ACS OMEGA

http://pubs.acs.org/journal/acsodf

Article

Machine Learning with Enormous "Synthetic" Data Sets: Predicting Glass Transition Temperature of Polyimides Using Graph Convolutional Neural Networks

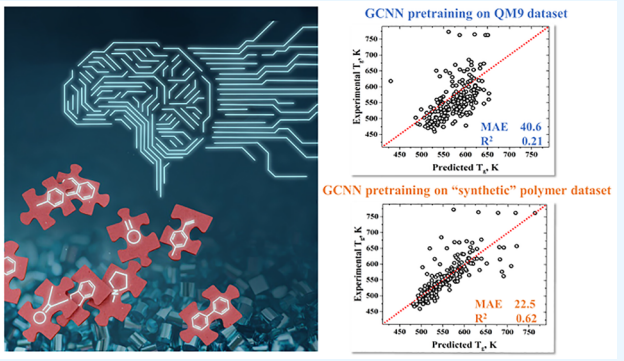
Igor V. Volgin, Pavel A. Batyr, Andrey V. Matseevich, Alexey Yu. Dobrovskiy, Maria V. Andreeva, Victor M. Nazarychev, Sergey V. Larin, Mikhail Ya. Goikhman, Yury V. Vizilter, Andrey A. Askadskii, and Sergey V. Lyulin*

Cite This: *ACS Omega* 2022, 7, 43678–43691

Read Online

ACCESS | Metrics & More | Article Recommendations | Supporting Information

ABSTRACT: In the present work, we address the problem of utilizing machine learning (ML) methods to predict the thermal properties of polymers by establishing "structure–property" relationships. Having focused on a particular class of heterocyclic polymers, namely polyimides (PIs), we developed a graph convolutional neural network (GCNN), being one of the most promising tools for working with big data, to predict the PI glass transition temperature T_g as an example of the fundamental property of polymers. To train the GCNN, we propose an original methodology based on using a "transfer learning" approach with an enormous "synthetic" data set for pretraining and a small experimental data set for its fine-tuning. The "synthetic" data set contains more than 6 million combinatorically generated repeating units of PIs and theoretical values of their T_g values calculated using the well-established Askadskii's quantitative structure–property relationship (QSPR) computational scheme. Additionally, an experimental data set for 214 PIs was also collected from the literature for training, fine-tuning, and validation of the GCNN. Both "synthetic" and experimental data sets are included into a PolyAskInG database (Polymer Askadskii's Intelligent Gateway). By using the PolyAskInG database, we developed GCNN which allows estimation of T_g of PI with a mean absolute error (MAE) of about 20 K, which is 1.5 times lower than in the case of Askadskii QSPR analysis (33 K). To prove the efficiency and usability of the proposed GCNN architecture and training methodology for predicting polymer properties, we also employed "transfer learning" to develop alternative GCNN pretrained on proxy-characteristics taken from the popular quantum-chemical QM9 database for small compounds and fine-tuned on an experimental T_g values data set from PolyAskInG database. The obtained results indicate that pretraining of GCNN on the "synthetic" polymer data set provides MAE which is almost twice as low as that in the case of using the QM9 data set in the pretraining stage (~41 K). Furthermore, we address the questions associated with the influence of the differences in the size of the experimental and "synthetic" data sets (so-called "reality gap" problem), as well as their chemical composition on the training quality. Our results state the overall priority of using polymer data sets for developing deep neural networks, and GCNN in particular, for efficient prediction of polymer properties. Moreover, our work opens up a challenge for the theoretically supported generation of large "synthetic" data sets of polymer properties for the training of the complex ML models. The proposed methodology is rather versatile and may be generalized for predicting other properties of different polymers and copolymers synthesized through the polycondensation reaction.



1. INTRODUCTION

The development of modern approaches to predict polymer properties is usually associated with the rapid development of computer-aided design. First of all, here we should mention atomistically detailed Molecular Dynamics simulations that accurately predict the properties of very complicated polymers, even before their synthesis.[1−3] However, the focus of current computational development of new materials is shifting to the use of less resource-consuming data-driven approaches based on machine learning (ML) methods.[4−6] Considering the fact that the total number of possible small organic molecules is estimated[7] at 10^{60}, and the number of currently known ones does not exceed[8] 10^8, it becomes obvious that a successful search

Received: July 22, 2022
Accepted: October 28, 2022
Published: November 17, 2022

for new chemical compounds, especially polymers, with the desired properties can only be achieved by a reasonable application of ML methods.

However, the solution of such an inverse (or backward) task, i.e., looking for the chemical structure of a polymer with required properties, should be based on the excellently operating ML models predicting the physical properties of any new polymer from its chemical structure (the so-called direct task or forward calculations). The standard approach to solve this direct task using ML methods is impossible without handling with "big data", which is a challenge in the case of polymers,[9] even for the temperature of the glass transition T_g.

When developing new polymers, T_g is one of the most important characteristics that should be predicted. T_g for thermoplastic polymers characterizes their heat resistance[10] and can be used to predict the limiting temperature value at which materials creep begins to play a significant role in determining their mechanical properties.[11] Moreover, T_g may serve as an intermediate ("proxy") characteristic for training ML models to solve the "inverse" task with the so-called "transfer learning" approach.[12] For this reason, T_g can be very handy to try new approaches in ML applications for polymer design.

Generally, the application of computational methods to predict T_g of polymers is a nontrivial issue that has a long-standing history. First of all, the classical quantitative structure−property relationship (QSPR) computational schemes proposed by van Krevelen,[13] Bicerano,[14] and Askadskii[15,16] should be mentioned among the first examples of applying ML philosophy to establish "structure−property" relationships for polymers including T_g prediction (for brief overview of the classical computational schemes for T_g prediction we refer the reader to the Section S1 in the Supporting Information file).

In recent years, classical QSPR schemes have been gradually replaced by various ML models serving as more versatile and optimizable tools for each specific case. Regarding polymer T_g prediction, plenty of such models were reported in the literature.[17−38] For example, a support vector regression (SVR) model was proposed by Varnek et al.[37] as a unified approach of T_g prediction of both cross-linked and linear polymers of any type. To construct the model, the authors used a data set of T_g values for 389 polymers collected from the literature (270 values for training and the remaining 119 for testing), which is still a quite small data set taking into account the diversity of polymer types. Performance tests showed a root-mean-square error (RMSE) of 35.9 K on the testing set. Another example is the study by Pilania et al.[36] who addressed the problem of predicting T_g of polyhydroxyalkanoate homopolymers and copolymers. They used a random forest ML model with 20 descriptors as a primary algorithm, which was shown to outperform the kernel ridge regression (KRR) model. As a result, they reported an RMSE value of 11.12 K on the testing set, which seems overly optimistic. Some criticism of these results is related to the extremely small size of the database used for training (120 values) and testing (13 values). The estimation of the T_g value is also available on the recently developed "Polymer Genome" online platform introduced by Ramprasad et al.[38] The underlying ML model is based on Gaussian process regression (GPR) with 68 descriptors for polymer representation. The database used for the development of this ML model consists of 451 values of T_g with 360 and 91 values in training and testing sets correspondingly. The best RMSE on the testing set was shown to be 24 degrees. Finally, Cheng et al.[35] examined different training methods to construct a predictive ML model for polyimide (PI) T_g prediction. The authors used a database of 225 PIs with 225 T_g values (160 and 65 values in training and testing sets, correspondingly). The best performance of the ML model with 197 descriptors (the mean absolute error is about 20 K) is achieved if the training and testing data sets are statistically representative of the entire data set, subject to additional optimization of the ML model.

The above-mentioned ML models are based on thorough calculations of various descriptors, which are typically performed by quantum chemistry or Molecular Dynamics simulations. This approach is rather complicated due to the variability of the results of additional simulations and statistical analysis of the descriptors sets and sometimes leads to the lost information about chemical structure. Deep neural networks (DNN) that directly operate with a graph representation (so-called graph neural networks, GNN) of polymer repeating units are free from these limitations and may, therefore, be extremely useful.[39]

However, the lack of experimental data hinders further intensive application of DNN for predicting polymer properties. An original solution of this problem is the so-called "transfer learning" approach,[40,41] wherein ML models are developed in two stages. First, the ML model is pretrained by using proxy characteristics, which correlate with the desired property, while at the second stage the model is fine-tuned under the available restricted data set in order to make final prediction of target property.

The "transfer learning" approach has been extensively tested in studies devoted to the prediction of the properties of small molecules.[42−48] Various DNN have appeared to be more accurate if they have been pretrained in advance on large amounts of "synthetic" (i.e., theoretically calculated) data, and only then fine-tuned with other more precise computational or experimental data.[45−48] For small molecules, these results were obtained using specifically developed quantum-chemical databases comprising up to hundreds of thousands of compounds and their properties (including, for example, the well-known quantum-chemical QM9 database,[49] the Materials Project database,[50] and the Open Quantum Materials Database[51]).

While analyzing these studies, an important question arises regarding the usefulness of such databases for the development of DNN for predicting polymer properties. One may assume that a large amount of uniformly obtained data for low molecular-weight compounds in quantum-chemical databases may be also used at the stage of pretraining.

An important test of this assumption was made by Yoshida et al., who developed a fully connected neural network for predicting the thermal conductivity of novel polymers.[12] The authors pretrained their model on a large data set of specific heat at constant volume from the QM9 database of small molecule properties (133 805 records), followed by fine-tuning (using "transfer learning") on a small experimental data set (28 records) on thermal conductivity values of known polymers. This enabled them to reduce the mean absolute error (MAE) of thermal conductivity by 40% compared to that of a random forest model trained directly using the 28 data points. Ramprasad et al. also have shown the advantage of using data of different fidelity for developing machine learning models.[52,53] Particularly, in ref 53 a multifidelity information fusion model based on the co-kriging method was developed for predicting crystallization tendency of polymers. During training of their model, the authors used database comprised of a low-fidelity data set for 429 polymers calculated by means of the van

https://doi.org/10.1021/acsomega.2c04649
ACS Omega 2022, 7, 43678−43691

Krevelen's group contribution method and high-fidelity experimental data set for 107 polymers. As a result, their model was found to be by 23% more accurate than the standard Gaussian process regression model trained only with high-fidelity data set. However, comprehensive testing of the applicability of the "transfer learning" approach was beyond the scope of these works, and consequently, rather simple architectures were used for the neural networks and a very limited amount of data was used for their development.

Nevertheless, more complex models and larger data sets have been considered by other authors.[54,55] St. John et al. applied "transfer learning" for the development of message passing neural networks (MPNN) to predict the electronic properties of polymers.[54] For this purpose, they used two data sets composed of the results of density functional theory (DFT) calculations performed on different levels. Both data sets contained monomer properties and corresponding approximated values for polymers. During "transfer learning", the authors pretrained their MPNN on the data set of lower level DFT data and fine-tuned it with a lower amount of high-quality DFT. The results of the study have shown that this MPNN provided three times more accurate predictions for polymer bandgap values than other MPNN whose weights were randomly initialized instead of being pretrained. Shi et al. also demonstrated the enhancement of the electronic property predictions for long oligomers if a graph convolutional neural network (GCNN) was pretrained in advance on the DFT data for monomers and then fine-tuned with a smaller amount of corresponding data for larger molecules composed of them.[55] This enabled them to improve the accuracy of the prediction by 37% compared with the results for the model developed without "transfer learning".

