Article

Transfer Learning Facilitates the Prediction of Polymer−Surface Adhesion Strength

Jiale Shi, Fahed Albreiki, Yamil J. Colón, Samanvaya Srivastava, and Jonathan K. Whitmer*

Cite This: https://doi.org/10.1021/acs.jctc.2c01314

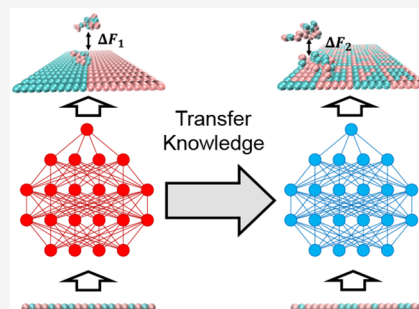Read Online

ACCESS | Metrics & More | Article Recommendations | Supporting Information

ABSTRACT: Machine learning (ML) accelerates the exploration of material properties and their links to the structure of the underlying molecules. In previous work [Shi et al. *ACS Applied Materials & Interfaces* **2022**, *14*, 37161−37169.], ML models were applied to predict the adhesive free energy of polymer−surface interactions with high accuracy from the knowledge of the sequence data, demonstrating successes in inverse-design of polymer sequence for known surface compositions. While the method was shown to be successful in designing polymers for a known surface, extensive data sets were needed for each specific surface in order to train the surrogate models. Ideally, one should be able to infer information about similar surfaces without having to regenerate a full complement of adhesion data for each new case. In the current work, we demonstrate a transfer learning (TL) technique using a deep neural network to improve the accuracy of ML models trained on small data sets by pretraining on a larger database from a related system and fine-tuning the weights of all layers with a small amount of additional data. The shared knowledge from the pretrained model facilitates the prediction accuracy significantly on small data sets. We also explore the limits of database size on accuracy and the optimal tuning of network architecture and parameters for our learning tasks. While applied to a relatively simple coarse-grained (CG) polymer model, the general lessons of this study apply to detailed modeling studies and the broader problems of inverse materials design.

INTRODUCTION

Numerous industrial applications and biological phenomena involve chemically specific polymer−surface interactions, from ink absorption on paper[1,2] and semiconductor fabrication and coating[3,4] to the design and synthesis of artificial tissues[5] and viruses recognizing receptors on a cell surface.[6−10] The use of highly tuned sequence-defined polymers is attractive in controlling phase behavior, stabilizing interfaces, and promoting adhesion. Sequence-dependent adsorption of polymers to patterned surfaces has been studied through traditional theoretical and computational approaches[11−18] and machine learning methods,[19] emphasizing the importance of polymer sequence in determining the adsorption energies.[17,19]

Machine learning (ML) and artificial intelligence (AI)[20−32] have achieved dramatic success in determining the behaviors and properties of polymer and biomacromolecule systems,[33−41] including predicting protein structure,[33−35] polymer structures (such as radius of gyration in solvent),[36,42] and thermodynamic properties (such as polymer glass transition temperature, T_g).[40,43,44] However, the wide-ranging chemical sequence, topological space, and mass distribution of the polymer are too extensive to explore.[42,45] For example, even for linear binary copolymers with 20 monomers, the number of possible sequences is approximately one million. The chemical space becomes exponentially large if more monomer types, variable degrees of polymerization, nonuniform topologies, and mass distributions enter the description. ML techniques can

help but often provide knowledge highly specific to the immediate problem and require significant new data sets to incorporate information outside the original scope. For example, our prior work (see ref 19) utilized ML models to predict the adhesive free energy of polymer−surface interactions with high accuracy and aid the inverse-design of polymer sequence for known surface compositions, but exploring adhesion of such a polymer to a substrate requires about 8000 data points to train an accurate ML model for each decorated surface. Often, ML models are inaccurate or overfit when trained on small data sets. At the same time, in both industrial applications and biological settings, the surface patterns vary substantially, both structurally and randomly. Collecting large data sets for every patterned surface from thousands or millions of new experiments or simulations is, therefore, prohibitively difficult and expensive. In realistic situations, it may only be feasible to collect tens to hundreds of new data points. Data-driven ML modeling is easier to implement but often necessitates large data sets that could be

difficult to obtain.[46−48] Therefore, our aim here is to determine the minimum amount of additional computation necessary to obtain an accurate binding model, building as much as possible on prior knowledge.

Transfer learning (TL) can be a valuable technique to overcome the dilemma of insufficient data.[46−48] In TL, an ML model initially pretrained for a given task on a large data set of the source domain is utilized as the base to train a model for a new task by fine-tuning a small data set of the target domain.[29,46−49] Typically, TL can improve the model's accuracy if the source and target domains are closely related.[29,46−48,50] TL has achieved considerable success in speech recognition,[51,52] image recognition,[53,54] and natural language processing.[55,56] In addition, TL has also been successfully utilized in materials informatics studies[57−59] such as structural prediction of gas adsorption in MOFs,[60] phonon properties in semiconductors,[61] and thermal conductivity[62] and electrochemical properties[29] of polymers. However, these studies typically do not explore the explicit inverse design problem involved in materials design: what molecular structures, subject to reasonable constraints, are best for a given application.

In this study, we demonstrate the ability of transfer learning to leverage the prediction performance of adhesive free energies between polymer chains with a defined sequence and patterned surfaces via fine-tuning a pretrained model. The source domain and learning task come from a large data set of polymer-surface interactions with one patterned surface.[19] The target domain and learning task come from a small data set of polymer-surface interactions with a different patterned surface.[19] We utilize a deep neural network architecture to perform transfer learning and characterize the improvements on three example cases. We also explore the limits of database size on accuracy and the optimal tuning of network architecture and parameters for our learning tasks.

■ METHODS

Data Set. The data sets used in this work are from our recent work, Shi et al. (ref 19). As shown in Figure 1 (a), every data point includes one sequence-defined polymer and its adhesive free energy ΔF with a patterned surface. The ΔF were generated by LAMMPS molecular dynamic simulations[63] coupled with adaptive biasing force (ABF) method[64] SSAGES.[27,65−69] The polymer chain and surface are both composed of two types of beads, denoted "red" beads and "green" by their visualization in Figure 1. The polymer is modeled as a flexible 20-bead linear chain. The surface is holonomically constrained, with a simple square lattice of beads having dimensions of $20\sigma \times 20\sigma$ for a total of 400 beads. Each data set contains 2×10^4 sequence-defined polymers and their adhesive free energies with one patterned surface. There are four different data sets, one for each pattern shown in Figure 1(b): PS1, which is composed of half red beads and half green beads in two stripes. $N_{red} = 200$ and $N_{green} = 200$; PS2, which is composed of 16 alternate small size squares ($5\sigma \times 5\sigma$) of red and green beads with the same overall composition as PS1; PS3, where each bead was randomly generated with a probability of 0.5 for each site to be red or green resulting in $N_{red} = 184$ and $N_{green} = 216$; and PS4, which is built upon PS2 but randomized within the interior of the $5\sigma \times 5\sigma$ squares resulting in a total of $N_{red} = 206$ and $N_{green} = 194$. PS3 and PS4 allow exploration of the role of randomizing effects on our adhesive models, with PS4 including randomness within an
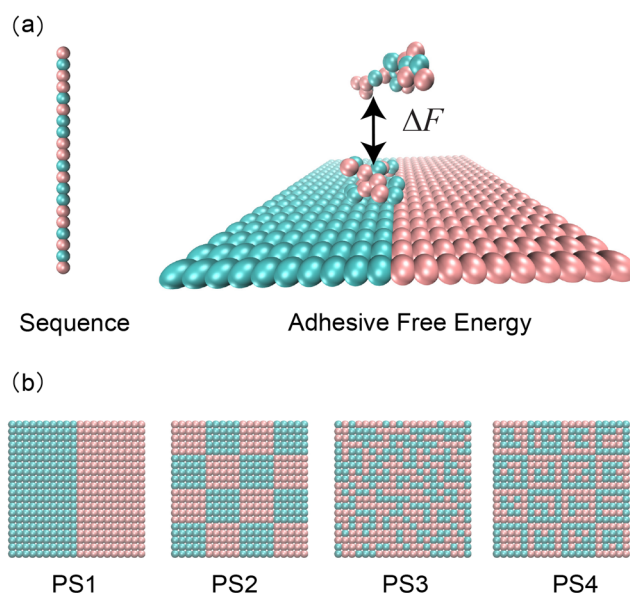

