

ARTICLE OPEN



Accelerated design and characterization of non-uniform cellular materials via a machine-learning based framework

Chunping Ma¹, Zhiwei Zhang², Benjamin Luce¹, Simon Pusateri¹, Binglin Xie³, Mohammad H. Rafiei⁴✉ and Nan Hu³✉

Cellular materials, widely found in engineered and nature systems, are highly dependent on their geometric arrangement. A non-uniform arrangement could lead to a significant variation of mechanical properties while bringing challenges in material design. Here, this proof-of-concept study demonstrates a machine-learning based framework with the capability of accelerated characterization and pattern generation. Results showed that the proposed framework is capable of predicting the mechanical response curve of any given geometric pattern within the design domain under appropriate neural network architecture and parameters. Additionally, the framework is capable of generating matching geometric patterns for a targeted response through a databank constructed from our machine learning model. The accuracy of the predictions was verified with finite element simulations and the sources of errors were identified. Overall, our machine-learning based framework can boost the design efficiency of cellular materials at unit level, and open new avenues for the programmability of function at system level.

npj Computational Materials (2020)6:40; <https://doi.org/10.1038/s41524-020-0309-6>

INTRODUCTION

Artificial intelligence (AI) is taking the world by storm, demonstrated when AlphaGo defeated the world champion Lee Sedol in the game of Go in 2016¹. Emerging AI methods such as machine learning (ML) has revolutionizing the way of doing research across almost all fields of study^{2–5}. In material science, computation-based approaches have been used to gain insights on material behaviors and properties. Thanks to recent advances in computational power and robust algorithms, we have witnessed the rise of an interdisciplinary field in computational material science^{6–9}. Specifically, researchers have already shown success in utilizing ML-based or other AI methods in two major categories: (1) to accelerate the prediction of material properties for specific applications^{10–24}, and (2) to accelerate the on-demand design and the optimization of material microstructure and composition for targeted properties^{25–42}. These promising studies have shown superior effectiveness of ML techniques compared to traditional computational modeling or experimental measurements on a variety of materials.

Motivated by these recent developments, we take initiative to apply ML techniques to accelerate the design and characterization of cellular materials. Cellular materials, recently identified as one type of architected material, are structured materials with high tunability of properties determined by the geometry and the assembly of unit cells^{43,44}. Such a unique function has been showcased in a wide range of length scales, from nanolattice to the Eiffel Tower. Frontier studies in this field have focused on advanced design approaches^{45,46}, geometric mechanics^{47–51}, fabrication^{52–56}, and potential applications at multiple length scales^{57–59}. Yet, most studies to date usually focus on unit cells with an assumption of a periodic pattern over a larger domain, while aperiodic patterns with non-uniform cells bring infinite possibilities for achieving high tailorability in failure mode, energy absorption, etc. As the design domain of geometric pattern expands, it becomes impossible for the conventional intuition-

based method to attain feasible patterns of desired properties across the entire domain.

In this work, we develop a ML-based framework (Fig. 1) by training a deep neural network (DNN) with minibatch stochastic gradient descent learning algorithm to predict various mechanical responses due to non-uniform geometric pattern; meanwhile build a databank through the developed ML-based framework to solve inverse pattern designs for given targeted responses in a highly efficient manner. Our baseline geometry is a square-shape cellular material with cross infills as local reinforcement, illustrated in Fig. 1a. The simplest 2×2 unit cell has a total of 16 possible patterns by considering various layouts of infills. As the number of cells becomes larger to $n \times n$, the total possible patterns is dramatically increased in terms of $2^{n \times n}$. We selected a 4×4 unit cell due to its reasonable size of data and the availability of computational power. The specific dimensions can be found in Fig. Supplementary Fig. 1a. In Fig. 1b, we demonstrated a typical non-uniform pattern with 12 infills along with its binary representation in a matrix form used later as the input data to feed the DNN. We used finite element (FE) simulation to estimate a response curve of the cellular material under axial compression and used as the grand truth to compare with the resulted predictions by the ML model. The FE simulations were conducted using commercial software ABAQUS. A total of 25 sampling points on the response curve were used as the output data of the ML model. Given that we have tested the 3D-printed cellular materials made of polylactic acid (PLA) in a pilot study⁶⁰, we used the same material with an ideal elastoplastic constitution (Supplementary Fig. 1b) to discover a larger possibility of failure modes. We also tested the response using a nonlinear elastoplastic constitution which has a clear softening stage and finally converges to the same plastic stress as the IEP, as shown in Supplementary Fig. 1b. However, though the nonlinear elastoplastic constitution may better reflect the PLA character, there is no significant difference between the response curves using the ideal and the nonlinear

¹Department of Mechanical and Aerospace Engineering, The Ohio State University, Columbus, OH 43210, USA. ²School of Civil Engineering, Harbin Institute of Technology, Harbin, China. ³School of Civil Engineering and Transportation, South China University of Technology, Guangzhou, China. ⁴Department of Mechanical Engineering, Johns Hopkins University, Baltimore, USA. ✉email: mrafiei1@jhu.edu; nanhu026@scut.edu.cn

