Article

Physics-inspired transfer learning for ML-prediction of CNT band gaps from limited data

Check for updates

Ksenia V. Bets ⓘ , Patrick C. O'Driscoll & Boris I. Yakobson ⓘ ✉

Recent years have seen a drastic increase in the scientific use of machine learning (ML) techniques, yet their applications remain limited for many fields. Here, we demonstrate techniques that allow overcoming two obstacles to the widespread adoption of ML, particularly relevant to nanomaterials and nanoscience fields. Using the prediction of the band gap values of carbon nanotubes as a typical example, we address the representation of the periodic data as well as training on extremely small datasets. We successfully showed that careful choice of the *activation function* allows capturing periodic tendencies in the datasets that are common in physical data and previously posed significant difficulty for neural networks. In particular, utilization of the recently proposed parametric periodic Snake activation function shows a dramatic improvement. Furthermore, tackling a typical lack of accurate data, we used the *transfer learning* technique utilizing more abundant low-quality computational data and achieving outstanding accuracy on a significantly expanded dataspace. This strategy was enabled by the use of a combination of the Snake and ReLU layers, capturing data periodicity and amplitude, respectively. Hence, retraining only ReLU layers allowed the transfer of the periodic tendencies captured from low-quality data to the final high-accuracy neural network. Those techniques are expected to expand the usability of ML approaches in application to physical data in general and the fields of nanomaterials in particular.

Over recent years, the use of various machine learning (ML) methods saw a drastic increase in material and nano-science fields. Overcoming the most common ML limitation—the necessity for massive amounts of data—the applications primarily focused on utilizing well-developed image processing approaches[1–3], optimizing automated experimentation techniques and existing data mining[4–6], and using more widely available computational data[7–11]. Such an approach, however, can only be of use in special cases, preventing the application of ML to typical systems with scarce experimental data. The reasonably accessible computational data commonly lack the required accuracy to naturally supplement experimental results. It is also common for computed and experimental data to not explicitly correspond to the same system, e.g., experimentally measured parameters can be strongly affected by the presence of structural defects that can't be accounted for in computations due to inherent scale differences. Furthermore, nanomaterial properties are highly dependent on the material's size and structure, resulting in a small set of discrete values rather than a large continuous data space, placing a hard limit on data availability. Another common complication to applying ML solutions to such problems is the periodic

nature of phenomena that need to be captured. ML solutions often have difficulty representing periodic behaviors and typically use larger, more complex models that require even more data to train.

Here, we demonstrate how those obstacles (small amount of accurate data, periodic nature of the physical properties, etc.) can be overcome with careful selection of the *activation function*, and use of the *transfer learning* approach, which is further improved with *physics-inspired limitations*. As a familiar and important example, we chose the prediction of the band gap values of the carbon nanotubes (CNT) from their chiral indices (n,m), which presents a well-known step-like periodic behavior with a characteristic period of 3 from (n-m) values. First, we will show that while the most common ML approaches fail to represent such periodic functions due to limited available data, the recently proposed Snake activation function shows greatly improved results.

Posing a second obstacle, the discrete nature of the data, characteristic of nanomaterials, significantly limits the upper limit of the available data, creating a very difficult task for ML. As is typical for many relevant nanosystems, the available experimental data is insufficient for use with ML

Department of Materials Science and NanoEngineering, Rice University, Houston, TX, USA. ✉e-mail: biy@rice.edu

methods; the thorough literature search produced only 137 experimental and high-accuracy computational values[12–17] (Fig. 1a) (see SI for a complete list of used values and sources). Consequently, it is common to attempt enrichment of the dataset through affordable computations; in our case, the DFTB (density functional tight binding) method provides the values for 851 CNTs[18,19] (Fig. 1a). Unfortunately, such data, even if precise per se, often lacks accuracy 'across the pools'—experimental and computed. Not only does the magnitude of band gaps significantly vary between experimental and DFTB results, but the fine details of the trend for semiconducting tubes do not match (Fig. 1b), ruling out the use of the mix of two available datasets together. In this work, we demonstrate that this low-quality data can still be useful, enabling the learning of general trends. Later employing the transfer learning (TL) approach, this rough model is re-trained to accurately represent experimental results despite extremely sparse data.

The physics of band gaps in CNTs is well understood, and it derives from linear band dispersion at the Fermi level near the K point in a rolled-up graphene sheet[20,21]. It must be mentioned that over the years, scientists proposed various equations predicting the CNT band gap fitted to experimental data. Such equations, based on a theoretical understanding of the band gap origins or purely empirical, provide reasonably accurate values with a computational speed that can not be matched even by the simplest ML approach[22–24]. The goal of this manuscript is not to compete with those equations or even to predict the band gap values but rather to illustrate a successful transfer learning ML approach capable of handling challenging periodic data while being trained on a realistically small dataset. The existence of the empirical equations provides a convenient way to prove the methodology's effectiveness and evaluate performance without being used to generate training data.

