Augmenting Polymer Datasets by Iterative Rearrangement

Stanley Lo[a], Martin Seifrid[a], Théophile Gaudin[b, c], Alán Aspuru-Guzik[a, b, d, e, f, g*]

[a]*Department of Chemistry, 80 St. George St., University of Toronto, Ontario M5S 3H6, Canada*

[b]*Department of Computer Science, University of Toronto, 40 St George St, Toronto, ON M5S 2E4*

[c]*IBM Research Zürich, 8803 Rüschlikon, Zürich, Switzerland*

[d]*Department of Chemical Engineering & Applied Chemistry, 200 College St., University of Toronto, Ontario M5S 3E5, Canada*

[e]*Department of Materials Science & Engineering, 184 College St., University of Toronto, Ontario M5S 3E4, Canada*

[f]*CIFAR Artificial Intelligence Research Chair, Vector Institute, Toronto, ON M5S 1M1, Canada*

[g]*Lebovic Fellow, Canadian Institute for Advanced Research (CIFAR), Toronto, ON M5S 1M1, Canada*

*Email: alan@aspuru.com

ABSTRACT: One of the biggest obstacles to successful polymer property prediction is an effective representation that accurately captures the sequence of repeat units in a polymer. Motivated by the successes of data augmentation in computer vision and natural language processing, we explore augmenting polymer data by rearranging the molecular representation while preserving the correct connectivity, revealing additional substructural information that is not present in a single representation. We evaluate the effects of this technique on the performance of machine learning models trained on three experimental polymer datasets and compare them to common molecular representations. Data augmentation improves deep learning property prediction performance compared to equivalent (non-augmented) representations. In datasets where the target property is primarily influenced by the polymer sequence rather than experimental parameters, this data augmentation technique provides the molecular embedding with more information to improve property prediction accuracy.

Keywords: data augmentation, machine learning, polymers, molecular representation

Introduction

Scientists increasingly often employ machine learning (ML) methods to accelerate discovery.[1] ML has seen many successes in polymer applications such as protein folding, polymer dielectrics, and polymer electrolytes.[2–4] However, polymers remain one of the most challenging domains for ML applications due to their relative complexity as opposed to small molecules.[5,6]

We identify two pressing problems in ML for polymers. First, the quality and quantity of data that is available is limited. Second, a molecular representation that accurately captures the stochastic and continuous nature of polymers is needed. Compared to datasets in other materials domains such as the Materials Project[7] and drug discovery,[8–11] polymer datasets are significantly smaller.[1,12,13] Additionally, establishing well-defined structure-property relationships is difficult because variations in experimental methods, processing history, and measuring conditions yield different results.[6] For instance, measuring the molecular weight of a polymer is different by nuclear magnetic resonance, gel permeation chromatography, and viscosity. Due to the high variability of results across the measurements of a single property, key experimental polymer properties such as molecular weight and dispersity are often absent in datasets, yet they are integral to the performance, self-assembly, and rheological properties of a polymer.[6,9,10–13]

An appropriate molecular representation for polymers is a non-trivial problem because polymers are repetitive, stochastic, and polydisperse. The dispersity and skewness of the molecular weight distribution influence the polymer's physical properties which illustrates the complexity of polymers,[19] and subsequently the difficulty of ML for polymers since a lot of this information is absent in datasets. However, other characteristics such as polymer topology can be represented via BigSMILES or a graph neural network.[20,21] There are many potential topologies such as block, alternating, random, and graft copolymers, which can assemble into different nanostructures.[22] By defining the frequency of each connection between the monomers of a copolymer in a graph representation, a variety of polymer topologies can be represented.[21]

Currently, there are effective but limited ways of encoding polymers. Line notations have been demonstrated to be an effective and relatively compact representation for polymers as demonstrated by Hierarchical Editing Language for Macromolecules (HELM),[23] and BigSMILES.[20] HELM represents a polymer as a hierarchy of atoms, monomers, and simple and complex polymers which is "SMILES-like" because of its computational- and human-readable format. The lower-level components (atoms, monomers) can combine to form higher-level components (simple, complex polymers).[23] HELM is popular among many pharmaceutical databases and companies because of its compact notation and versatility across several combinations of structure types.[24] BigSMILES addresses the stochasticity of a polymer and represents complex polymer topologies (random and block copolymers, and ring and branched polymers) by adding additional syntax and symbols effectively extending the conventional SMILES notation.[20,25]

Molecular fingerprints encode basic substructural information such as functional groups and connectivity but do not capture the stochastic, repetitive, or polydisperse nature of polymers.[26] Polymeric fingerprints are similar, but fragment a polymer into monomers, dimers, and trimers instead of generating fragments within a specific radius of each atom.[3,27] The success of fingerprint representations is known to depend on the task.[28] Another alternative is a fingerprint created from an oligomer with more than five repeat units which were shown to perform relatively well in property prediction tasks because they can partially represent the repeating nature of polymers.[29] Despite such advances, representing the repetitive nature of a polymer remains a long-standing problem.[29] We aim to tackle this problem with data augmentation.

Data augmentation has been extremely successful in computer vision (CV) and natural language processing (NLP)[30,31] because it maximizes the amount of information that can be learned from scarce data by performing transformations that allow ML models to ignore irrelevant variances. Empirically, models trained on augmented data are less error-prone to transformations in real (non-augmented) data.[30] SMILES representations can be augmented by generating new non-canonical SMILES from the same molecule by changing the order in which the molecular graph is traversed each time.[32,33] Augmenting SMILES improves the property prediction performance of a long short-term memory (LSTM) neural network (NN) because it can learn the grammar of SMILES notation.[32]

In this work, we present a new data augmentation technique for polymers via iterative rearrangement of fragments. This data augmentation technique can be used on any polymer because the molecular identity of the repeat unit is not restricted to a specific sequence. The only condition for this data augmentation technique is for the polymer to have a linear backbone that can be decomposed into 2 or more unique fragments. Augmenting data in this way captures the repetitive nature of polymers and reveals new substructural information which may improve model performance. First, we compare the performance of our augmented representations to that of common molecular representations with well-established ML and deep learning (DL) models across three different datasets. These datasets cover different structure-property relationships in polymers: CO_2 Solubility,[34] Pervaporation Separation,[35] and Swelling.[36] To this end, we built a rigorous ML workflow which incorporates stratified cross-validation, and automated hyperparameter optimization. Next, we evaluate trends in the performance of the augmented representations across the three datasets. Finally, we compare the accuracy of our models against the results reported in the literature.

Results and Discussion

Augmenting Polymer Representations

Augmenting polymer data by rearranging the fragments of a polymer backbone while preserving correct chemical connectivity could expose an ML model to previously unseen substructures (**Figure 1**), showing that the absolute position of a fragment is irrelevant to the properties of the polymer. Additionally, fragments need not correspond to the monomer but could be monomer substructures or be spread across co-monomers. As mentioned before, the previously unseen substructures along a polymer backbone could also be captured by a sufficiently long oligomer.[29] However, the length of an oligomer sequence is an additional feature that the ML model will implicitly learn when presented with an oligomeric polymer representation which could confuse the model. On the other hand, augmenting polymer representations by iterative rearrangement yields representations of uniform size while providing additional information.