Figure 1. A schematic of the data sets about adhesive free energies of sequence defined polymers with patterned surfaces from the work of Shi et al.[19] (a) Every data point includes the following: A sequence-defined polymer and its adhesive free energy with a patterned surface. Each data set contains 2×10^4 sequence defined polymers and their adhesive free energies with one patterned surface. Therefore, for simplification, we use the name of the patterned surface to represent each data set. (b) There are four such data sets (Data-PS1, Data-PS2, Data-PS3, and Data-PS4) for four different patterned surfaces (PS1, PS2, PS3, and PS4).

overall structure rather than only randomness. For simplicity, we use the name of the patterned surface to represent each data set, called Data-PS1, Data-PS2, Data-PS3, and Data-PS4. Detailed distributions and analysis of the adhesive free energy data sets are available (see ref 19); reduced metrics corresponding to Gaussian fit parameters for each free energy distribution Data-PS1, Data-PS2, Data-PS3, and Data-PS4 are shown in Table 1. Additional details for generating the data sets are discussed in the previous work.[19] All data sets are available online at https://github.com/shijiale0609/ML_PSI.

Table 1. Gaussian Fitting Details[19] of Distributions of Adhesive Free Energies for Data-PS1, Data-PS2, Data-PS3, and Data-PS4

Data set	μ (k_BT)	σ (k_BT)
Data-PS1	15.66	2.89
Data-PS2	13.84	1.55
Data-PS3	8.96	0.77
Data-PS4	8.20	0.31

Transfer Learning Architecture. In this work, a deep neural network (DNN) architecture[29,60] with one input layer, three hidden layers, and one output layer was used to quantify the relationship between the polymer sequence information and polymer−surface adhesive free energy, ΔF. The input was one hot encoding of the polymer sequence. The output was the adhesive free energy. The DNN architecture is shown in Figure 2.

First, we trained a source DNN with the source data set. We used the Data-PS1, 2×10^4 data points of polymer sequences and their ΔF with PS1 as the source data, as the ML model
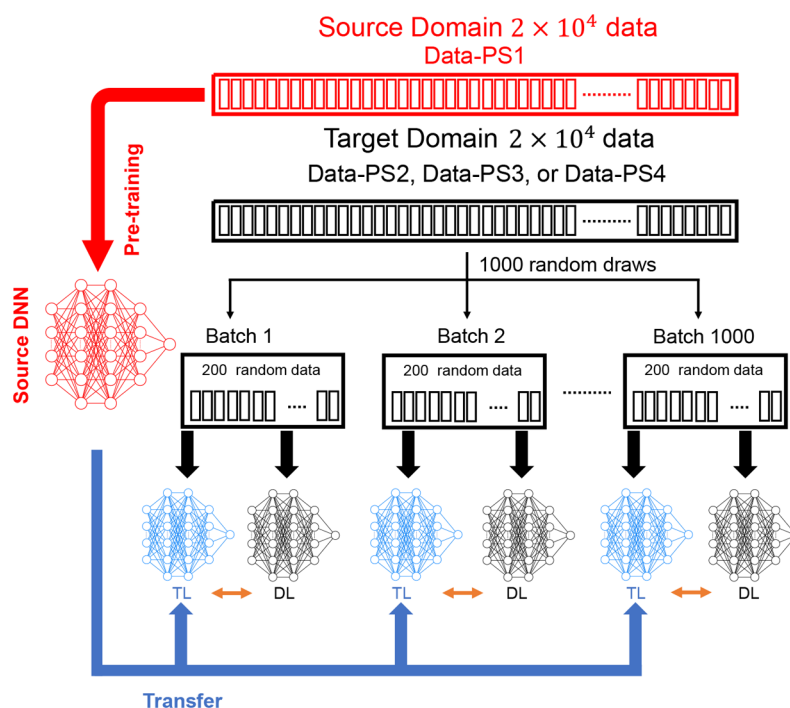
Figure 2. A schematic of the procedure for testing the performance of transfer learning from source domain (Data-PS1) to target domain (Data-PS2, Data-PS3, or Data-PS4). A total of 2×10^4 polymer sequences and the corresponding ΔF with PS1 are used as the source data. We train a fully connected deep neural network whose architecture is (20,64,64,32,1), using all the 2×10^4 source data points and save its weights. When training the DNN for target data, as a transfer learning framework, we fine-tune a subset of the weights in the pretrained source DNN using 200 data points (TL) and compare with the learning from a randomly initialized DNN using direct learning (DL).

applied to this data set achieved the highest accuracy among the four original data sets.[19] Then we randomly separated the 2×10^4 data points into 1.6×10^4 as the training set and 4×10^3 as the validation set. A 4:1 ratio is a commonly used ratio in machine learning.[20,21,70] The training set is the set of data that was used to train and make the model learn the hidden features/patterns in the data. In each epoch, the same training data was fed to the neural network architecture repeatedly, and the model continued to learn the features of the data. The validation set is a set of data that was used to validate our model performance during training. This validation process provided information that helped tune the model's hyperparameters and configurations. A test set is not required for this initial task as we are seeking a baseline trained on PS1 to extend to the other data sets. Without the need to leave data points for a test set, we were able to have more data points for training and validation. The hyperparameters of the DNN are optimized on the source task PS1 by promoting the accuracy and robustness of the DNN. Utilizing an n-tuple description for the hidden layers of a fully connected DNN, our network was represented by (20,64,64,32,1). The learning rate, which serves as the step size for updating the DNN parameters, was set to 0.00002 to make the learning process stable. LeakyReLU[71] with a negative slope of 0.1 was used as the activation function, and the Adam algorithm[72] was used to optimize weights. The number of learning epochs was set to 10^4, and the training process can be early ended by a converging check function applied on the validation data to terminate the training process, if appropriate. We trained a source DNN using the training set of source domain and selected the epoch with the highest accuracy on the validation set as the base DNN for the subsequent TL task, which were

referred to as the pretrained source DNN (depicted as the red DNN in Figure 2). An open-source machine learning framework, Pytorch,[73] was used to implement the DNN. All the parameters are stored on Github as described in the Data Availability Statement.