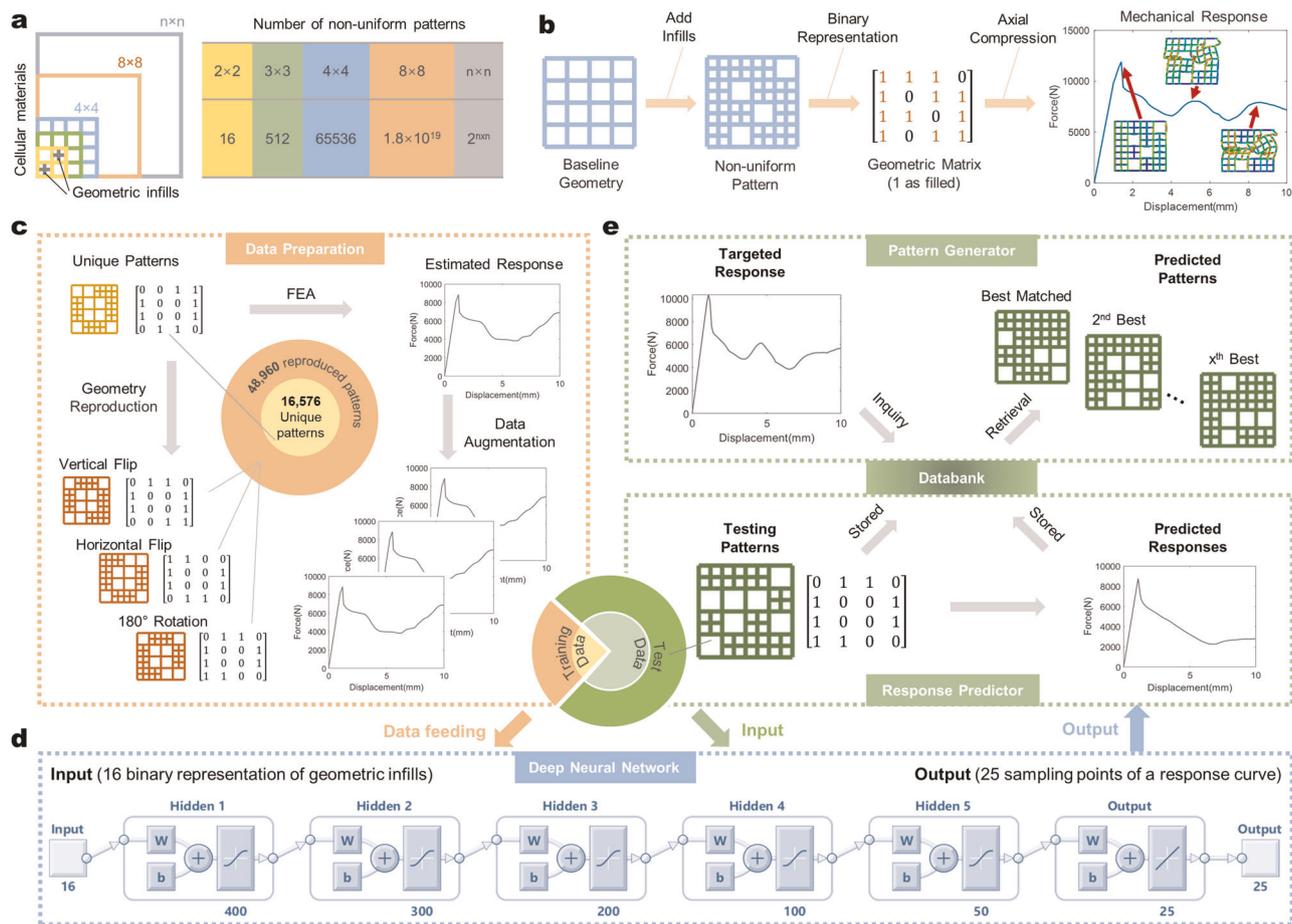


Fig. 1 Machine-learning based framework for non-uniform cellular materials. **a** Baseline geometry and design domain. **b** Formation of non-uniform pattern, binary representation and simulated response. Flowchart for the machine learning-based framework: **c** Data preparation. **d** Architecture of the deep neural network. **e** Accelerated response prediction and pattern generation.

elastoplastic constitutions (Supplementary Fig. 1c). The reason might be that the major plastic deformation and the subsequent contact happen under large strain, thus the constitutive difference in small strain has no significant influence on the response curve. Therefore, in this study we simplify the nonlinear elastoplastic constitution to a similar ideal elastoplastic constitution for PLA. Additional details of FE model can be found in *Methods*.

ML models are usually statistics-based data-driven algorithms which are trained to recognize and predict relations in data. Once properly trained, the ML model is capable of predicting solutions to abstract problems in a miniscule amount of computational time. These approaches are especially useful for solving problems that exist in a large design space and have numerous non-intuitive parameters. Our ML-based framework contains three major components: data preparation (Fig. 1c), training the DNN (Fig. 1d), and implementation with the trained model (Fig. 1e). In the data preparation, we generated all 65,536 possible patterns and their corresponding response curves for the 4×4 unit cell. Any given pattern may share the same physical phenomenon under the axial compression with its counterparts reproduced by flipping the original pattern along the vertical and/or horizontal axis as shown in Fig. 1c. In other words, due to their geometric symmetry there are only 16,576 unique patterns that can exhibit different mechanical responses under the axial compression for the 4×4 unit cells. As a result, we only run 16,576 FE simulations for the “unique” patterns to obtain response curves as reference. Then, we instructed the ML model by conducting simple geometric operations on unique patterns and reproduced all 48,960

counterparts that share identical responses, which we called “reproduced patterns”. It is noted that the training data comprise of both the unique patterns and their corresponding reproduced patterns, because the DNN does not naturally “know” that unique patterns and its reproduced counterparts should have identical response (See *Methods* for additional descriptions).

After the data preparation, we feed the training data (pattern-response) for the DNN and used the test data to verify the performance of the training. It is noteworthy that the birth of DNN traces back to 80’s, referring to networks that have more than one hidden layer, whether that the hidden layer is a simple perceptron layer of neurons, Boltzmann machine, or convolutional layer. The more the number of hidden layers, the deeper the network is. Training the DNN, however, remains a challenging problem till today. The network architecture has become more applicable in recent years due to (1) the invention of new training/learning methods for faster and less computationally intensive training^{61–67} and (2) the advent of faster computer hardware such as GPUs. In this study, the DNN comprises of 5 hidden layers, and number of neurons of each layer are given in Fig. 1d. We built the DNN in MATLAB® using the stochastic conjugate gradient backpropagation as the learning algorithm^{21,68}. Since we used the hyperbolic tangent sigmoid transfer function, the outputs (uniformly sampled forces) are scaled to $[-0.9, 0.9]$ for better performance. The proposed model does not depend on the learning method, and one may use other or newer learning techniques to fast-train the proposed model and increase its accuracy. In Fig. 1e, first we applied the trained DNN as a response predictor to obtain the