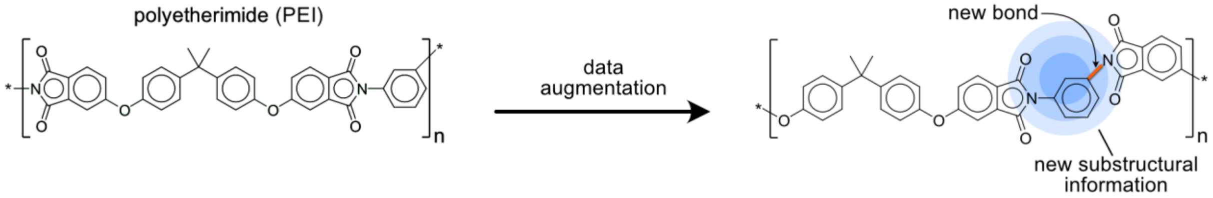

Figure 1. Advantages of the data augmentation technique demonstrated on a polyetherimide. After data augmentation, hidden substructural information is revealed, indicating that the absolute position of fragments can be ignored.

Encoding augmented polymer representations manually would be a tedious process. Unfortunately, no molecular fragmentation tools are available that could automatically fragment polymer backbones. We created a semi-automated workflow for augmenting polymer representations that is available as a GitHub repository.[37] We define the backbone as a linear chain of fragments that results in the simplest representation, indicated by the green bonds in **Figure 2a**.[38] Automatically identifying ideal backbone fragmentation patterns is very challenging because this task requires a well-defined set of rules. However, sidechains or other functional groups may contain the same substructures. The user can manually determine the bonds to be broken. The resulting fragments are then numbered (**Figure 2b**), iteratively re-ordered, and one-hot encoded (**Figure 2c**). This rearrangement preserves the polymer sequence while generating new encodings thereof.

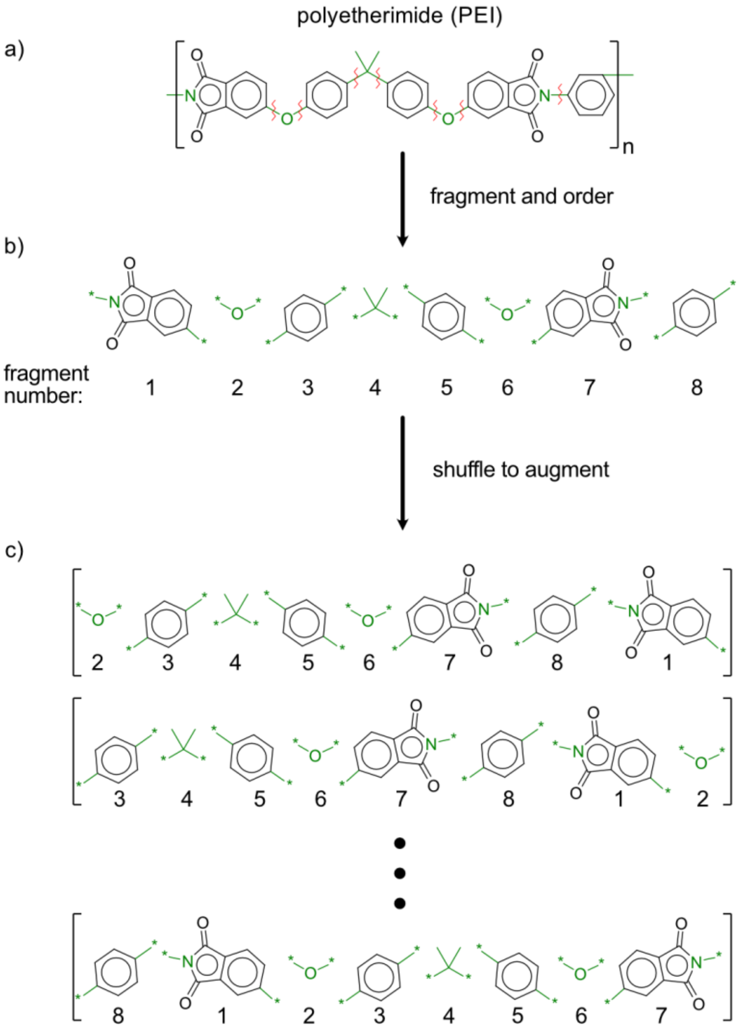

Figure 2*. Illustration of data augmentation for a polyetherimide. a) The polymer is fragmented along the backbone. b) The fragments are assigned numbers. c) The fragments are iteratively re-ordered such that the polymer sequence is preserved.*

Augmenting with one-hot encoded fragments can effectively reveal that the absolute position of fragments is not relevant to structure-property relationships. However, this approach cannot reveal previously unseen substructures. Fingerprints are a robust method of segmenting a molecule because they are order-invariant and generate substructures from each atom of the molecule within a specific radius while incorporating atomic features such as the number of "heavy" atoms, the valence minus the number of hydrogens, atomic number, atomic mass, atomic charge, the number of implicit and explicit hydrogens, and whether the atom is in a ring.[27] However, as discussed earlier, fingerprints generated from a single representation may miss important substructural information. We recombined the augmented fragments into single structures and generated the corresponding fingerprints to create a representation that benefits from the advantages of both iterative rearrangement and additional substructural information from fingerprints.

An alternative to the fragment-based method discussed above is to encode the augmented fragments and the recombined augmented fragments as SMILES. The advantage is that the augmented fragments are no longer represented as arbitrarily related numbers but rather as SMILES which have some structural information. However, the disadvantage is that the model must learn the grammar and syntax of SMILES in addition to the structure-property relationships. For this reason, it is important to consider the data augmentation of SMILES and the choice of an appropriate model which can effectively learn sequential information such as an LSTM.[39,40]

Model-Representation Pair Performance Within Datasets

Below we compare different models (support vector machines (SVM), random forest (RF), gradient-boosted tree (XGBoost), NN, and LSTM) and representations (SMILES, SELFIES, BigSMILES, BRICS, fragments, extended-connectivity fingerprints (ECFP_6), augmented SMILES, augmented fragments, and recombined augmented fragments) as they pertain to different experimental polymer datasets.