Next, we turned to the target data set and applied the DNN with the same hyperparameters. The small target data set was composed of 200 data points which were randomly drawn from existing data on the new domain (Data-PS2, Data-PS3, or Data-PS4). The data set was then divided into training, validation, and test sets in the ratio of 72:18:10, to be consistent with previous transfer learning studies.[60] 144 training data points were used for training the model, and 36 validation data points were used to determine when the training should be stopped and to avoid overfitting. The validation data set was used to select the training epoch. Since the validation data set was involved in the training process, the model's performance is toward it. Therefore, we additionally tested our model on the untouched test data set to provide unbiased final model performance metrics. Our use of this protocol enabled us to address the core question: "How well does the model perform on the small data set of Data-PS2, Data-PS3, or Data-PS4 without bias?". To illustrate the power of transfer learning, with the same 200 data points and the same separation for training, validation, and test sets, we performed direct learning (DL) (black DNN in Figure 2) and transfer learning (TL) (blue DNN in Figure 2). For direct learning, we trained the DNN model from randomly initialized weights. For transfer learning (blue DNN in Figure 2), we alternatively fine-tuned the weights of all layers in the pretrained DNN from the source task. There are three reasons that we choose to fine-tune the weights of all layers. First, we

sought to build an end-to-end model which is more friendly to other users who are not familiar with deep learning. In an end-to-end model, users only need to focus on the input and output and do not need to worry about how to modify the inside architecture of the model. We want to show that starting from a pretrained DNN without fixing the weights can get improvements. Second, we tested other fine-tuning formats, like fixing the weights of the first n layers and fine-tuning the weights of the remaining m layers.[60] We found that those formats do not provide competitive improvements and sometimes behaved worse than when fine-tuning all layers. Third, when the size of the training set increases, fixing the weights of some layers might lead to underfitting. Fine-tuning the weights of all layers is more robust to the size of the training data.

The comparison between performances of DL and TL scenarios was evaluated by comparing their respective coefficients of determination (R^2 values) on the same test sets (20 data points).

$$R^2 = 1 - \frac{\sum (\Delta F_i - \Delta \widehat{F_i})^2}{\sum (\Delta F_i - \Delta \overline{F})^2} \tag{1}$$

The maximum performance score of $R^2 = 1.0$ occurs when every prediction is correct ($\Delta F_i \equiv \Delta \widehat{F_i}$). Note that R^2 can be negative because the model can be arbitrarily poor. The choice of R^2 as an evaluation metric was reasonable, as R^2 can provide a natural baseline for judging the performance of models.[60] For each small data set, we obtained two coefficients: R^2_{DL}, which shows the performance of DL, and R^2_{TL}, which characterizes the performance of TL. The small data set resulted in highly variable accuracies in models due to the random data drawing. Therefore, we did not limit testing to a single small target data case and randomly drew sample data from the target space 1000 times for both DL and TL scenarios, subsequently obtaining 1000 pairs of R^2_{DL} and R^2_{TL}. This enabled us to gain a statistically robust understanding of the behavior of TL, mitigating the effects of outlier data sets on training.

■ RESULTS AND DISCUSSION

A summary of performance (captured via R^2 values) of direct (DL) and transfer (TL) learning for 1000 target data sets at sizes of 200, 4000, and 20000 from Data-PS2, Data-PS3, and Data-PS4 is given in Table 2. We step through specific cases below. We note that there appears to be a saturation of the

Table 2. R^2 Characteristics from DL and TL for 1000 Target Data Sets for Data Sets Data-PS2, Data-PS3, and Data-PS4 at Size 200, 4000, and 20000[a]

Data set	Data set Size	R^2_{DL}	R^2_{TL}
Data-PS2	200	-0.0089 ± 0.1956	0.8303 ± 0.0747
Data-PS2	4000	0.9256 ± 0.0085	0.9541 ± 0.0044
Data-PS2	20000	0.9587 ± 0.0023	0.9646 ± 0.0014
Data-PS3	200	0.6338 ± 0.2079	0.7998 ± 0.1173
Data-PS3	4000	0.8794 ± 0.0123	0.8973 ± 0.0101
Data-PS3	20000	0.8994 ± 0.0043	0.9058 ± 0.0041
Data-PS4	200	0.4502 ± 0.1849	0.6578 ± 0.1341
Data-PS4	4000	0.8251 ± 0.0145	0.8319 ± 0.0132
Data-PS4	20000	0.8528 ± 0.0053	0.8548 ± 0.0051

[a]Transfer learning proceeds using a neural network trained on Data-PS1 applied to data on the target surface.

improvement between transfer and direct learning strategies, though direct learning never statistically outperforms transfer learning on these data sets. Overall, we can conclude that transfer learning is a sound strategy in the studied cases, in particular if transferring to a small data set. We discuss the performance of transfer learning from PS1 to other data sets in more detail in what follows.

Knowledge Transfer from Data-PS1 to Data-PS2. We first investigate the application of TL in transferring knowledge from Data-PS1 over to Data-PS2, which links data acquired from one regular patterned surface to another regular patterned surface. PS1 and PS2 have the same overall composition (red:green = 200:200) and similar patterning when accounting for periodic boundaries, though PS2 uses smaller squares of uniform chemistry rather than two large stripes. The results of 1000 trials are plotted as pairs R^2_{DL} and R^2_{TL} for comparison in Figure 3. As shown in Figure 3(a), 624
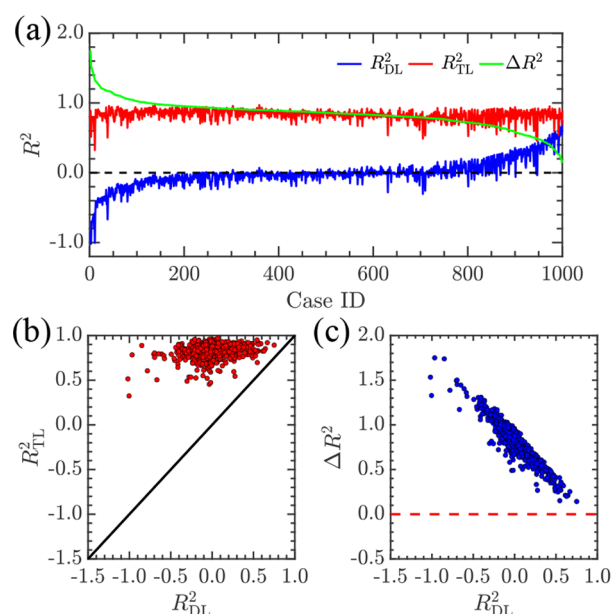

Figure 3. Transfer learning applied adhesive free energies of sequence defined polymers using a DNN fit to Data-PS1 adapted to Data-PS2 using a small data set. (a) 1000 pairs of R^2_{DL} for direct learning (blue line) and R^2_{TL} for transfer learning (red line). The improvement from transfer learning ($\Delta R^2 = R^2_{TL} - R^2_{DL}$) is represented by the green line. The Case ID numbers on the x axis are sorted by the value ΔR^2 in descending order. (b) R^2_{TL} plotted against R^2_{DL}. (c) Improvement ΔR^2 as a function of R^2_{DL}.

DL R^2 values are negative, implying poor model performance for those cases. All R^2 values for TL cases are positively correlated, and many are close to one, meaning that the models' performances are excellent in those cases. Collectively, the average R^2 on the 1000 test sets through DL is -0.0089 ± 0.1956, while the same metric for TL is 0.8303 ± 0.0747. Therefore, TL both improved the mean value of R^2 and decreased its standard deviation (SD). The diminishing SD shows that TL is less sensitive to the random selection of the small data set than the DL, which can be ascribed to the weights of the pretrained DNN being close to the optimized weights of the target DNN. From the dashed line depicting ΔR^2 in Figure 3(c), where all the ΔR^2 are greater than zero, and Figure 3(b), where all points (R^2_{TL} vs R^2_{DL}) are above the line $y = x$, it can be inferred that in all the 1000 target cases, TL

improved the accuracy of the model prediction. In Figure 3(c), a strong negative linear relationship between the improvement in ΔR^2 and the model accuracy from DL demonstrated that TL contributed improved knowledge in situations where DL yielded low accuracy. Since TL transfers a pretrained network rather than initializing weights randomly, the additional small data set acts to refine the weights rather than generate them wholesale—hence, even when DL yields low accuracy, the performance of TL remains stable, as is strongly evident in the improvement of ΔR^2 for these surfaces. At the same time, when DL already achieved very high model accuracy on the target tasks, the transfer knowledge from the source task offered only a slight improvement. For these cases, the randomly initialized weights of the DNN for DL happened to be close to the optimized weights.