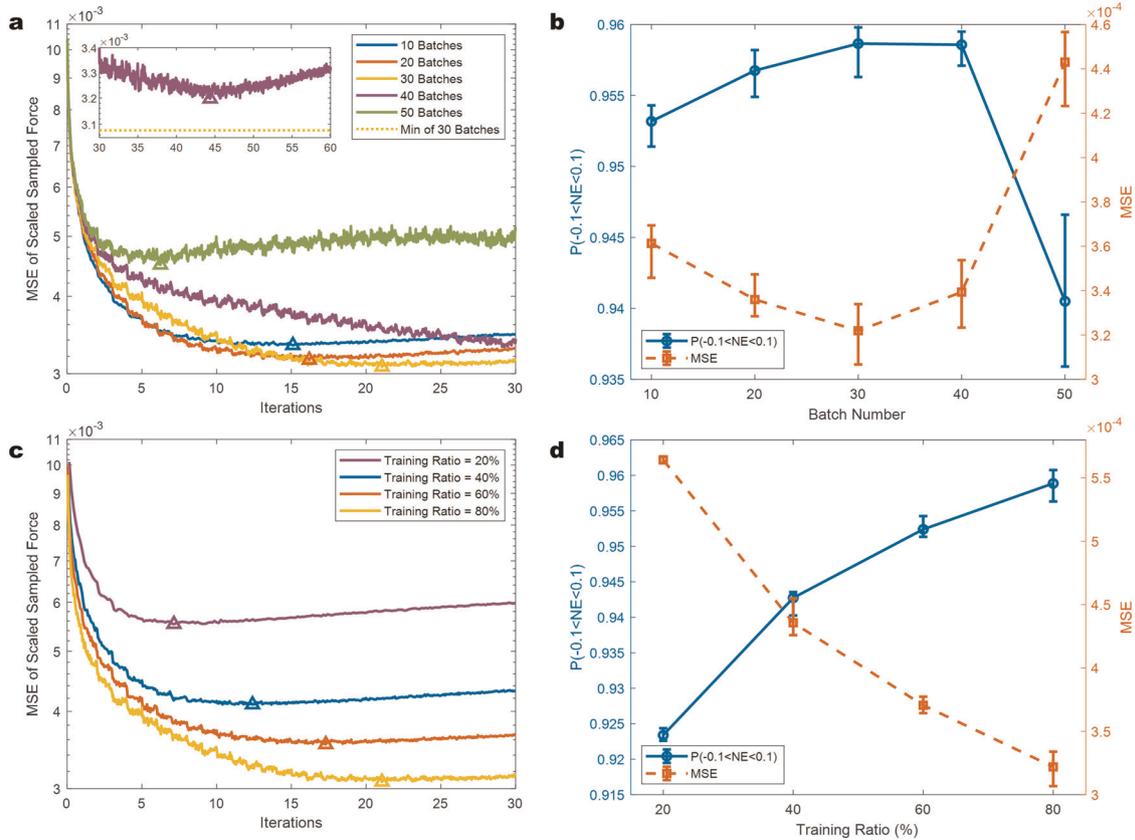


Fig. 2 DNN Performance under different batch numbers and training ratios. a Effect of batch number on convergence. **b** Effect of batch number on the performance in terms of probability $P(-0.1 < NE < 0.1)$ and MSE. **c** Effect of training ratio on convergence. **d** Effect of batch number on the performance in terms of probability $P(-0.1 < NE < 0.1)$ and MSE.

response curves of the test patterns; then we used these test patterns and their predictions (as well as the training data if in a real-world application) to form a databank playing the role of pattern generator. For the response prediction, we compared the performance of four different batch numbers/sizes and five training ratios to get the optimal parameters, and the epoch number for each batch is fixed to 250. For the pattern generation, the databank can efficiently recommend multiple feasible geometric patterns satisfying a targeted response.

RESULTS

Network parameters and architecture

To test the efficiency and robustness of the defined DNN architecture, we started with identifying the “optimal” batch number/size and training ratio, to find the optimal setting. The performance of the DNN is evaluated by two indices. One is the mean squared error (MSE) of all the predictions and the corresponding references, which can be written as

$$\text{MSE} = \frac{1}{mn} \sum_{i=1}^n \sum_{j=1}^m (f_{ij} - \hat{f}_{ij})^2 \quad (1)$$

where m is the number of sampled forces from a response curve, and n is the number of the test patterns. \hat{f}_{ij} and f_{ij} are the j -th normalized predicted sampled force and j -th normalized reference sampled force of the i -th test pattern, respectively. A smaller MSE means a better performance. Furthermore, we defined the

normalized error (NE) as:

$$\text{NE} = \frac{\hat{f}_{ij} - f_{ij}}{m} \quad (2)$$

$$\max_{j=1} f_{ij}$$

then we can obtain the probability density function (PDF) of NE and use the probability $P(-0.1 < NE < 0.1)$ as the second index. The larger the probability, the better the performance. Note that each parameter value was independently run five times using different training data, and the average performance is used to represent the effect of each value.

First, we fixed the training ratio to 80% and varied the batch number from 10 to 50 with an increment of 10. In Fig. 2a, the average convergence lines of the training process are given, where the triangles indicate the convergence points at which the MSE of scaled sampled forces reaches the lowest value. One iteration means that all batches are looped for 250 epochs. For the cases of 10, 20, and 30 batches, the larger the batch number is, the more iterations it takes to converge, and the better final performance it can get. By further increasing the batch number to 40, the model starts to take more iterations than 30 batches to converge yet does not provide a better performance. As the batch number becomes 50, the batch size is too small to represent the true gradient, leading to a premature convergence with the worst performance. In Fig. 2b, we provided a direct correlation between the batch number and the performance in terms of probability and MSE. The probability $P(-0.1 < NE < 0.1)$ of 50 batches has a much larger variation range than other batch numbers, indicating that the performance is unstable and strongly relies on the choice of training/test data. It is obvious to see that the optimal batch number is 30, meaning that the optimal batch size is $80\% \times 16576/$