Results and discussion
Representation of periodic functions with machine learning
First, we must address the ability of ML to represent periodic physical data, which often poses a challenge and requires the use of overly complex models and, hence, an increased amount of data. For this first stage, we used the
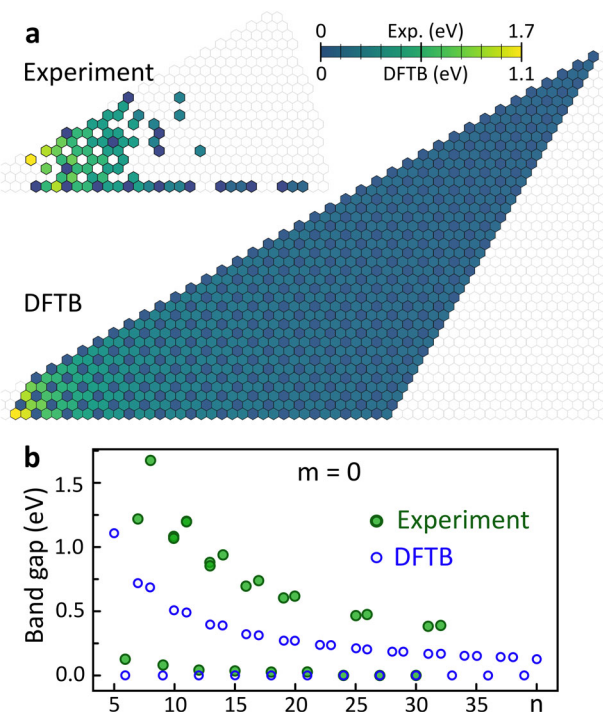

Fig. 1 | Data. a The range and values of the available experimental and DFTB data for the CNT band gap (the color bar is shown in the top right). **b** Comparison of the experimental (hollow blue) and DFTB (filled green) values of $(n,0)$ CNTs highlighting the difference in magnitude and the fine details of the trend.

more numerous DFTB data, randomly split into training and testing datasets in a 70:30 ratio (see Methods section for details). To establish a baseline and establish the failure of the common approach in this situation, we trained a simple 2-layer neural network (NN) with a variable width layer using a popular ReLU (Rectified Linear Unit)[25,26] activation function, $ReLU(x) = \max(0, x)$. For simplicity, the averaged absolute error value was used as a loss function (L1 loss). The plots of training and testing loss for this 2xReLU NNs display all common training characteristics (Fig. 2a): underfitting for smaller NNs (width below 200 neurons per layer) where the model simplicity prevents accurate data representation, overfitting for larger NN (above 512 neurons per layer) where the training data memorization prevents generalization and somewhat accurate prediction of testing data for the moderate size of NN. The NNs of optimal size were able to only achieve the relatively low accuracy of $\varepsilon_{max} \cong 0.45$ eV and, more importantly, failed to accurately capture the periodicity. Note that we use the maximal absolute error ε_{max} for an individual CNT band gap prediction of the best-performing NN to characterize the performance through the ultimate guaranteed accuracy of each band gap prediction. To simplify the visualization, we plot DFTB data and ML predictions for only zigzag CNTs ($m = 0$), which, in the case of 2xReLU NNs, clearly show the absence of any kind of periodic behavior, a shortcoming typical for conventional activation functions (Fig. 2b).

It should be mentioned that in simpler cases, one can devise a scheme that would separate the data into distinct sets that do not display periodicity, for example, using a period of 3 for our data, completely sidestepping the problem. However, this does not represent a generic solution and, significantly, would even further reduce the amount of data available for each set.

Recently, Ziyin et al.[27] addressed the problem of representing periodic functions with NNs by creating a parametric periodic activation function named Snake $:= x + \sin^2(ax)/a$, where parameter a can be learned within the optimization algorithm or set by the user. While being less computationally efficient, 2-layer NN with Snake activation functions significantly outperforms ReLU in representing periodic data (Fig. 2c, d). Not only the performance of 2xSnake NNs is improved to $\varepsilon_{max} \cong 0.2$ eV, but more importantly, the periodicity is accurately captured (Fig. 2d). Interestingly, the loss value changes with the width of the layer does not show typical overfitting behavior.

For comparison, we also evaluated the simpler traditional periodic activation function - $\sin(ax)$, which showed performance slightly below that of Snake (Supplementary Fig. 1). Notably, NN with $\sin(ax)$ showed a tendency to be trapped in local minimums[28], resulting in reduced stability manifested in significantly increased error bars in Supplementary Fig. 1e.

Further, by combining two layers with Snake and two layers with ReLU activation functions, and varying the width of layers (both Snake and both ReLU layers are set to the same width), we create the most complex model that would be considered. In principle, such architecture should separate the periodic trend to be captured by Snake layers and the magnitude trends in ReLU layers and allow for better performance, as well as the transfer learning approach we will discuss later.

