Machine Learning-Guided Discovery of Polymer Membranes for CO_2 Separation

Yasemin Basdogan,[1] Dylan R. Pollard,[2] Tejus Shastry,[3] Matthew R. Carbone,[4] Sanat K. Kumar,[3] and Zhen-Gang Wang[1*]

[1]Chemical Engineering, California Institute of Technology, Pasadena, CA 91125, USA
[2]Chemical Engineering, Auburn University, Auburn, AL 36830, USA
[3]Chemical Engineering, Columbia University, New York, NY 10027, USA
[4]Computational Science Initiative, Brookhaven National Laboratory, Upton, NY 11973, USA

[*]To whom correspondence should be addressed; E-mail: zgw@caltech.edu.

Designing polymer membranes with high gas permeability and selectivity remains a grand challenge for energy, the environment, and economic sustainability. Increasing both the selectivity and permeability is a difficult multi-task constrained design problem for polymer membranes due to the trade-off between these two properties. The complexity of chemical composition and morphology of polymers makes this problem especially hard to attack with trial-and-error or intuition-based strategies. In this work, we instead present a machine learning (ML)-driven genetic algorithm to tackle the design problem of polymer membranes for CO_2 separation from N_2 and O_2. Using literature data of permeability for three gases, CO_2, N_2, and O_2, we constructed multiple ML models using different fingerprinting featurization schemes to predict all three gas permeabilities as well as the CO_2/N_2 and CO_2/O_2 selectivity values. Then, we employed a genetic algorithm to design new polymers and evaluated their performance with respect to the Robeson upper bounds using our machine learning models. We were able to identify new polymer membranes that are promising for both CO_2/N_2 and CO_2/O_2 separations. The

https://doi.org/10.26434/chemrxiv-2023-5h4s7 ORCID: https://orcid.org/0000-0002-2071-9675

top discovered polymers are predicted to have high glass transition temperatures, T_g. Similarly, the pyridine functionality was found in $\approx 20\%$ of the predicted polymers. Both of these facts are well in line with currently accepted experimental wisdom for CO_2 based separations. The framework developed here can be used to design polymers for any application involving constrained optimization. Finally, we outlined the strengths and limitations of this approach, as well as the imminent challenges and opportunities with using machine learning guided data-driven inverse design of polymers.

Introduction

The increased concentration of CO_2 in the atmosphere is the single most important anthropogenic cause of global warming. Decreasing the release of CO_2 into the atmosphere requires efficient CO_2 capture and separation technologies. Decades of research have been devoted to improving existing gas separation technologies, but there is still an imminent need to find new methodologies given the current course of climate change (*1*). Traditional unit operations have the ability to isolate high-purity products, but they have a high carbon footprint due to the high energy requirements. Membrane-based technologies are an attractive alternative because they provide savings in capital and energy-related operating costs, and offer advantages related to the ease of operation and compact environmental footprint (*2–4*). Polymer membranes have been successfully investigated for H_2 recovery, N_2 generation, but there is still a significant opportunity to improve polymer membrane technology for CO_2 separations. Although hundreds of new materials are synthesized each year, most of the commercial membranes used today are from the 1990s, and they rely on a dozen or so common polymer structures. This is largely because the two properties that are important for a membrane material – high flux (permeability) and high gas purity (selectivity) – are inversely correlated. This inverse relationship between gas selectivity and permeability was first examined by Robeson in 1991 (*5*) and revisited in 2008 for pure homopolymer membranes (*6*) and is famously known as the Robeson Upper Bound. Since then, there have been considerable efforts in designing polymers that are above the empirically determined upper bound for a given application (*7–9*).