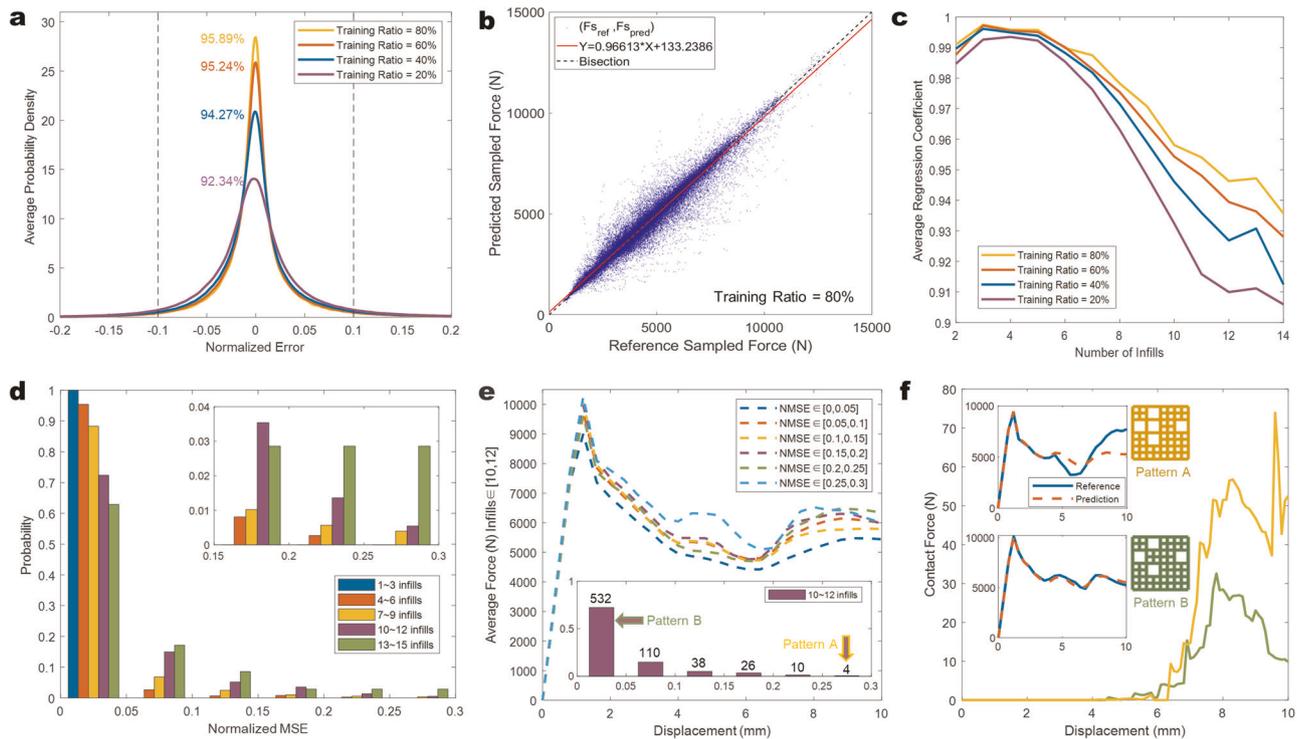


Fig. 3 Mechanical response prediction by DNN. **a** Average probability density function of the normalized error under different training ratios. **b** The comparisons between predicted sampled forces and reference sampled forces. **c** Effect of infill numbers on response prediction. **d** Detailed categorization of infill numbers against performance. **e** Average response curves of different error levels. **f** Predicted and reference response curves of two representative patterns along with total contact force under axial compression.

$30 \approx 442$. Then, we fixed the optimal batch size and altered the training ratio from 20% to 80% with an increment of 20%. Similar to Fig. 2a, we show the average training convergence lines of different training ratios in Fig. 2c. Apparently, the larger the training ratio, the more iterations it takes to converge, and also a better performance it can reach. In Fig. 2d, we gave the direct correlation between the training ratio and the performance in terms of probability and MSE. It can be seen that $P(-0.1 < NE < 0.1)$ value only drops slightly (still above 0.92) as the training ratio drops from 80% to 20%. The training time varied depending on batch number and training ratio. For example, it took about 5 hours for a case with a 80% training ratio, 30 batches and 30 iterations using a standard desktop with an Intel i7-7700@3.6 GHz, 8 cores CPU.

Model validation and response prediction

Having identified an appropriate batch number (30) and the training ratio (20%), we then focused on the validation and accuracy of response prediction to those referenced FEM results. Compared to existing studies on predicting single property such as strength, stiffness, and toughness, we evaluated the entire response curve which contains multiple characteristics. As mentioned in Fig. 1, with the trained DNN, we can estimate the response curve of any pattern from the test data. In Fig. 3a, we plotted the average PDF of each training ratio, and marked the corresponding values of probability $P(-0.1 < NE < 0.1)$. It can be observed that the PDF is more elevated and concentrated (more accurate) with a higher training ratio, but the probability is not significantly influenced. We further conducted a regression analysis for each pair of predicted curves and the corresponding reference curve. From this, the obtained regression coefficient is a good estimator of whether a predicted curve follows the trend (i.e., increase and decrease) of the reference curve. The regression coefficient ranges from 0 to 1, the higher the value, the better

matching of the trend. Supplementary Fig. 3a illustrates the average results of the regression analysis. Basically, all training ratios can yield satisfying results where the majority of the regression coefficients is larger than 0.9. Thus, a low training ratio is acceptable and capable in prediction. Another common representation of the overall performance for a ML model is the comparison between each single predicted value and its reference value. In Fig. 3b, we present an example of 80% training ratio, where the coordinate of each blue dot comprises a predicted sampled force $F_{S_{pred}}$ and the corresponding reference value $F_{S_{ref}}$. A closer position of a dot to the bisection line ($y = x$) indicates a more accurate prediction. The band width formed by these blue dots can intuitively reflect the overall performance. Clearly, the predictions under an 80% training ratio result in a narrow band width, indicating a more satisfying performance. The comparison plots generated by the same DNN using other training ratios are shown in Supplementary Fig. 3b–d.

Despite the acceptable prediction by the DNN, we attempted to identify the source of error by analyzing the results under an 80% training ratio. Our first hypothesis is that geometric infills and their layouts can affect the prediction result. Recall that the infill number can range from one to fifteen for the 4×4 unit cell with non-uniform patterns. In Fig. 3c, we plotted the average values of regression coefficients versus a varying number of infills for all patterns from the test data set. It can be seen that under all tested training ratios the regression coefficient decreases with an increasing number of infills. To further examine this observation, we calculated the mean squared error between each predicted response curve and its reference FE response curve from the test data set (i.e., 20% of the total data, 3316 unique patterns) and normalized the mean squared error values into $[0, 1]$. A high normalized mean squared error (NMSE) value indicates a poor prediction. In Fig. 3d, we grouped all data into five subsets based on their number of infills. Since the probability $P(-0.1 < NE < 0.1)$