Knowledge Transfer from Data-PS1 to Data-PS3. Next, we investigated the application of TL from Data-PS1 to Data-PS3. While PS1 is very regular, PS3 is a fully randomized surface with composition (red:green = 184:216), generated using a random probability $P(\text{red}) = P(\text{green}) = 0.5$ for the beads in the square lattice. The average R^2 on 1000 test sets modeled using DL was 0.6338 ± 0.2079, while the same metric from TL was improved to 0.7998 ± 0.1173. We note that DL's performance for Data-PS3 was better than that for Data-PS2, attributable to the reason that the standard variation ($\sigma = 0.77$ k_BT) of the whole 2×10^4 Data-PS3's ΔF is smaller than that of Data-PS2 ($\sigma = 1.55$ k_BT).[19] We conclude that the improvement from TL is less robust on this data set than Data-PS2, likely because of the tighter distribution for adhesive energies (see ref 19 for context). Still there remains a marked improvement. Another significant reason for the differences in this case is that the source and data sets are the more dissimilar adhesion properties related to the randomization of the surface pattern. Still, DL has 15 cases where R^2_{DL} is not greater than zero, while TL only has 2 cases where R^2_{TL} is not greater than zero. The green line in Figure 4(a) and data in Figure 4(b) show that in most examined cases (910 out of 1000), TL gives positive improvement. In Figure 4(c), there is a generally negative linear relationship between ΔR^2 and model accuracy from DL, though the linear relationship is not as strong as the prior data set in Figure 3(c). Thus, we infer that when the target tasks have very high model accuracy from DL already, the transfer of knowledge from the source task does not always help further improve the model accuracy.

Knowledge Transfer from Data-PS1 to Data-PS4. Finally, we test the application of TL from Data-PS1 to Data-PS4; the surface PS4 is a randomized version of PS2 whose composition (red:green = 206:194) differs slightly from the 1:1 composition of PS2.[19] The average R^2 on 1000 test sets through DL is 0.4502 ± 0.1849, while the same metric from TL is improved to 0.6578 ± 0.1341. DL's performance for Data-PS4 was better than that for Data-PS2, attributable to the relative tightness of the free energy distribution of Data-PS4 ($\sigma = 0.31$ k_BT) compared to Data-PS2 ($\sigma = 1.55$ k_BT).[19] The improvement is not as evident as in the first case (Data-PS1 to Data-PS2) but is overall much better than the performance using TL on the completely randomized PS3 surface. The green line in Figure 5(a) and Figure 5(b) shows most cases (939 out of 1000) are positively impacted by TL. In Figure 5(c), the negatively correlated relationship between the improvement in ΔR^2 and the model accuracy from DL appeared weaker than the previous two cases in Figure 3(c) and Figure 4(c), though we note that a single testing point
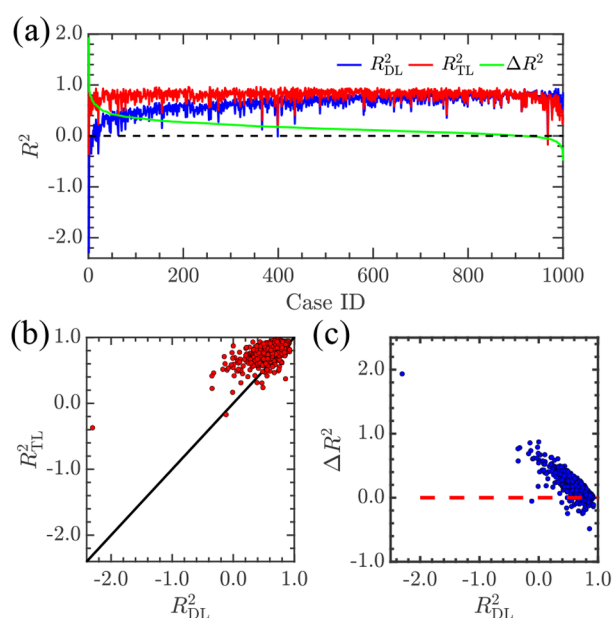

Figure 4. Transfer learning applied adhesive free energies of sequence defined polymers using a DNN fit to Data-PS1 adapted to Data-PS3 using a small data set. (a) R^2 values for direct learning (R^2_{DL}, blue line), transfer learning (R^2_{TL}, red line), and improvement from transfer learning ($\Delta R^2 = R^2_{TL} - R^2_{DL}$, green line) of the 1000 target cases. Case ID numbers on the x axis are sorted by the value ΔR^2 in descending order. (b) R^2_{TL} plotted against R^2_{DL}. (c) Improvement ΔR^2 as a function of R^2_{DL}.
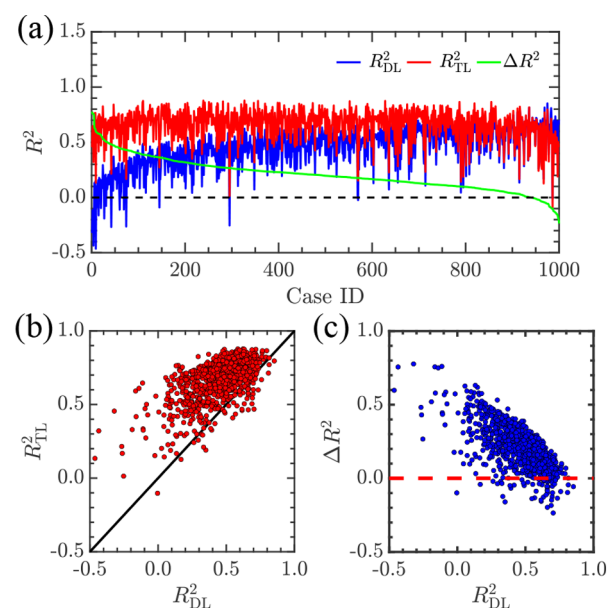

Figure 5. Transfer learning applied adhesive free energies of sequence defined polymers using a DNN fit to Data-PS1 adapted to Data-PS4 using a small data set. (a) R^2 values for direct learning (R^2_{DL}, blue line), transfer learning (R^2_{TL}, red line), and improvement from transfer learning ($\Delta R^2 = R^2_{TL} - R^2_{DL}$, green line) of the 1000 target cases. Case ID numbers on the x axis are sorted by the value ΔR^2 in descending order. (b) R^2_{TL} plotted against R^2_{DL}. (c) Improvement ΔR^2 as a function of R^2_{DL}.

with good TL and poor DL performance skews the plots visually.

Feature Importance Analysis. The structure of the one-hot encoding of our sequence-defined polymers permitted the interrogation of the feature importance of various sites on the polymer backbone. We utilize the entire data set of Data-PS1, Data-PS2, Data-PS3, and Data-PS4. The details of the training process were identical to those stated in the Methods section. Permutation feature importance, which is defined to be the decrease in predictive accuracy (ΔR^2) when a single feature value is randomly shuffled, was used to evaluate descriptor importance.[60,74] The feature importance for the feature i is computed by

$$\mathrm{FI}_i = \Delta R^2 = R^2 - R_i^2$$

where R^2 is the predictive accuracy without randomly shuffling, and R_i^2 is the predictive accuracy after randomly shuffling the ith dimensional feature. We used the permutation feature importance implementation in the Python package ELI5[75] to perform this analysis. Since our input is the one-hot encoding of the polymer sequence, a 20-dimensional vector, we alternatively shuffled the feature value of each dimension and calculated the descriptor importance for every input variable.