However, in the aforementioned studies, the data sets used were primarily constructed on the basis of time-consuming DFT calculations for oligomers, while "transfer learning" was tested only inside the "synthetic" domain of data. Obviously, this is related to the fact that the amount of data on polymers in the literature which could be effectively used at the pretraining stage is very limited. To avoid this limitation, an alternative strategy was recently proposed by Hasebe, who suggested the use of a novel DNN architecture called the knowledge-embedded message-passing neural network (KEMPNN) for the prediction of the polymer glass-transition temperature.[56] The key idea of this approach is to introduce manual annotations about the influence of the chemical graph composition on the target properties. The performance of the KEMPNN was also tested for polymers. In this case, Bicerano's database comprising 315 polymers and their T_g values was used for training. As a result, RMSE = 33.6 ± 5.2 K was achieved, which is lower by almost 5 K than RMSE = 38.5 ± 6.4 or a baseline MPNN model without knowledge embedding. However, these results lay within the same interval if one takes the RMSE uncertainty into account.

Therefore, despite the results of the works discussed above, the potential of "transfer learning" in developing DNN for polymer properties predictions remains unexplored due to the absence of purely "synthetic" huge data sets of polymer properties which may be used at the pretraining stage.

In the present work, we attempt to answer the question regarding the possibility of applying "transfer learning" to develop GCNN for polymer property prediction by using enormous "synthetic" data sets of artificially generated polymer structures and their macroscopic properties, calculated by means of classical computational schemes, instead of huge databases of small molecules, at the stage of ML model pretraining. In spite of previously used "transfer learning" from lower to higher fidelity methods[52,53] our training methodology is tested on GCNN of a novel architecture. This class of models appears to be one of the versatile models for operating with large amounts of data.[57] The models have been used to solve various tasks in different areas, including materials chemistry. However, GCNN were mainly applied to predict the properties of small molecules or crystals,[44,46,58−62] while a few works consider polymers,[55,63−65] and just one of these consider "transfer learning"[55] restricted, however, by a "synthetic" domain of data.

For this purpose, we apply a specially developed "synthetic" data set of PI T_g values from our PolyAskInG (Polymer Askadskii's Intelligent Gateway) database (http://polycomplab.org/index.php/ru/database.html). The choice of PIs as a class of testable heterocyclic polymers in the present study is dictated by the possibility of generating their chemical structure from simple chemical groups, such as in the LEGO set[66] and by their high potential in various industrial applications.[67] Due to its universality, for calculating the PI T_g values in our "synthetic" data set, we chose Askadskii's computational scheme.[15,16] It is not parametrized for some particular class of compounds, as, for example, in the case of simple regression ML models, and provides more versatility for predicting the properties of various polymers having diverse chemical structures without the need to take into account a large number of correction factors.

As a result, we compare MAE for three methods of predicting PI T_g values: Askadskii's computational scheme,[15,16] GCNN pretrained on the QM9 database,[59,60] and GCNN pretrained on the "synthetic" data set of our PolyAskInG database (both of which are fine-tuned with the experimental data set of the PolyAskInG database of PI T_g values).

In addition, we address the "reality gap" problem that arises from the differences in the size of the experimental and "synthetic" data sets influencing the GCNN training quality. This problem has been first explored in the area of deep learning in computer vision tasks:[68−76] "synthetic datasets" may contain rather huge values with fail-proof ground-truth labeling; a large amount of data could significantly decrease the model performance on real data. Regarding this issue, we derive important estimates about the necessary amount of data on pretraining stage for the effective development of GCNN to predict polymer properties.

The rest of the paper is organized as follows. In section 2 (Materials and Methods) we describe the data sets from the QM9 database and the PolyAskInG database and discuss the principles of a combinatorial generation of PI repeating units, our GCNN architecture, its training, and testing methods. In section 3 (Results and Discussion) we present the obtained results on the efficiency of GCNN pretrained on a data set from QM9 or on a "synthetic" part of the PolyAskInG database by comparing the predictions of these GCNNs with the experimental data. The conclusions are given in section 4.

2. MATERIALS AND METHODS

2.1. PolyAskInG Database of Polyimide T_g Values. In order to develop GCNNs, in the present work, we have used different data sets. The PolyAskInG database is made up of the experimental and "synthetic" data sets of PI T_g values.

2.1.1. Experimental Data Set. The collection of data on the macroscopic properties of polymers represents a challenging task, even in the case of T_g. It is well-known that polymer T_g depends on many specific factors, such as polymer molecular weight, polydispersity, the presence of chemical cross-links,

https://doi.org/10.1021/acsomega.2c04649
ACS Omega 2022, 7, 43678−43691

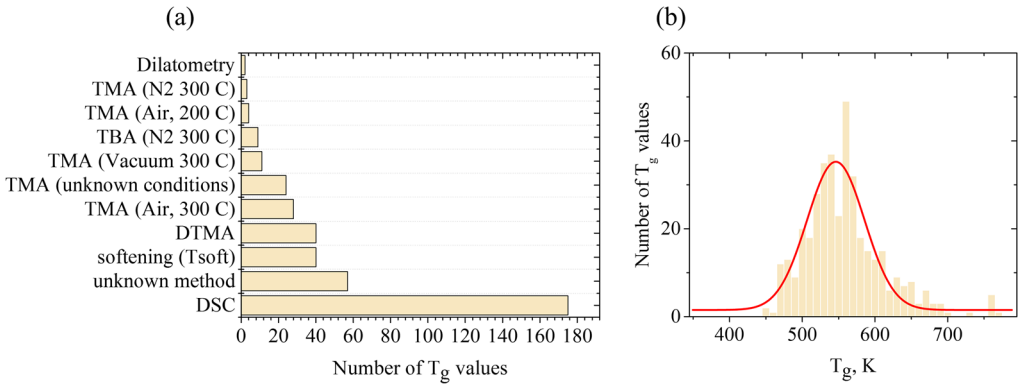


Figure 1. (a) Number of experimental T_g values for PIs in the experimental data set of the PolyAskInG database. (b) Distribution of PI T_g values in the experimental data set of the PolyAskInG database. The red line indicates the fitting with the Gaussian function with $R^2 = 0.88$ (peak at 547 K).

residual solvents, thermal prehistory of the sample, presence of crystalline regions, etc.[77−83] Moreover, the experimental T_g value may be measured by various methods: differential scanning calorimetry (DSC), thermal mechanical analysis (TMA), dynamic mechanical thermal analysis (DMTA), softening experiments, etc. Also, it is worth mentioning the early experiments reporting the so-called softening temperature (T_{soft}) closely related to the glass transition temperature of polymers but not equal to it. The dependence of T_g on the method of its determination is well-known not only in experimental studies but also in computational approaches, for example, in Molecular Dynamics simulations.[84] Moreover, T_g measured by using each of the above methods depends on a particular experimental setup used: cooling rate (in DSC experiments or in simulations), a type of atmosphere (in TMA or DMTA experiments), etc. Particularly, the effect of the cooling rate has been shown for polystyrene[85] and in our previous work on Molecular Dynamics simulations of polyimides.[86] Finally, the temperature range of the glass transition region of a particular class of polymers under investigation may also be an important factor, since the width of this region may alternate for different classes of polymers. Thus, the variability of T_g values obtained for the same polymer by different experimental methods can reach 20−25 degrees.

The problems discussed above introduce a certain arbitrariness in the collected data. Since many of the influencing factors are simply neither mentioned nor thoroughly discussed in the relevant works, the size of uniformly obtained databases that could be developed on the basis of such data will be extremely small. From this point of view our "synthetic" data set is free from this shortcoming.

The experimental data set of the PolyAskInG database was compiled on the basis of the published results[87−90] and the references therein and contains 214 PI repeating units composed of 7 atom types (C, H, O, N, F, S, and Cl) and 607 values of PI T_g. Some PIs were not selected from the published data because their chemical structure included rarely occurring types of atoms (such as Br or Si) or extra-large bulky groups.

Taking into account the diversity of the experimental approaches used to investigate glass transition, we have classified T_g values regarding the measurement method and the corresponding conditions: TMA (in various environments), DMTA, DSC, TBA, dilatometry, softening experiments (to measure so-called softening temperature T_{soft}), and "unknown method" (when information is not presented in the original source). All structures and corresponding T_g values, experimental methods, and measurement conditions were additionally checked and verified. This verification allowed us to correct many errors and misprints that are present in the values of T_g or even in the chemical structure of the PIs (for the details we refer the reader to Section S2 in the Supporting Information).

The chemical structures of the PIs in the database are represented in the SMILES format[91] for convenience in operating with the database in the future. Figure 1(a) contains a frequency plot of different experimental methods' exploitation, while the distribution of T_g values is shown in Figure 1(b).

The distribution of PI T_g values in experimental data set is almost Gaussian ($R^2 = 0.88$) with a peak at 547 K and a mean T_g value of 557 (±59) K. Furthermore, the analysis of the experimental database shows that the T_g values obtained by various methods or conditions (within the same method) may differ between each other by up to several tens of degrees.

2.1.2. "Synthetic" Data Set. The "synthetic" data set contains chemical structures of PI repeating units composed of seven atom types (as in the experimental data set) in SMILES format[91] and the corresponding T_g values. The database was developed in two stages.

At the first step, the ChemLG program (version 0.6.0)[66,92] was used to generate separately diamines (5075 units) and dianhydrides (322 units) according to fixed combinatorial rules, followed by combinatorial generation of 6 726 950 PIs. The repeating unit of any PIs is composed of diamine and dianhydride, Figure 2(a). In turn, diamines and dianhydrides usually consist of an alternating sequence of flexible ("linkers") and bulky ("moieties") groups, Figure 2(b,c), as was recently rightly emphasized by Afzal et al.[66] Based on this work, as well as by analyzing the PI structures in the experimental data set of the PolyAskInG database, we have identified 7 "linkers" and 28 "moieties" most frequently appearing in the PI repeating units, Figure 2(b,c). For further details about the generation of polyimide repeating units we refer the reader to the Section S3.1 in the Supporting Information.