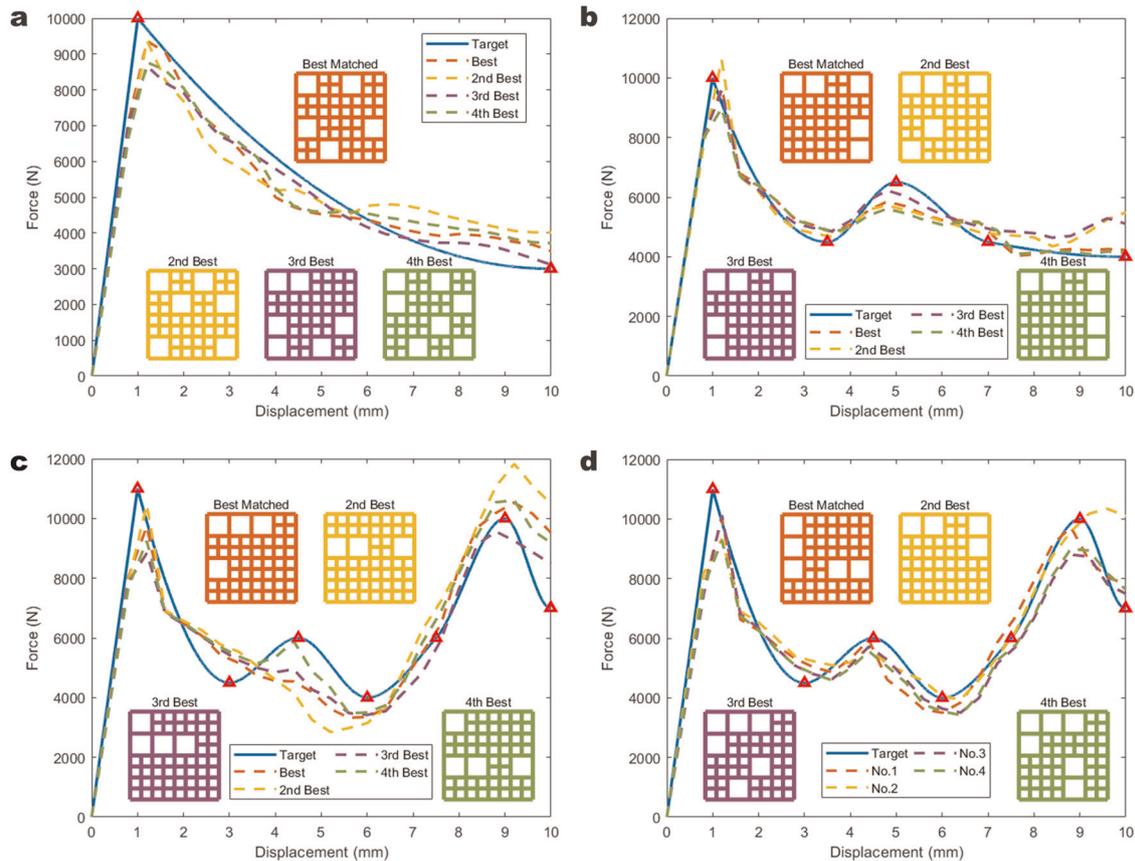


Fig. 4 Patten generation for targeted response curves using ML-based databank. **a–c** Top four patterns and their reference curves for a targeted response with one, two and three load-dropping points only using the predicted responses by 20% training ratio. **d** Top four feasible patterns and their reference curves for the same target response in **c** using the reference responses of all available patterns.

of the DNN has reached 95.89% under the training ratio of 80%, data amount is limited when the NMSE value is above 0.3. We thus only showed the data with a NMSE value from 0 to 0.3 and divided them by six error intervals for each subset. It is obvious that the majority of data for all five subsets is located between the NMSE value from 0 to 0.05, indicating a good prediction by our ML model. Above the NMSE value of 0.05, in particular between 0.15 to 0.3 (see the inset in Fig. 3d), it can be seen that the two subsets with 10–12 (green) and 13–15 (purple) infills are more dominant compared to the other subsets, confirming that the prediction error is mainly contributed from the patterns with larger infill numbers.

To better connect the numerical prediction with the physical phenomenon, we extracted the subset with 10–12 infills with 720 unique patterns in total, and the pattern number in each error interval is given in the inset of Fig. 3e. Due to varied response curves in each error level, we examined the average reference response curves to identify the source of error in Fig. 3e. It can be seen that six curves show a similar general trend decreasing in stiffness after the first critical load that related to buckling of cell walls and later regaining stiffness near the axial displacement of 6 mm. The average response curve of the high error level (NMSE 0.25–0.3) has the largest force. Our hypothesis is that this error could be resulted from contacts and interactions among cell walls in the material, which we have observed in our prior study. In other words, the complex interaction among cell walls under axial compression is a highly nonlinear behavior and can lead to wrong prediction of the mechanical response even in FEM, not to mention the DNN which is only a surrogate model. These average curves also confirm why the prediction error increases with a larger infill number, because the patterns with higher infill

numbers are more likely to exhibit a biaxial cell interaction other than the lateral buckling observed in patterns with lower infill numbers.

In addition to the infill number, we also identified that the spatial heterogeneity and geometric complexity of the cellular materials have created challenges for our DNN although only 4×4 unit cells were evaluated. Based on the discussion in Fig. 3d, e, we discovered two representative patterns within the subset of 10–12 infills, pattern A from a high error level (NMSE 0.25–0.3) and pattern B from a low error level (NMSE 0–0.05), indicated by the arrows in Fig. 3e. In Fig. 3f, we plotted contact forces-displacement curve, reference and predicted curves (two insets) and actual infill layout of pattern A and B. Pattern B has one extra infill in the second cell from the left on the second row compared to pattern A. Theoretically, both patterns should buckle at the second row where the weakest cross section is, while the FE simulation (blue reference curves of two insets) shows that their response become significantly different after the axial displacement of 6 mm. This discrepancy can be well explained through the total contact force history for both patterns in Fig. 3f in which we summed up the contact force within each cell. It can be seen that the pattern A has significantly larger total contact forces than the pattern B, indicating stronger cell interactions, which is consistent with our previous hypothesis. As a result, the DNN cannot distinguish pattern A and B because the training data was not enriched enough to relay the discrepancy between slightly different case A and B. Such lack of enrichment can be due to either the choice of training data or the limit of cellular material itself. Thus, the DNN considered their responses in a similar fashion (indicated by red predicted curves of two insets) and thus failed to capture the actual response of pattern A. By far, the DNN, with the proposed

network architecture and training parameters, could not completely eliminate this issue, but we might be able to alleviate by selecting a more suitable ML algorithm (CNN, Deep Boltzmann Machine, Regression SVM, Ensembles, etc.), using repeated random sampling, obtaining the material properties in priori, and carefully choosing training data.

Databank building and pattern generation

Having validated the DNN and gained confidence on the prediction, we further exploited the trained DNN to construct a databank that containing all the possible patterns and their corresponding responses. In this study, we were able to use the test patterns and their predicted responses to construct the databank to accurately reflect the performance of the pattern generator and the response predictor. For a given targeted response, we can find the best matched predicted response and the corresponding pattern in the databank; and can use the difference between the targeted response and the pattern's reference response to evaluate the performance of the pattern generator. Since the pattern generator was built based on the results of the response predictor, its average PDFs under different training ratios (Supplementary Fig. 4) has a similar trend yet more elevated shape than their counterparts (Fig. 3a) due to that fact that non-uniform patterns with different infill layouts can yield similar responses, which enhances pattern generator's robustness to satisfy a targeted response.

To better demonstrate the capability of the pattern generator, we generated three arbitrary but reasonable targeted responses, featuring various stiffness-dropping points that are directly related to buckling events in the cellular material. In Fig. 4, these targeted responses were illustrated using solid blue lines and control points (the red triangles). We used the databank based on predicted responses by our framework under only 20% training ratio to generate top four feasible patterns by searching the predicted response curves that has lowest deviation from the targeted response curve. To showcase the accuracy of the pattern generator, we plotted reference response curves of generated patterns (dash lines) in Fig. 4. Clearly, the pattern generator can provide feasible patterns for all three arbitrary targeted responses with one (Fig. 4a), two (Fig. 4b), and three (Fig. 4c) stiffness-dropping points, respectively, although the accuracy of the reference curve decreases in Fig. 4c compared to the one in Fig. 4a because the targeted response curve has much higher complexity in terms of multiple stiffness-dropping points. In Fig. 4d, we showed four reference curves with a better agreement for the same targeted curve in Fig. 4c because we implemented the pattern generator that built directly from the reference responses of all possible patterns. However, to build a databank by running simulations for all possible patterns would take more effort; thus we satisfied with our prediction improvement with the databank constructed by DNN architecture with only 20% training ratio.