Plotting the ε_{max} for the best-performing NN of a given architecture against the widths of Snake and ReLU layers, we find the region of optimal performance for 2xSnake-2xReLU NNs (highlighted in green in Fig. 2e). The periodicity of the data was well captured by those NN (Fig. 2f) with the accuracy even further improved significantly to $\varepsilon_{max} \cong 0.0075$ eV. Note that overfitting for larger NN is again present due to the use of ReLU layers (top right corner of Fig. 2e).

Originally, Ziyin et al.[27] showed the use of the Snake for conventional, continuous periodic functions. Obtaining those promising results, we have clearly confirmed the ability of NN with the Snake[27] activation function to reproduce almost step-like periodic trends in the discrete physical data where more common activation functions fail. The rather remarkable ability of Snake to extrapolate should also be mentioned. The importance of this for physical data ML cannot be overstated. While we use a rather simple case here, significantly more complex periodic
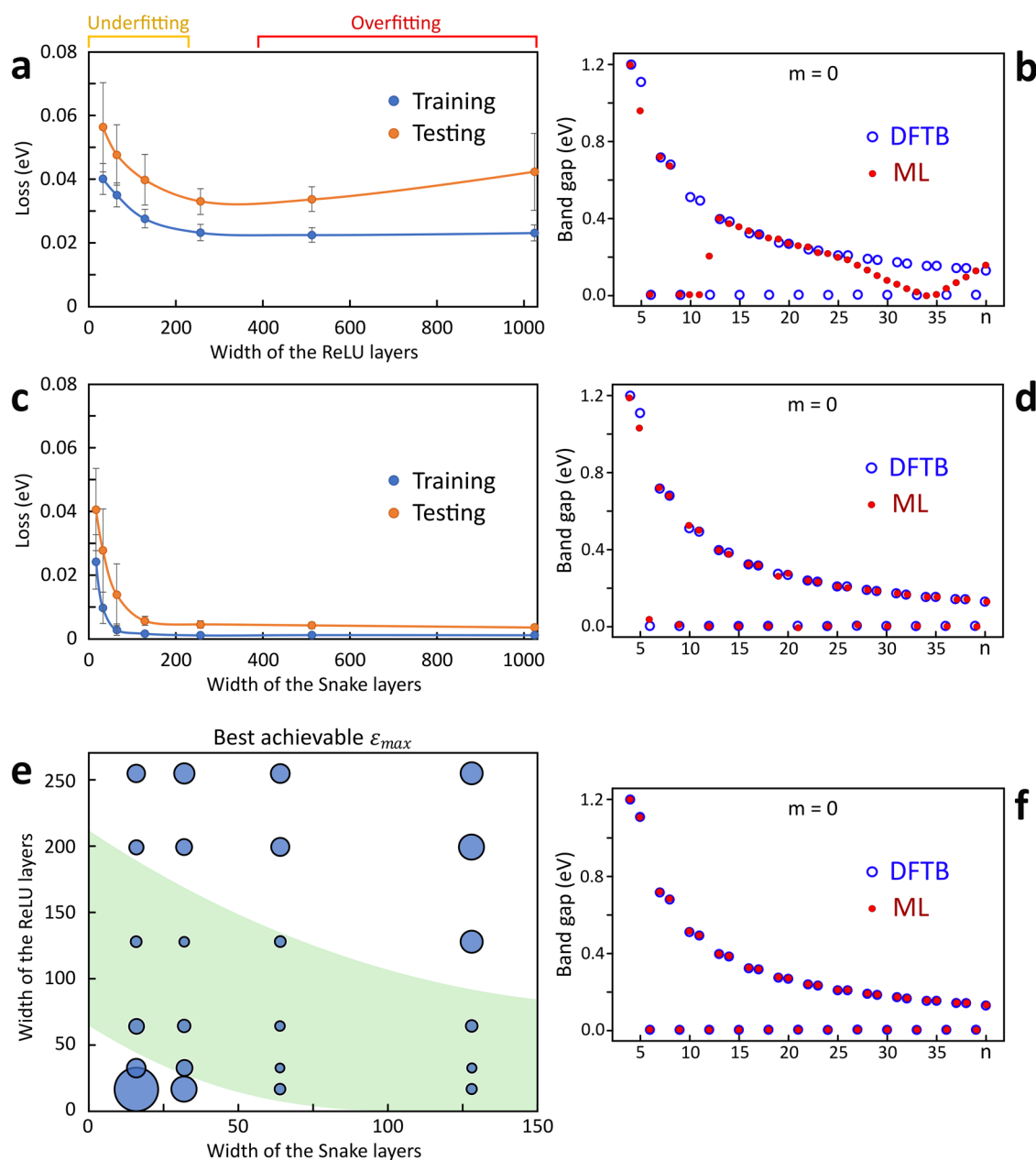
Fig. 2 | Representation of the periodic function with ML, results on DFTB data. ML results of two-layer NN with ReLU **a**, **b** and Snake **c**, **d** activation functions and four-layer NN with two Snake and two ReLU layers **e**, **f**. The best achievable ε_{max} by the 4-layer NN (circle size) **e** with corresponding widths of the Snake and ReLU layers was used to identify combinations of widths resulting in the best performance (shaded green area). Simplified (m = 0) visualization of the best performance 2xReLU **b**, 2xSnake **d**, and 2xSnake-2xReLU **f** networks demonstrating representation of the periodic data. NN with just ReLU layers fails to capture the general trend of the data **b** with $\varepsilon_{max} > 0.45$ eV. The performance with just Snake is greatly improved, well-representing periodic data, $\varepsilon_{max} > 0.2$ eV **d**. The combination of Snake and ReLU shows the best performance with $\varepsilon_{max} > 0.0075$ eV **f**. Error bars indicate standard deviation.