At the second stage, the calculation of the T_g values for each PI was performed according to Askadskii's computational scheme implemented in the "Cascade" program.[15,16] A comprehensive description of Askadskii's computational scheme for calculating T_g is given in the Section S3.2 of the Supporting Information. It should be noted that the PIs with T_g values greater than 800 K (15 repeating units in total) were excluded from the final version of the "synthetic" data set because of the absence of such T_g values for PIs in the experimental data set of the database. As a result, the final number of PIs in the "synthetic" data set of the

https://doi.org/10.1021/acsomega.2c04649
ACS Omega 2022, 7, 43678−43691

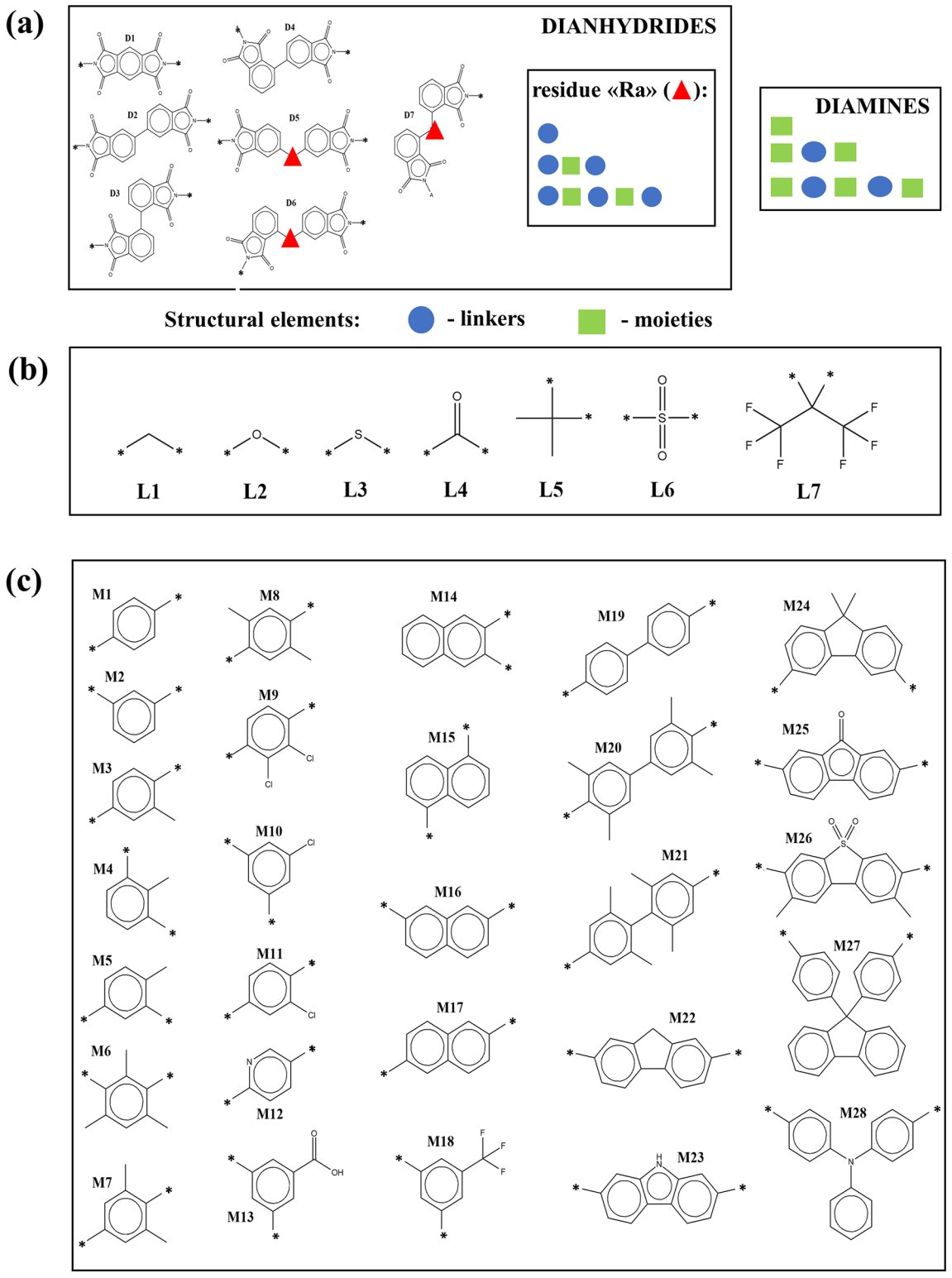


Figure 2. (a) Principal structures of diamines and dianhydrides used in the combinatorial generation of PI repeating units in the "synthetic" data set of the PolyAskInG database. Diamines and "Ra" residues in dianhydrides represent the combinations of "linkers" (b) and "moieties" (c).

database is 6 726 935. The distribution of PI T_g values in "synthetic" data set is almost Gaussian (correlation coefficient R^2 = 0.997) with a peak at 493 K and a mean T_g value of 501(±43) K, Figure 3.

2.1.3. Quantum-Chemical Data Set. The QM9 quantum-chemical database comprises the geometric, energetic, electronic, and thermodynamic properties of 133 885 small organic molecules calculated using the density functional theory (B3LYP/6-31G(2df,p) level of theory).[49] During recent years, the developing ML model using the QM9 database has become a so-called "golden standard" in relevant areas of chemistry. The elemental composition of molecules includes only 5 elements (H, C, N, O, and F). However, for the polymers and for the PIs, in particular, repeating units could also contain S and Cl atoms.

https://doi.org/10.1021/acsomega.2c04649
ACS Omega 2022, 7, 43678−43691

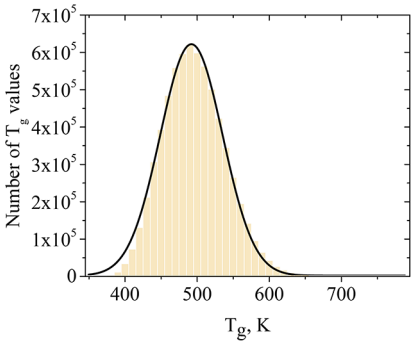


Figure 3. Distribution of T_g values of the PI in the "synthetic" data set of the database. The black line indicates the fitting with the Gaussian function with R^2 = 0.997 (peak at 493 K).

Such repeating units were also included in our "synthetic" data set of PI T_g values described above (see Section 2.1.2 for details). For our GCNN pretraining, we used subsets corresponding to isotropic polarizability (α), energy of highest occupied molecular orbital (ϵ_{HOMO}), and energy of lowest unoccupied molecular orbital (ϵ_{LUMO}). Previously, these characteristics were successfully used as descriptors in different ML models for the prediction of polymers T_g.[23,24,26]

2.2. Neural Network Architecture and Training. *2.2.1. Data representation.* Recently, Lee et al. showed that GCNNs better predict the properties of polymers if the input molecular graphs represent repeating units of polymers rather than their oligomers or monomers end-capped with hydrogen atoms.[65] Thus, in the presented model, the molecular graph of the PI repeating unit is used as input; see Figure 4.

The molecular graph contains the most important information about the chemical structure of small molecules (QM9) or polymer repeating units (PolyAskInG database). Each molecular graph is represented by a set of vertices (atoms) and edges (valence bonds). To describe the molecular graph, we encode the following characteristics of the vertices: atom type (C, N, O, S, F, Cl, or H); valence of atom; number of bonded hydrogen atoms; belonging of atom to aromatic ring (bool: 0—no, 1—yes). In turn, to encode the properties of the edges, we use the following features: bond type (single, double, or triple) and bond length.

For encoding atom and bond types, as well as valences of atoms, we use one-hot encoding, meaning that a vector of a fixed length is filled with 0 (if property does not match) and 1 (if property matches). Since we would like to predict the T_g value of the polymers, we also encode a specific cyclic connection in the molecular graph for them (see the gray line in Figure 4).

For the simplicity of the data collection and representation we have not used the information about the phase state of polyimides during developing our experimental data set. We supposed that all the data was obtained for amorphous polymer samples. Also, no information about the configuration of polymers or their dynamics was used.

2.2.2. Proposed GCNN Architecture. Our GCNN has a classical architecture for graph classification/regression. It contains three main parts: Graph convolution part, i.e., the sequence of graph convolutional layers (GCL); Feature aggregation, i.e. the pool (aggregation) function that maps a set of hidden vectors to output vector; Multilayer perceptron (MLP) part, i.e., MLP at the top of GCNN, for the final prediction.

We use the modified Gated graph convolution[93] with gated recurrent unit[94] as a basic operation for our GCNNs. We account the edge features using the learnable message function, which uses the edge feature vector as an input and applies the $k \times k$ matrix A implemented by MLP to transform it. This MLP is trained simultaneously with the GCL layer.

Since a typical molecular graph of a polyimide repeating unit may contain up to about several tens of vertexes, a lot of iterations are required to pass information from one vertex to another. For this reason, along with the graph neural network (GNN) part we use an additional 2-GNN part in our GCNNs, which distinguishes our network from that used in refs 54 and 63. Previously, Grohe et al. have shown that adding a 2-GNN part increases the expressive ability of ordinary graph neural networks, which is usually limited.[95] Our preliminary tests have also demonstrated that adding a 2-GNN part in parallel to the original graph neural networks improved its performance. A schematic illustration of the proposed GCNNs architecture is presented in Figure 5.

Each of our GCNNs has five GCLs. There are three GCLs before the "2-graph" conversion procedure and two graph convolutional layers after it. After each GCL, we use Rectified Linear Unit (ReLU) activation.

For each GCL, we apply three message passing sessions. In addition, we use the two-layer MLP with a linear layer consisting of 256 neurons followed by a linear layer consisting of 4096

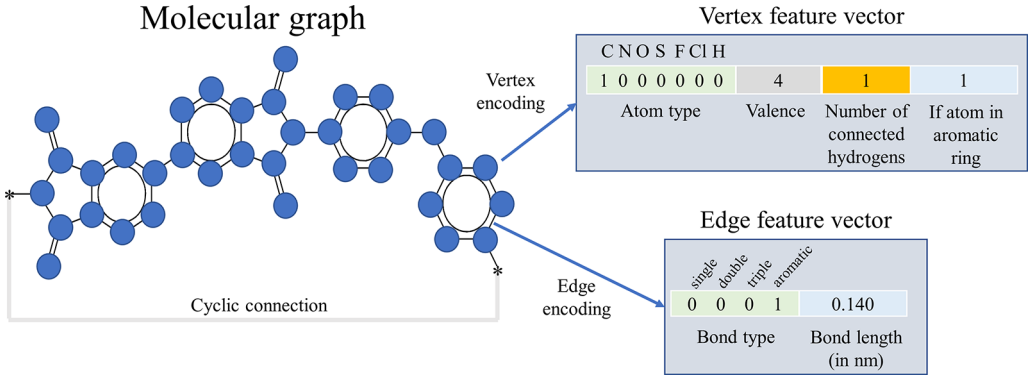


Figure 4. Illustration of the encoding procedure of a molecular graph. Each vertex (atom) is described within a feature vector with parameters: the atom type, valence of atom, number of connected hydrogen atoms, and belonging of an atom to an aromatic ring. Each edge (bond) is described within a feature vector with parameters: bond type (single, double, triple, or aromatic) and bond length (in nanometers). The gray line indicates a cyclic connection between repeating units.

