and G.X. Gu, *Effect of constituent materials on composite performance: Exploring design strategies via machine learning.* Advanced Theory and Simulations, 2019. **2**(6): p. 1900056.
23. Gu, G.X., et al., *Bioinspired hierarchical composite design using machine learning: simulation, additive manufacturing, and experiment.* Materials Horizons, 2018. **5**(5): p. 939-945.
24. Gu, G.X., C.-T. Chen, and M.J. Buehler, *De novo composite design based on machine learning algorithm.* Extreme Mechanics Letters, 2018. **18**: p. 19-28.
25. Kim, Y., et al., *Designing an Adhesive Pillar Shape with Deep Learning-Based Optimization.* ACS Applied Materials & Interfaces, 2020. **12**(21): p. 24458-24465.
26. Mansouri Tehrani, A., et al., *Machine learning directed search for ultraincompressible, superhard materials.* Journal of the American Chemical Society, 2018. **140**(31): p. 9844-9853.
27. Simpson, T.W., et al., *Kriging models for global approximation in simulation-based multidisciplinary design optimization.* AIAA journal, 2001. **39**(12): p. 2233-2241.
28. Jeong, S., M. Murayama, and K. Yamamoto, *Efficient optimization design method using kriging model.* Journal of aircraft, 2005. **42**(2): p. 413-420.
29. Li, Z., et al., *The optimization via response surface method for micro hydrogen gas actuator.* International Journal of Hydrogen Energy, 2019. **44**(59): p. 31633-31643.
30. Peurifoy, J., et al., *Nanophotonic particle simulation and inverse design using artificial neural networks.* Science advances, 2018. **4**(6): p. eaar4206.
31. Sanchez-Lengeling, B. and A. Aspuru-Guzik, *Inverse molecular design using machine learning: Generative models for matter engineering.* Science, 2018. **361**(6400): p. 360-365.
32. Chen, C.T. and G.X. Gu, *Generative deep neural networks for inverse materials design using backpropagation and active learning.* Advanced Science, 2020. **7**(5): p. 1902607.
33. Kim, B., S. Lee, and J. Kim, *Inverse design of porous materials using artificial neural networks.* Science advances, 2020. **6**(1): p. eaax9324.
34. Hsu, Y.-C., C.-H. Yu, and M.J. Buehler, *Tuning Mechanical Properties in Polycrystalline Solids Using a Deep Generative Framework.* Advanced Engineering Materials.
35. Lim, J., et al., *Molecular generative model based on conditional variational autoencoder for de novo molecular design.* Journal of cheminformatics, 2018. **10**(1): p. 1-9.
36. Marcus, G., *Deep learning: A critical appraisal.* arXiv preprint arXiv:1801.00631, 2018.
37. Martius, G. and C.H. Lampert, *Extrapolation and learning equations.* arXiv preprint

arXiv:1610.02995, 2016.

38. Barnard, E. and L. Wessels, *Extrapolation and interpolation in neural network classifiers.* IEEE Control Systems Magazine, 1992. **12**(5): p. 50-53.
39. Mitchell, J., et al., *Extrapolation in NLP.* arXiv preprint arXiv:1805.06648, 2018.
40. Greydanus, S., M. Dzamba, and J. Yosinski, *Hamiltonian neural networks.* Advances in Neural Information Processing Systems, 2019. **32**: p. 15379-15389.
41. Long, Z., Y. Lu, and B. Dong, *PDE-Net 2.0: Learning PDEs from data with a numeric-symbolic hybrid deep network*. Journal of Computational Physics, 2019. **399**: p. 108925.
42. Kloss, A., S. Schaal, and J. Bohg, *Combining learned and analytical models for predicting action effects.* arXiv preprint arXiv:1710.04102, 2017.
43. Lopedoto, E. and T. Weyde. *ReLEx: Regularisation for Linear Extrapolation in Neural Networks with Rectified Linear Units*. in *International Conference on Innovative Techniques and Applications of Artificial Intelligence*. 2020. Springer.
44. Xu, K., et al., *How neural networks extrapolate: From feedforward to graph neural networks.* arXiv preprint arXiv:2009.11848, 2020.
45. Hooker, G. and S. Rosset, *Prediction-based regularization using data augmented regression.* Statistics and Computing, 2012. **22**(1): p. 237-249.
46. He, K., et al. *Identity mappings in deep residual networks*. in *European conference on computer vision*. 2016. Springer.
47. He, K., et al. *Deep residual learning for image recognition*. in *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.
48. Li, L. and C. Ma. *Transfer Regression with Data-Augmented Ensemble Learning Framework*. in *IOP Conference Series: Earth and Environmental Science*. 2019. IOP Publishing.
49. Campagna, R. and E. Perracchione, *Data-driven extrapolation via feature augmentation based on variably scaled thin plate splines.* arXiv preprint arXiv:2012.12552, 2020.
50. Jeong, H., et al., *Phase field modeling of crack propagation under combined shear and tensile loading with hybrid formulation.* Computational Materials Science, 2018. **155**: p. 483-492.