CO_2 Solubility

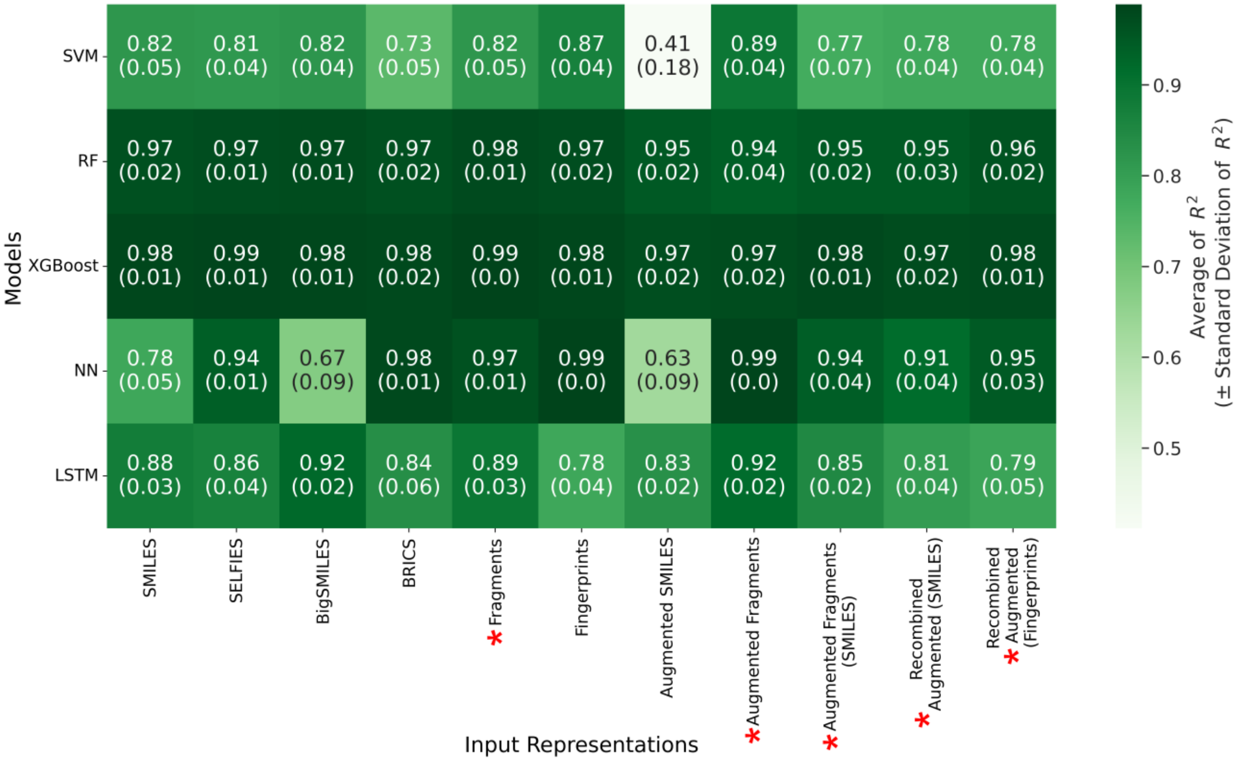

Figure 3*. Heatmap of model-representation pair performance for the CO_2 Solubility dataset. New representations presented in this work are marked by red asterisks.*

The CO_2 Solubility dataset contains 13 different polymers common to gas solubility experiments (**Table S1**).[34] The two continuous variables, temperature and pressure, have a wide distribution of values (**Figure S1**), which is suitable for ML because the data is sufficiently diverse for the models to extract information from it. Because temperature and pressure are strongly correlated with the solubility of CO_2 independent of polymer structure (**Figure S2**), all model-representation pairs are relatively accurate (most $R^2 > 0.7$). However, this suggests that the molecular representation will have minimal impact on the prediction of CO_2 solubility.

RF and XGBoost methods consistently perform very well for the different representations (R^2 = 0.94-0.99). RF and XGBoost perform better than other models because tree-based methods are robust models for capturing relevant features while ignoring other ones.[41] DL methods such as NNs and LSTMs are more sensitive to molecular representation. As opposed to tree-based models, uninformative features can have adverse effects on DL model output.[41] For example, the SMILES and BigSMILES representations are uninformative features for the NN model because it requires the NN to learn the grammar and syntax of these representations. NNs typically do not perform well on sequential data because of their inherent architecture and lack of temporal encoding, unlike recurrent NNs or LSTMs.[40,42]

Pervaporation

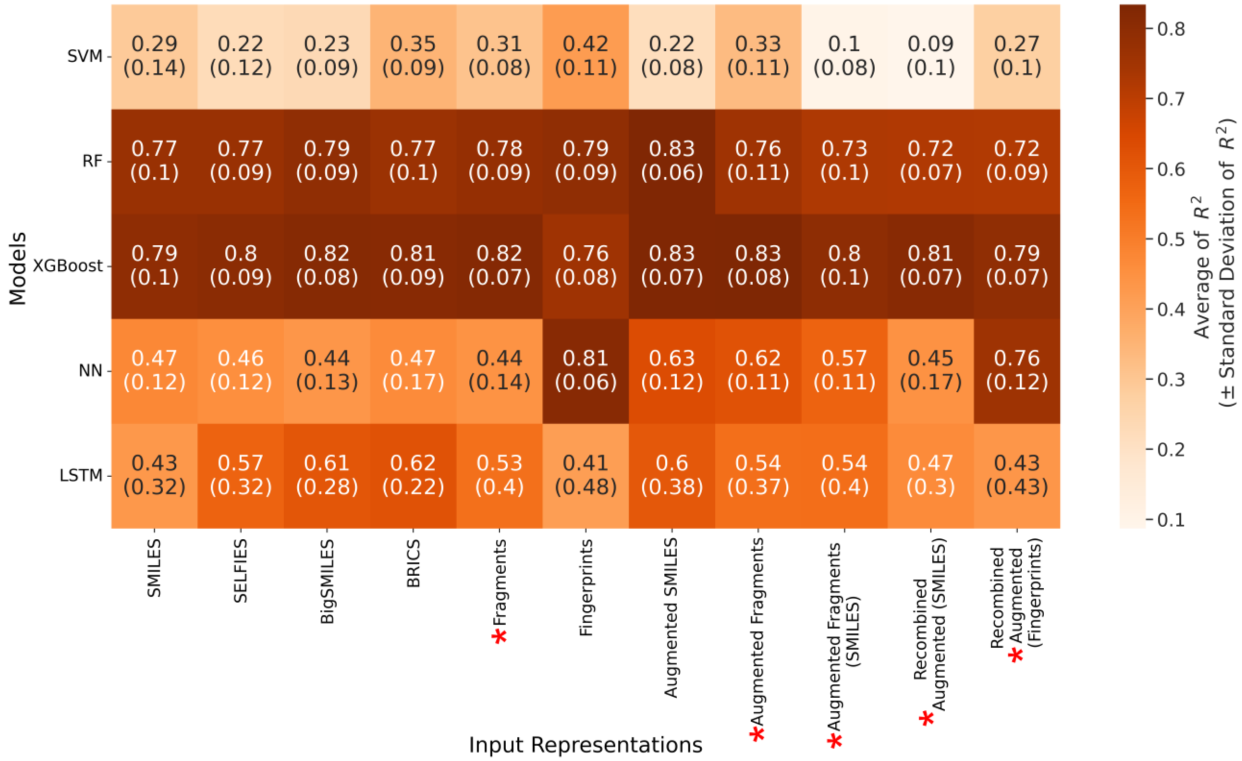

__Figure 4__. Heatmap of model-representation pair performance for the Pervaporation dataset. New representations presented in this work are marked by red asterisks.

The Pervaporation dataset contains 16 polymers used in pervaporation separation membranes (**Table S2**) and 6 solvents (**Table S3**).[35] Four polymers – poly(vinyl acetate), alginate, chitosan, and poly(dimethyl siloxane) – constitute 73.5% of the data points, which could limit model generalizability to other polymers. Many of the continuous input features (**Figure S5a**) such as temperature, contact angle, and weight percent of water on the feed side (xw_wt_%) have a concentration of data points around a few values. However, values for temperature and weight percent of water on the feed side are clustered around specific numbers (e.g. multiples of 5 and 10) to a statistically unlikely degree, suggesting that the true numbers may have been rounded when reported. The solvent solubility parameter values are unevenly distributed because there are only 6 solvents in the dataset. The output variable (total flux, J) is unimodal, smooth, and heavily skewed toward small values (**Figure S5b**). Relatively low correlations between the gross descriptors and total flux suggest that ML is appropriate for structure-property predictions (**Figure S6**).[43]

RF and XGBoost perform well across most of the molecular representations, similar to the CO_2 Solubility dataset. NNs perform worse than RF and XGBoost on property prediction tasks with a non-smooth feature space, as seen by the uneven distribution of input variables (**Figure S5**), because a tree-based method can segment the feature space which makes it easy to learn piece-wise functions.[41] The lack of effect from the changes in molecular representation suggests that RF and XGBoost rely on the gross descriptors rather than the molecular representation to make predictions. As seen in the previous dataset, NN and LSTM are more sensitive to molecular representation. For the same reason that a poor representation will negatively affect the DL model output, a good molecular representation will have positive effects on the model output which is shown by the fingerprint representation on the NN model.