Designing polymers with targeted structural and functional properties is challenging due to the practically infinite polymer chemistry design space. Trial-and-error or intuition-based strategies are not efficient, and they are likely to miss optimal solutions due to the complexity of chemical composition and morphology of polymers. Furthermore, these strategies with traditional experimental and computational routes are time and resource consuming. Machine Learning (ML) models trained on polymer data sets can mitigate this problem, as it is possible to predict a new material's properties instantaneously by interpolating within an existing dataset (*10*). There have been a number of studies in the recent literature that leveraged ML to predict the properties of polymers. For example, Alves *et al.* developed models to discover polymeric micelle formulations for poorly soluble drugs using micellar solubilization data (*11*). Tao *et al.* used ML models to predict the glass transition temperature of a polymer based on its structural formulation (*12*). Later, these authors also did a benchmarking study to compare the predictive power of numerous ML models and showed the importance of structure and feature representations (*13*). Xu *et al.* used ML models to study swelling of polymer membranes in different solvents with chemically informed molecular representations and descriptors (*14*). Wang *et al.* used ML models to screen polymers for pervaporation separation (*15*) and developed a data-driven approach to predict the fractional free volume of polymers (*16*). There are also excellent review articles published in the last couple of years that summarize the recent developments in ML studies of polymer properties (*17–26*).

The success of applying ML models to design new polymer membranes for gas separation has been comparatively lacking, largely owing to limitations in data availability. Barnett *et al.* used experimental gas permeability data to develop a ML model to predict gas separation in polymer membranes (*27*). They have successfully identified several polymers for improved CO_2/CH_4 separation and synthesized two of them to experimentally validate the ML predictions. Yuan *et al.* used ML algorithms to predict the missing values for the permeability of different gases in the online Polymer Gas Separation Membrane Database of the Membrane Society of Australasia (*28*). Yang *et al.* used the same data set and leveraged ML models to predict gas permeability based on the polymer chemistry (*29*). However, a ML model that predicts polymer properties by itself does not lead to the discovery of new polymer membranes with optimal properties. In principle, one can propose many candidate polymers, possibly at random, and use ML to predict their performance. This is obviously not an efficient strategy. A ML "forward model" needs to be coupled with an inverse design/generative algorithm to efficiently explore the polymer material space. Genetic algorithms (GA) are an example of a data-driven inverse design method, which can be effectively coupled with an ML model. Srinivasan *et al.* used GA to design single-stranded DNA grafted colloids (*30*). These authors were able to reproduce the experimentally validated phase diagram and additionally identify the formation of four previously unobserved crystal structures. Kim *et al.* demonstrated one of the first data-driven inverse design methods of new polymers having high band gap and high glass transition temperature that is relevant for high-temperature and high-energy density dielectrics (*31*). They successfully identified new polymer structures with the desired properties.

In this work, we follow a similar procedure to Kim *et al.* to design new polymer membranes with the desired selectivity and permeability for CO_2 separation from N_2 and O_2. First, we start by assembling a library of gas permeabilities corresponding to the experimental studies of various polymers. Next, we train multiple ML models based on various fingerprints to determine which ML model performs the best in predicting gas permeability. Then, we use our ML models to drive a GA for 100 generations and create more than 16000 new polymer structures. We also use different fitness functions to design the best possible polymers given our initial data set.

Application of this combined ML-GA framework results in the discovery of more than 20 new polymers that are above both the CO_2/N_2 and CO_2/O_2 Robeson upper bounds, many of which contain aromatic functional groups along with oxygen- and nitrogen- motifs, aligning with experimental observations that show imines and polyethers as promising polymer membranes for CO_2 separation (*32–36*).

Results

We compiled a literature database of permeability for three gases – CO_2, N_2, and O_2 – in a variety of polymers at a temperature range of 300–330 K. The number of data points for each gas is different due to the availability of data in the literature, so we only considered polymers that have permeability measurements for all three gasses. This resulted in 780 different polymers in our library, which represent a sizable portion of the polymers that are typically included in the most up-to-date Robeson plots. The selectivity versus permeability data for CO_2/N_2 and CO_2/O_2 are shown in Figure 1. We see that there are only three polymers that are above the CO_2/N_2 Robeson upper bound (*37, 38*). These polymers have a benzene ring and ether oxygen functional groups in common, which are known to be favorable for CO_2 separation. There are more than ten polymers that are above the CO_2/O_2 upper bound as shown in Figure 1b. The list of all polymers in our library and permeability measurements are provided in the SI.