https://doi.org/10.1021/acsomega.2c04649
ACS Omega 2022, 7, 43678−43691

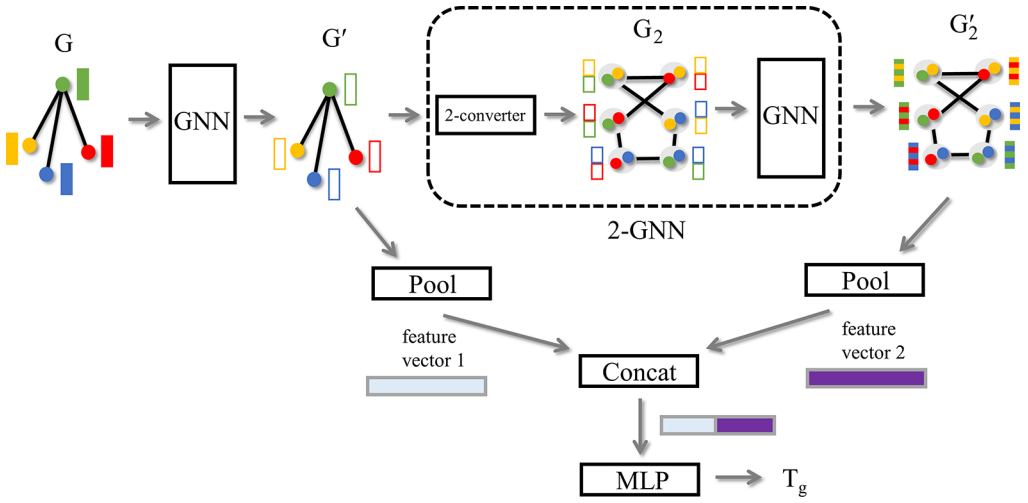


Figure 5. Schematic illustration of the proposed GCNN architecture. Colored bars indicate feature vectors of vertexes. G denotes input molecular graph of the polyimide repeating unit, G' is the input graph processed with the first GNN block (first subnet), G_2 is the "2-graph" formed from the G', G_2' is the "2-graph" processed with the second GNN block (second subnet), and MLP is the multilayer perceptron. For the convenience of drawing the feature vectors of the edges are not shown.

neurons (the dimension of A matrix is 64 × 64) with ReLU nonlinearity in between in order to compute A matrix. For readout, we use ordinary sum operation. After readout, stage, and concatenation, the resulting vector is processed by a linear layer with 256 neurons with ReLU to get the final feature vector. Then we use a one-neuron layer to get the final prediction. The dimensions of the input graph features are reduced to a fixed size with 64 channels using two linear layers (one for the node features and one for the edge features).

For a more comprehensive review about our GCNN architecture, we refer the reader to Section S4 in the Supporting Information. Testing code and pretrained models are available on https://github.com/polycomplab/GCNN_PI_glass_transition.git.

2.2.3. Training and Testing Details. Modern deep neural networks require really big data sets for training because they contain a huge parameter set that needs to be tuned. As noted above, public data sets in chemistry are much smaller than, for example, computer vision data sets (e.g., the most popular ImageNet data set contains 1 million images[96]). Only several huge databases are known in chemistry (such as QM9 for low molecular-weight compounds mentioned above[49]). Contrary to such theoretical data sets, the creation of comprehensive experimental data sets for ML is almost impossible in this research area, since it requires time-consuming and expensive experiments. Even databases containing several-thousands polymers of a single type are practically unavailable. To overcome this problem, one may use the "transfer learning" approach.

On the other hand, one of the most popular ML ideas for operating with small experimental data sets is to expand them using mathematical simulation (artificial data modeling). We use a similar approach for T_g prediction. Our "synthetic" data set, which is described above (see Section 2.1.2), contains over 6 726 935 polyimides with the corresponding T_g values. We use this "synthetic" data set as an alternative for the data set from QM9 for pretraining our neural network.

The development of each GCNN consisted of the following stages: pretraining GCNN (using the data set from the QM9 database or the "synthetic" data set from the PolyAskInG database) and fine-tuning GCNN on a small amount of available experimental data (using an experimental data set of the PolyAskInG database).

Thus, we will compare the results for the GCNN pretrained on the "synthetic" data set of PolyAskInG database (using the T_g values calculated by Askadksii's computational scheme) and GCNN pretrained on the QM9 data set (using proxy characteristics similar to that used in ref 12), both of which are fine-tuned using the experimental data set of the PolyAskInG database. Note that GCNN pretrained on the QM9 database was chosen as the benchmark model. Additionally, we will obtain results for one more GCNN trained only on the experimental data set of the PolyAskInG database. Note that the data in the "synthetic" data sets used for pretraining were normalized by subtracting the corresponding averaged values of T_g and dividing them by standard deviation. This allowed us to account for the differences in the characteristics of T_g distributions in experimental and "synthetic" data sets.

Fine-tuned GCNN models were tested on PI T_g experimental values, also included as a part in PolyAskInG database. However, since our experimental data set of the PI T_g is relatively small, we use a 10-fold cross-validation technique to obtain statistically meaningful results about the performance of GCNNs (unless otherwise stated). In 10-fold cross-validation, the training data set is randomly partitioned into 10 equally sized nonoverlapping subsets. Then we use one subset for testing and the remaining 9 subsets for training. We repeating this procedure N times, until each of the N subsets is tested once. Then we average the results to produce a single estimate of the mean average error (MAE) used as a quality metrics.

Note that our experimental data set contains T_g values obtained by different experimental techniques (TMA, DMTA, DSC, Dilatometry, TBA, etc.). However, due to the small amount of records, we use a single T_g value (averaged over all experimental values for a particular polymer) as a prediction target for each polymer in the data set. Recently, it was shown that such approach is the best choice for developing ML models.[97]

We use the PyTorch Framework[98] and PyTorch Geometric extension library[99] for implementing and training our GCNNs. The training source code is available on https://github.com/polycomplab/GCNN_PI_glass_transition.git. The training

https://doi.org/10.1021/acsomega.2c04649
ACS Omega 2022, 7, 43678−43691

time is about 3 days for pretraining on the "synthetic" data set and about 1 h for fine-tuning on the experimental data (for a 2-GPU server with Tesla P100 accelerators). In the case of using the "synthetic" data set from the PolyAskInG database 90% of the data set was in the training set and 10% in the validation set, unless otherwise stated. In turn, the data set from the QM9 database was split into training, validation, and test, following an 80/10/10 ratio.

3. RESULTS AND DISCUSSION

3.1. Testing of the Pretrained GCNNs. Testing of GCNN pretrained on the QM9 data set was performed against the corresponding validation set. As a result, we have reached MAE = 0.07 for α, 37.1 meV for ϵ_{HOMO}, and 34.1 meV for ϵ_{LUMO}. These values are comparable with those commonly reached by applying other ML models.[54,100] We should emphasize that during GCNN pretraining on QM9 database we used a slightly different architecture of GCNN compared to its final variant—with the last layer containing 3 outputs. However, at the fine-tuning stage, the last layer was replaced by inner product layer with only 1 output in order to predict the T_g value.

The results of testing of GCNN pretrained on the "synthetic" data set of the PolyAskInG database are presented in Table 1. Testing was performed against theoretical T_g values of PIs in the experimental data set obtained by Askadskii's computational scheme.

Table 1. Mean Absolute Error (MAE) of T_g Predicted by Askadksii's Computational Scheme for GCNNs Pretrained on "Synthetic" Datasets of Various Sizes from the PolyAskInG Database[a]

No. of PI in "synthetic" data set used for pretraining	MAE, K
1000	14.9
5000	10.2
100000	6.5
1000000	7.4

[a]MAE was calculated directly between the predicted and target sets.

Taking into account the significantly larger size of the "synthetic" part of the PolyAskInG database compared to the size of the QM9 data set, we have estimated MAE values with respect to the size of the "synthetic" data set used during pretraining. Our results presented in Table 1 show that MAE reduces with increasing the size of the pretraining data set until the number of entities in the data set reaches 100 000. A small increase of the MAE in the case of the largest data set (from 6.5 to 7.4) may be explained by overfitting of GCNN. Nevertheless, we should emphasize that for only pretrained GCNN the best MAE is achieved on a "synthetic" data set comprised of 100 000 or even more entries (see the results in Table 1). Therefore, using a massive "synthetic" data set is of considerable significance in reaching the best accuracy for the GCNN predictions. We assume that for the entire "synthetic" data set from the PolyAskInG database we will obtain even smaller values of MAE.

However, reaching the minimal values of MAE on the validation sets was not a priority task, since the final accuracy of the developed GCNN is dictated by fine-tuning on the experimental data set.

3.2. Testing Fine-Tuned GCNNs: "Reality Gap Problem". After testing pretrained GCNNs, we evaluated the performance of the fine-tuned models. To this end, we have performed three independent learning experiments for GCNNs pretrained on data sets of various sizes to estimate the MAE for experimentally predicted T_g values. During testing we used a single value of T_g for each PI. This value is obtained by averaging over multiple experimental values (if available) corresponding to a certain PI.

As a result, we observe no significant differences in the MAE values being not less than 22.5 K, Table 2.

Table 2. Mean Absolute Error (MAE) and Standard Deviation (SD) Values Obtained for GCNNs Pretrained on "Synthetic" Dataset of Various Sizes from the PolyAskInG Database[a]

No. of PI in "synthetic" data set	MAE, K	SD, K
5000	23.2	5.3
100000	22.5	5.1
1000000	23.2	5.7

[a]MAE was calculated using a 10-fold cross-validation technique. Target T_g for each PI is an average over all corresponding experimental values.

Given that Askadskii's computational scheme was originally parametrized for TMA experiments, we also used an additional experimental set for validation. In this set, there are those T_g values which were obtained only by TMA methods. Averaging was also performed if several TMA experiments were performed to characterize the T_g value of a particular PI. As a result, we observed similar MAE values (see Section S5 in the Supporting Information).

Another important result is related to the influence of the size of the "synthetic" data set on the GCNNs performance, Table 2 demonstrates that the "synthetic" pretraining data set containing at least 5000 records is already enough to achieve the optimal accuracy of the proposed GCNN regardless of the testing method. This result is in line with the conclusions of previous works addressing the problem of a so-called "reality" gap, existing in the case of using both real (experimental) and "synthetic" data for deep neural network training. Namely, the results obtained during the application of convolutional neural networks in the field of computer vision[68−76] allow one to conclude that the optimal ratio of real data to "synthetic" data in any deep learning task could be about 5−20% to 80−95% (for a more comprehensive review we refer the reader to Section S6 in the Supporting Information). Since our current experimental data set contains 214 PIs with 607 values of their T_g values, one may assume that starting with about 214 real samples at the lowest possible real-to-synthetic data ratio 5% to 95% we will obtain just about 5000 samples in total effective real-and-synthetic training set. Thus, we experimentally demonstrate that such a real-to-synthetic data ratio meets the "reality gap" problem in our case as well as in other known cases. Our conclusion is also found to be independent of the similarity of the "synthetic" and experimental data sets composition (for the details we refer the reader to the Section S7 in the Supporting Information).