Figure set

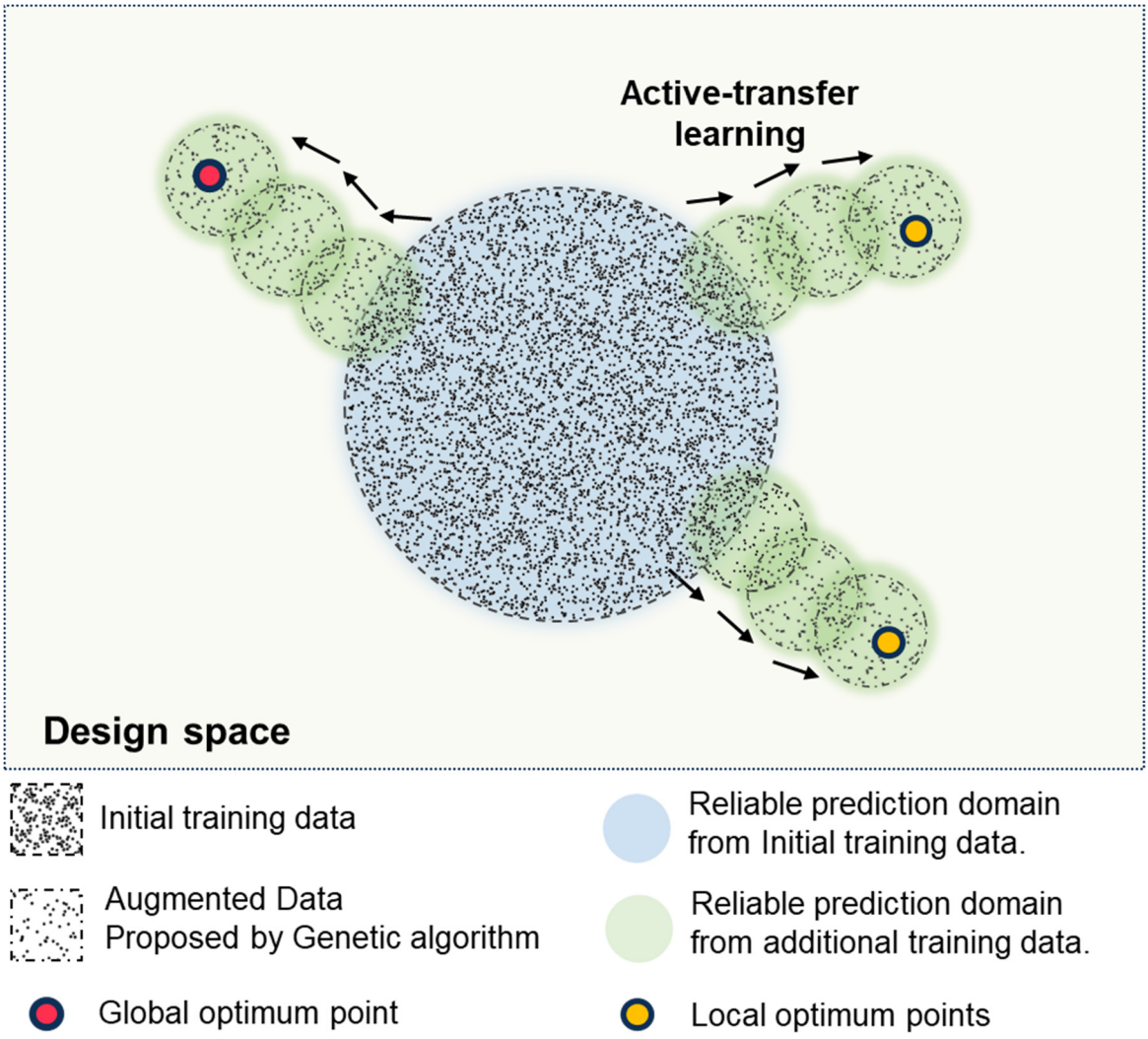

Figure 1. Schematic of gradual expansion of reliable prediction domain of DNN based on addition of data generated from hyper-heuristic genetic algorithm and active-transfer learning.

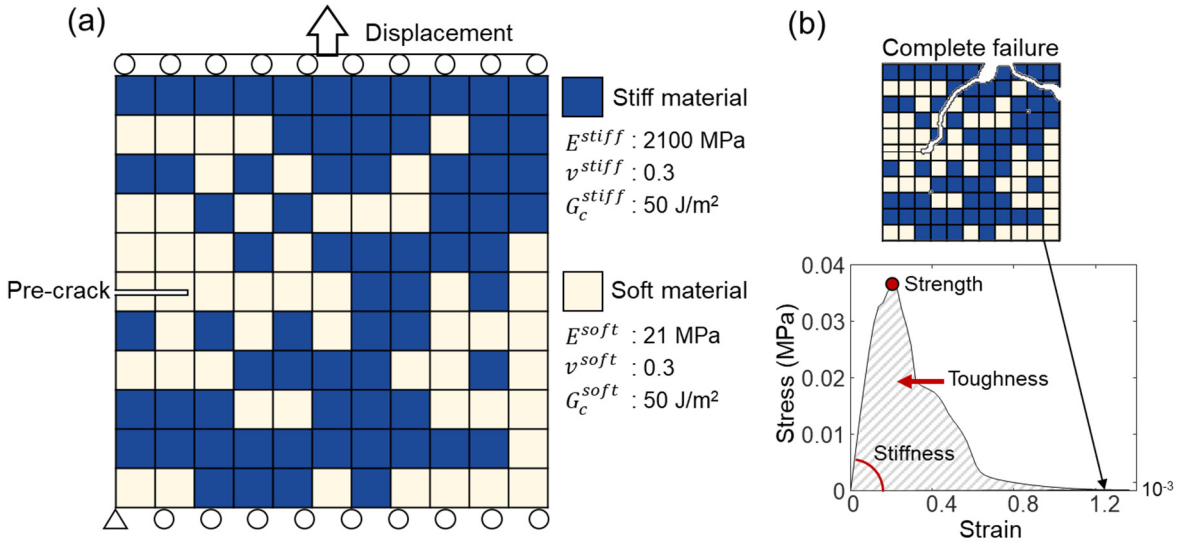

Figure 2. (a) Schematic of 11×11 grid composite composed of stiff and soft materials, and (b) the process for obtaining mechanical properties of composite based on the stress-strain curves. The tensile deformations are applied until the complete failure of composite, i.e., no load bearing capacity.