Swelling

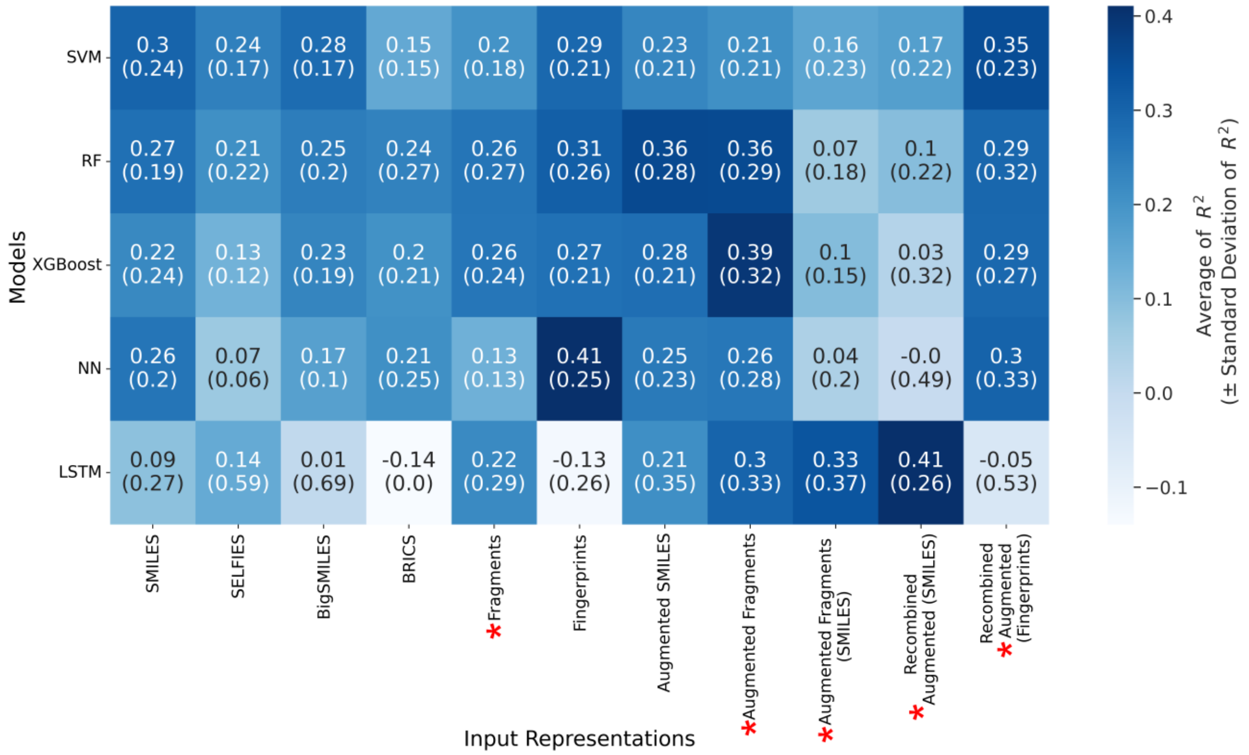

Figure 5. *Heatmap of model-representation pair performance for the Swelling dataset. New representations presented in this work are marked by red asterisks.*

The Swelling dataset contains 9 different polymers (**Table S4**) and 9 solvents (**Table S5**).[36] The distribution of swelling degree (SD) is unimodal and skewed towards an SD of 0-2% (**Figure S9**). The poor performance across all the models and representations can be explained by the limited diversity and size of the dataset (only 77 data points).

One of the most notable differences between the Swelling dataset and the other two datasets is the lack of consistent performance across the molecular representations. Without other input variables, RF and XGBoost models are dependent on molecular representation for property prediction. Augmented fragments perform best for RF and XGBoost which suggests that it can be an appropriate molecular representation, leveraging the advantages of our data augmentation technique. However, the results from augmented fragments with RF and XGBoost have considerably higher standard deviation and fall within one standard deviation of other representations which make it difficult to conclude the performance improvement. LSTMs perform poorly on order-invariant representations such as BRICS and Fingerprints because LSTMs process information sequentially where the order is important to consider and they have limited memory for long-term information throughout the representation.[39] The standard deviation of performance across training folds is large, suggesting that performance is highly dependent on the specific fold since there is such little data. The overall poor performance and high standard deviation can be attributed to the lack of data.

General Trends Across Datasets

Data augmentation can slightly decrease performance because it exaggerates bias within a dataset. Since property prediction in the CO_2 Solubility and Pervaporation datasets is dependent on the input variables rather than the molecular representation, data augmentation of the molecular representation has direct negative effects. This data augmentation technique introduces data bias because the number of augmented data points depends on the length of the polymer and the diversity of fragments in the polymer. More data points can be created for polymers composed of a larger number of fragments (i.e., more possible rearrangements). Consequently, the original distribution of data will be biased towards data points with polymers that yield more fragments (**Figure S3, S4, S7, S8, S10, S11**), resulting in overfitting. With the Swelling dataset, the adverse effects are more pronounced. Since the dataset is small, the original distribution will be shifted relatively more than other datasets with larger amounts of data (**Figure S10, S11**). Furthermore, SVMs perform poorly because they are limited by data imbalance and this data augmentation technique exaggerates this effect.[44]

Across all the datasets, it is apparent that LSTMs perform poorly with order-invariant representations because they are designed for temporal sequence data.[45] While LSTMs perform well with line notations (SMILES, BigSMILES, SELFIES, and augmented SMILES), they perform poorly on fingerprints and BRICS because the position of bits (fingerprints) or fragments (BRICS) is irrelevant. In contrast, the position of characters in SMILES is integral to encoding the molecule’s structure.[46]

Fingerprint representations consistently perform best across the three experimental datasets (**Figure 3-6**) because they are order-invariant and can capture substructures of different sizes.[27,46] The success of ECFP can be attributed to the extraction of relevant molecular substructures, and the inclusion of molecular features as mentioned before.[27,46] These molecular features are not included in the non-fingerprint representations used in this study which provides ECFP with more information about the molecular structure. Other representations must rely on the ML model to implicitly extract this information from the structure which is non-trivial.

Impact of Data Augmentation on ML Model Performance

NN models are most likely to benefit from data augmentation. The other models will not benefit as much because the molecular representation has little effect on RF and XGBoost models, and SVMs perform poorly on imbalanced datasets. NNs can take advantage of this data augmentation technique by learning to ignore the absolute position of fragments and capturing previously unseen structural information.

Across the three datasets we examined, data augmentation via iterative rearrangement improves the performance of NN models slightly (**Figure 6**). This improvement is most pronounced in the CO_2 Solubility dataset, where performance is already high, while differences between conventional and augmented fragments are within one standard deviation for the Pervaporation and Swelling datasets. This is reasonable because a DL model will improve with a smoother feature space.[41] The feature space is smoother from data augmentation because the molecular representation slightly changes while mapping to the same output which means that each data point occupies a larger area in the feature space. The augmented fragments representation improves model performance because it ignores irrelevant features, captures the repetitive nature of the polymer structure, and offers new substructural information.