functions, such as Hamiltonians, could be of interest to the nanomaterials community.

Transfer learning for ML of small experimental datasets

Our second goal was to overcome the scarcity of accurate computational and experimental data. Characteristically for discrete physical data, one is not interested in predictive interpolation between available data points; for example, in our case, such predictions would correspond to non-existent CNTs with partial chiral indexes. It would be of significant benefit, however, to predict the values of the band gaps for a much wider range of CNTs than is already described by the experimental data, looking to create an extrapolative model. To this end, we are going to use the transfer learning technique, where the NN previously trained on the low-quality data (DFTB) is then partially re-trained on limited, accurate experimental data. In our particular case, we start with the best-performing 2xSnake-2xReLU NNs trained on DFTB data and re-train only the ReLU layers (Fig. 3a). The distribution of learned Snake period a for both layers is shown in Supplementary Fig. 4. The motivation for this particular architecture is to preserve the pre-learned periodicity of the data present in DFTB results and captured by the parameters of the Snake layers. To evaluate the extrapolative performance, we would consider both maximal error ε_{max} and average error $\langle\varepsilon\rangle$ for the training set (comparison with the experimental data) and testing set (comparison with the values predicted with empirical equations[22–24] on the range of DFTB data), for convenience we would refer to those characteristics as an experimental (Exp.) and extended range (Ext. Range). We train 50 NN instances with randomized
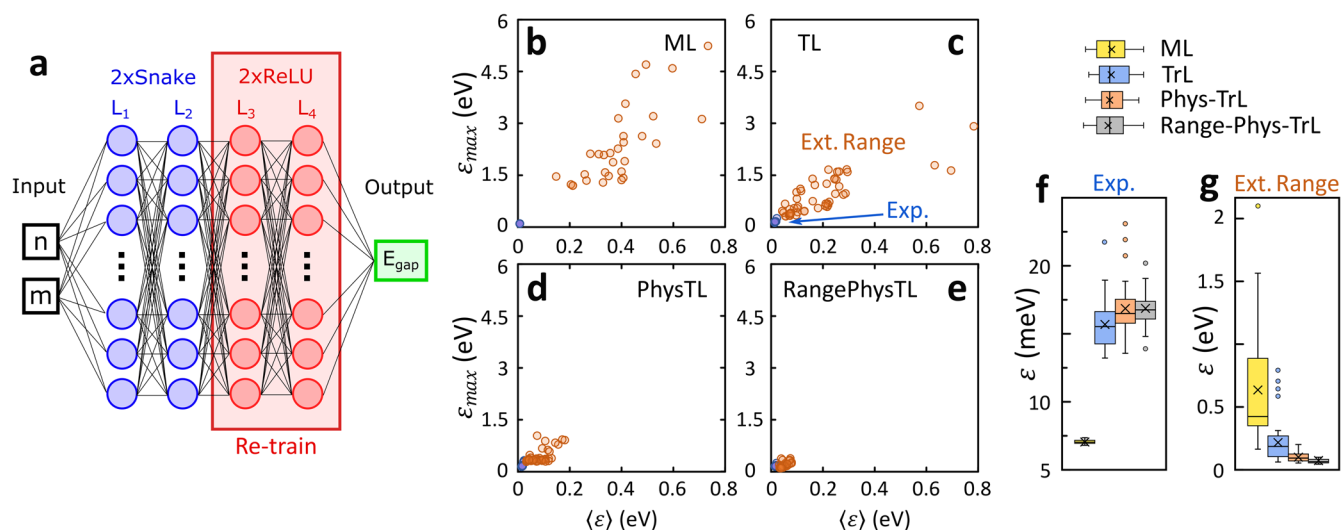
Fig. 3 | Transfer learning ML. a Schematic of the 2xSnake-2xReLU NN used for the transfer learning, where the parameters of the first two layers are kept constant after pre-training on DFTB data while 2 ReLU layers are re-trained using only the experimental dataset. **b–e** The performance of 2xSnake-2xReLU NNs (Snake layer width of 16 and ReLU layer width of 64 nodes) performance on experimental (Exp., blue) in comparison to the extrapolative performance of the extended range (Ext. Range, orange) trained with simple training (ML, **b**), transfer learning (TrL, **c**), transfer learning with physics-inspired restrictions (Phys-TrL, **d**), and transfer learning with physics inspired restriction on the reduced range of CNTs with the diameter smaller than (40,0) (Range-Phys-TrL, **e**). The zoomed-in version of panels **b–e** is shown in Supplementary Fig. 3. Whisker plots (showing outliers, mean, median and all quartile boundaries) of results for simple and transfer learning approaches on experimental **f** and expanded ranges **g** at different approximations.
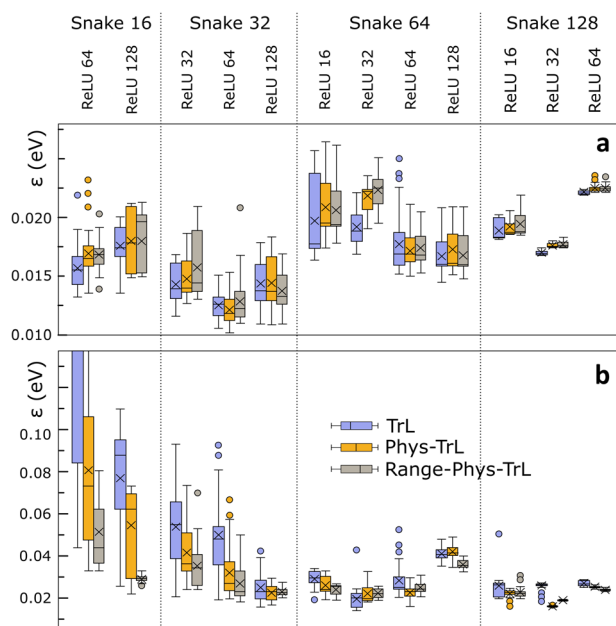