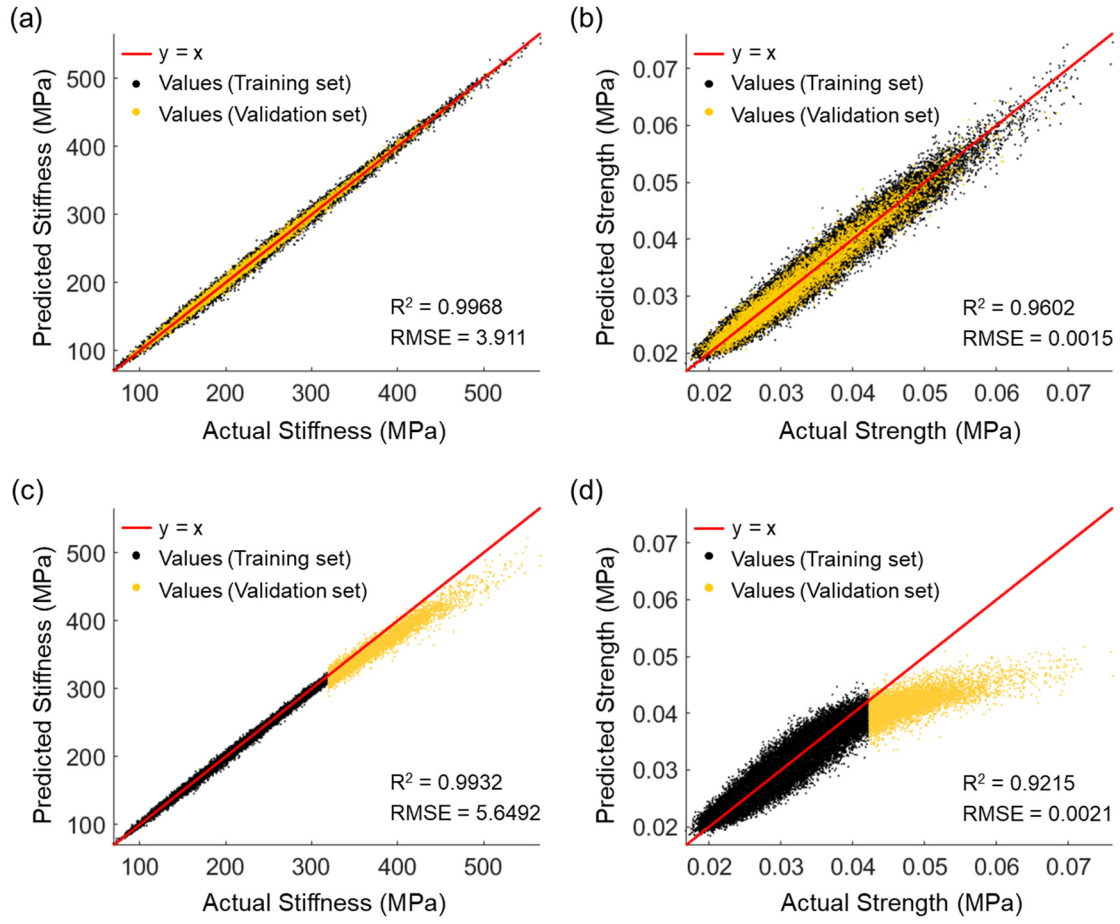

Figure 3. (a-b) DNN training results in terms of stiffness and strength with the validation sets randomly selected 10% of data in the data distribution. (c-d) DNN training results to check generalization on unseen domain by setting the training sets and the validation sets as 90% of data with lowest output value and 10% of data having highest output value, respectively. The coefficient of determination (R^2) are calculated by following equation: $R^2 = 1 - \sum(y_i - \bar{y})^2 / \sum(y_i - f_i)^2$. y_i, f_i and $\bar{y}$ represent the actual value, fitted value and mean of actual values. The root mean squared error (RMSE) are calculated by following equation: $\mathrm{RMSE} = \sqrt{\sum(y_i - f_i)\ /N}$.

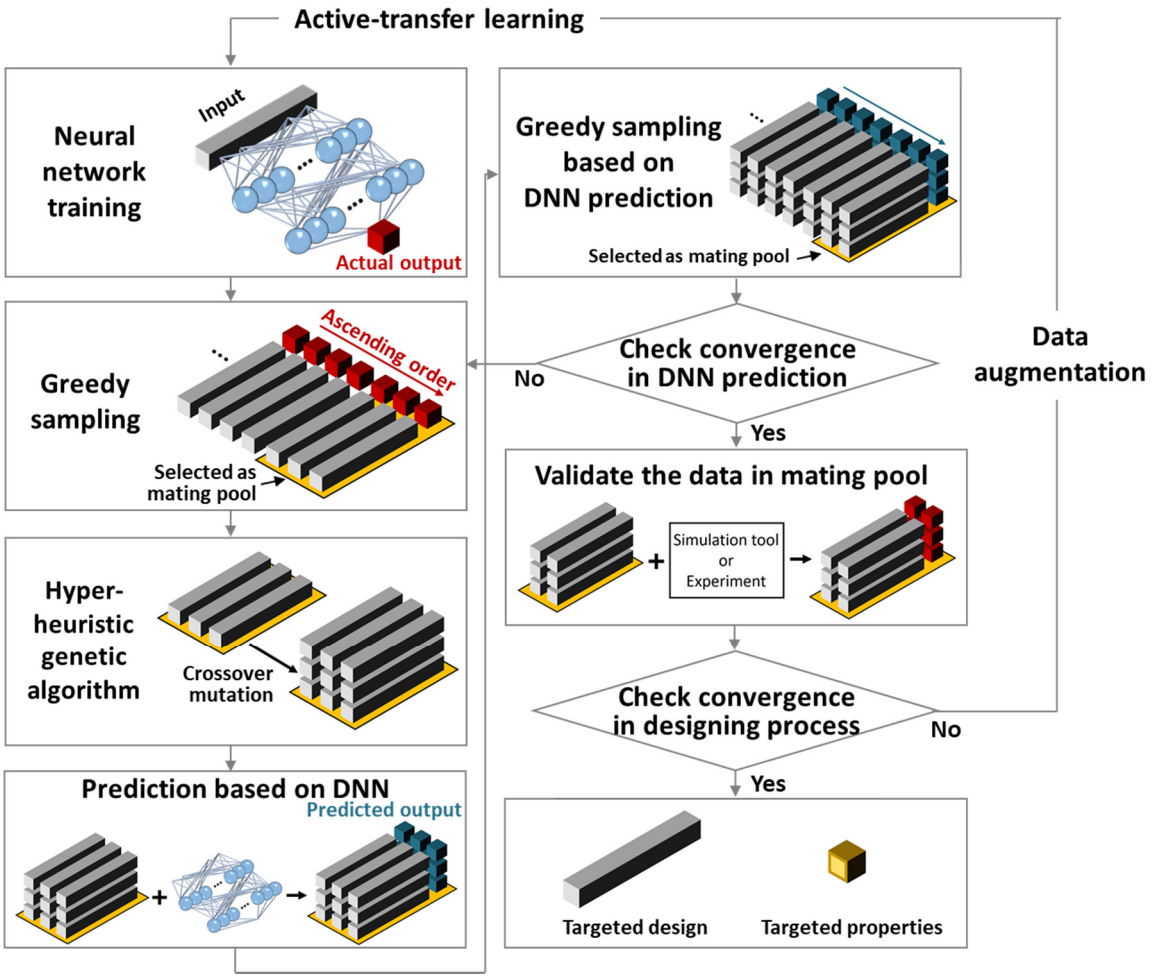

Figure 4. Flowchart of the hyper-heuristic genetic optimization algorithm based on active-transfer learning.

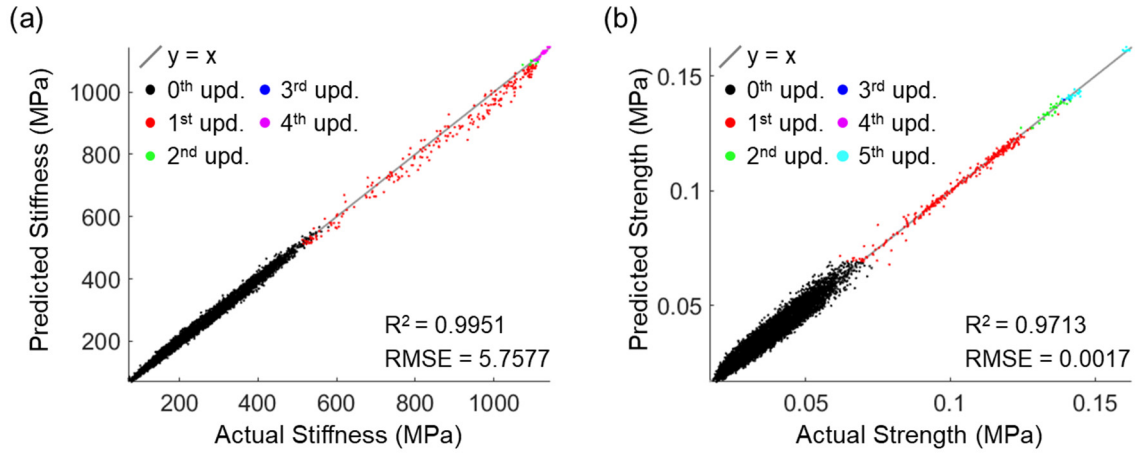

Figure 5. Designing progress plot for maximum (a) stiffness and (b) strength. The values are predicted from DNN trained at last update. The data obtained from each iteration are represented with different color.