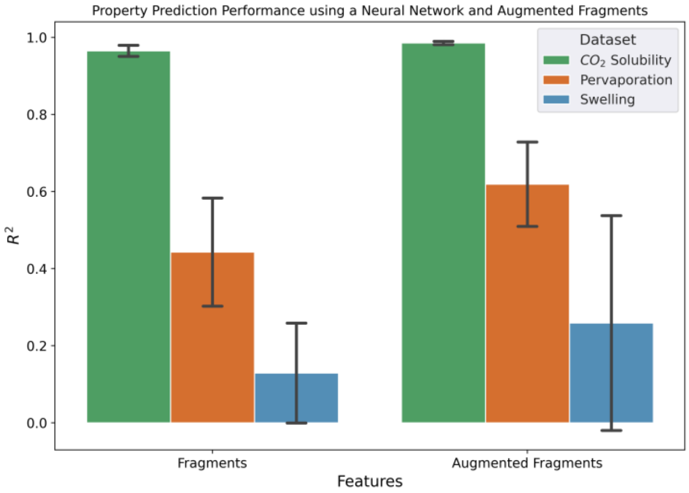

***Figure 6**. Comparison of the performance of neural network models for the CO_2 solubility, pervaporation, and swelling datasets with and without data augmentation. Error bars represent the standard deviation of R^2.*

As one-hot encoded fragments have limited generalizability,[26] recombining the augmented fragments into a molecular structure makes them more generalizable. To leverage the advantages of both fingerprints and data augmentation, the augmented fragments are recombined into a molecular structure (**Figure 2**) from which fingerprint representations could be generated. The resulting fingerprint contains new substructural information because the iterative rearrangement creates new bonds that were not previously present.

However, recombined augmented fingerprints do not outperform conventional fingerprint representations. The NN models perform equivalently on the Pervaporation and Swelling datasets, while their performance decreases on the CO_2 Solubility dataset. The lack of improvement could likely be attributed to the fact that the extra substructural information generated by augmenting fingerprints is not significant enough to yield better model performance. For example, a change in one bit out of 1024 may not help the model learn something new.

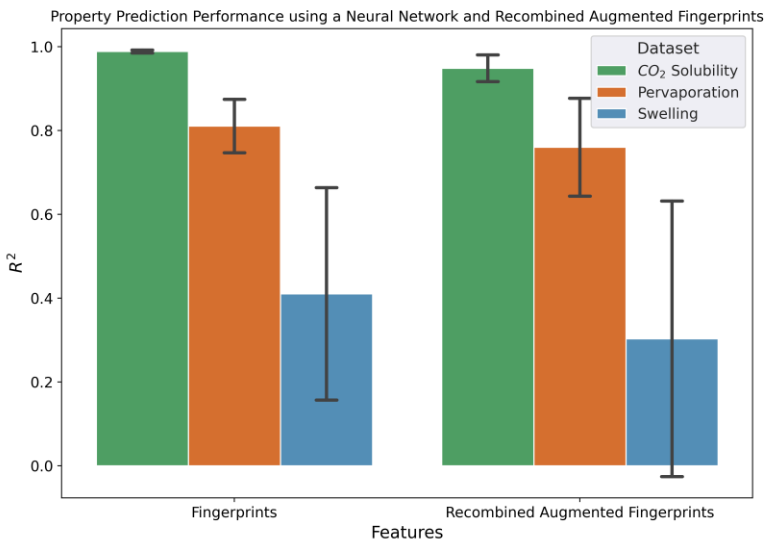

***Figure** 7. Comparison of property prediction performance between the fingerprints and recombined augmented fingerprints representation for the CO_2 solubility, pervaporation, and swelling datasets using the neural network model. Error bars represent the standard deviation of R^2.*

Comparing to the Reported Model Performance

ML model performance was rigorously evaluated using stratified cross-validation which avoids inflated performance by preventing the biased selection of easy data points in the out-of-distribution test set. Here, we compare the performance of our best models for each dataset to those reported in the literature.

CO_2 Solubility

Soleimani *et al.* trained a new model for each polymer using its corresponding data.[34] Therefore, the resulting models cannot generalize beyond the specific polymer used for training. For the sake of comparison, we calculate a global R^2 value across all the model predictions for each polymer. Training a single model using the aggregated data from each polymer makes our models more generalizable and is more challenging because the model cannot overfit to a specific polymer. We found that our best model (NN trained on fingerprints) performs significantly better than the reported models.

Pervaporation

Wang *et al.* evaluate their models via 50 random train/test splits (80%/20%), precluding a direct comparison with our models. Nevertheless, the approximate model performance from the 50 splits reported by Wang *et al.* is within a single standard deviation of our best model. Additionally, we do not know the exact variance of their models' performance because the results from the 50 splits could only be estimated from a figure.

Swelling

Xu *et al.* use several different sampling methods: 500 random train/test (90%/10%) splits, leave-one-out splitting, and evaluation on a single split. These splitting techniques offer better results than stratified cross-validation (~80%/~20%) splits because there is more training data and less testing data. In addition, random split is not a reliable sampling strategy for evaluating a model's performance because data in one split may be easier than in another. The reported models perform significantly better than the models in this study which suggests that the task is highly dependent on the specific model and input representation configurations. Additionally, we were only able to reproduce the results using a sum of fragments-based kernel ridge regression (KRR) model because sufficient details were not provided to fully reproduce the reported methods.

Table 1. Comparison of the best-performing model-representation pairs against reported results.

Dataset	**Model**		**Input Representation**	**R^2 (± S.D.)**
CO_2 Solubility	Reference[34]	Gradient Boosting Regression	Pressure and Temperature	0.9754†
	This work	NN	Fingerprints	0.9886 (± 0.0035)
Pervaporation	Reference[35]	Gradient Boosting Regression	Sum of Fragments (custom fingerprint)	ca. 0.90† (ca. ± 0.07)
	This work	XGBoost	Augmented Manual Fragments	0.8339 (± 0.0820)
Swelling	Reference[36]	Kernel Ridge Regression (Laplacian)	Sum of Fragments (custom fingerprint)	OSN: 0.91† PDMS: 0.94†
	This work	NN	Fingerprint	0.4103 (± 0.2534)

† – Authors did not report results from cross-validation.

OSN: organic solvent nanofiltration membrane, PDMS: polydimethylsiloxane.

Summary and Future Work

Augmenting polymer data via iterative rearrangement of molecular representations achieves comparable performance to that of the best previously reported representations. In polymer datasets that are heavily influenced by structure-property relationships, data augmentation provides a better molecular representation by extracting more structural information.

Polymer data augmentation via iterative rearrangement improves the predictive performance of NNs trained on the augmented fragments representation because DL models require more data for a smoother feature space. Data augmentation increases the amount of data which is a blessing and a curse. With more data, more information can be extracted but it requires more computational resources for training. Despite the costs associated with data augmentation, it is negligible compared to acquiring more data from experiments or computer simulations.