Fig. 4 | Transfer learning results. Experimental **a** and extended range **b** results for 2xSnake-2xReLU NNs with various layer widths partially re-trained on the experimental data. Whisker plots show outliers, mean, median, and all quartile boundaries.

initiation for all considered approaches and NN architectures to evaluate achievable performance.

For an initial comparison, we start with the NN architecture with 2 Snake layers 16 nodes (neurons) wide, and 2 ReLU layers 64 nodes wide (performance results shown in Fig. 3b, e and Fig. 3f, g)—the smallest NN within the optimal region highlighted in Fig. 2e. We perform a from-scratch training of 2xSnake-2xReLU NN on experimental data for completeness (marked as ML). Due to the significant complexity of the model and the small size of the dataset, the results are plagued by overfitting, showing

outstanding results in the prediction of the data points within the experimental dataset (the best performance $\varepsilon_{max} = 0.097$ eV and $\langle\varepsilon\rangle = 0.007$ eV) and poor performance on the extended range (the best performance $\varepsilon_{max} = 1.20$ eV and $\langle\varepsilon\rangle = 0.15$ eV) (Fig. 3b). The experimental and extended ranges are shown in Supplementary Fig. 2. The employment of the transfer learning procedure (marked as TrL in Fig. 3 and Fig. 4) described above markedly improve the extrapolative performance of the model, lowering achieved error level to $\varepsilon_{max} = 0.31$ eV and $\langle\varepsilon\rangle = 0.043$ eV (Fig. 3c).

Even further improvement can be achieved by incorporating some restrictions based on the understanding of the physical nature of the data. Including such additional rules during the optimization process is a common approach that allows scientists to leverage preexisting knowledge of the phenomena, only requiring that the restrictions are formulated to be compatible with the form of the loss function. Such physics-inspired restrictions generally are accomplished through the design of both restrictions and loss functions ahead of time. In our case, relying on an understanding of the band gap nature, we include the additional condition that punishes the prediction of any negative values (marked as Phys-TrL in Figs. 3 and 4). Despite being relatively simple, this modification is effective in improving the results on the extended range to $\varepsilon_{max} = 0.28$ eV and $\langle\varepsilon\rangle = 0.032$ eV (Fig. 3d). Even further improvement can be achieved by limiting the range of extrapolation (marked as Range-Phys-TrL in Figs. 3 and 4). Considering only CNTs with a size below the diameter of (40,0) nanotube, the performance can reach $\varepsilon_{max} = 0.098$ eV and $\langle\varepsilon\rangle = 0.030$ eV (Fig. 3e). Note that while the performance on the extended range is improved, the restrictions applied to the optimization inevitably results in reduced performance on the training data (Fig. 3f, g).

For easier comparison, we also provide the whisker charts of results on the experimental (Fig. 3f) and extended range (Fig. 3g) for simple ML, transfer learning, transfer learning with physics-inspired restrictions, and transfer learning with physics-inspired restrictions on the reduced range. As mentioned above, it is easy to see that the performance on the training set worsens with additional restrictions, while the results on the extended range drastically improve.