Having obtained the above results, hereinafter, we will discuss the results for GCNN pretrained on the data set comprised of 100000 PIs randomly chosen from the original "synthetic" data set, which is almost equal in size to the QM9 data set (107 108 records).

3.3. Testing Fine-Tuned GCNNs: Polymer Vs Monomer Pretraining Data Sets. Our main results of testing GCNNs

https://doi.org/10.1021/acsomega.2c04649
ACS Omega 2022, 7, 43678−43691

and Askadskii's computational scheme on the experimental data set of the PolyAskInG database are shown in Table 3.

Table 3. Mean Absolute Error (MAE) and Standard Deviation (SD) Values for Prediction of PI T_g Using Different Models

Predicting model	MAE,[d] K	SD, K
Askadskii's computational scheme	33.4	5.3
GCNN + exp[a]	28.1	5.7
GCNN + QM9 + exp[b]	40.6	6.9
GCNN + PolyAskInG[c]	22.5	5.1

[a]GCNN + exp: GCNN trained only on the experimental data set of the PolyAskInG database. [b]GCNN + QM9 + exp: GCNN pretrained on the QM9 database and fine-tuned with the experimental data set of the PolyAskInG database. [c]GCNN + PolyAskInG: GCNN pretrained on the synthetic data set of the PolyAskInG database and fine-tuned with its experimental part. [d]MAE for Askadskii's computational scheme was calculated over all experimental datasets. MAE for GCNNs was calculated using a 10-fold cross-validation technique. MAE for Askadskii's computational scheme was calculated directly between the predicted and target sets. Target T_g for each PI is an average over all corresponding experimental values.

First of all, from Table 3 it follows that Askadskii's computational scheme provides MAE = 33.4 K in the case when the predicted T_g values for each PI are averaged over all experimental methods. Similar results were obtained for GCNNs testing with respect to TMA and DSC methods (see the Section S8 in Supporting Information).

The main reason for such rather high MAE values comes from the fact that originally Askadskii's computational scheme was calibrated to predict T_g values that should be obtained by TMA experiments. We assume that the large number of experimental results obtained by DSC, TBA, DMTA, and dilatometry (the number of T_g is 253 out of 607 in total) may cause a high MAE value in the case of testing Askadskii's computational scheme.

Comparable prediction accuracy is obtained for GCNN, trained only on the experimental data set of the PolyAskInG database (MAE = 28.1 K). Interestingly, this result indicates better performance of the GCNN compared to a classical Askadskii's computational scheme. Nevertheless, due to limited size of the experimental data set, further investigations are required to test developed GCNNs on a larger experimental database (which will be performed in the future).

Further analysis of the results in Table 3 indicates that the GCNN developed within the "transfer learning" approach using the data set from the PolyAskInG database is even more accurate: MAE is 22.5 K, which is 11 degrees less than MAE for Askadskii's computational scheme.

Bearing in mind that the MAE for GCNN trained only on the experimental data set of the PolyAskInG database; this result proves the necessity of pretraining while developing more efficient GCNN. On the other hand, we also observe better performance of GCNNs in comparison with Askadskii's computational scheme, which may be due to the accounting in GCNN for complicated relationships between T_g and chemical structure of the PI repeating unit, as well as due to the more accurate calibration of GCNN for prediction of experimental results obtained by different methods.

Finally, GCNN pretrained using the QM9 data set and fine-tuned with experimental data on PI T_g values provides MAE = 40.6 K, which is almost two times larger than for GCNN pretrained with "synthetic" polymer data. This conclusion is also supported by visual analysis of the parity plots and corresponding coefficient of determination R^2 values, Figure 6.

Otherwise speaking, using QM9 for pretraining provides a worse result than if we used a "synthetic" polymer data set. The reason for this may be 2-fold. On the one hand, we suppose that accounting of the polymeric nature of the compounds under investigation may play a crucial role in predicting polymer properties. Obviously, pretraining on a data set of small molecules properties does not provide such an advantage. On the other hand, even if we ignore the above assumption, the reason for a worse performance of QM9 may be explained by the fact that it does not contain T_g values itself, only the characteristics that implicitly correlate with T_g.

Additionally, we make a note about the dependence of GCNNs performance on different accuracies of training data. On the one hand, T_g values in the "synthetic" data set are homogeneous; i.e., they are obtained by the application of Askadskii's computational scheme as a single method. From this point of view, our experimental data set is rather heterogeneous since T_g values obtained using various methods are reported. This fact entails the problem of the different accuracy of training data, which is hard to avoid due to the limited amount of experimental data available in the literature sources which has been obtained using a single method. However, Askadskii's

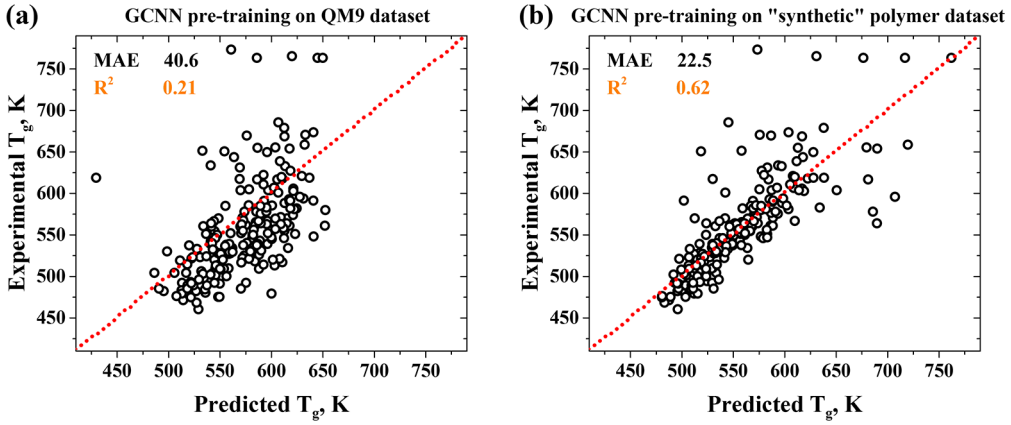


Figure 6. Parity plots for the T_g predictions of GCNNs pretrained on the (a) QM9 data set and (b) "synthetic" polymer data set of the PolyAskInG database. The coefficient of determination R^2 is given in each subplot. Red dotted line serves as a guide for the eyes.

https://doi.org/10.1021/acsomega.2c04649
ACS Omega 2022, 7, 43678−43691

computational scheme was initially calibrated for predicting T_g values predicted by the TMA method. Therefore, we could make rough estimates about the influence of the different accuracy of the training data if we compared T_g values predicted by GCNN with corresponding experimental values averaged: (a) over all methods (Table 2) or (b) only over the TMA method (Table S4). Testing our best GCNN over the experimental values averaged over all methods gives MAE = 22.5 K, while the corresponding MAE for the GCNN tests over experimental values averaged over only TMA methods is 24.9 K. Thus, our results indicate that different accuracy of training data has a negligible influence on the final performance of the GCNN.

All in all, the above results clearly indicate that using "synthetic" data sets of polymer properties for pretraining GCNN is more preferable than applying data sets of small molecule properties for this purpose. This result states the overall priority of using our methodology for training deep neural networks, and GCNN in particular, for efficient predicting other polymer properties.

3.4. Limitations and Future outlook. Obviously, when analyzing the results of our study, certain limitations should be taken into account.

The efficiency of the "transfer learning" is dictated by the ratio between the sizes of the "synthetic" and experimental data sets (the so-called "reality gap" problem). On the one hand, more data used during deep neural network training provide it with better accuracy. On the other hand, better accuracy for the predicting model will be achieved in cases where there is a larger amount of experimental data with a fixed amount of "synthetic" data. Therefore, one should take into account that the overall advantage of "transfer learning" in the approach suggested is limited by the size of the experimental database.

The graph convolutional neural networks developed are optimized for a particular class of polymers, i.e., polyimides. The transferability of our model to the other polymer classes, whose structure is composed from a similar set of chemical groups, was beyond the scope of the study. However, if one attempts to use our neural network model for other polymer classes, we suggest to start from a pretrained model and then fine-tune it with a database for the polymer class under investigation.

Polyimide repeating units in a "synthetic" data set are composed of similar sets of elementary structural building blocks presented in experimental structures with combination rules underlying their generation. Certainly, there exist plenty of other polyimide repeating units which contain other building blocks combined according to different rules. However, including additional experimentally reported polyimides will immediately lead to a considerable increase in the number of polyimide structures in the "synthetic" data set. In the present study, we limited ourselves to the case of affinitive structures in the experimental and synthetic data sets in order to avoid dealing with even larger "synthetic" data sets.

Our graph convolutional neural network model was trained to establish correlations between the most basic (primary) features of polyimides repeating units as molecular graphs. In other words, no information was taken into account about the polymers phase state, the spatial configuration, or their dynamics. However, the phase state of polymers may be also taken into account during GCNN development in order to improve its accuracy. For example, GCNN may be fine-tuned using an experimental data set extended with additional parameters, such as, for example, the difference in the 3D coordinates of macromolecular configurations in the amorphous and ordered states from computer simulations, as well as values of crystallinity degree from experiments. These investigations will be performed in our future works.

Taking into account the aforementioned limitations, we plan to solve the following tasks in the future works:

(1) to test other architectures that take into account additional "synthetic" data during neural network pretraining (for example, the 3D structure of a molecular graph or the dynamic behavior of the molecule) to improve the predictive power of the models;
(2) to apply our neural network models to other classes of polymers (e.g., polyamides, polyamidoimides) and/or properties (e.g., permeability), composed of similar sets of building blocks, in order to test the transferability of the models developed;
(3) to expand the training/testing database in order to refine the results already obtained, including also the testing of the developed neural network model for sets of compounds which are not in the database;
(4) to compare various computational schemes for developing "synthetic" data sets of polymer properties and choose the best one to be used for developing the most accurate neural network model.

4. CONCLUSIONS

DNN are breakthrough models in the current ML, which is successfully applied to solve many different tasks. In recent years, "transfer learning" has appeared as an effective approach for developing such models. However, the potential of this approach has not been fully explored in the field of polymer science, especially for the case of transfer from "synthetic" to the real domain of data having different fidelity.

In the present work, we addressed this problem by introducing an unprecedentedly large "synthetic" data set of polyimides T_g values which was applied while developing GCNN as an example of the most promising class of DNN.

For this purpose, we have developed GCNN with more complex 2-GNN architecture which have not been previously applied for predicting polymer properties and/or to the testing "transfer learning" approach. The training of the GCNN was performed by using an unprecedentedly large set of polyimides structures, whose generation is supported by the analysis of the polyimide repeating units reported experimentally, and corresponding T_g values calculated using the classical computational scheme proposed by Askadskii.