The results of the feature importance analysis are shown in Figure 6. Even though the absolute value of feature importance
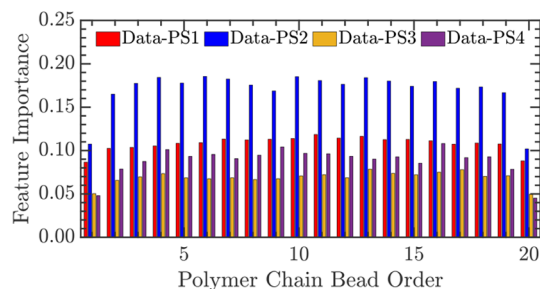

Figure 6. Permutation feature importance of a 20-dimensional input vector representing beads in the sequence-defined polymer for four data sets: Data-PS1 (red), Data-PS2 (blue), Data-PS3 (orange), and Data-PS4 (purple). Essentially, only the end points are significantly different with the 18 interior beads having rough similar importance to one another.

is different for each patterned surface, some common features exist for all four patterned surfaces. The head (first) and the tail (twentieth) beads had relatively lower values of feature importance, and the other 18 beads' feature importance were almost the same within the individual surface data set. Statt et al.[37] also found that the ends of an intrinsically disordered protein (IDP) have a distinct effect on the phase behavior (critical temperature) compared with mutations in the middle of the chain, though the ends are seen there to have a more pronounced effect on the proteins' phase behavior. The common features we found among Data-PS1, Data-PS2, Data-PS3, and Data-PS4 represent the shareable knowledge from TL and can explain the successful application of TL in these cases. Pretrained models were able to obtain these features before fine-tuning with the small data sets.

Size Effects for TL Improvements. From the above investigations, we illustrated that transfer learning can improve the accuracy of the DNN models trained on a small target data set (200 data points). It is of interest to see how this scales with the amount of the available data; thus, we also explored the effects of the size of the target data set on the

improvements from TL, increasing the "small" data set to values between 100 and 4000 points. As previously, other training settings were kept the same as for the $N = 200$ data sets.

Figure 7(a) illustrates the prediction performance of DL and TL for Data-PS2 as a function of the size of the target data set,
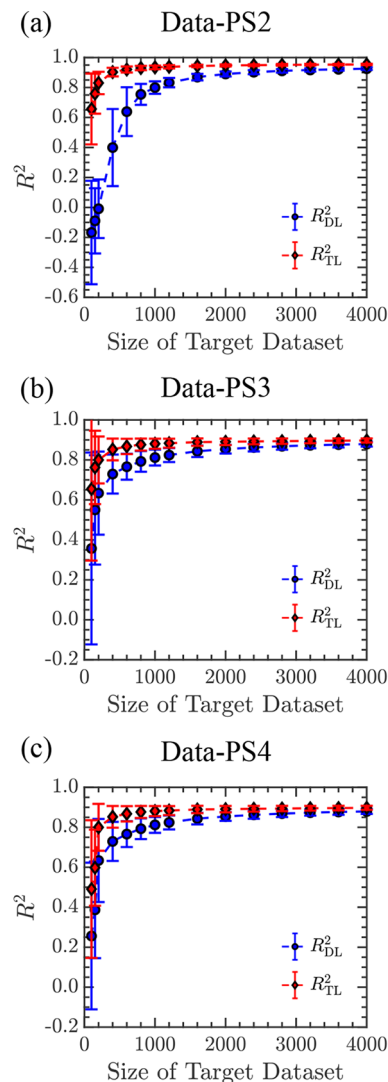

Figure 7. (a) R_{DL}^2 (blue) and R_{TL}^2 (red) as a function of the size of the target data set. Data-PS2 was used for this comparison. The error bars reflect the SD of the R^2 score from 1000 random draws. As the size of the target data set increases, R_{DL}^2 and R_{TL}^2 both increase, and their SD values decrease. R_{DL}^2 increases more dramatically relative to R_{TL}^2. Though the improvement ΔR^2 decreases with the increasing size of the data set, R_{TL}^2 is always larger than R_{DL}^2. Similar improvements of DL and TL were seen in Data-PS3 (b) and Data-PS4 (c), though the behavior of TL saturated at lower accuracy in each relative to Data-PS2.

as quantified by the R^2 score. The mean value increases, and the SD decreases for the R^2 score in both DL and TL as the size of the target data set increases. These values are seen to change more rapidly in DL than TL, which is perhaps to be expected, as more data should drastically improve the behavior of DL given the initial data sets are so small; the pretrained TL model is able to approach an optimal fit more easily. We note R_{TL}^2 quickly converged with increasing size. After the target

data size exceeded 800, accuracy (measured by R_{TL}^2) saturated. Thus, the improvement between TL and DL decreased as the target data set size increased. Still, the performance of the TL models was always better on average than DL models. This indicates that the transfer of knowledge from the source task can offer significant improvement when scant target data is available. The DNN model can learn sufficiently directly from the feed data in large target data sets, and TL does not offer significant improvement. In our data sets, 2×10^4 independent points are available, and TL's efficacy saturates relative to DL when approximately 20% of the target data set is used. This suggests a threshold below which TL should always be used. Similar improvements of DL and TL were seen in Data-PS3 and Data-PS4 [see Figure 7(b,c)], though the behavior of TL saturated at lower accuracy in each relative to Data-PS2. Nonetheless, improvements on the order of one standard deviation in R^2 were seen up to the same 20% threshold applied to the target data size.

To assess the robustness of our transfer learning results to the choice of training patterns for the pretrained model, we selected Data-PS4 as the source data, which has a lower relative accuracy. We trained a pretrained model on Data-PS4 and then applied it to Data-PS1, Data-PS2, and Data-PS3. As shown in Figure S5 in the SI, we found that transfer learning with a pretrained model from Data-PS4 also resulted in improved predictions from the transfer learning model. Detailed results and discussions are included in the SI.

CONCLUSION

In summary, a comprehensive TL study of the polymer–surface interaction between polymers with defined sequences and different surfaces was conducted through pretraining a DNN with a large data set from the source domain (Data-PS1) and fine-tuning the pretrained DNN with the small data set from the target domain (Data-PS2, Data-PS3, or Data-PS4). Knowledge was transferable among the polymer interactions with different patterned surfaces. TL significantly upgraded the performance of the model trained on the small data set. In addition, our results showed that TL's model is more stable than the DL and less likely to be affected by the random selection of data. The study of permutation feature importance revealed that the four patterned surfaces have some similar features, representing part of the reasons the transferable knowledge can work. We also demonstrate that the increase in target data size can diminish the improvement from TL, ascribable to the fact that DL learns more knowledge from the feed data directly when the size of the target data increases. However, the TL model with the full-fine-tuning architecture always performs better than the DL model, even though the improvement diminishes at large sizes of data sets.

Our work highlights the importance of transfer learning in elevating the performance of an ML model regarding the polymer–surface interaction under insufficient data dilemmas. Usually, tens or hundreds of data points are insufficient to train an accurate ML model. With transfer learning tools, the shareable knowledge from a pretrained model can help the ML model trained for polymer–surface interaction with a new surface to obtain higher performance. Our test cases are all simulation data sets, and the knowledge is shared among different surfaces in simulation. But the benefit of knowledge sharing is not only limited to simulation data sets. For example, Briceno-Mena et al.[29] have utilized transfer learning techniques to increase the performance of an ML model trained with insufficient experimental data by transferring knowledge from an ML model trained with a large simulation data set.[29] Similarly, for the prediction and optimization of adhesive energies, transfer learning can be used to maximize our knowledge within a new chemical domain from a smaller amount of simulations or experiments, perhaps allowing purely computational and coarse-grained models to cheaply explore compositional space and predictive models to be refined by directed experimentation. As transfer learning, especially few-shot learning, has achieved dramatic successes in computer vision and language models by building a series of sizable pretrained ML models, such as YOLO,[76] BERT,[56] and GPT-3,[55] we anticipate knowledge about network structure and problem complexity may be used to guide these algorithms and their applications to materials problems. With more experimental and simulation data about polymer–surface interactions being produced and collected in the future, it is expected to obtain a sizable pretrained ML model for polymer surface interaction.