Data augmentation can increase the amount of data from reasonable manipulations which provides ML models with more information. Data augmentation in ML for chemistry is uncommon with only a few examples such as randomized SMILES,[32,47] and crystal structure augmentation.[48] The challenge with data augmentation is building informative manipulations for the relevant chemical information. Unlike visual data which has many strategies for augmentation,[49] generating multiple novel molecular representations is difficult because molecules can be naturally represented as graphs which are rotation, translation, and permutation invariant.[50–52] Graph neural networks could potentially benefit from the new substructural information generated by augmentation via iterative rearrangement. In experimental domains where data is scarce and costly, data augmentation extracts more information from the available data which should be an important consideration for ML workflows.

Methods

Datasets

The datasets are derived from experimentally measured polymer properties and were collected from the supporting information of the original work.

CO_2 Solubility

The CO_2 Solubility dataset has 515 data points, 13 unique polymers, and 3 input features: molecular representation, pressure (mPa), and temperature (K) in a sorption cell for a pressure-decay experiment.[34,53] The target property is CO_2 solubility in polymers (g of CO_2 dissolved / g of polymer).

Pervaporation

The Pervaporation dataset has 681 data points, 16 unique polymers, 6 solvents, and 7 input features: molecular representation, and the gross descriptors: weight percent of the water on the feed side (%), operating temperature (^oC), pressure on the permeate side (mbar), membrane thickness (μm), water contact angle of the membrane (°), and the Hildebrand solubility parameter ($Mpa^{1/2}$).[35] The target property is the total flux of permeation (kg/m^2h).

Swelling

The Swelling dataset has 77 data points, 9 unique polymers, 9 solvents, and 1 input feature: molecular representation.[36] The target property is the degree of swelling (*l* %).

Machine Learning

Data Augmentation

Recombining Augmented Fragments into Recombined Augmented Fingerprints

The polymer is segmented into fragments along its backbone by breaking single bonds (**Figure S12a**). The two atoms previously connected by a broken bond are labelled with an arbitrary element that is absent from the dataset in ascending atomic number (**Figure S12b**). The end groups are assigned arbitrary elements in the same manner. The order of the backbone is iterated by rearranging the first fragment to the last fragment in the sequence (**Figure S12c**). The polymer backbone is reconstituted by applying reaction SMARTS on the first fragment and the next fragment in the sequence which reacts the arbitrary elements together to form a single bond (C-C, C-N, or C-O). The reaction SMARTS is applied to the resulting fragment and the next fragment until the whole molecule is reformed. This ensures that the relative order of the polymer is preserved (**Figure S12d**). Only the unique SMILES of the recombined polymers are kept to generate new fingerprints. It is important to note that the recombined fragments form many symmetrical structures which were eliminated, hence a reduction in the number of data points for the recombined augmented fragments.

Machine Learning Workflow

The dataset is pre-processed from the SMILES representation into the other representations, and the feature and target properties are scaled using min-max scaling. Before training, the dataset is split using stratified cross-validation using an 80% train, 20% test split. Hyperparameters of the scikit-learn[54] models (SVM, RF, and XGBoost[55]) are optimized using 5-fold cross-validation from the training data to create a ~64% training fold and a ~16% validation fold. The hyperparameters of the PyTorch models (NN, and LSTM)[56] were initially hand-tuned, but the networks were not sensitive to any further hyperparameter optimization. The PyTorch models were run for 100 epochs to ensure that the models reach convergence. To ensure a fair comparison, the training, validation, and test folds of the PyTorch models were configured in the same manner as the scikit-learn models which have ~64% of the data for training, ~16% for validation, and ~20% for test folds. After training, model performance is evaluated based on the prediction accuracy for the test set. Note that the test set does not contain augmented data when evaluating the data augmentation technique.

Molecular Representation and Model Evaluation

There are numerous molecular representations and ML models to choose from, and the combination's performance can be task-dependent.[26,57] We have included some of the most common ML (SVM, RF, XGBoost) and DL (NN, LSTM) models to evaluate the impact of polymer representation on model performance. To understand the impact of data augmentation, we compared canonical SMILES,[46] augmented SMILES,[32,47] one-hot encoded molecular fragments (both augmented and recombined augmented), and other common representations (BigSMILES,[20] SELFIES,[58] BRICS,[59] ECFP6).[27]

Conflicts of Interest

A.A.-G. is the chief visionary officer and a board member of Kebotix, Inc., a company that carries out closed-loop molecular materials discovery.

Acknowledgement

The authors thank Dr. Kjell Jorner, Prof. Connor Coley, and Prof. Helen Tran for helpful discussions. S. Lo acknowledges the support from the Natural Sciences and Engineering Research Council of Canada (NSERC) through the Undergraduate Student Research Award (Application ID: 561994), and funding from the National Research Council Canada (NRC) under the "CHEMOS: a software for self-driving laboratories" project (Fund number: 509300). A. Aspuru-Guzik thanks Anders G. Frøseth for his generous support. A. Aspuru-Guzik also acknowledges the generous support of Natural Resources Canada and the Canada 150 Research Chairs program.

Code and Data Availability

The code and data required to reproduce these findings are available at https://github.com/Stanlo229/da_for_polymers.

References

(1) Sha, W.; Li, Y.; Tang, S.; Tian, J.; Zhao, Y.; Guo, Y.; Zhang, W.; Zhang, X.; Lu, S.; Cao, Y.-C.; Cheng, S. Machine Learning in Polymer Informatics. *InfoMat* **2021**, *3* (4), 353–361. https://doi.org/10.1002/inf2.12167.

(2) 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* (7873), 583–589. https://doi.org/10.1038/s41586-021-03819-2.

(3) Mannodi-Kanakkithodi, A.; Pilania, G.; Huan, T. D.; Lookman, T.; Ramprasad, R. Machine Learning Strategy for Accelerated Design of Polymer Dielectrics. *Sci. Rep.* **2016**, *6* (1), 20952. https://doi.org/10.1038/srep20952.

(4) Wang, Y.; Xie, T.; France-Lanord, A.; Berkley, A.; Johnson, J. A.; Shao-Horn, Y.; Grossman, J. C. Toward Designing Highly Conductive Polymer Electrolytes by Machine Learning Assisted Coarse-Grained Molecular Dynamics. *Chem. Mater.* **2020**, *32* (10), 4144–4151. https://doi.org/10.1021/acs.chemmater.9b04830.

(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* (1), 163. https://doi.org/10.3390/polym12010163.

(6) Audus, D. J.; de Pablo, J. J. Polymer Informatics: Opportunities and Challenges. *ACS Macro Lett.* **2017**, *6* (10), 1078–1082. https://doi.org/10.1021/acsmacrolett.7b00228.

(7) 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* (1), 011002. https://doi.org/10.1063/1.4812323.

(8) Richard, A. M.; Judson, R. S.; Houck, K. A.; Grulke, C. M.; Volarath, P.; Thillainadarajah, I.; Yang, C.; Rathman, J.; Martin, M. T.; Wambaugh, J. F.; Knudsen, T. B.; Kancherla, J.; Mansouri, K.; Patlewicz, G.; Williams, A. J.; Little, S. B.; Crofton, K. M.; Thomas, R. S. ToxCast Chemical Landscape: Paving the Road to 21st Century Toxicology. *Chem. Res. Toxicol.* **2016**, *29* (8), 1225–1251. https://doi.org/10.1021/acs.chemrestox.6b00135.

(9) Kim, S.; Chen, J.; Cheng, T.; Gindulyte, A.; He, J.; He, S.; Li, Q.; Shoemaker, B. A.; Thiessen, P. A.; Yu, B.; Zaslavsky, L.; Zhang, J.; Bolton, E. E. PubChem 2019 Update: Improved Access to Chemical Data. *Nucleic Acids Res.* **2019**, *47* (D1), D1102–D1109. https://doi.org/10.1093/nar/gky1033.