We performed comprehensive testing of our GCNN and its training methodology against other approaches by comparing the results from Askadskii's computational scheme, GCNN developed with the popular QM9 database of small molecules properties, as well as GCNN developed without using a "transfer learning" approach (i.e., trained only on experimental data). The outcomes are as follows:

- we have proven that using extremely large "synthetic" data sets of properties is of critical importance for developing GCNN, as well as for predicting polymer properties (as indicated by the results presented in Table 1);
- we have shown that using "synthetic" polymer data sets of properties at the pretraining stage of "transfer learning" is more preferable for enhancing the final accuracy of the fine-tuned GCNN in comparison to using databases of small molecules, such as the benchmarking QM9 data set (as indicated by the results presented in Table 2);

https://doi.org/10.1021/acsomega.2c04649
ACS Omega 2022, 7, 43678−43691

- we have addressed the "reality gap" problem for polymer property prediction that has been previously explored in the area of deep learning in computer vision tasks: too large amount of "synthetic" data could significantly decrease the model performance on real data. Our estimates show that no more than 95% of the overall amount training data should be "synthetic" in order to provide a reasonable accuracy of GCNN if the "transfer learning" approach is applied to its development.

These results significantly enrich the body of knowledge on using "transfer learning" for DNN, and in particular GCNN, and will impact the strategy of their development through the use of the "transfer learning" approach. On the basis of our results, we recommend the extensive application of "synthetic" data sets of polymer properties instead of data sets for the properties of small molecules (such as, for example, the well-known QM9 database). To develop such "synthetic" databases, classical computational schemes for polymer properties are the most suitable tools to be used. At the same time, particular attention should be paid to the collection and organization of experimental databases of polymer properties in order to eliminate the "reality gap" problem. By following these recommendations, the problem of data scarcity may be overcome, while the development and application of DNN for solving both direct and inverse tasks of the "structure−property" relationship in polymer science continues.

To facilitate an active development in this area, we make available our PolyAskInG database, as well as the source code of the developed GCNN models, on the following web servers: http://polycomplab.org/index.php/ru/database.html and https://github.com/polycomplab/GCNN_PI_glass_transition.git.

We suppose that the presented PolyAskInG database will be just the first step toward creating a comprehensive, publicly available database for machine learning in the area of polymer science. However, even now our "synthetic" data set of the PolyAskInG database could be used to train multifidelity models, along with other existing databases.[101] On the other hand, the approaches proposed in the present work also pave the way toward solving the inverse task in the "structure−property" relationship, which will be the subject of our future publications. We hope that the wide community of polymer researchers will support such activity on public polymer database assembly for ML and that the wide ML community will be interested in solving this attractive and sophisticated ML task.

■ ASSOCIATED CONTENT

Data Availability Statement

The QM9 database is publicly available at http://quantum-machine.org/datasets. The source code of the original ChemLG program is available at https://github.com/hachmannlab/chemlg. The PolyAskInG database including "synthetic" and experimental data sets of PI T_g is available at http://polycomplab.org/index.php/ru/database.html (both as SQlite database file). The source code of the developed GCNN is available at https://github.com/polycomplab/GCNN_PI_glass_transition.git.

Supporting Information

The Supporting Information is available free of charge at https://pubs.acs.org/doi/10.1021/acsomega.2c04649.

Brief overview of the classical computational schemes for T_g prediction (section S1), details about misprints in experimental data (section S2) and the development of "synthetic" data set of polyimides T_g (section S3), details about proposed deep neural network architecture (section S4), literature review about the "reality gap" problem in the field of deep neural networks development (section S6), and additional results of GCNNs testing (sections S5, S7, S8) (PDF)

■ AUTHOR INFORMATION

Corresponding Author

Sergey V. Lyulin − *Institute of Macromolecular Compounds of the Russian Academy of Sciences (IMC RAS), St. Petersburg 199004, Russian Federation;* orcid.org/0000-0002-3743-4457; Email: sergey.v.lyulin@gmail.com

Authors

Igor V. Volgin − *Institute of Macromolecular Compounds of the Russian Academy of Sciences (IMC RAS), St. Petersburg 199004, Russian Federation*

Pavel A. Batyr − *Federal State Unitary Enterprise "State Research Institute of Aviation Systems" (GosNIIAS), Moscow 125167, Russian Federation*

Andrey V. Matseevich − *A.N. Nesmeyanov Institute of Organoelement Compounds of Russian Academy of Sciences (INEOS RAS), Moscow 119991, Russian Federation*

Alexey Yu. Dobrovskiy − *Institute of Macromolecular Compounds of the Russian Academy of Sciences (IMC RAS), St. Petersburg 199004, Russian Federation*

Maria V. Andreeva − *Institute of Macromolecular Compounds of the Russian Academy of Sciences (IMC RAS), St. Petersburg 199004, Russian Federation*

Victor M. Nazarychev − *Institute of Macromolecular Compounds of the Russian Academy of Sciences (IMC RAS), St. Petersburg 199004, Russian Federation;* orcid.org/0000-0003-2448-8584

Sergey V. Larin − *Institute of Macromolecular Compounds of the Russian Academy of Sciences (IMC RAS), St. Petersburg 199004, Russian Federation;* orcid.org/0000-0002-1570-9333

Mikhail Ya. Goikhman − *Institute of Macromolecular Compounds of the Russian Academy of Sciences (IMC RAS), St. Petersburg 199004, Russian Federation*

Yury V. Vizilter − *Federal State Unitary Enterprise "State Research Institute of Aviation Systems" (GosNIIAS), Moscow 125167, Russian Federation*

Andrey A. Askadskii − *A.N. Nesmeyanov Institute of Organoelement Compounds of Russian Academy of Sciences (INEOS RAS), Moscow 119991, Russian Federation; Moscow State University of Civil Engineering (MGSU), Moscow 129337, Russian Federation*

Complete contact information is available at: https://pubs.acs.org/10.1021/acsomega.2c04649

Author Contributions

The manuscript was written through contributions of all authors. All authors have given approval to the final version of the manuscript.

Funding

This work was supported by the Russian Science Foundation Grant No. 22-13-00066, https://rscf.ru/en/project/22-13-00066/. Neural network training was performed using the resources of the computing cluster of GosNIIAS.

Notes

The authors declare no competing financial interest.

■ ACKNOWLEDGMENTS

The authors would like to thank Prof. A. A. Polotsky, Dr. A. A. Mercurieva, V. S. Gorbatsevich, and I. V. Perunov for the comments and fruitful discussions, and V. A. Adigamova for the art design of graphical abstract.

■ REFERENCES

(1) Nazarychev, V. M.; Larin, S. V.; Yakimansky, A. V.; Lukasheva, N. V.; Gurtovenko, A. A.; Gofman, I. V.; Yudin, V. E.; Svetlichnyi, V. M.; Kenny, J. M.; Lyulin, S. V. Parameterization of Electrostatic Interactions for Molecular Dynamics Simulations of Heterocyclic Polymers. *J. Polym. Sci., Part B: Polym. Phys.* **2015**, *53*, 912−923.

(2) Lukasheva, N. V.; Tolmachev, D. A.; Nazarychev, V. M.; Kenny, J. M.; Lyulin, S. V. Influence of Specific Intermolecular Interactions on the Thermal and Dielectric Properties of Bulk Polymers: Atomistic Molecular Dynamics Simulations of Nylon 6. *Soft Matter* **2017**, *13*, 474−485.

(3) Glova, A. D.; Falkovich, S. G.; Dmitrienko, D. I.; Lyulin, A. V.; Larin, S. V.; Nazarychev, V. M.; Karttunen, M.; Lyulin, S. V. Scale-Dependent Miscibility of Polylactide and Polyhydroxybutyrate: Molecular Dynamics Simulations. *Macromolecules* **2018**, *51*, 552−563.

(4) Zhou, T.; Song, Z.; Sundmacher, K. Big Data Creates New Opportunities for Materials Research: A Review on Methods and Applications of Machine Learning for Materials Design. *Engineering* **2019**, *5*, 1017−1026.

(5) Chen, G.; Shen, Z.; Iyer, A.; Ghumman, U. F.; Tang, S.; Bi, J.; Chen, W.; Li, Y. Machine-Learning-Assisted de Novo Design of Organic Molecules and Polymers: Opportunities and Challenges. *Polymers* **2020**, *12*, 163.

(6) Morgan, D.; Jacobs, R. Opportunities and Challenges for Machine Learning in Materials Science. *Annu. Rev. Mater. Res.* **2020**, *50*, 71−103.

(7) Kim, S.; Thiessen, P. A.; Bolton, E. E.; Chen, J.; Fu, G.; Gindulyte, A.; Han, L.; He, J.; He, S.; Shoemaker, B. A.; Wang, J.; Yu, B.; Zhang, J.; Bryant, S. H. PubChem Substance and Compound Databases. *Nucleic Acids Res.* **2016**, *44*, D1202−D1213.

(8) Bohacek, R. S.; McMartin, C.; Guida, W. C. The Art and Practice of Structure-Based Drug Design: A Molecular Modeling Perspective. *Med. Res. Rev.* **1996**, *16*, 3−50.

(9) Audus, D. J.; De Pablo, J. J. Polymer Informatics: Opportunities and Challenges. *ACS Macro Lett.* **2017**, *6*, 1078−1082.

(10) Rubinstein, M.; Colby, R. H. *Polymer Physics*, 1st ed.; Oxford University Press: New York, 2003.

(11) Lee, C. J. Polyimides, Polyquinolines and Poly-Quinoxalines: T_g-Structure Relationships. *J. Macromol. Sci. Part C* **1989**, *29*, 431−560.

(12) Wu, S.; Kondo, Y.; Kakimoto, M.; 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 Comput. Mater.* **2019**, *5*, 66.

(13) van Krevelen, D. W.; te Nijenhuis, K. *Properties of Polymers*, 4th ed.; Elsevier: Amsterdam, 2009.

(14) Bicerano, J. *Prediction of Polymer Properties*, 3rd ed.; CRC Press: Boca Raton, 2002.

(15) Askadskii, A. A. *Computational Materials Science of Polymers*, 1st ed.; Cambridge International Science Publishing: Cambridge, 2003.

(16) Askadskii, A. A. Methods for Calculating the Physical Properties of Polymers. *Rev. J. Chem.* **2015**, *5*, 83−142.

(17) Koehler, M. G.; Hopfinger, A. J. Molecular Modelling of Polymers: 5. Inclusion of Intermolecular Energetics in Estimating Glass and Crystal-Melt Transition Temperatures. *Polymer* **1989**, *30*, 116−126.

(18) Cao, C.; Lin, Y. Correlation between the Glass Transition Temperatures and Repeating Unit Structure for High Molecular Weight Polymers. *J. Chem. Inf. Comput. Sci.* **2003**, *43*, 643−650.

(19) Mattioni, B. E.; Jurs, P. C. Prediction of Glass Transition Temperatures from Monomer and Repeat Unit Structure Using Computational Neural Networks. *J. Chem. Inf. Comput. Sci.* **2002**, *42*, 232−240.

(20) Cypcar, C. C.; Camelio, P.; Lazzeri, V.; Mathias, L. J.; Waegell, B. Prediction of the Glass Transition Temperature of Multicyclic and Bulky Substituted Acrylate and Methacrylate Polymers Using the Energy, Volume, Mass (EVM) QSPR Model. *Macromolecules* **1996**, *29*, 8954−8959.