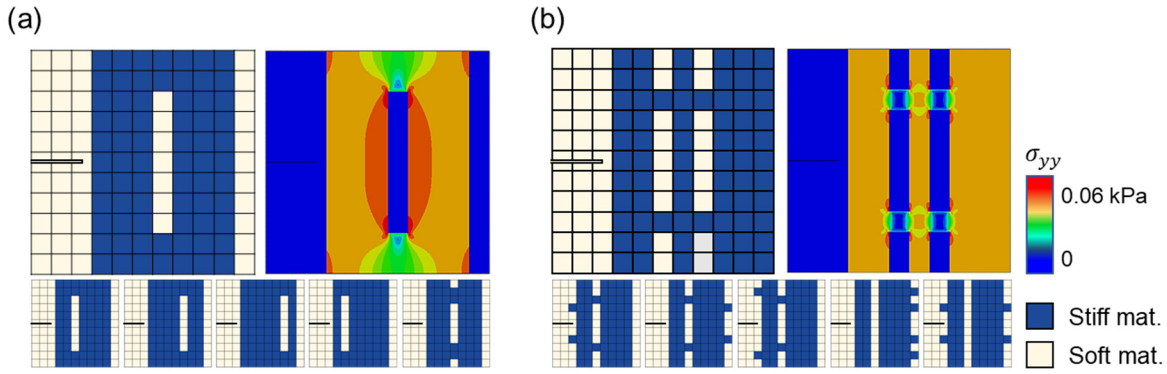

Figure 6. Top 6 composite structures optimized for maximum (a) stiffness and (b) strength. The enlarged structures on the top are structures having highest stiffness and strength value, and the corresponding axial stress (σ_{yy}) distribution colored at $\varepsilon_{yy} = 2.2727 \times 10^{-5}$. The maximum stiffness and strength of composite having optimized microstructures are 1144.7 MPa and 0.16276 MPa, respectively. The five structures having following highest value of stiffness and strength are represented below.

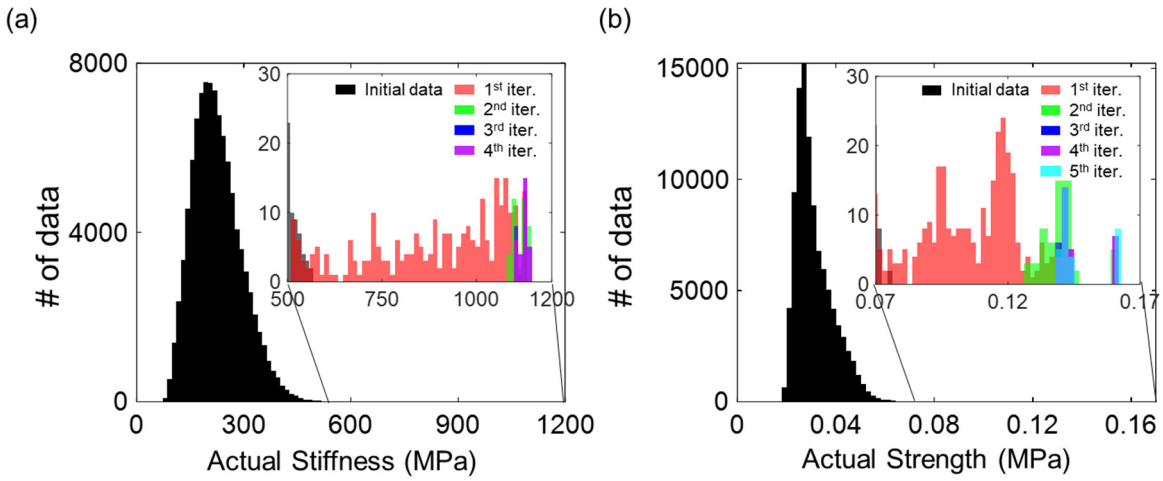

Figure 7. Histogram of (a) stiffness and (b) strength after the optimization. The initial data set are colored black, the data from 1st, 2nd, 3rd, 4th, and 5th updates of DNN are colored red, green, blue, purple, and cyan, respectively.

Supplementary Information

Deep Learning Framework for Material Design Space Exploration using Active Transfer Learning and Data Augmentation

Yongtae Kim[1†], Youngsoo Kim[2†], Charles Yang[3], Kundo Park[1], Grace X. Gu[3], Seunghwa Ryu[1*]

Affiliations

[1]Department of Mechanical Engineering & KI for the NanoCentury, Korea Advanced Institute of Science and Technology (KAIST), 291 Daehak-ro, Yuseong-gu, Daejeon 34141, Republic of Korea

[2]Department of Nature-Inspired System and Application, Korea Institute of Machinery and Materials, Daejeon, (34103) 156, Republic of Korea

[3]Department of Mechanical Engineering, University of California, Berkeley, CA 94720, USA

[†]These authors contributed equally to this work.

*corresponding author: ryush@kaist.ac.kr

Keywords: Deep Learning; Active-Transfer learning; Data augmentation; Regression; Material design;

1. Prediction of deep neural network (DNN) upon unseen output value ranges with different activation functions.

To analyze the effect of different activation functions, we compared the predictive power of deep neural networks(DNNs) upon unseen output value ranges by employing different activation functions including hyperbolic tangent (Tanh), Rectified linear activation function (ReLU), Leaky ReLU, Exponential Linear Unit (eLU), Clipped ReLU. The activation functions are given as follows:

$$f^{\mathrm{Tanh}} = tanh(x)\,, \qquad f^{Clipped\ ReLU} = \begin{cases} ceil & x > ceil \\ max(0,x) & x \le ceil \end{cases}$$

$$f^{ReLU} = max(0,x)\,, \qquad f^{Leaky\ ReLU} = \begin{cases} x & x > 0 \\ ax & x \le 0 \end{cases}, \qquad f^{eLU} = \begin{cases} x & x > 0 \\ a(e^x - 1) & x \le 0 \end{cases}$$