Now that the potential of the transfer learning is shown on a single NN architecture, we proceed to test various combinations of Snake and ReLU layer widths, using the best-performing DFTB-trained NN as a starting

Fig. 5 | DFTB data preparation. The chiral map showing the full DFTB dataset (851 points) and randomly chosen 592 data points (~70% of the full set) that were further randomly upsampled to 1024 data points used as a training set.
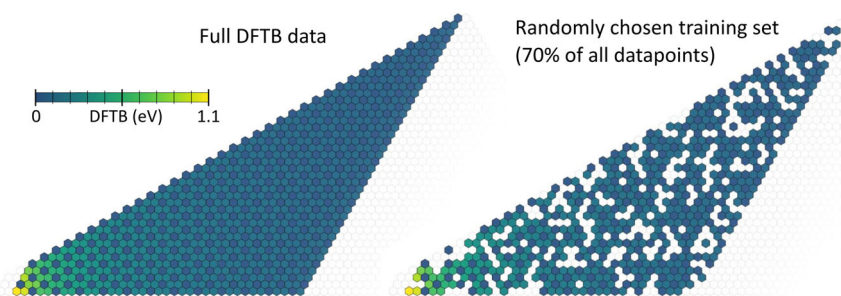
point. We limit our investigation to the NN highlighted in green in Fig. 2e. As previously, for each architecture, 50 instances of NN are trained following TrL, Phys-TrL, and Range-Phys-TrL approaches.

It is clear that the performance on the experimental range is universally good, with the error level slightly dipping at the Snake width of 32 nodes (Fig. 4a). At the same time, the prediction accuracy on the extended range is progressively and significantly improved with the increased complexity of the model, reaching a plateau at a Snake width of 64 (Fig. 4b). Interestingly, not only the best performance is enhanced with higher complexity, but the variability of the results from the produced NNs is also decreased, suggesting more robust models. The accuracy achieved through the transfer learning approach on the extended range reaches $\varepsilon_{max} = 0.091$ eV and $\langle\varepsilon\rangle = 0.016$ eV, on par with that of conventional ML training on DFTB data despite a very small dataset size.

To further investigate the potential usefulness of the outlined methodology on experimental data, we consider its performance of the data with imperfect periodicity. While the local deviations from periodicity are very typical for material science datasets (e.g., localized structural defects within perfect lattice), they are absent in our test case. We artificially introduced such deviations to an increasing number of points within the DFTB dataset (Supplementary Fig. 5), finding only a slight performance decrease for the data with up to 10% non-periodic data (Supplementary Fig. 6). This level of tolerance towards imperfect periodicity in pre-training data opens a variety of possible applications. While used as an example here the CNT bandgaps are indeed a representative and relevant dataset. It is worth reminding that the band gap variability with chiral symmetry of nanotubes has been seen as both opportunity and a hurdle to electronics applications, a decades-long challenge to reveal the origins of chiral type distributions[29,30] and especially to achieve chiral-selective synthesis[31].

In conclusion, we have successfully demonstrated a methodology to overcome several common obstacles to the use of ML on datasets in the nanomaterials field. First, the use of the recently proposed Snake activation function enables the learning of the periodic functions quite common in physical data. Here Snake's effectiveness is illustrated on a *discrete step-wise periodic* function of the CNT band gap that is common for electronic and optical properties of nanostructures (the Periodic Table of the chemical elements is also a compelling example of this kind); yet its use on more conventional continuous periodic functions, such as Hamiltonians[32], can prove to be important for the field of nanomaterials. It can also find application in many other tasks that remain challenging for NN, such as learning symmetry from diffraction images[33,34]. Furthermore, we employed transfer learning techniques by re-using NNs pre-trained on the numerous but inaccurate DFTB data. This approach allowed us to successfully represent accurate experimental data from just 137 data points, clearly illustrating transfer learning capabilities for the typical case of extremely limited data availability. Moreover, the represented range significantly exceeded that of the used data. We believe that the demonstrated approach should significantly expand the usability of ML techniques in the nanomaterial research field.

Methods
Dataset preparation
We used three distinct data sources to compose two different data sets. The first data set was composed of DFTB data[18,19] for training and testing. The DFTB data were used to evaluate the ability of the network to learn characteristically periodic patterns. The second was composed of experimental and high-accuracy DFT[12–17] for training, and the testing was performed using empirical formulas[22–24]. This second dataset was used primarily to evaluate the transfer learning potential of the neural network. The exact datasets used are available upon request.

The DFTB dataset included all valid n and m combinations in [4,40], [0,n], respectively, a total of 851 points (see Fig. 1). The data were then split into training and testing subsets in approximately 70:30 ratio (592 and 259 points respectively) (Fig. 5 visualizes full and training subsets). The training set was randomly upsampled, sampled with replacement to 1024 points from 592 points, and the testing set was left at native 259 points.

The transfer learning training set was composed of 137 training points[12–17] (see Supplementary Information for a complete list of used values and sources) that were upsampled to 342 points by resampling the less represented dataspace (larger diameter and larger chiral angle CNTs). The oversampling of the large diameter CNTs corrected for the underrepresentation in the dataspace with smaller absolute energy values. The testing set was evaluated over the entire n, m range of interest in that experiment. The full range of DFTB data was used except for the transfer learning evaluation on the reduced extended range, where only nanotubes with a diameter below (40,0) nanotubes were included.