(21) Camelio, P.; Cypcar, C. C.; Lazzeri, V.; Waegell, B. A Novel Approach toward the Prediction of the Glass Transition Temperature: Application of the EVM Model, a Designer QSPR Equation for the Prediction of Acrylate and Methacrylate Polymers. *J. Polym. Sci. Part A Polym. Chem.* **1997**, *35*, 2579−2590.

(22) Camelio, P.; Lazzeri, V.; Waegell, B.; Cypcar, C.; Mathias, L. J. Glass Transition Temperature Calculations for Styrene Derivatives Using the Energy, Volume, and Mass Model. *Macromolecules* **1998**, *31*, 2305−2311.

(23) Morrill, J. A.; Jensen, R. E.; Madison, P. H.; Chabalowski, C. F. Prediction of the Formulation Dependence of the Glass Transition Temperatures of Amine-Epoxy Copolymers Using a QSPR Based on the AM1Method. *J. Chem. Inf. Comput. Sci.* **2004**, *44*, 912−920.

(24) Yu, X.; Wang, X.; Wang, H.; Liu, A.; Zhang, C. Prediction of the Glass Transition Temperatures of Styrenic Copolymers Using a QSPR Based on the DFT Method. *J. Mol. Struct. THEOCHEM* **2006**, *766*, 113−117.

(25) Liu, A.; Wang, X.; Wang, L.; Wang, H.; Wang, H. Prediction of Dielectric Constants and Glass Transition Temperatures of Polymers by Quantitative Structure Property Relationships. *Eur. Polym. J.* **2007**, *43*, 989−995.

(26) Liu, W.; Cao, C. Artificial Neural Network Prediction of Glass Transition Temperature of Polymers. *Colloid Polym. Sci.* **2009**, *287*, 811−818.

(27) Xu, J.; Zhu, L.; Fang, D.; Liu, L.; Xu, W.; Li, Z. Prediction of Glass Transition Temperatures for Polystyrenes from Cyclic Dimer Structures Using Artificial Neural Networks. *Fibers Polym.* **2012**, *13*, 352−357.

(28) Yu, X.; Wang, X.; Li, X.; Gao, J.; Wang, H. Prediction of Glass Transition Temperatures for Polystyrenes by a Four-Descriptors QSPR Model. *Macromol. Theory Simulations* **2006**, *15*, 94−99.

(29) Yu, X.; Yi, B.; Wang, X.; Xie, Z. Correlation between the Glass Transition Temperatures and Multipole Moments for Polymers. *Chem. Phys.* **2007**, *332*, 115−118.

(30) Palomba, D.; Vazquez, G. E.; Díaz, M. F. Novel Descriptors from Main and Side Chains of High-Molecular-Weight Polymers Applied to Prediction of Glass Transition Temperatures. *J. Mol. Graph. Model.* **2012**, *38*, 137−147.

(31) Afantitis, A.; Melagraki, G.; Makridima, K.; Alexandridis, A.; Sarimveis, H.; Iglessi-Markopoulou, O. Prediction of High Weight Polymers Glass Transition Temperature Using RBF Neural Networks. *J. Mol. Struct. THEOCHEM* **2005**, *716*, 193−198.

(32) Keshavarz, M. H.; Esmaeilpour, K.; Taghizadeh, H. A New Approach for Assessment of Glass Transition Temperature of Acrylic and Methacrylic Polymers from Structure of Their Monomers without Using Any Computer Codes. *J. Therm. Anal. Calorim.* **2016**, *126*, 1787−1796.

(33) Reynolds, C. H.; Druker, R.; Pfahler, L. B. Lead Discovery Using Stochastic Cluster Analysis (SCA): A New Method for Clustering Structurally Similar Compounds. *J. Chem. Inf. Comput. Sci.* **1998**, *38*, 305−312.

(34) Reynolds, C. H. Designing Diverse and Focused Combinatorial Libraries of Synthetic Polymers. *J. Comb. Chem.* **1999**, *1*, 297−306.

(35) Wen, C.; Liu, B.; Wolfgang, J.; Long, T. E.; Odle, R.; Cheng, S. Determination of Glass Transition Temperature of Polyimides from Atomistic Molecular Dynamics Simulations and Machine-Learning Algorithms. *J. Polym. Sci.* **2020**, *58*, 1521−1534.

(36) Pilania, G.; Iverson, C. N.; Lookman, T.; Marrone, B. L. Machine-Learning-Based Predictive Modeling of Glass Transition

https://doi.org/10.1021/acsomega.2c04649
ACS Omega 2022, 7, 43678−43691

Temperatures: A Case of Polyhydroxyalkanoate Homopolymers and Copolymers. *J. Chem. Inf. Model.* **2019**, *59*, 5013−5025.
(37) Higuchi, C.; Horvath, D.; Marcou, G.; Yoshizawa, K.; Varnek, A. Prediction of the Glass-Transition Temperatures of Linear Homo/Heteropolymers and Cross-Linked Epoxy Resins. *ACS Appl. Polym. Mater.* **2019**, *1*, 1430−1442.
(38) Kim, C.; Chandrasekaran, A.; Huan, T. D.; Das, D.; Ramprasad, R. Polymer Genome: A Data-Powered Polymer Informatics Platform for Property Predictions. *J. Phys. Chem. C* **2018**, *122*, 17575−17585.
(39) Korolev, V.; Mitrofanov, A.; Korotcov, A.; Tkachenko, V. Graph Convolutional Neural Networks as "General-Purpose" Property Predictors: The Universality and Limits of Applicability. *J. Chem. Inf. Model.* **2020**, *60*, 22−28.
(40) Weiss, K.; Khoshgoftaar, T. M.; Wang, D. D. *A Survey of Transfer Learning; Journal of Big Data* **2016**, *3*, 9.
(41) Zhuang, F.; Qi, Z.; Duan, K.; Xi, D.; Zhu, Y.; Zhu, H.; Xiong, H.; He, Q. A Comprehensive Survey on Transfer Learning. *Proc. IEEE* **2021**, *109*, 43−76.
(42) Vermeire, F. H.; Green, W. H. Transfer Learning for Solvation Free Energies: From Quantum Chemistry to Experiments. *Chem. Eng. J.* **2021**, *418*, 129307.
(43) Panapitiya, G.; Girard, M.; Hollas, A.; Sepulveda, J.; Murugesan, V.; Wang, W.; Saldanha, E. Evaluation of Deep Learning Architectures for Aqueous Solubility Prediction. *ACS Omega* **2022**, *7*, 15695−15710.
(44) Roszak, R.; Beker, W.; Molga, K.; Grzybowski, B. A. Rapid and Accurate Prediction of p K a Values of C-H Acids Using Graph Convolutional Neural Networks. *J. Am. Chem. Soc.* **2019**, *141*, 17142−17149.
(45) Jha, D.; Choudhary, K.; Tavazza, F.; Liao, W.-k.; Choudhary, A.; Campbell, C.; Agrawal, A. Enhancing Materials Property Prediction by Leveraging Computational and Experimental Data Using Deep Transfer Learning. *Nat. Commun.* **2019**, *10*, 5316.
(46) Chen, C.; Ye, W.; Zuo, Y.; Zheng, C.; Ong, S. P. Graph Networks as a Universal Machine Learning Framework for Molecules and Crystals. *Chem. Mater.* **2019**, *31*, 3564−3572.
(47) Grambow, C. A.; Li, Y. P.; Green, W. H. Accurate Thermochemistry with Small Data Sets: A Bond Additivity Correction and Transfer Learning Approach. *J. Phys. Chem. A* **2019**, *123*, 5826−5835.
(48) Zhang, D.; Xia, S.; Zhang, Y. Accurate Prediction of Aqueous Free Solvation Energies Using 3D Atomic Feature-Based Graph Neural Network with Transfer Learning. *J. Chem. Inf. Model.* **2022**, *62*, 1840−1848.
(49) Ramakrishnan, R.; Dral, P. O.; Rupp, M.; Von Lilienfeld, O. A. Quantum Chemistry Structures and Properties of 134 Kilo Molecules. *Sci. Data* **2014**, *1*, 1−7.
(50) Jain, A.; Ong, S. P.; Hautier, G.; Chen, W.; Richards, W. D.; Dacek, S.; Cholia, S.; Gunter, D.; Skinner, D.; Ceder, G.; Persson, K. A. Commentary: The Materials Project: A Materials Genome Approach to Accelerating Materials Innovation. *APL Mater.* **2013**, *1*, 011002.
(51) Kirklin, S.; Saal, J. E.; Meredig, B.; Thompson, A.; Doak, J. W.; Aykol, M.; Rühl, S.; Wolverton, C. The Open Quantum Materials Database (OQMD): Assessing the Accuracy of DFT Formation Energies. *npj Comput. Mater.* **2015**, *1*, 15010.
(52) Patra, A.; Batra, R.; Chandrasekaran, A.; Kim, C.; Huan, T. D.; Ramprasad, R. A Multi-Fidelity Information-Fusion Approach to Machine Learn and Predict Polymer Bandgap. *Comput. Mater. Sci.* **2020**, *172*, 109286.
(53) Venkatram, S.; Batra, R.; Chen, L.; Kim, C.; Shelton, M.; Ramprasad, R. Predicting Crystallization Tendency of Polymers Using Multifidelity Information Fusion and Machine Learning. *J. Phys. Chem. B* **2020**, *124*, 6046−6054.
(54) St. John, P. C.; Phillips, C.; Kemper, T. W.; Wilson, A. N.; Guan, Y.; Crowley, M. F.; Nimlos, M. R.; Larsen, R. E. Message-Passing Neural Networks for High-Throughput Polymer Screening. *J. Chem. Phys.* **2019**, *150*, 234111.
(55) Lee, C. K.; Lu, C.; Yu, Y.; Sun, Q.; Hsieh, C. Y.; Zhang, S.; Liu, Q.; Shi, L. Transfer Learning with Graph Neural Networks for Optoelectronic Properties of Conjugated Oligomers. *J. Chem. Phys.* **2021**, *154*, 024906.
(56) Hasebe, T. Knowledge-Embedded Message-Passing Neural Networks: Improving Molecular Property Prediction with Human Knowledge. *ACS Omega* **2021**, *6*, 27955−27967.
(57) Zhang, S.; Tong, H.; Xu, J.; Maciejewski, R. Graph Convolutional Networks: A Comprehensive Review. *Comput. Soc. Networks* **2019**, *6*, 11.
(58) Park, C. W.; Wolverton, C. Developing an Improved Crystal Graph Convolutional Neural Network Framework for Accelerated Materials Discovery. *Phys. Rev. Mater.* **2020**, *4* (6), 63801.
(59) Xie, T.; Grossman, J. C. Crystal Graph Convolutional Neural Networks for an Accurate and Interpretable Prediction of Material Properties. *Phys. Rev. Lett.* **2018**, *120*, 145301.
(60) Wallach, I.; Dzamba, M.; Heifets, A. AtomNet: A Deep Convolutional Neural Network for Bioactivity Prediction in Structure-Based Drug Discovery. *Arxiv* **2015**, DOI: 10.48550/arXiv.1510.02855.
(61) Kearnes, S.; McCloskey, K.; Berndl, M.; Pande, V.; Riley, P. Molecular Graph Convolutions: Moving beyond Fingerprints. *J. Comput. Aided. Mol. Des.* **2016**, *30*, 595−608.
(62) Louis, S. Y.; Zhao, Y.; Nasiri, A.; Wang, X.; Song, Y.; Liu, F.; Hu, J. Graph Convolutional Neural Networks with Global Attention for Improved Materials Property Prediction. *Phys. Chem. Chem. Phys.* **2020**, *22*, 18141−18148.
(63) Park, J.; Shim, Y.; Lee, F.; Rammohan, A.; Goyal, S.; Shim, M.; Jeong, C.; Kim, D. S. Prediction and Interpretation of Polymer Properties Using the Graph Convolutional Network. *ACS Polym. Au* **2022**, *2*, 213−222.
(64) Zeng, M.; Kumar, J. N.; Zeng, Z.; Savitha, R.; Chandrasekhar, V. R.; Hippalgaonkar, K. Graph Convolutional Neural Networks for Polymers Property Prediction. *Arxiv* **2018**, DOI: 10.48550/arXiv.1811.06231.
(65) Tao, L.; Varshney, V.; Li, Y. Benchmarking Machine Learning Models for Polymer Informatics: An Example of Glass Transition Temperature. *J. Chem. Inf. Model.* **2021**, *61*, 5395−5413.
(66) Afzal, M. A. F.; Haghighatlari, M.; Ganesh, S. P.; Cheng, C.; Hachmann, J. Accelerated Discovery of High-Refractive-Index Polyimides via First-Principles Molecular Modeling, Virtual High-Throughput Screening, and Data Mining. *J. Phys. Chem. C* **2019**, *123*, 14610−14618.
(67) Ghosh, M. *Polyimides: Fundamentals and Applications*, 1st ed.; Marcel Dekker: New York, 1996.
(68) Nowruzi, F. E.; Kapoor, P.; Kolhatkar, D.; Hassanat, F. Al.; Laganiere, R.; Rebut, J. How Much Real Data Do We Actually Need: Analyzing Object Detection Performance Using Synthetic and Real Data. *Arxiv* **2019**, DOI: 10.48550/arXiv.1907.07061.
(69) Kiefer, B.; Ott, D.; Zell, A. Leveraging Synthetic Data in Object Detection on Unmanned Aerial Vehicles. *Arxiv* **2021**, DOI: 10.48550/arXiv.2112.12252.
(70) Tremblay, J.; Prakash, A.; Acuna, D.; Brophy, M.; Jampani, V.; Anil, C.; To, T.; Cameracci, E.; Boochoon, S.; Birchfield, S. Training Deep Networks with Synthetic Data: Bridging the Reality Gap by Domain Randomization. *IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Work.* **2018**, 1082−1090.
(71) Gorbachev, V.; Nikitin, A.; Basharov, I. Adversarial Learning for Effective Detector Training via Synthetic Data. *CEUR Workshop Proc.* **2020**, short16-1.
(72) Kniaz, V. V.; Knyaz, V. A.; Mizginov, V.; Papazyan, A.; Fomin, N.; Grodzitsky, L. Adversarial Dataset Augmentation Using Reinforcement Learning and 3D Modeling. In *Advances in Neural Computation, Machine Learning, and Cognitive Research IV. NEUROINFORMATICS 2020. Studies in Computational Intelligence*, 1st ed.; Kryzhanovsky, B., Dunin-Barkowski, W., Redko, V., Tiumentsev, Y., Eds; Springer: Cham, 2021.
(73) Kniaz, V. V.; Moshkantsev, P. V.; Mizginov, V. A. Deep Learning a Single Photo Voxel Model Prediction from Real and Synthetic Images. In *Advances in Neural Computation, Machine Learning, and Cognitive Research III. NEUROINFORMATICS 2019. Studies in Computational*