The Tanh and Clipped ReLU have bounded output values ranges $(-1, 1)$, and $(0,\ ceil)$, respectively. We tested two different ceil values of 1 and 6 for Clipped ReLU. In contrary, ReLU, Leaky ReLU, and eLU have unbounded output value ranges. The coefficients for the Leaky ReLU and eLU are arbitrarily assigned as 0.1, and 1, respectively. For the fair comparison, other training conditions such as architectures, number of epochs, learning rate, and etc. are equally assigned. The DNN composed of these activation functions are trained on the dataset having lowest 90% stiffness values, and are tested to predict the dataset with highest 10% stiffness values. Because the DNN are trained on the dataset with low output value, the validation results can be regarded as the predictive performance of the DNN in unseen domain during the optimization process. The test results are visualized in Figure S1. The DNN composed of bounded activation functions show relatively weak predictive power upon the unseen output value range, as represented in Figure S1 (a-c). Especially, DNN composed of Clipped ReLU with $\mathrm{ceil} = 1$ show clearly bounded output values prediction. In contrary, as shown in Figure S1 (d-f), unbounded activation functions show better predictive performances

upon unseen domain with higher R^2 values and smaller root mean squared error (RMSE), as consistent with previous research [1-3]. In the present study, we employed the DNN with the leaky ReLU because it shows the least RMSE value.

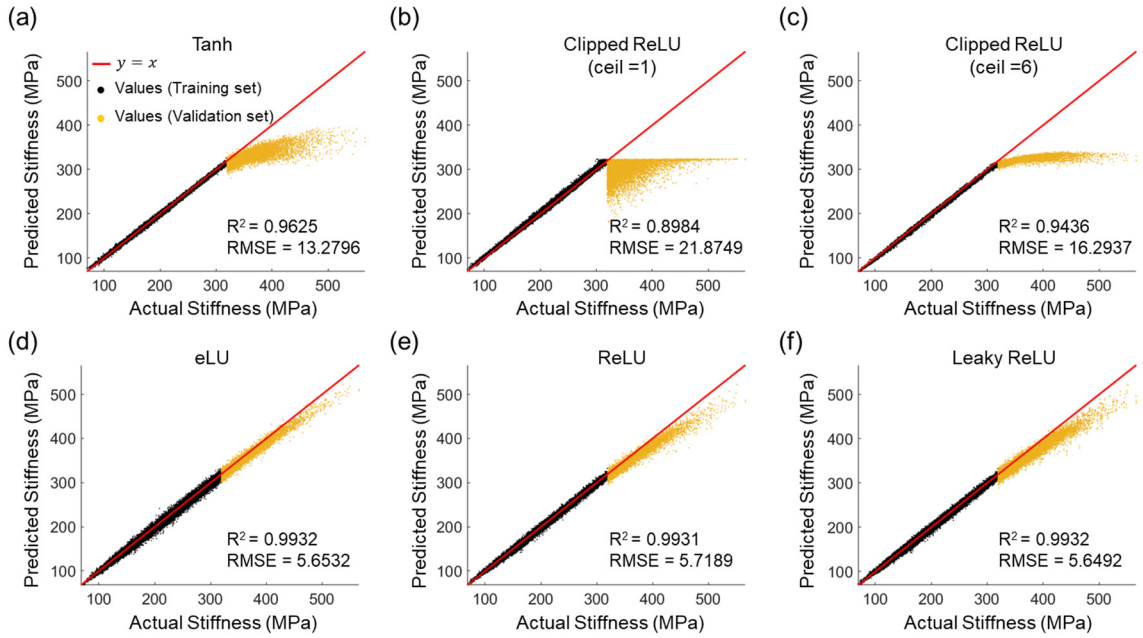

Figure S1. Training results on the stiffness with DNNs composed of different activation functions. The training sets and the validation sets are selected as the dataset with the lowest 90% stiffness values and the highest 10% stiffness values, respectively. The coefficient of determination (R^2) is calculated by following equation: $R^2 = 1 - \sum(y_i - \bar{y})^2 / \sum(y_i - f_i)^2$. y_i, f_i and $\bar{y}$ represent the actual value, fitted value and mean of actual values. The root mean squared error (RMSE) is calculated by following equation:RMSE $= \sqrt{\sum(y_i - f_i)\ /N}$.

2. Architecture of deep neural network for stiffness, strength, and toughness prediction

The architectures of deep neural network (DNN) adopted for stiffness, strength, toughness predictions are represented in Figure S1 (a), (b), (c). These architectures are constructed based on the residual network [4], to secure enough number of learnable parameters without exploding or vanishing gradient problems. In addition, because it is known that the full pre-activation networks have good generalization performance, each residual unit is constructed as depicted in Figure S2(d). The regularizations are mainly controlled by modifying the complexity of the architectures, i.e., the number of layers, the number of neurons at fully connected layer, the size of filters and the number of filters at each convolutional layer. The flowchart representing the process for constructing the DNN architecture is depicted in Figure S3. In case of toughness prediction represented in Figure S4, we found that the interpolation task on seen domain was possible (despite of relatively large R^2 value), but the predictive performance on unseen domain turns out to be very weak.