(10) Kuhn, M.; Letunic, I.; Jensen, L. J.; Bork, P. The SIDER Database of Drugs and Side Effects. *Nucleic Acids Res.* **2016**, *44* (D1), D1075-1079. https://doi.org/10.1093/nar/gkv1075.

(11) Wu, Z.; Ramsundar, B.; Feinberg, E. N.; Gomes, J.; Geniesse, C.; Pappu, A. S.; Leswing, K.; Pande, V. MoleculeNet: A Benchmark for Molecular Machine Learning. *Chem. Sci.* **2018**, *9* (2), 513–530. https://doi.org/10.1039/C7SC02664A.

(12) Nakata, M.; Shimazaki, T.; Hashimoto, M.; Maeda, T. PubChemQC PM6: Data Sets of 221 Million Molecules with Optimized Molecular Geometries and Electronic Properties. *J. Chem. Inf. Model.* **2020**, *60* (12), 5891–5899. https://doi.org/10.1021/acs.jcim.0c00740.

(13) Otsuka, S.; Kuwajima, I.; Hosoya, J.; Xu, Y.; Yamazaki, M. PoLyInfo: Polymer Database for Polymeric Materials Design. In *2011 International Conference on Emerging Intelligent Data and Web Technologies*; IEEE: Tirana, Albania, 2011; pp 22–29. https://doi.org/10.1109/EIDWT.2011.13.

(14) Whitfield, R.; P. Truong, N.; Messmer, D.; Parkatzidis, K.; Rolland, M.; Anastasaki, A. Tailoring Polymer Dispersity and Shape of Molecular Weight Distributions: Methods and Applications. *Chem. Sci.* **2019**, *10* (38), 8724–8734. https://doi.org/10.1039/C9SC03546J.

(15) Gentekos, D. T.; Sifri, R. J.; Fors, B. P. Controlling Polymer Properties through the Shape of the Molecular-Weight Distribution. *Nat. Rev. Mater.* **2019**, *4* (12), 761–774. https://doi.org/10.1038/s41578-019-0138-8.

(16) Rosenbloom, S. I.; Fors, B. P. Shifting Boundaries: Controlling Molecular Weight Distribution Shape for Mechanically Enhanced Thermoplastic Elastomers. *Macromolecules* **2020**, *53* (17), 7479–7486. https://doi.org/10.1021/acs.macromol.0c00954.

(17) Gentekos, D. T.; Jia, J.; Tirado, E. S.; Barteau, K. P.; Smilgies, D.-M.; DiStasio, R. A. Jr.; Fors, B. P. Exploiting Molecular Weight Distribution Shape to Tune Domain Spacing in Block Copolymer Thin Films. *J. Am. Chem. Soc.* **2018**, *140* (13), 4639–4648. https://doi.org/10.1021/jacs.8b00694.

(18) Chatzigiannakis, E.; Vermant, J. Dynamic Stabilisation during the Drainage of Thin Film Polymer Solutions. *Soft Matter* **2021**, *17* (18), 4790–4803. https://doi.org/10.1039/D1SM00244A.

(19) Gentekos, D. T.; Dupuis, L. N.; Fors, B. P. Beyond Dispersity: Deterministic Control of Polymer Molecular Weight Distribution. *J. Am. Chem. Soc.* **2016**, *138* (6), 1848–1851. https://doi.org/10.1021/jacs.5b13565.

(20) 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 Cent. Sci.* **2019**, *5* (9), 1523–1531. https://doi.org/10.1021/acscentsci.9b00476.

(21) Aldeghi, M.; Coley, C. W. A Graph Representation of Molecular Ensembles for Polymer Property Prediction. 29.

(22) *Electron Microscopy of Polymers*; Springer Berlin Heidelberg: Berlin, Heidelberg, 2008. https://doi.org/10.1007/978-3-540-36352-1.

(23) Zhang, T.; Li, H.; Xi, H.; Stanton, R. V.; Rotstein, S. H. HELM: A Hierarchical Notation Language for Complex Biomolecule Structure Representation. *J. Chem. Inf. Model.* **2012**, *52* (10), 2796–2806. https://doi.org/10.1021/ci3001925.

(24) David, L.; Thakkar, A.; Mercado, R.; Engkvist, O. Molecular Representations in AI-Driven Drug Discovery: A Review and Practical Guide. *J. Cheminformatics* **2020**, *12* (1), 56. https://doi.org/10.1186/s13321-020-00460-5.

(25) Lin, T.-S.; Rebello, N. J.; Lee, G.-H.; Morris, M. A.; Olsen, B. D. Canonicalizing BigSMILES for Polymers with Defined Backbones. *ACS Polym. Au* **2022**, acspolymersau.2c00009. https://doi.org/10.1021/acspolymersau.2c00009.

(26) Pattanaik, L.; Coley, C. W. Molecular Representation: Going Long on Fingerprints. *Chem* **2020**, *6* (6), 1204–1207. https://doi.org/10.1016/j.chempr.2020.05.002.

(27) Rogers, D.; Hahn, M. Extended-Connectivity Fingerprints. *J. Chem. Inf. Model.* **2010**, *50* (5), 742–754. https://doi.org/10.1021/ci100050t.

(28) Ramprasad, R.; Batra, R.; Pilania, G.; Mannodi-Kanakkithodi, A.; Kim, C. Machine Learning in Materials Informatics: Recent Applications and Prospects. *Npj Comput. Mater.* **2017**, *3* (1), 54. https://doi.org/10.1038/s41524-017-0056-5.

(29) Chen, L.; Pilania, G.; Batra, R.; Huan, T. D.; Kim, C.; Kuenneth, C.; Ramprasad, R. Polymer Informatics: Current Status and Critical next Steps. *Mater. Sci. Eng. R Rep.* **2021**, *144*, 100595. https://doi.org/10.1016/j.mser.2020.100595.

(30) Khosla, C.; Saini, B. S. Enhancing Performance of Deep Learning Models with Different Data Augmentation Techniques: A Survey. In *2020 International Conference on Intelligent Engineering and Management (ICIEM)*; IEEE: London, United Kingdom, 2020; pp 79–85. https://doi.org/10.1109/ICIEM48762.2020.9160048.

(31) Liu, P.; Wang, X.; Xiang, C.; Meng, W. A Survey of Text Data Augmentation. In *2020 International Conference on Computer Communication and Network Security (CCNS)*; IEEE: Xi'an, China, 2020; pp 191–195. https://doi.org/10.1109/CCNS50731.2020.00049.

(32) Bjerrum, E. J. SMILES Enumeration as Data Augmentation for Neural Network Modeling of Molecules. arXiv May 17, 2017. http://arxiv.org/abs/1703.07076 (accessed 2022-08-12).

(33) Vogt, M.; de la Vega de Leon, A.; Bajorath, J. Algorithmic Chemoinformatics. In *Tutorials in Chemoinformatics*; John Wiley & Sons, Ltd, 2017; pp 393–448. https://doi.org/10.1002/9781119161110.ch24.

(34) Soleimani, R.; Saeedi Dehaghani, A. H.; Rezai-Yazdi, A.; Hosseini, S. A.; Hosseini, S. P.; Bahadori, A. Evolving an Accurate Decision Tree-Based Model for Predicting Carbon Dioxide Solubility in Polymers. *Chem. Eng. Technol.* **2020**, *43* (3), 514–522. https://doi.org/10.1002/ceat.201900096.