Intelligence, 1st ed.; Kryzhanovsky, B., Dunin-Barkowski, W., Redko, V., Tiumentsev, Y., Eds.; Springer: Cham, 2020.
(74) Kniaz, V. V.; Knyaz, V. A.; Remondino, F. The Point Where Reality Meets Fantasy: Mixed Adversarial Generators for Image Splice Detection. *Adv. Neural Inf. Process. Syst.* **2019**, *32*, 1−12.
(75) Tripathi, S.; Chandra, S.; Agrawal, A.; Tyagi, A.; Rehg, J. M.; Chari, V. Learning to Generate Synthetic Data via Compositing. *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.* **2019**, 461−470.
(76) Zhao, Y.; Kong, S.; Shin, D.; Fowlkes, C. Domain Decluttering: Simplifying Images to Mitigate Synthetic-Real Domain Shift and Improve Depth Estimation. *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.* **2020**, 3327−3337.
(77) Fox, T. G.; Flory, P. J. The Glass Temperature and Related Properties of Polystyrene. Influence of Molecular Weight. *J. Polym. Sci.* **1954**, *14*, 315−319.
(78) Barrat, J. L.; Baschnagel, J.; Lyulin, A. Molecular Dynamics Simulations of Glassy Polymers. *Soft Matter* **2010**, *6*, 3430−3446.
(79) Stutz, H.; Illers, K. -H; Mertes, J. A Generalized Theory for the Glass Transition Temperature of Crosslinked and Uncrosslinked Polymers. *J. Polym. Sci., Part B: Polym. Phys.* **1990**, *28*, 1483−1498.
(80) Reyes-Mayer, A.; Constant, A.; Romo-Uribe, A.; Jaffe, M. The Influence of Thermal Annealing on Microstructure and Mechanical Properties in High Performance Liquid Crystal Copolyesters. *Mater. Res. Soc. Symp. Proc.* **2012**, *1373*, 185−189.
(81) Mano, J. F.; Gómez Ribelles, J. L.; Alves, N. M.; Salmerón Sanchez, M. Glass Transition Dynamics and Structural Relaxation of PLLA Studied by DSC: Influence of Crystallinity. *Polymer* **2005**, *46*, 8258−8265.
(82) Mark, J. E. *Physical Properties of Polymers Handbook*, 1st ed.; Oxford University Press: New York, 1996.
(83) Abiad, M. G.; Carvajal, M. T.; Campanella, O. H. A Review on Methods and Theories to Describe the Glass Transition Phenomenon: Applications in Food and Pharmaceutical Products. *Food Eng. Rev.* **2009**, *1*, 105−132.
(84) Baljon, A. R. C.; Van Weert, M. H. M.; Degraaff, R. B.; Khare, R. Glass Transition Behavior of Polymer Films of Nanoscopic Dimensions. *Macromolecules* **2005**, *38*, 2391−2399.
(85) Lyulin, A. V.; Balabaev, N. K.; Michels, M. A. J. Molecular-Weight and Cooling-Rate Dependence of Simulated T_g for Amorphous Polystyrene. *Macromolecules* **2003**, *36*, 8574−8575.
(86) Lyulin, S. V.; Larin, S. V.; Gurtovenko, A. A.; Nazarychev, V. M.; Falkovich, S. G.; Yudin, V. E.; Svetlichnyi, V. M.; Gofman, I. V.; Lyulin, A. V. Thermal Properties of Bulk Polyimides: Insights from Computer Modeling versus Experiment. *Soft Matter* **2014**, *10*, 1224−1232.
(87) Ding, M. Isomeric Polyimides. *Prog. Polym. Sci.* **2007**, *32*, 623−668.
(88) Alentiev, A.; Yampolskii, Y.; Ryzhikh, V.; Tsarev, D. The Database "Gas Separation Properties of Glassy Polymers" (Topchiev Institute): Capabilities and Prospects. *Pet. Chem.* **2013**, *53*, 554−558.
(89) Liu, W. Prediction of Glass Transition Temperatures of Aromatic Heterocyclic Polyimides Using an ANN Model. *Polym. Eng. Sci.* **2010**, *50*, 1547−1557.
(90) Bessonov, M. I.; Koton, M. M.; Kudryavtsev, V. V.; Laius, L. A. *Polyimides - Thermally Stable Polymers*, 1st ed.; Springer Science +Business Media: New York, 1987.
(91) Weininger, D. SMILES, a Chemical Language and Information System: 1: Introduction to Methodology and Encoding Rules. *J. Chem. Inf. Comput. Sci.* **1988**, *28*, 31−36.
(92) Hachmann, J.; Afzal, M. A. F.; Haghighatlari, M.; Pal, Y. Building and Deploying a Cyberinfrastructure for the Data-Driven Design of Chemical Systems and the Exploration of Chemical Space. *Mol. Simul.* **2018**, *44*, 921−929.
(93) Gilmer, J.; Schoenholz, S. S.; Riley, P. F.; Vinyals, O.; Dahl, G. E. Neural Message Passing for Quantum Chemistry. *Arxiv* **2017**, DOI: 10.48550/arXiv.1704.01212.
(94) Cho, K.; van Merriënboer, B.; Bahdanau, D.; Bengio, Y. On the Properties of Neural Machine Translation: Encoder-Decoder Approaches. *Arxiv* **2014**, DOI: 10.48550/arXiv.1409.1259.
(95) Morris, C.; Ritzert, M.; Fey, M.; Hamilton, W. L.; Lenssen, J. E.; Rattan, G.; Grohe, M. Weisfeiler and Leman Go Neural: Higher-Order Graph Neural Networks. *Arxiv* **2018**, DOI: 10.48550/arXiv.1810.02244.
(96) Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; Berg, A. C.; Fei-Fei, L. ImageNet Large Scale Visual Recognition Challenge. *Int. J. Comput. Vis.* **2015**, *115*, 211−252.
(97) Jha, A.; Chandrasekaran, A.; Kim, C.; Ramprasad, R. Impact of Dataset Uncertainties on Machine Learning Model Predictions: The Example of Polymer Glass Transition Temperatures. *Model. Simul. Mater. Sci. Eng.* **2019**, *27*, 024002.
(98) https://pytorch.org/. (Accessed November 10, 2022)
(99) https://pytorch-geometric.readthedocs.io/en/latest/. (Accessed November 10, 2022)
(100) Shui, Z.; Karypis, G. Heterogeneous Molecular Graph Neural Networks for Predicting Molecule Properties. *Arxiv* **2020**, DOI: 10.48550/arXiv.2009.12710.
(101) Chen, C.; Zuo, Y.; Ye, W.; Li, X.; Ong, S. P. Learning Properties of Ordered and Disordered Materials from Multi-Fidelity Data. *Nat. Comput. Sci.* **2021**, *1*, 46−53.

https://doi.org/10.1021/acsomega.2c04649
ACS Omega 2022, 7, 43678−43691