DISCUSSION

In summary, we implemented a machine learning framework to achieve two common goals in computational materials, i.e., the forward property/response prediction and the backward composition/pattern design, applying in particular to the case of non-uniform cellular materials under axial compression. We identified an appropriate batch number and training ratio for DNN that can be used as a baseline parameter for other cellular materials with different cell geometries. Our ML model is capable of predicting the mechanical response of patterns from the testing data under a 20% training ratio. We also identified the source of error which is primarily related to various geometric infills and their layout, as well as the complexity raised from the material and geometric

nonlinearity due to strong interaction and contact among cell walls. After validation of the DNN, we created a databank, allowing further generation of feasible geometric patterns for a given response. The proposed framework has demonstrated using a cellular architecture that could align with the authors' accessible computer resources. However, the framework is not limited to certain material architectures or the number of configurations, although we only validated the proposed framework using 4×4 unit cells and limited the number of combinations (only 65,536 possibilities and 16,576 with symmetry). We believe that our framework is comprehensive and can be incorporated into a larger design domain with more complicated architectures if enough computational resources, such as high-performance parallel computing resources, are available. Our ML-based framework should be capable of accelerating the material design with tailorable properties at the local level and improving the performance and/or functionality at the global level.

There are several advantages regarding the framework proposed in this paper. First, it only needs a relatively simple ML model for the response prediction, based on which the pattern generation can be directly realized, thus the training is not computationally expensive compared with similar studies^{30,69,70}. Second, while the architecture of the ML model is not complicated, it is capable of predicting the entire nonlinear response history with plasticity and large deformation other than individual properties^{10,18,28}. Third, from the geometric perspective, it is able to predict the response for the majority of the patterns although the ML model alone as a surrogate method run into difficulty to provide accurate response for patterns with a larger infill number. Last but not the least, using the databank we can not only provide multiple candidate patterns for a targeted response, but also can easily apply some extra constraints, such as the number and the location of infills, in addition to the targeted response, which are difficult for conventional methods to realize. In addition, it can provide more robust results for the pattern generation if we construct the databank using all the possible patterns.

To improve the efficiency of our ML-based framework, we can apply more sophisticated ML models than the DNN. A promising candidate can be the physics-encoded ML models that are capable of understanding complex material properties and geometric topology. The rotation-invariant convolutional neural network can be another choice to deal with the situation that some rotated/flipped patterns have the same response⁷¹. Dual encoder is another possible option which has been used in natural language processing where the binary inputs and the response outputs are being encoded into a common domain⁷². The advantage of dual encoder is to minimize the discrepancy between the encoded features coming from binary inputs and the encoded features from the response outputs. Once the dual encoders are trained enough, one can unfold the two encoders to create an autoencoder that receives either the binary inputs and gives the response or receives the response to give the binary inputs.

Another interesting and practical direction for the future study is to introduce the effect of imperfections that are random and mostly inevitable in experiments. To address this issue, one of the feasible ways is to consider it as the variation of the beam thickness or the material property (such as yield stress) of each cell and infill, assuming that such variation follows a certain distribution, e.g., Gaussian distribution. While since the buckling of beams and columns play a significant role during the deformation, another suitable way to represent the imperfections is to superpose suitable mode shapes to a perfect geometry, and the proportion of each mode shape is a random variable. As a result, extra parameters will be required in the input to describe the imperfections, and a more sophisticated machine learning model should be applied to reflect the expanded design domain.

The Bayesian machine learning model² can be a promising choice for such a probabilistic problem, and the power of which has been demonstrated in material design⁷³. It is also noteworthy that the buckling directions of beams and columns can have significant influence on the subsequent global behavior, and thus we can further manipulate the response if the buckling direction can be controlled. For example, the most common case under axial compression is that all the columns on the same level buckle towards the same direction, in which the global stiffness is smaller than the case that the adjacent columns buckle towards each other forming internal support and leading to smaller lateral drift. Our recent study on deterministic buckling and energy trapping can be a promising methodology to propel this future direction⁷⁴.

METHODS

Finite element modeling

In order to generate the mechanical responses for all the possible patterns under axial compression, we developed a Python script for the commercial software ABAQUS/Explicit® 2018 to automatically read the binary geometric matrices and build the corresponding FE models. The compression tests were solved as a 2D quasi-static problem with an assigned out-of-plane depth of 22 mm. The Explicit solver was applied due to its outstanding efficiency and convergence under complex contacts and large deformations. The kinetic energy was confined to be less than 1% of the internal energy so that the compression was guaranteed to be quasi-static. An elastoplastic constitution was used to describe the PLA, detailed elastoplastic parameters are shown in Supplementary Table 1. The density, elastic modulus, Poisson's ratio, and yield stress are 0.00124 g/mm³, 2114.6 MPa, 0.3, and 72.116 MPa, respectively. Frictionless and "hard" self-contact is applied for each enclosed inner boundary and two vertical outer lines. The FE simulations were parallelly performed on a 16-core Intel® Xeon® E5-2698 V4 @2.20 GHz PC with 64GB Ram. It took approximately 20 seconds on average for the simulation of each pattern.

Correction of predicted responses

As discussed in Fig. 1c, the ML model does not naturally know that a unique pattern and the corresponding reproduced counterparts should have identical response, it would predict similar but different responses for these patterns. These different predictions are usually offset from the reference, but the average of the predictions is usually much closer to the reference. Thus, in this study, we used the averaged prediction as the final result for the unique pattern and its reproduced patterns. A typical example is shown in Supplementary Fig. 2a using 80% training ratio. Supplementary Fig. 2b-d show the comparisons between using the averaged predictions and the non-averaged. It is obvious that the averaging process significantly enhance the performance. Comparing the probability $P(-0.1 < NE < 0.1)$ in Supplementary Fig. 2b with that in Fig. 3a, the improvement of using averaged predictions is similar to increasing training ratio from 40% to 80%. In this study, we used averaged predictions if no specific description.

DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding author upon request.