(35) Wang, M.; Xu, Q.; Tang, H.; Jiang, J. Machine Learning-Enabled Prediction and High-Throughput Screening of Polymer Membranes for Pervaporation Separation. *ACS Appl. Mater. Interfaces* **2022**, acsami.1c22886. https://doi.org/10.1021/acsami.1c22886.

(36) Xu, Q.; Jiang, J. Machine Learning for Polymer Swelling in Liquids. *ACS Appl. Polym. Mater.* **2020**, *2* (8), 3576–3586. https://doi.org/10.1021/acsapm.0c00586.

(37) Lo, S. Stanlo229/Da_for_polymer: Preprint Release, 2022. https://doi.org/10.5281/ZENODO.7362799.

(38) *The IUPAC Compendium of Chemical Terminology: The Gold Book*, 4th ed.; Gold, V., Ed.; International Union of Pure and Applied Chemistry (IUPAC): Research Triangle Park, NC, 2019. https://doi.org/10.1351/goldbook.

(39) Yu, Y.; Si, X.; Hu, C.; Zhang, J. A Review of Recurrent Neural Networks: LSTM Cells and Network Architectures. *Neural Comput.* **2019**, *31* (7), 1235–1270. https://doi.org/10.1162/neco_a_01199.

(40) Flam-Shepherd, D.; Zhu, K.; Aspuru-Guzik, A. Language Models Can Learn Complex Molecular Distributions. *Nat. Commun.* **2022**, *13* (1), 3293. https://doi.org/10.1038/s41467-022-30839-x.

(41) Grinsztajn, L.; Oyallon, E.; Varoquaux, G. Why Do Tree-Based Models Still Outperform Deep Learning on Tabular Data? 33.

(42) Sundermeyer, M.; Ney, H.; Schlüter, R. From Feedforward to Recurrent LSTM Neural Networks for Language Modeling. *IEEEACM Trans. Audio Speech Lang. Process.* **2015**, *23* (3), 517–529. https://doi.org/10.1109/TASLP.2015.2400218.

(43) Dhall, D.; Kaur, R.; Juneja, M. Machine Learning: A Review of the Algorithms and Its Applications. In *Proceedings of ICRIC 2019*; Singh, P. K., Kar, A. K., Singh, Y., Kolekar, M. H., Tanwar, S., Eds.; Lecture Notes in Electrical Engineering; Springer International Publishing: Cham, 2020; pp 47–63. https://doi.org/10.1007/978-3-030-29407-6_5.

(44) Akbani, R.; Kwek, S.; Japkowicz, N. Applying Support Vector Machines to Imbalanced Datasets. In *Machine Learning: ECML 2004*; Boulicaut, J.-F., Esposito, F., Giannotti, F., Pedreschi, D., Eds.; Hutchison, D., Kanade, T., Kittler, J., Kleinberg, J. M., Mattern, F., Mitchell, J. C., Naor, M., Nierstrasz, O., Pandu Rangan, C., Steffen, B., Sudan, M., Terzopoulos, D., Tygar, D., Vardi, M. Y., Weikum, G., Series Eds.; Lecture Notes in Computer Science; Springer Berlin Heidelberg: Berlin, Heidelberg, 2004; Vol. 3201, pp 39–50. https://doi.org/10.1007/978-3-540-30115-8_7.

(45) Van Houdt, G.; Mosquera, C.; Nápoles, G. A Review on the Long Short-Term Memory Model. *Artif. Intell. Rev.* **2020**, *53* (8), 5929–5955. https://doi.org/10.1007/s10462-020-09838-1.

(46) Weininger, D.; Weininger, A.; Weininger, J. L. SMILES. 2. Algorithm for Generation of Unique SMILES Notation. *J. Chem. Inf. Comput. Sci.* **1989**, *29* (2), 97–101. https://doi.org/10.1021/ci00062a008.

(47) Arús-Pous, J.; Johansson, S. V.; Prykhodko, O.; Bjerrum, E. J.; Tyrchan, C.; Reymond, J.-L.; Chen, H.; Engkvist, O. Randomized SMILES Strings Improve the Quality of Molecular Generative Models. *J. Cheminformatics* **2019**, *11* (1), 71. https://doi.org/10.1186/s13321-019-0393-0.

(48) Magar, R.; Wang, Y.; Lorsung, C.; Liang, C.; Ramasubramanian, H.; Li, P.; Farimani, A. B. AugLiChem: Data Augmentation Library of Chemical Structures for Machine Learning. *ArXiv211115112 Phys.* **2021**.

(49) Shorten, C.; Khoshgoftaar, T. M. A Survey on Image Data Augmentation for Deep Learning. *J. Big Data* **2019**, *6* (1), 60. https://doi.org/10.1186/s40537-019-0197-0.

(50) Gasteiger, J.; Groß, J.; Günnemann, S. Directional Message Passing for Molecular Graphs. arXiv April 5, 2022. http://arxiv.org/abs/2003.03123 (accessed 2022-11-13).

(51) Sanchez-Lengeling, B.; Reif, E.; Pearce, A.; Wiltschko, A. B. A Gentle Introduction to Graph Neural Networks. *Distill* **2021**, *6* (9), e33. https://doi.org/10.23915/distill.00033.

(52) Mercado, R.; Rastemo, T.; Lindelöf, E.; Klambauer, G.; Engkvist, O.; Chen, H.; Bjerrum, E. J. Graph Networks for Molecular Design. *Mach. Learn. Sci. Technol.* **2021**, *2* (2), 025023. https://doi.org/10.1088/2632-2153/abcf91.

(53) Sato, Y.; Yurugi, M.; Fujiwara, K.; Takishima, S.; Masuoka, H. Solubilities of Carbon Dioxide and Nitrogen in Polystyrene under High Temperature and Pressure. *Fluid Phase Equilibria* **1996**, *125* (1), 129–138. https://doi.org/10.1016/S0378-3812(96)03094-4.

(54) Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; Vanderplas, J.; Passos, A.; Cournapeau, D. Scikit-Learn: Machine Learning in Python. *Mach. Learn. PYTHON* **2022**, 6.

(55) XGBoost: EXtreme Gradient Boosting. https://github.com/dmlc/xgboost.

(56) Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; Desmaison, A.; Köpf, 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. **2019**. https://doi.org/10.48550/ARXIV.1912.01703.

(57) Huang, B.; von Lilienfeld, O. A. Communication: Understanding Molecular Representations in Machine Learning: The Role of Uniqueness and Target Similarity. *J. Chem. Phys.* **2016**, *145* (16), 161102. https://doi.org/10.1063/1.4964627.

(58) Krenn, M.; Häse, F.; Nigam, A.; Friederich, P.; Aspuru-Guzik, A. Self-Referencing Embedded Strings (SELFIES): A 100% Robust Molecular String Representation. *Mach. Learn. Sci. Technol.* **2020**, *1* (4), 045024. https://doi.org/10.1088/2632-2153/aba947.

(59) Degen, J.; Wegscheid-Gerlach, C.; Zaliani, A.; Rarey, M. On the Art of Compiling and Using “Drug-Like” Chemical Fragment Spaces. *ChemMedChem* **2008**, *3* (10), 1503–1507. https://doi.org/10.1002/cmdc.200800178.