Neural network methods
The networks were built using python 3.7.9, pytorch 1.7, and cuda 11.0. We evaluated three different networks in this work, two versions of the two-layer and one four-layer fully connected feed-forward network. All of these are traditional neural networks that include the bias term as part of their topology. For brevity, we will denote the topology of the network by the number of elements in each of the feed-forward layers and by the transfer function used in that layer, as the input and output were the same across all networks studied.

All networks in this work utilized AdamW[35] optimization methodology with an initial learning rate of 10^{-3}. ReLU layers were initialized using He initialization[25]. The Snake layers were initialized using He initialization for the weight component and for the period component the Uniform [0,3] distribution. Unlike the implementation used by Ziyin et al.[27], we allowed both of these components to update with the network. The networks were all trained to minimize the L1 loss between the prediction and the band gap in the dataset. Networks were stopped after they ran for 20×10^6 epochs of the training set data, and the best-performing network was evaluated by L1 loss over the testing set. We also recorded the absolute value of the maxim deviation of any given prediction vs. the actual to evaluate the worst possible prediction of the network.

The first variation of the two-layer network utilized the ReLU transfer function. The second variation of the two-layer network utilized the Snake transfer function. For both types of two-layer networks, we evaluated

networks with the same number of neurons in each of the two hidden layers. The examined sizes are 16, 32, 64, 128, 256, 512, and 1024.

We observed that the Snake layers did an excellent job of learning the underlying periodic behavior; therefore, we decided to use a four-layer network that consisted of two Snake layers followed by two ReLU layers. The Snake layers learned the underlying phenomena, and the ReLU layers would learn the appropriate scale. This allowed us to transfer train the network by relearning only the ReLU layers using the much smaller, highly accurate datasets. To help ensure that these networks were learning physically meaningful outputs, we modified our L1 loss to also penalize negative band gaps energies equivalent to 10x the negative value. This strongly discouraged the network from learning any non-physical solutions. The evaluation of the retrained networks over the range significantly exceeding the limited range of the accurate dataset was performed using empirical data[22–24] (see SI for details) in the range of the DFTB dataset or slightly reduced, as described above. To simplify the evaluation, the widths of the ReLU and Snake layers were varied independently while keeping two ReLU layers and two Snake layers of the same width. We evaluated the four-layer networks in the following layer widths: 16, 32, 64, and 128 for both ReLU and Snake layers, and 200 and 256 for ReLU layers only.

Data availability
The dataset used in this study is included in this published article and its Supplementary Information files. The model parameters generated during the current study are available from the corresponding author upon a reasonable request.

Code availability
The code used in the current study is available from the corresponding author upon a reasonable request.