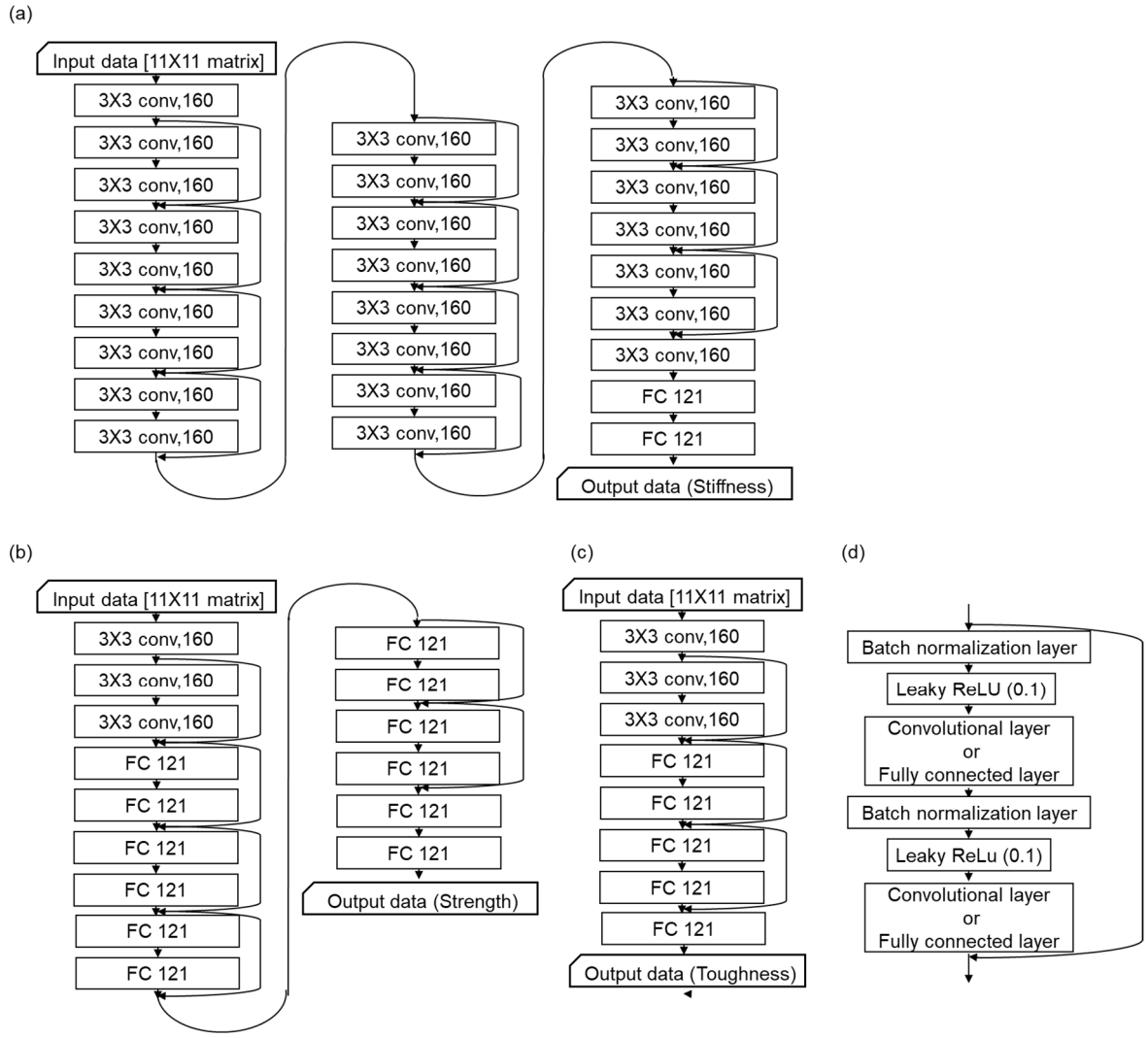

Figure S2. (a) Neural network architecture for prediction of stiffness. The architectures are constructed based on Residual network, and (b) skipped layer are constructed with pre-activation to improve generalization of neural network.

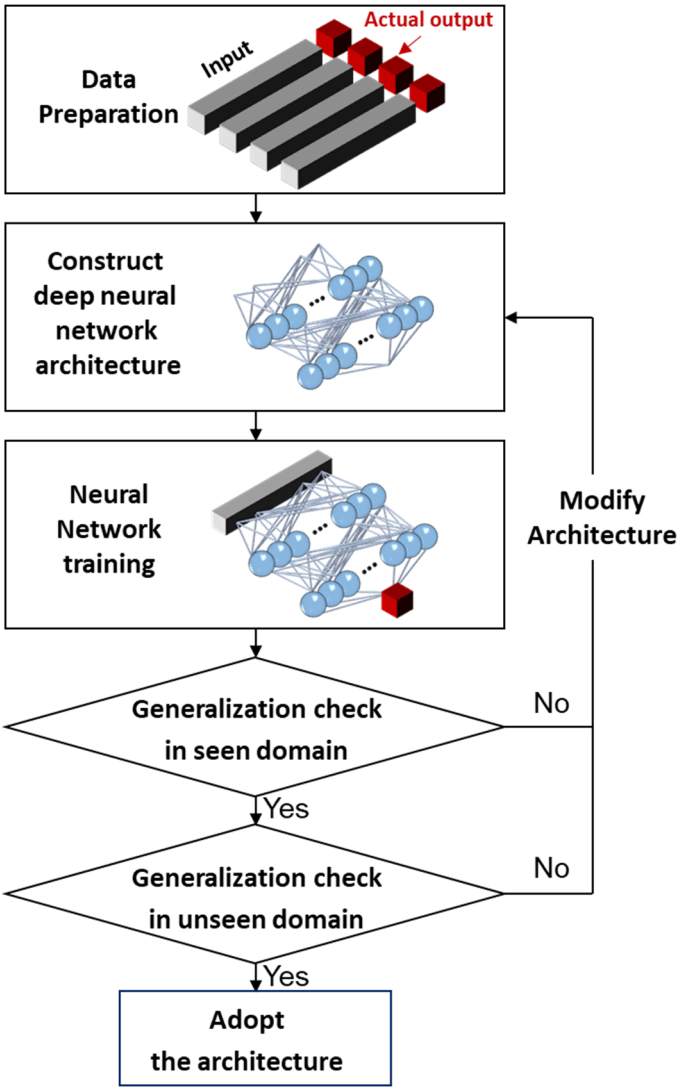

Figure S3. Flow chart of constructing DNN architecture adopted for optimization leveraging the prediction upon unseen domain. The generalization performance of DNN should be double-checked on the basis of seen domain and unseen domain.

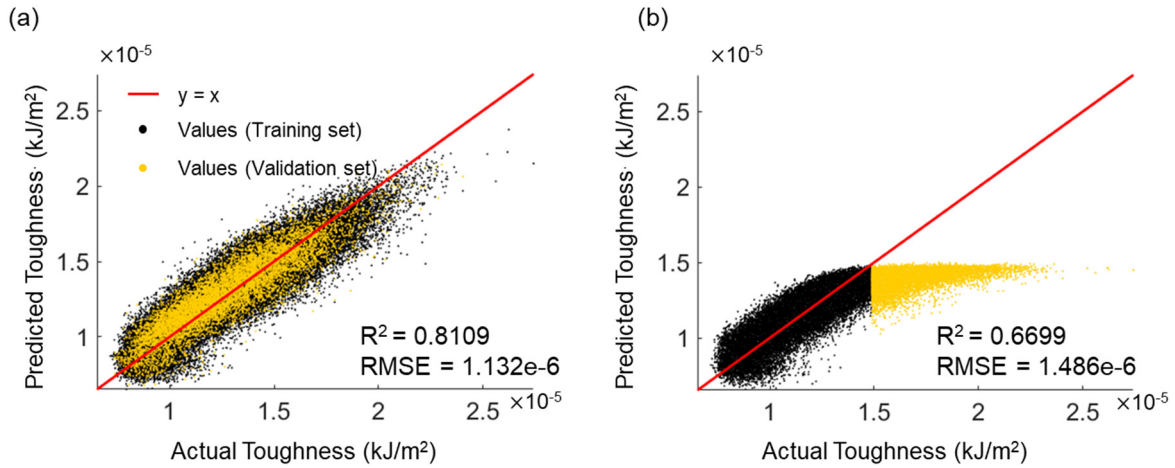

Figure S4. (a) The DNN training result for toughness. The coefficient of determination (R^2) are calculated by following equation: $R^2 = 1 - \sum(y_i - \bar{y})^2 / \sum(y_i - f_i)^2$. y_i, f_i and $\bar{y}$ represent the actual value, fitted value and mean of actual values. The root mean squared error (RMSE) are calculated by following equation:$\mathrm{RMSE} = \sqrt{\sum(y_i - f_i)\ /N}$. (b) extrapolation validity check of trained architecture. The DNN is trained using 90 % of data having low values and validated using 10 % of data having high value in terms of stiffness and strength, respectively.