Received: 17 November 2019; Accepted: 20 March 2020;

Published online: 23 April 2020

REFERENCES

- Silver, D. et al. Mastering the game of Go without human knowledge. *Nature* **550**, 354 (2017).
- Ghahramani, Z. Probabilistic machine learning and artificial intelligence. *Nature* **521**, 452 (2015).
- Melnikov, A. A. et al. Active learning machine learns to create new quantum experiments. *Proc. Natl Acad. Sci. USA* **115**, 1221–1226 (2018).
- Salehi, H. & Burgueño, R. Emerging artificial intelligence methods in structural engineering. *Eng. Struct.* **171**, 170–189 (2018).
- Topol, E. J. High-performance medicine: the convergence of human and artificial intelligence. *Nat. Med.* **25**, 44–56 (2019).
- Chen, L.-Q. et al. Design and discovery of materials guided by theory and computation. *npj Comput. Mater* **1**, 1–2 (2015).
- Sumpter, B. G., Vasudevan, R. K., Potok, T. & Kalinin, S. V. A bridge for accelerating materials by design. *npj Comput Mater* **1**, 1–11 (2015).
- Liu, Y., Zhao, T., Ju, W. & Shi, S. Materials discovery and design using machine learning. *J. Materiomics* **3**, 159–177 (2017).
- Butler, K. T., Davies, D. W., Cartwright, H., Isayev, O. & Walsh, A. Machine learning for molecular and materials science. *Nature* **559**, 547–555 (2018).
- Mannodi-Kanakkithodi, A., Piloni, G., Huan, T. D., Lookman, T. & Ramprasad, R. Machine learning strategy for accelerated design of polymer dielectrics. *Sci. Rep.* **6**, 20952 (2016).
- Raccuglia, P. et al. Machine-learning-assisted materials discovery using failed experiments. *Nature* **533**, 73–76 (2016).
- Xue, D. et al. Accelerated search for materials with targeted properties by adaptive design. *Nat. Commun.* **7**, 11241 (2016).
- Bartók, A. P. et al. Machine learning unifies the modeling of materials and molecules. *Sci. Adv.* **3**, e1701816 (2017).
- Carrasquilla, J. & Melko, R. G. Machine learning phases of matter. *Nat. Phys.* **13**, 431–434 (2017).
- Zhou, Q. et al. Learning atoms for materials discovery. *Proc. Natl Acad. Sci. USA* **115**, E6411–E6417 (2018).
- Jie, J. et al. Discovering unusual structures from exception using big data and machine learning techniques. *Sci. Bull.* **64**, 612–616 (2019).
- Lookman, T., Balachandran, P. V., Xue, D. & Yuan, R. Active learning in materials science with emphasis on adaptive sampling using uncertainties for targeted design. *npj Comput Mater* **5**, 1–17 (2019).
- Nyshadham, C. et al. Machine-learned multi-system surrogate models for materials prediction. *npj Comput Mater* **5**, 1–6 (2019).
- Umehara, M. et al. Analyzing machine learning models to accelerate generation of fundamental materials insights. *npj Comput Mater* **5**, 1–9 (2019).
- Wu, Y.-J., Fang, L. & Xu, Y. Predicting interfacial thermal resistance by machine learning. *npj Comput Mater* **5**, 56 (2019).
- Rafei, M. H., Khushefati, W. H., Demirboga, R. & Adeli, H. Neural network machine learning, and evolutionary approaches for concrete material characterization. *ACI Mater. J* **113**, 781–789 (2016).
- Rafei, M. H., Khushefati, W. H., Demirboga, R. & Adeli, H. Supervised deep restricted Boltzmann machine for estimation of concrete. *ACI Mater. J* **114**, 237–244 (2017).
- Yang, K. et al. Predicting the Young's modulus of silicate glasses using high-throughput molecular dynamics simulations and machine learning. *Sci. Rep.* **9**, 8739 (2019).
- Liu, H. et al. Predicting the dissolution kinetics of silicate glasses by topology-informed machine learning. *npj Mater. Degrad.* **3**, 32 (2019).
- Liu, R. et al. A predictive machine learning approach for microstructure optimization and materials design. *Sci. Rep.* **5**, 11551 (2015).
- Ward, L., Agrawal, A., Choudhary, A. & Wolverton, C. A general-purpose machine learning framework for predicting properties of inorganic materials. *npj Comput Mater* **2**, 16028 (2016).
- Bassman, L. et al. Active learning for accelerated design of layered materials. *npj Comput Mater* **4**, 1–9 (2018).
- Gu, G. X., Chen, C.-T. & Buehler, M. J. De novo composite design based on machine learning algorithm. *Extrem. Mech. Lett.* **18**, 19–28 (2018).
- Liu, Z., Zhu, D., Rodrigues, S. P., Lee, K. T. & Cai, W. Generative model for the inverse design of metasurfaces. *Nano Lett.* **18**, 6570–6576 (2018).
- Ma, W., Cheng, F. & Liu, Y. Deep-learning-enabled on-demand design of chiral metamaterials. *ACS Nano* **12**, 6326–6334 (2018).
- Peurifoy, J. et al. Nanophotonic particle simulation and inverse design using artificial neural networks. *Sci. Adv.* **4**, eaar4206 (2018).
- Pilozzi, L., Farrelly, F. A., Marcucci, G. & Conti, C. Machine learning inverse problem for topological photonics. *Commun. Phys* **1**, 1–7 (2018).
- Sanchez-Lengeling, B. & Aspuru-Guzik, A. Inverse molecular design using machine learning: generative models for matter engineering. *Science* **361**, 360–365 (2018).
- Hamel, C. M. et al. Machine-learning based design of active composite structures for 4D printing. *Smart Mater. Struct.* **28**, 065005 (2019).
- Jennings, P. C., Lysgaard, S., Hummelshøj, J. S., Vegge, T. & Bliigaard, T. Genetic algorithms for computational materials discovery accelerated by machine learning. *npj Comput Mater* **5**, 1–6 (2019).
- Rafei, M. H., Khushefati, W. H., Demirboga, R. & Adeli, H. Novel approach for concrete mixture design using neural dynamics Model and Virtual Lab Concept. *ACI Mater. J* **114**, 117–127 (2017).
- Bessa, M. A. & Pellegrino, S. Design of ultra-thin shell structures in the stochastic post-buckling range using Bayesian machine learning and optimization. *Int. J. Solids Struct.* **139–140**, 174–188 (2018).