ASSOCIATED CONTENT

Data Availability Statement

The adhesive free energy data sets that used in this article are available online at https://github.com/shijiale0609/ML_PSI. Example scripts and information necessary to run the examples contained in this article are posted at https://github.com/shijiale0609/TL_PSI.

Ⓢ Supporting Information

The Supporting Information is available free of charge at https://pubs.acs.org/doi/10.1021/acs.jctc.2c01314.

Performance of different transfer learning formats and results with an alternative source model (PDF)

AUTHOR INFORMATION

Corresponding Author

Jonathan K. Whitmer − Department of Chemical and Biomolecular Engineering, University of Notre Dame, Notre Dame, Indiana 46556, United States; Department of Chemistry and Biochemistry, University of Notre Dame, Notre Dame, Indiana 46556, United States; ⓞ orcid.org/0000-0003-0370-0873; Email: jwhitme1@nd.edu

Authors

Jiale Shi − Department of Chemical and Biomolecular Engineering, University of Notre Dame, Notre Dame, Indiana 46556, United States

Fahed Albreiki − Department of Chemical and Biomolecular Engineering, University of California, Los Angeles, Los Angeles, California 90095, United States

Yamil J. Colón − Department of Chemical and Biomolecular Engineering, University of Notre Dame, Notre Dame, Indiana 46556, United States; ⓞ orcid.org/0000-0001-5316-9692

Samanvaya Srivastava − Department of Chemical and Biomolecular Engineering, University of California, Los Angeles, Los Angeles, California 90095, United States; California NanoSystems Institute, Center for Biological Physics, Institute for Carbon Management, and Center for Biological Physics, University of California, Los Angeles, Los Angeles, California 90095, United States; ⓞ orcid.org/0000-0002-3519-7224

Complete contact information is available at:
https://pubs.acs.org/10.1021/acs.jctc.2c01314

Author Contributions

JS, FA, SS, and JKW conceived the study. JS and JKW designed the research plan. YJC provided suggestions and guidance on the transfer learning framework. JS conducted transfer learning model training and analyzed the results. JS, FA, YJC, SS, and JKW interpreted the results and wrote the paper.

Notes

The authors declare no competing financial interest.

■ REFERENCES

(1) Chakraborty, A. K.; Golumbfskie, A. Polymer AdsorptionDriven Self-Assembly of Nanostructures. *J. Annu. Rev. Phys. Chem.* **2001**, *52*, 537−573.

(2) Li, L.; Lin, Q.; Tang, M.; Duncan, A. J. E.; Ke, C. Advanced Polymer Designs for Direct-Ink-Write 3D Printing. *Chem.—Eur. J.* **2019**, *25*, 10768−10781.

(3) Ouk Kim, S.; Solak, H. H.; Stoykovich, M. P.; Ferrier, N. J.; De Pablo, J. J.; Nealey, P. F. Epitaxial self-assembly of block copolymers on lithographically defined nanopatterned substrates. *Nature* **2003**, *424*, 411−414.

(4) Ikawa, M.; Yamada, T.; Matsui, H.; Minemawari, H.; Tsutsumi, J.; Horii, Y.; Chikamatsu, M.; Azumi, R.; Kumai, R.; Hasegawa, T. Simple push coating of polymer thin-film transistors. *Nat. Commun.* **2012**, *3*, 1176.

(5) Annabi, N.; Tamayol, A.; Shin, S. R.; Ghaemmaghami, A. M.; Peppas, N. A.; Khademhosseini, A. Surgical materials: Current challenges and nano-enabled solutions. *Nano Today* **2014**, *9*, 574−589.

(6) Xiu, S.; Dick, A.; Ju, H.; Mirzaie, S.; Abdi, F.; Cocklin, S.; Zhan, P.; Liu, X. Inhibitors of SARS-CoV-2 Entry: Current and Future Opportunities. *J. Med. Chem.* **2020**, *63*, 12256−12274.

(7) Wong, J. P.; Damania, B. SARS-CoV-2 dependence on host pathways. *Science* **2021**, *371*, 884−885.

(8) Callaway, E. Making sense of coronavirus mutations. *Nature* **2020**, *585*, 174−177.

(9) Plante, J. A.; Liu, Y.; Liu, J.; Xia, H.; Johnson, B. A.; Lokugamage, K. G.; Zhang, X.; Muruato, A. E.; Zou, J.; Fontes-Garfias, C. R.; Mirchandani, D.; Scharton, D.; Bilello, J. P.; Ku, Z.; An, Z.; Kalveram, B.; Freiberg, A. N.; Menachery, V. D.; Xie, X.; Plante, K. S.; Weaver, S. C.; Shi, P.-Y. Spike mutation D614G alters SARS-CoV-2 fitness. *Nature* **2021**, *592*, 116−121.

(10) Hie, B.; Zhong, E. D.; Berger, B.; Bryson, B. Learning the language of viral evolution and escape. *Science* **2021**, *371*, 284−288.

(11) Ozboyaci, M.; Kokh, D. B.; Corni, S.; Wade, R. C. Modeling and simulation of protein- surface interactions: achievements and challenges. *Q. Rev. Biophys.* **2016**, *49*, e4.

(12) Chakraborty, A. K.; Bratko, D. A simple theory and Monte Carlo simulations for recognition between random heteropolymers and disordered surfaces. *J. Chem. Phys.* **1998**, *108*, 1676−1682.

(13) Muthukumar, M. Pattern recognition by polyelectrolytes. *J. Chem. Phys.* **1995**, *103*, 4723−4731.

(14) Chakraborty, A. K. Disordered heteropolymers: models for biomimetic polymers and polymers with frustrating quenched disorder. *Phys. Rep.* **2001**, *342*, 1−61.

(15) Muthukumar, M. Pattern recognition in self-assembly. *Curr. Opin. Colloid Interface Sci.* **1998**, *3*, 48−54.

(16) Chauhan, G.; Simpson, M. L.; Abel, S. M. Crowding-induced interactions of ring polymers. *Soft Matter* **2021**, *17*, 16−23.

(17) Kriksin, Y. A.; Khalatur, P. G.; Khokhlov, A. R. Adsorption of multiblock copolymers onto a chemically heterogeneous surface: A model of pattern recognition. *J. Chem. Phys.* **2005**, *122*, 114703.

(18) Tereshkin, E. V.; Tereshkina, K. B.; Krupyanskii, Y. F. Predicting Binding Free Energies for DPS Protein-DNA Complexes and Crystals Using Molecular Dynamics. *Supercomputing Frontiers and Innovations* **2022**, *9*, 33−45.

(19) Shi, J.; Quevillon, M. J.; Amorim Valença, P. H.; Whitmer, J. K. Predicting Adhesive Free Energies of Polymer-Surface Interactions with. *Machine Learning. ACS Applied Materials & Interfaces* **2022**, *14*, 37161−37169.

(20) Bishop, C. M. *Pattern Recognition and Machine Learning (Information Science and Statistics)*; Springer-Verlag: Berlin, Heidelberg, 2006.

(21) Murphy, K. P. *Machine learning: a probabilistic perspective*; MIT Press: 2012.

(22) Murphy, K. P. *Probabilistic Machine Learning: An introduction*; MIT Press: 2022.

(23) Murphy, K. P. *Probabilistic Machine Learning: Advanced Topics*; MIT Press: 2023.

(24) Artrith, N.; Butler, K. T.; Coudert, F.-X.; Han, S.; Isayev, O.; Jain, A.; Walsh, A. Best practices in machine learning for chemistry. *Nat. Chem.* **2021**, *13*, 505−508.