3. Effects of heuristics on the optimization process.

The intuition from the field of solid mechanics indicates that the symmetric structure is efficient for load bearing, and that the soft material at the crack is able to relieve the stress concentration at the crack tip, as shown in Figure S4. These heuristic design principles are implemented in hyper-heuristic genetic algorithm by constraining the explorable design space to satisfy these two conditions. It accelerated the optimization process compared to conventional particle swarm optimization and conventional genetic algorithm without any heuristics, as represented Figures S5. The comparisons are conducted with the DNN trained on initial data set without the updates on DNN. The conventional particle swarm optimization process proceeds by random swapping stiff and soft blocks. The conventional genetic algorithm adopts the prescribed procedure in the main text without aforementioned heuristics. We find that the optimization with hyper-heuristic genetic algorithm generally converged to a higher optimal point with a smaller number of iterations compared to other algorithms. The composite structures designed by each optimization algorithms and corresponding stiffness values are represented in Figure S6. Because the optimized structures are proposed by DNN on unseen domain, there are relatively large discrepancy between predicted stiffness and actual stiffness value. We note that most structures from optimization algorithms without the heuristics also has a soft material at the crack tip, and most stiff material blocks to the right side of structures in relatively symmetric manner, which is consistent with our intuition from solid mechanics.

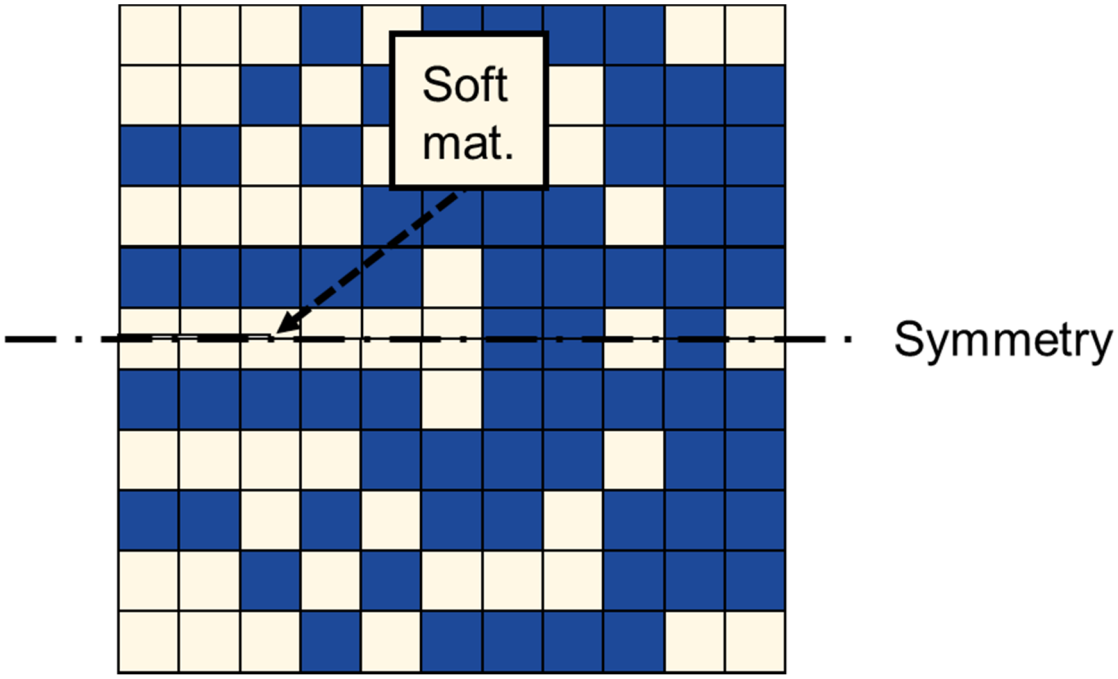

Figure S5. The composite structures have symmetrical structure with respect to the dashed line, and have soft material at the crack tip.

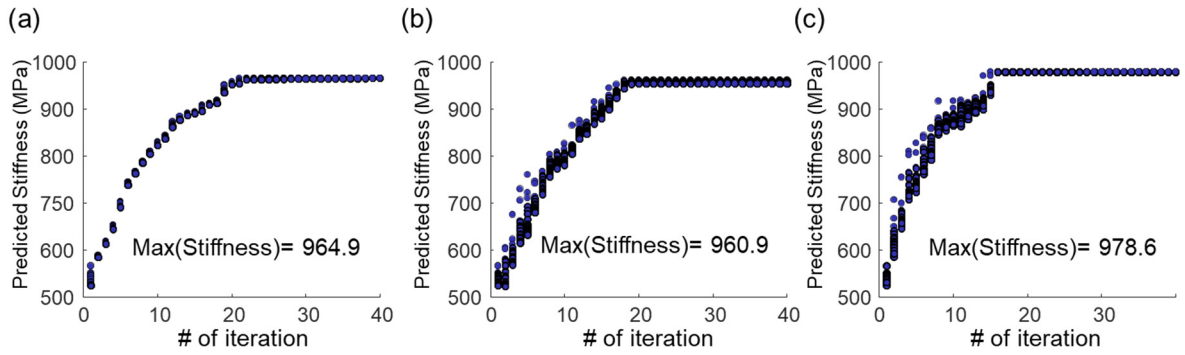

Figure S6. The progress of stiffness optimization process are compared with (a) particle swarm optimization, (b) genetic optimization, and (c) hyper-heuristic genetic optimization. The number of data chosen at each iteration is 30. The converged values can be varied due to the randomness in the optimization.