The first step in applying ML models to evaluate physical properties is choosing an appropriate mathematical form to be used as input. This is commonly known as featurization (or fingerprinting in the chemo-informatics literature) and it is of critical importance to the quality and interpretability of the ML models. We start with generating the simplified molecular-input line-entry system (SMILES) (*39, 40*) representations of our polymers based on their repeating units. We cap the two ends of the monomer structure with hydrogen atoms to create a consistent data set. Based on our SMILES strings, we use two common fingerprints in the literature, the

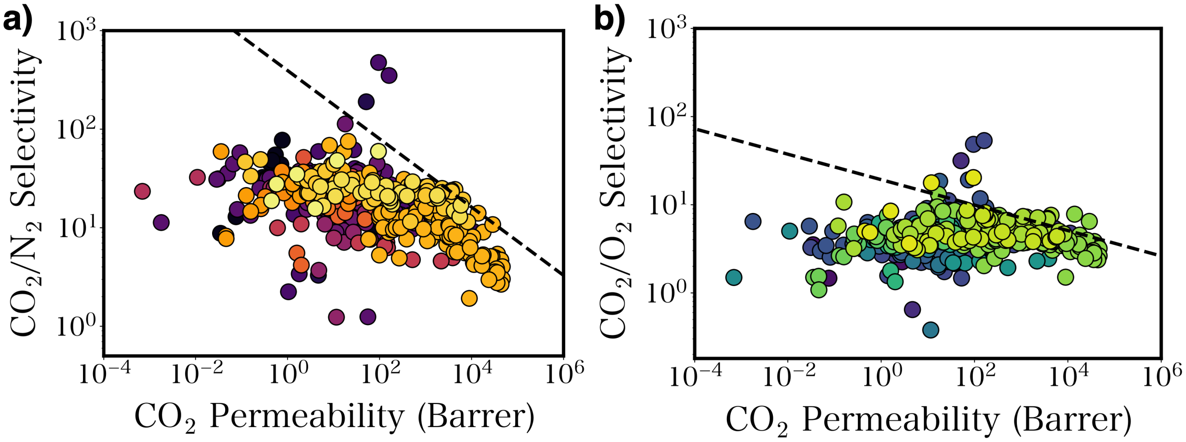

Figure 1: Robeson plot of selectivity versus permeability for (a) CO_2/N_2 and (b) CO_2/O_2 separations. The 2008 Robeson upper bounds (*6*) are shown as dashed black lines. Color code represents the different classes of polymers. Each data points represent a single polymer.

Extended Connectivity Fingerprint with bond diameter four Angstroms (ECFP4) (*41, 42*) and the Molecular ACCess System (MACCS) (*43, 44*) fingerprint. MACCS is a common substructure keys-based fingerprint consisting of a binary vector of 166 bits depending on the presence of certain substructures or features from a given list of structural keys (*45*). ECFP4 is an example of a topological fingerprint that is based on analyzing all the fragments of the molecule by looking at the environment of each atom up to a set radius, and then hashing every one of these environments to create the fingerprint. One needs to be careful when using hashed fingerprints because a bit cannot be traced back to a given feature, and this may result in a given bit corresponding to more than one different feature, which is called "bit collision" (*46*). We use ECFP4, based on the Morgan algorithm (*47*), which is a 2048 bit fingerprint as implemented in RDKit. Figure 2 shows the comparison for predicting CO_2 permeability with the random forest regression model using both fingerprints. We fit and plot the logarithmic permeability values to better visualize the data set. Both fingerprints result in R^2 value of 0.982 for the training set. However, R^2 for the test set is considerably higher when we use ECFP4 as shown in Figure 2. We also compare the root mean square error (RMSE) of the fits for both fingerprints. The test set RMSE with ECFP4 fingerprint is 0.131, and the test set RMSE with MACCS fingerprint is 0.161. Thus, we use ECFP4 to train and test our ML models for the rest of this paper.