(25) de Pablo, J. J.; Jackson, N. E.; Webb, M. A.; Chen, L.-Q.; Moore, J. E.; Morgan, D.; Jacobs, R.; Pollock, T.; Schlom, D. G.; Toberer, E. S.; Analytis, J.; Dabo, I.; De- Longchamp, D. M.; Fiete, G. A.; Grason, G. M.; Hautier, G.; Mo, Y.; Rajan, K.; Reed, E. J.; Rodriguez, E.; Stevanovic, V.; Suntivich, J.; Thornton, K.; Zhao, J.-C. New frontiers for the materials genome initiative. *npj Computational Materials* **2019**, *5*, 41.

(26) Gormley, A. J.; Webb, M. A. Machine learning in combinatorial polymer chemistry. *Nature Reviews Materials* **2021**, *6*, 642.

(27) Huang, K.; Fu, T.; Glass, L. M.; Zitnik, M.; Xiao, C.; Sun, J. DeepPurpose: a deep learning library for drug-target interaction prediction. *Bioinformatics* **2021**, *36*, 5545−5547.

(28) Liang, Z.; Li, Z.; Zhou, S.; Sun, Y.; Yuan, J.; Zhang, C. Machine-learning exploration of polymer compatibility. *Cell Reports Physical Science* **2022**, *3*, 100931.

(29) Briceno-Mena, L. A.; Romagnoli, J. A.; Arges, C. G. PemNet: ATransfer Learning-Based Modeling Approach of High-Temperature Polymer Electrolyte Membrane Electrochemical Systems. *Ind. Eng. Chem. Res.* **2022**, *61*, 3350−3357.

(30) Sattari, K.; Xie, Y.; Lin, J. Data-driven algorithms for inverse design of polymers. *Soft Matter* **2021**, *17*, 7607−7622.

(31) Sevgen, E.; Guo, A. Z.; Sidky, H.; Whitmer, J. K.; de Pablo, J. J. Combined Force-Frequency Sampling for Simulation of Systems Having Rugged Free Energy Landscapes. *J. Chem. Theory Comput.* **2020**, *16*, 1448−1455.

(32) Sidky, H.; Whitmer, J. K. Learning free energy landscapes using artificial neural networks. *J. Chem. Phys.* **2018**, *148*, 104111.

(33) Jumper, J.; Evans, R.; Pritzel, A.; Green, T.; Figurnov, M.; Ronneberger, O.; Tunyasuvunakool, K.; Bates, R.; Žídek, A.; Potapenko, A.; Bridgland, A.; Meyer, C.; Kohl, S. A. A.; Ballard, A. J.; Cowie, A.; Romera-Paredes, B.; Nikolov, S.; Jain, R.; Adler, J.; Back, T.; Petersen, S.; Reiman, D.; Clancy, E.; Zielinski, M.; Steinegger, M.; Pacholska, M.; Berghammer, T.; Bodenstein, S.; Silver, D.; Vinyals, O.; Senior, A. W.; Kavukcuoglu, K.; Kohli, P.; Hassabis, D. Highly accurate protein structure prediction with AlphaFold. *Nature* **2021**, *596*, 583−589.

(34) Tunyasuvunakool, K.; Adler, J.; Wu, Z.; Green, T.; Zielinski, M.; Žídek, A.; Bridgland, A.; Cowie, A.; Meyer, C.; Laydon, A.; Velankar, S.; Kleywegt, G. J.; Bateman, A.; Evans, R.; Pritzel, A.;

Figurnov, M.; Ronneberger, O.; Bates, R.; Kohl, S. A. A.; Potapenko, A.; Ballard, A. J.; Romera-Paredes, B.; Nikolov, S.; Jain, R.; Clancy, E.; Reiman, D.; Petersen, S.; Senior, A. W.; Kavukcuoglu, K.; Birney, E.; Kohli, P.; Jumper, J.; Hassabis, D. Highly accurate protein structure prediction for the human proteome. *Nature* **2021**, *596*, 590−596.

(35) Baek, M.; DiMaio, F.; Anishchenko, I.; Dauparas, J.; Ovchinnikov, S.; Lee, G. R.; Wang, J.; Cong, Q.; Kinch, L. N.; Schaeffer, R. D.; Millán, C.; Park, H.; Adams, C.; Glassman, C. R.; DeGiovanni, A.; Pereira, J. H.; Rodrigues, A. V.; van Dijk, A. A.; Ebrecht, A. C.; Opperman, D. J.; Sagmeister, T.; Buhlheller, C.; Pavkov-Keller, T.; Rathinaswamy, M. K.; Dalwadi, U.; Yip, C. K.; Burke, J. E.; Garcia, K. C.; Grishin, N. V.; Adams, P. D.; Read, R. J.; Baker, D. Accurate prediction of protein structures and interactions using a three-track neural network. *Science* **2021**, *373*, 871−876.

(36) Webb, M. A.; Jackson, N. E.; Gil, P. S.; de Pablo, J. J. Targeted sequence design within the coarse-grained polymer genome. *Sci. Adv.* **2020**, *6*, eabc6216.

(37) Statt, A.; Casademunt, H.; Brangwynne, C. P.; Panagiotopoulos, A. Z. Model for disordered proteins with strongly sequence-dependent liquid phase behavior. *J. Chem. Phys.* **2020**, *152*, 075101.

(38) Statt, A.; Kleeblatt, D. C.; Reinhart, W. F. Unsupervised learning of sequence-specific aggregation behavior for a model copolymer. *Soft Matter* **2021**, *17*, 7697−7707.

(39) Meenakshisundaram, V.; Hung, J.-H.; Patra, T. K.; Simmons, D. S. Designing Sequence-Specific Copolymer Compatibilizers Using a Molecular-Dynamics-Simulation-Based Genetic Algorithm. *Macromolecules* **2017**, *50*, 1155−1166.

(40) Ma, R.; Huang, D.; Zhang, T.; Luo, T. Determining influential descriptors for polymer chain conformation based on empirical force-fields and molecular dynamics simulations. *Chem. Phys. Lett.* **2018**, *704*, 49−54.

(41) Arora, A.; Lin, T.-S.; Rebello, N. J.; Av-Ron, S. H. M.; Mochigase, H.; Olsen, B. D. Random Forest Predictor for Diblock Copolymer Phase Behavior. *ACS Macro Lett.* **2021**, *10*, 1339−1345.

(42) Patel, R. A.; Borca, C. H.; Webb, M. A. Featurization strategies for polymer sequence or composition design by machine learning. *Mol. Syst. Des. Eng.* **2022**, *7*, 661−676.

(43) Ma, R.; Zhang, H.; Luo, T. Exploring High Thermal Conductivity Amorphous Polymers Using Reinforcement Learning. *ACS Appl. Mater. Interfaces* **2022**, *14*, 15587−15598.

(44) Ma, R.; Liu, Z.; Zhang, Q.; Liu, Z.; Luo, T. Evaluating Polymer Representations via Quantifying Structure-Property Relationships. *J. Chem. Inf. Model.* **2019**, *59*, 3110−3119.

(45) Lin, T.-S.; Coley, C. W.; Mochigase, H.; Beech, H. K.; Wang, W.; Wang, Z.; Woods, E.; Craig, S. L.; Johnson, J. A.; Kalow, J. A.; Jensen, K. F.; Olsen, B. D. BigSMILES: A Structurally-Based Line Notation for Describing Macromolecules. *ACS Central Science* **2019**, *5*, 1523−1531.

(46) Wang, J.; Chen, Y. *Introduction to Transfer Learning: Algorithms and Practice*; Springer Nature: 2023.

(47) Yang, Q.; Zhang, Y.; Dai, W.; Pan, S. J. *Transfer Learning*; Cambridge University Press: 2020.