References
1. Unruh, D., Kolluru, V. S. C., Baskaran, A., Chen, Y. & Chan, M. K. Y. Theory + AI/ML for microscopy and spectroscopy: challenges and opportunities. *MRS Bull.* **47**, 1024 (2022).
2. Chowdhury, A., Kautz, E., Yener, B. & Lewis, D. Image driven machine learning methods for microstructure recognition. *Comput. Mater. Sci.* **123**, 176 (2016).
3. Ziatdinov, M., Ghosh, A., Tommy Wong, C. Y. & Kalinin, S. V. AtomAI framework for deep learning analysis of image and spectroscopy data in electron and scanning probe microscopy. *Nat. Mach. Intell.* **4**, 12 (2022).
4. Reyes, K. G. & Maruyama, B. The machine learning revolution in materials? *MRS Bull.* **44**, 530 (2019).
5. Batra, R., Song, L. & Ramprasad, R. Emerging materials intelligence ecosystems propelled by machine learning. *Nat. Rev. Mater.* **6**, 8 (2021).
6. Blaiszik, B. et al. A data ecosystem to support machine learning in materials science. *MRS Commun.* **9**, 1125 (2019).
7. Dragoni, D., Daff, T. D., Csányi, G. & Marzari, N. Achieving DFT accuracy with a machine-learning interatomic potential: thermomechanics and defects in Bcc ferromagnetic iron. *Phys. Rev. Mater.* **2**, 013808 (2018).
8. Deringer, V. L., Caro, M. A. & Csányi, G. Machine learning interatomic potentials as emerging tools for materials science. *Adv. Mater.* **31**, 1902765 (2019).
9. Batzner, S. et al. E(3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials. *Nat. Commun.* **13**, 1 (2022).
10. Zuo, Y. et al. Performance and cost assessment of machine learning interatomic potentials. *J. Phys. Chem. A* **124**, 731 (2020).
11. Goldman, N., Fried, L. E., Lindsey, R. K., Pham, C. H. & Dettori, R. Enhancing the accuracy of density functional tight binding models through ChIMES many-body interaction potentials. *J. Chem. Phys.* **158**, 144112 (2023).
12. Ouyang, M., Huang, J.-L., Cheung, C. L. & Lieber, C. M. Energy gaps in "metallic" single-walled carbon nanotubes. *Science* **292**, 702 (2001).
13. Bachilo, S. M. et al. Structure-assigned optical spectra of single-walled carbon nanotubes. *Science* **298**, 2361 (2002).
14. Telg, H., Maultzsch, J., Reich, S. & Thomsen, C. Resonant-Raman intensities and transition energies of the ${E}_{11}$ transition in carbon nanotubes. *Phys. Rev. B* **74**, 115415 (2006).
15. Liu, K. et al. An Atlas of carbon nanotube optical transitions. *Nat. Nanotechnol.* **7**, 5 (2012).
16. Barone, V., Peralta, J. E., Wert, M., Heyd, J. & Scuseria, G. E. Density functional theory study of optical transitions in semiconducting single-walled carbon nanotubes. *Nano Lett.* **5**, 1621 (2005).
17. Matsuda, Y., Tahir-Kheli, J. & Goddard, W. A. Definitive band gaps for single-wall carbon nanotubes. *J. Phys. Chem. Lett.* **1**, 2946 (2010).
18. *Kataura Plot by S. Maruyama*, http://www.photon.t.u-tokyo.ac.jp/~maruyama/kataura/kataura.html.
19. Jorio, A. et al. The Kataura plot over broad energy and diameter ranges. *Phys. Status Solidi B* **243**, 3117 (2006).
20. Terrones, M. et al. Electronic, thermal and mechanical properties of carbon nanotubes. *Philos. Trans. A Math. Phys. Eng. Sci.* **362**, 2065 (2004).
21. Spataru, C. D., Ismail-Beigi, S., Benedict, L. X. & Louie, S. G. Excitonic effects and optical spectra of single-walled carbon nanotubes. *Phys. Rev. Lett.* **92**, 077402 (2004).
22. Gelao, G., Marani, R. & Perri, A. G. A formula to determine energy band gap in semiconducting carbon nanotubes. *ECS J. Solid State Sci. Technol.* **8**, M19 (2019).
23. Kleiner, A. & Eggert, S. Band gaps of primary metallic carbon nanotubes. *Phys. Rev. B* **63**, 073408 (2001).
24. Ding, J. W., Yan, X. H. & Cao, J. X. Analytical relation of band gaps to both chirality and diameter of single-wall carbon nanotubes. *Phys. Rev. B* **66**, 073401 (2002).
25. He, K. Zhang, X., Ren, S. & Sun, J. Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. *Proc. IEEE Int. Conf. Comput. Vis.* 1026–1034 (2015).
26. Strang, G. *Linear Algebra and Learning from Data*, 1st ed. (Wellesley-Cambridge Press, Wellesley, MA, 2019).
27. Ziyin, L., Hartwig, T. & Ueda, M. Neural Networks Fail to Learn Periodic Functions and How to Fix It. *Adv. Neural. Inf. Process. Syst.* **33**, 1583–1594 (2020).
28. Parascandolo, G., Huttunen, H. & Virtanen, T. *Taming the Waves: Sine as Activation Function in Deep Neural Networks*, https://openreview.net/forum?id=Sks3zF9eg (2016).
29. Penev, E. S., Bets, K. V., Gupta, N. & Yakobson, B. I. Transient kinetic selectivity in nanotubes growth on solid Co–W catalyst. *Nano Lett.* **18**, 5288–5293 (2018).
30. Bets, K. V., Penev, E. S. & Yakobson, B. I. Janus segregation at the carbon nanotube–catalyst interface. *ACS Nano* **13**, 8836–8841 (2019).
31. Yakobson, B. I. & Bets, K. V. Single-chirality nanotube synthesis by guided evolutionary selection. *Sci. Adv.* **8**, eadd4627 (2022).
32. Li, H. et al. Deep-learning density functional theory Hamiltonian for efficient ab initio electronic-structure calculation. *Nat. Comput. Sci.* **2**, 6 (2022).
33. Ra, M. et al. Classification of crystal structures using electron diffraction patterns with a deep convolutional neural network. *RSC Adv.* **11**, 38307 (2021).
34. Funk, C. & Liu, Y. *EscherNet 101*, preprint at https://arxiv.org/abs/2303.04208 (2023).
35. Loshchilov, I. & Hutter, F. *Decoupled Weight Decay Regularization*, preprint at https://arxiv.org/abs/1711.05101 (2017).

Article

Author contributions

All authors contributed to the development of ideas. K.B. collected the data, performed calculations and analysis, and wrote the manuscript. P.O. implemented a computational approach, performed calculations, and contributed to the manuscript writing. All authors contributed to the editing of the final manuscript. All authors read and approved the final manuscript.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s41524-024-01247-0.

Correspondence and requests for materials should be addressed to Boris I. Yakobson.

Reprints and permissions 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.