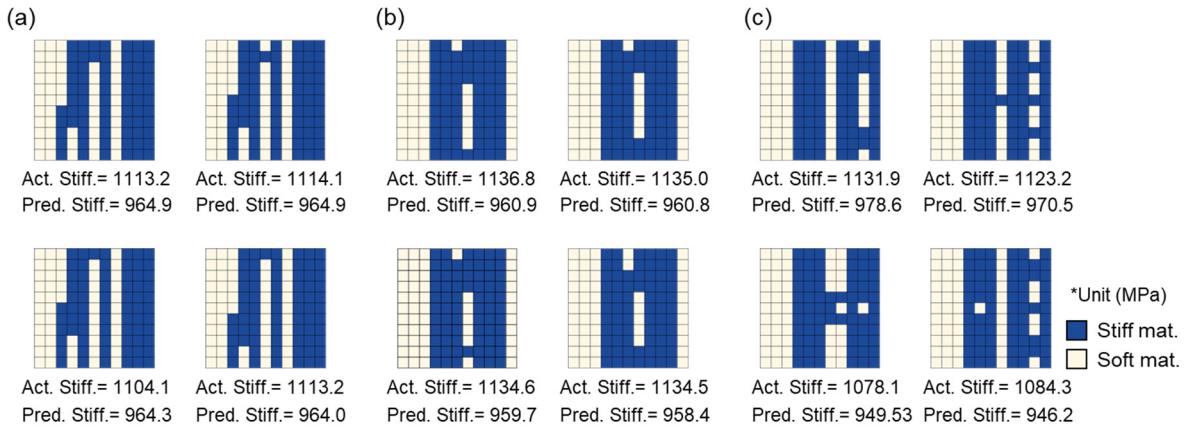

Figure S7. The best composite structures designed by (a) particle swarm optimization, (b) genetic optimization, and (c) hyper-heuristics genetic optimization, respectively. The structures are selected at the final iteration of each optimization algorithm based on the DNN trained on initial data set.

4. Stress distribution of each composite microstructure.

The optimized structures have less degree of stress concentration compared to the randomly chosen structures in trainig data set represented in Figure S7. Because the severe stress concentration promote early fracture at lower stress level, it is important to relive the stress concentration to achieve high strength. To validate our results, the composte structures, which are expected to have weak stress concentration are manully designed, as represented in Figure S8. We found that manually designed structures in Figure S8 (a), (b), and (c) show relatively severe stress concentration compared to deep learning based optimized structures. Interestingly, although Figure S8 (d) shows almost uniform stress distribution (which is much better than optimal design from the DNN-based search in Figure 8(b)), it has lower strength value of 0.15728 MPa due to its relatively low stiffness of 1109.7 MPa. Our manual design examples imply that the appropriate combination of high stiffness and small stress concentration is necessary to achieve the high strength. Our deep-learning-based forward design frameworks enables the search of the optimal design with the optimal compensation of each.

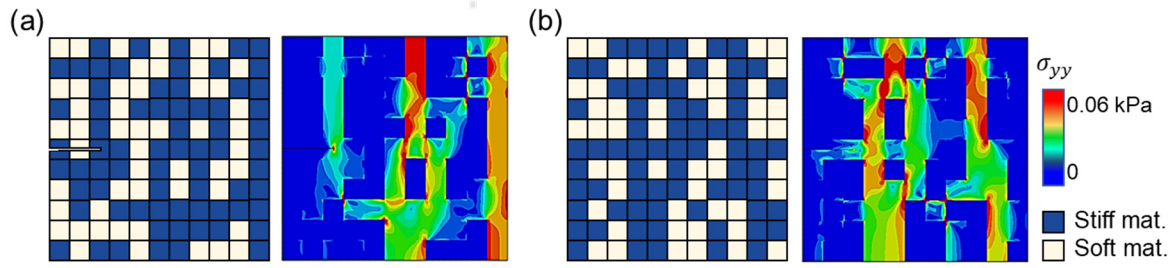

Figure S8. (a-b) Randomly chosen composite structure and corresponding axial stress (σ_{yy}) are colored at $\varepsilon_{yy} = 2.2727 \times 10^{-5}$.

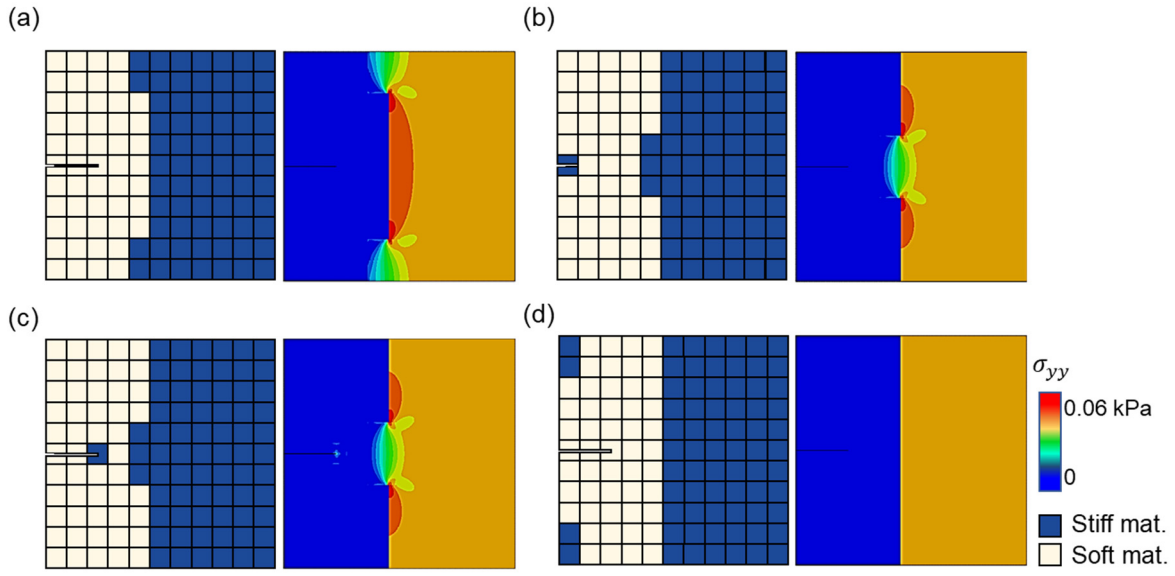

Figure S9. (a-d) Composite structures based on heuristics for maximum strength, and corresponding axial stress (σ_{yy}) are colored at $\varepsilon_{yy} = 2.2727 \times 10^{-5}$. The strength of each composites structure are 0.15798 MPa, 0.15794 MPa, 0.15823 MPa and 0.15728 MPa respectively.

Reference

1. Sonoda, S. and N. Murata, *Neural network with unbounded activation functions is universal approximator.* Applied and Computational Harmonic Analysis, 2017. **43**(2): p. 233-268.
2. Arora, R., et al., *Understanding deep neural networks with rectified linear units.* arXiv preprint arXiv:1611.01491, 2016.
3. Hein, M., M. Andriushchenko, and J. Bitterwolf. *Why relu networks yield high-confidence predictions far away from the training data and how to mitigate the problem*. in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019.
4. He, K., et al. *Deep residual learning for image recognition*. in *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.