(48) Yosinski, J.; Clune, J.; Bengio, Y.; Lipson, H. How transferable are features in deep neural networks?. *Advances in Neural Information Processing Systems 27*; The MIT Press: 2014; pp 1−9.

(49) Zhuang, F.; Qi, Z.; Duan, K.; Xi, D.; Zhu, Y.; Zhu, H.; Xiong, H.; He, Q. A comprehensive survey on transfer learning. *Proceedings of the IEEE* **2021**, *109*, 43−76.

(50) Pan, S. J.; Yang, Q. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering* **2010**, *22*, 1345−1359.

(51) Wang, D.; Zheng, T. F. Transfer learning for speech and language processing. *2015 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*; 2015; pp 1225−1237.

(52) Kunze, J.; Kirsch, L.; Kurenkov, I.; Krug, A.; Johannsmeier, J.; Stober, S. Transfer Learning for Speech Recognition on a Budget.

2017, arXiv:1706.00290. *arXiv preprint.* https://arxiv.org/abs/1706.00290 (accessed 2023-04-07).

(53) Ng, H.-W.; Nguyen, V. D.; Vonikakis, V.; Winkler, S. Deep Learning for Emotion Recognition on Small Datasets Using Transfer Learning. *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*; Association for Computing Machinery: New York, NY, USA, 2015; pp 443−449,

(54) Yang, X.; Zhang, Y.; Lv, W.; Wang, D. Image recognition of wind turbine blade damage based on a deep learning model with transfer learning and an ensemble learning classifier. *Renewable Energy* **2021**, *163*, 386−397.

(55) Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J. D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; Agarwal, S.; Herbert-Voss, A.; Krueger, G.; Henighan, T.; Child, R.; Ramesh, A.; Ziegler, D.; Wu, J.; Winter, C.; Hesse, C.; Chen, M.; Sigler, E.; Litwin, M.; Gray, S.; Chess, B.; Clark, J.; Berner, C.; McCandlish, S.; Radford, A.; Sutskever, I.; Amodei, D. Language Models are Few-Shot Learners. *Advances in Neural Information Processing Systems*; 2020; pp 1877−1901.

(56) Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. 2018, arXiv:1810.04805. *arXiv preprint.* https://arxiv.org/abs/1810.04805 (accessed 2023-04-07).

(57) Tsubaki, M.; Mizoguchi, T. Quantum Deep Descriptor: Physically Informed Transfer Learning from Small Molecules to Polymers. *J. Chem. Theory Comput.* **2021**, *17*, 7814−7821.

(58) Sultan, M. M.; Wayment-Steele, H. K.; Pande, V. S. Transferable Neural Networks for Enhanced Sampling of Protein Dynamics. *J. Chem. Theory Comput.* **2018**, *14*, 1887−1894.

(59) Käser, S.; Boittier, E. D.; Upadhyay, M.; Meuwly, M. Transfer Learning to CCSD(T): Accurate Anharmonic Frequencies from Machine Learning Models. *J. Chem. Theory Comput.* **2021**, *17*, 3687−3699.

(60) Ma, R.; Colón, Y. J.; Luo, T. Transfer Learning Study of Gas Adsorption in Metal-Organic Frameworks. *ACS Appl. Mater. Interfaces* **2020**, *12*, 34041−34048.

(61) Liu, Z.; Jiang, M.; Luo, T. Leverage electron properties to predict phonon properties via transfer learning for semiconductors. *Science Advances* **2020**, *6*, eabd1356.

(62) Wu, S.; Kondo, Y.; Kakimoto, M.-a.; Yang, B.; Yamada, H.; Kuwajima, I.; Lambard, G.; Hongo, K.; Xu, Y.; Shiomi, J.; Schick, C.; Morikawa, J.; Yoshida, R. Machine-learning-assisted discovery of polymers with high thermal conductivity using a molecular design algorithm. *npj Computational Materials* **2019**, *5*, 66.

(63) Thompson, A. P.; Aktulga, H. M.; Berger, R.; Bolintineanu, D. S.; Brown, W. M.; Crozier, P. S.; in 't Veld, P. J.; Kohlmeyer, A.; Moore, S. G.; Nguyen, T. D.; Shan, R.; Stevens, M. J.; Tranchida, J.; Trott, C.; Plimpton, S. J. LAMMPS - a flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales. *Comput. Phys. Commun.* **2022**, *271*, 108171.

(64) Darve, E.; Rodríguez-Gómez, D.; Pohorille, A. Adaptive biasing force method for scalar and vector free energy calculations. *J. Chem. Phys.* **2008**, *128*, 144120.

(65) Sidky, H.; Colón, Y. J.; Helfferich, J.; Sikora, B. J.; Bezik, C.; Chu, W.; Giberti, F.; Guo, A. Z.; Jiang, X.; Lequieu, J.; Li, J.; M?ller, J.; Quevillon, M. J.; Rahimi, M.; Ramezani-Dakhel, H.; Rathee, V. S.; Reid, D. R.; Sevgen, E.; Thapar, V.; Webb, M. A.; Whitmer, J. K.; de Pablo, J. J. SSAGES: Software Suite for Advanced General Ensemble Simulations. *J. Chem. Phys.* **2018**, *148*, 044104.

(66) Shi, J.; Huang, S.; Gygi, F.; Whitmer, J. K. Free-Energy Landscape and Isomerization Rates of Au4 Clusters at Finite Temperatures. *J. Phys. Chem. A* **2022**, *126*, 3392−3400.

(67) Shi, J.; Sidky, H.; Whitmer, J. K. Automated determination of n-cyanobiphenyl and n-cyanobiphenyl binary mixtures elastic constants in the nematic phase from molecular simulation. *Mol. Syst. Des. Eng.* **2020**, *5*, 1131−1136.

(68) Leonhard, A. C.; Whitmer, J. K. Accurate Determination of Cavitand Binding Free Energies via Unrestrained Advanced Sampling. *J. Chem. Theory Comput.* **2019**, *15*, 5761−5768.

(69) Cortés-Morales, E. C.; Rathee, V. S.; Ghobadi, A.; Whitmer, J. K. A molecular view of plasticization of polyvinyl alcohol. *J. Chem. Phys.* **2021**, *155*, 174903.

(70) Zhang, A.; Lipton, Z. C.; Li, M.; Smola, A. J. Dive into Deep Learning. 2021, arXiv:2106.11342. *arXiv preprint.* https://arxiv.org/abs/2106.11342 (accessed 2023-04-07).

(71) Xu, B.; Wang, N.; Chen, T.; Li, M. Empirical evaluation of rectified activations in convolutional network. 2015, arXiv:1505.00853. *arXiv preprint.* https://arxiv.org/abs/1505.00853 (accessed 2023-04-07).

(72) Kingma, D. P.; Ba, J. Adam: A method for stochastic optimization. 2014, arXiv:1412.6980. *arXiv preprint.* https://arxiv.org/abs/1412.6980 (accessed 2023-04-07).

(73) Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; Desmaison, A.; Kopf, A.; Yang, E.; DeVito, Z.; Raison, M.; Tejani, A.; Chilamkurthy, S.; Steiner, B.; Fang, L.; Bai, J.; Chintala, S. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems 32*; Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché Buc, F., Fox, E., Garnett, R., Eds.; Curran Associates, Inc.: 2019; pp 8024−8035.

(74) Altmann, A.; Toloşi, L.; Sander, O.; Lengauer, T. Permutation importance: a corrected feature importance measure. *Bioinformatics* **2010**, *26*, 1340−1347.

(75) *TeamHG-Memex, ELI5.* 2020. https://github.com/TeamHG-Memex/eli5 (accessed 2023-04-07).

(76) Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. *Proceedings of the IEEE conference on computer vision and pattern recognition*; 2016; pp 779−788.