38. Bessa, M. A. et al. A framework for data-driven analysis of materials under uncertainty: countering the curse of dimensionality. *Computer Methods Appl. Mech. Eng.* **320**, 633–667 (2017).
39. Chen, C.-T. & Gu, G. X. Effect of constituent materials on composite performance: exploring design strategies via machine Learning. *Adv. Theory Simul.* **2**, 1900056 (2019).
40. Hanakata, P. Z., Cubuk, E. D., Campbell, D. K. & Park, H. S. Accelerated search and design of Stretchable graphene kirigami using machine learning. *Phys. Rev. Lett.* **121**, 255304 (2018).
41. Gu, G. X., Chen, C.-T., Richmond, D. J. & Buehler, M. J. Bioinspired hierarchical composite design using machine learning: simulation, additive manufacturing, and experiment. *Mater. Horiz.* **5**, 939–945 (2018).
42. Xu, H., Liu, R., Choudhary, A. & Chen, W. A machine learning-based design representation method for designing heterogeneous microstructures. *J. Mech. Des.* **137**, 051403 (2015).
43. Schaedler, T. A. & Carter, W. B. Architected cellular materials. *Annu. Rev. Mater. Res.* **46**, 187–210 (2016).
44. Bertoldi, K., Vitelli, V., Christensen, J. & van Hecke, M. Flexible mechanical metamaterials. *Nat. Rev. Mater.* **2**, 17066 (2017).
45. Coulais, C., Teomy, E., de Reus, K., Shokef, Y. & van Hecke, M. Combinatorial design of textured mechanical metamaterials. *Nature* **535**, 529 (2016).
46. Overvelde, J. T. B., Weaver, J. C., Hoberman, C. & Bertoldi, K. Rational design of reconfigurable prismatic architected materials. *Nature* **541**, 347 (2017).
47. Overvelde, J. T. B., Shan, S. & Bertoldi, K. Compaction through buckling in 2D periodic, soft and porous structures: effect of pore shape. *Adv. Mater.* **24**, 2337–2342 (2012).
48. Berger, J. B., Wadley, H. N. G. & McMeeking, R. M. Mechanical metamaterials at the theoretical limit of isotropic elastic stiffness. *Nature* **543**, 533 (2017).
49. Liu, S., Azad, A. I. & Burgueño, R. Architected materials for tailorable shear behavior with energy dissipation. *Extrem. Mech. Lett.* **28**, 1–7 (2019).
50. Pham, M.-S., Liu, C., Todd, I. & Lerthanasarn, J. Damage-tolerant architected materials inspired by crystal microstructure. *Nature* **565**, 305–311 (2019).
51. Liu, J. et al. Harnessing buckling to design architected materials that exhibit effective negative swelling. *Adv. Mater.* **28**, 6619–6624 (2016).
52. Meza, L. R. et al. Resilient 3D hierarchical architected metamaterials. *Proc. Natl Acad. Sci.* **112**, 11502–11507 (2015).
53. Kim, Y., Yuk, H., Zhao, R., Chester, S. A. & Zhao, X. Printing ferromagnetic domains for untethered fast-transforming soft materials. *Nature* **558**, 274–279 (2018).
54. Ding, H. et al. Controlled microstructural architectures based on smart fabrication strategies. *Adv. Funct. Mater.* **30**, 1901760 (2019).
55. Muth, J. T., Dixon, P. G., Woish, L., Gibson, L. J. & Lewis, J. A. Architected cellular ceramics with tailored stiffness via direct foam writing. *Proc. Natl Acad. Sci. USA* **114**, 1832–1837 (2017).
56. Xia, X. et al. Electrochemically reconfigurable architected materials. *Nature* **573**, 205–213 (2019).
57. Zadpoor, A. A. Mechanical meta-materials. *Mater. Horiz.* **3**, 371–381 (2016).
58. Surjadi, J. U. et al. Mechanical metamaterials and their engineering applications. *Adv. Eng. Mater.* **21**, 1800864 (2019).
59. Kadic, M., Milton, G. W., van Hecke, M. & Wegener, M. 3D metamaterials. *Nat. Rev. Phys.* **1**, 198–210 (2019).
60. Ma, C. et al. Exploiting spatial heterogeneity and response characterization in non-uniform architected materials inspired by slime mould growth. *Bioinspiration Biomim.* **14**, 064001 (2019).
61. Goldberg, Y. Neural network methods for natural language processing. *Synth. Lectures Hum. Lang. Technol.* **10**, 1–309 (2017).
62. Sarikaya, R., Hinton, G. E. & Deoras, A. Application of deep belief networks for natural language understanding. *IEEE/ACM Trans. Audio, Speech Lang. Process.* (TASLP) **22**, 778–784 (2014).
63. Aich, A., Dutta, A. & Chakraborty, A. A scaled conjugate gradient backpropagation algorithm for keyword extraction. In *Information Systems Design and Intelligent Applications* (Springer, 2018).
64. Heravi, A. R. & Hodtani, G. A. A new correntropy-based conjugate gradient backpropagation algorithm for improving training in neural networks. *IEEE Trans. Neural Netw. Learn. Syst.* **29**, 6252–6263 (2018).
65. Kim, G., Hwang, C. S. & Jeong, D. S. Stochastic Learning with Back Propagation. In *2019 IEEE International Symposium on Circuits and Systems (ISCAS)* (IEEE, 2019).
66. Shi, X. & Shen, M. A new approach to feedback feed-forward iterative learning control with random packet dropouts. *Appl. Math. Comput.* **348**, 399–412 (2019).
67. Cho, K. H., Raiko, T. & Ilin, A. Gaussian-bernoulli deep boltzmann machine. In *The 2013 International Joint Conference on Neural Networks (IJCNN)* (IEEE, 2013).
68. Liu, Z. & Wu, C. T. Exploring the 3D architectures of deep material network in data-driven multiscale mechanics. *J. Mech. Phys. Solids* **127**, 20–46 (2019).
69. Liu, D., Tan, Y., Khoram, E. & Yu, Z. Training deep neural networks for the inverse design of nanophotonic structures. *ACS Photon.* **5**, 1365–1369 (2018).
70. Malkiel, I. et al. Plasmonic nanostructure design and characterization via deep learning. *Light Sci. Appl.* **7**, 60 (2018).
71. Cheng, G., Han, J., Zhou, P. & Xu, D. Learning rotation-invariant and fisher discriminative convolutional neural networks for object detection. *IEEE Trans. Image Process.* **28**, 265–278 (2018).
72. Young, T., Hazarika, D., Poria, S. & Cambria, E. Recent trends in deep learning based natural language processing [Review Article]. *IEEE Comput. Intell. Mag.* **13**, 55–75 (2018).
73. Bessa, M. A., Glowacki, P. & Houlder, M. Bayesian machine learning in meta-material design: fragile Becomes supercompressible. *Adv. Mater.* **31**, 1904845 (2019).
74. Zhao, Y. et al. Deterministic snap-through buckling and energy trapping in axially-loaded notched strips for compliant building blocks. *Smart Mater. Struct.* **29**, 02LT03 (2020).

ACKNOWLEDGEMENTS

This work was supported by the startup fund from the College of Engineering at The Ohio State University (N.H.). The authors would like to acknowledge the assistance from Mr. Andrew Schellenberg for gathering literature for the project. Finally, the simulations and machine learning training were run through the computational facilities at OSU Simulation Innovation and Modeling Center.

AUTHOR CONTRIBUTIONS

C.M., M.R., and N.H. conceived the idea; C.M. and M.R. developed the algorithm; C.M. and Z.Z. performed numerical simulations; B.L., S.P., and B.X. supported pilot experiments and simulations for the study; and N.H. wrote the paper with inputs from all authors. C.M. and Z.Z. are co-first author.

COMPETING INTERESTS

The authors declare no competing interests.

ADDITIONAL INFORMATION

Supplementary information is available for this paper at <https://doi.org/10.1038/s41524-020-0309-6>.

Correspondence and requests for materials should be addressed to M.H.R. or N.H.

Reprints and permission information is available at <http://www.nature.com/reprints>

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this license, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2020