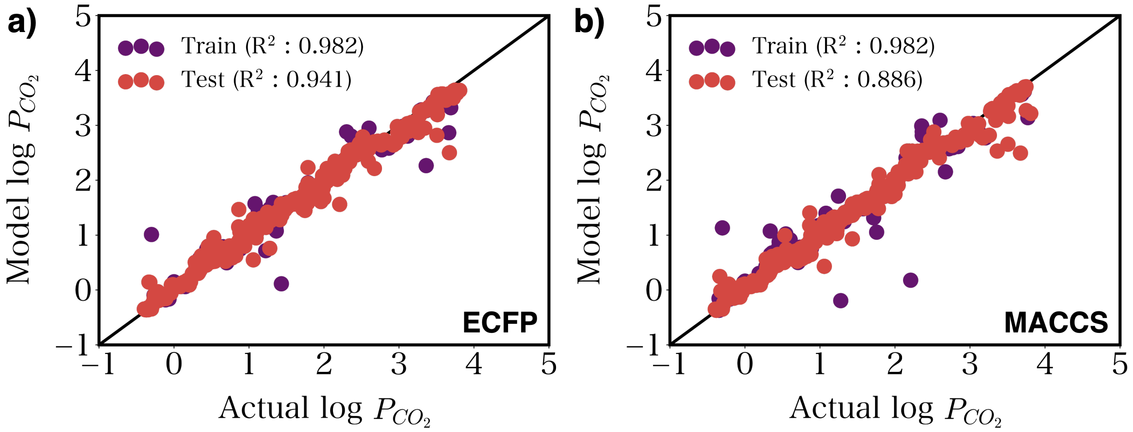

Figure 2: Comparison of CO_2 permeability model predictions using (a) ECFP and (b) MACCS fingerprints with random forest regression algorithm.

We start with randomly splitting our data set into one of two categories for each gas; one is used for training the ML model, while the other is initially withheld during training. The training data sets were 80% of our total database for each gas, which represents more than 600 polymers for each gas. We then apply the trained model to the remaining 20% of the polymers (test set) and use these data as verification of the model's accuracy. Then, we employ various ML models on the training sets including support vector regression (SVR), k-nearest neighbors (KNN), decision tree, and random forest regression. Next, we compare the predictive power of these popular ML regression models on CO_2 permeability values.

First we study SVR, which has the ability to consider non-linearity in the permeability data (*48*). SVR results in R^2 value of 0.84 and 0.203 RMSE on the test set. KNN regression, which predicts the target value by local interpolation of the targets associated with the nearest neighbors in the training set, results in R^2 value of 0.822 and 0.242 RMSE on the test set. Then, we employed a decision tree regression model, which uses a tree structure and inference layer to achieve the final decision of the modeling results (*18*). Decision tree regression performs better than both the SVR and KNN regression with R^2 value of 0.881 and 0.148 RMSE on the test set. Finally, to make a more accurate prediction, we used a random forest regression model, which is an ensemble learning method for regression that combines predictions from multiple decision tree models. As expected, random forest predictions are better than all the other algorithms that we have tried with R^2 value of 0.941 and 0.135 RMSE on the test set. Figure 3 summarizes the different ML regression models that we have tried. We note that Yang *et al.* compared random forest regression models with deep neural networks (DNN) and showed DNN model performs better than the random forest regression model (*29*). However, DNNs typically require much more data than what is available for this study.

To determine where on the Robeson plot a polymer is located, we need to be able to predict the CO_2/N_2 and CO_2/O_2 selectivity as well as the CO_2 permeability. The ideal selectivity $\alpha_{i/j}$ for the gas pair is the ratio of the permeabilities P_i and P_j. Thus, we need ML models to predict N_2 and O_2 permeability as well. Because the random forest regression model is the best performing model for the CO_2 permeability, we have continued using random forest regression for the N_2 and O_2 permeability. Figure 4 shows model predictions for the N_2 and O_2 permeability. For both gases we can predict the gas permeability with R^2 values higher than 0.9. The RMSE for N_2 and O_2 are 0.171 and 0.147, respectively. This demonstrates that we can predict all three gas permeabilities accurately with the random forest regression model. Now that we have established an accurate ML model to predict a polymer membrane's performance with respect to the Robeson upper bounds, we start designing new polymers with a GA and evaluate their performance on the fly with these ML models.

The first step of the GA is to construct the ”gene pool” that will be used to create the initial parent polymers. We used the “Breaking of Retrosynthetically Interesting Chemical Substructures” (BRICS) algorithm as implemented in the RDKit Python package to get the chemical

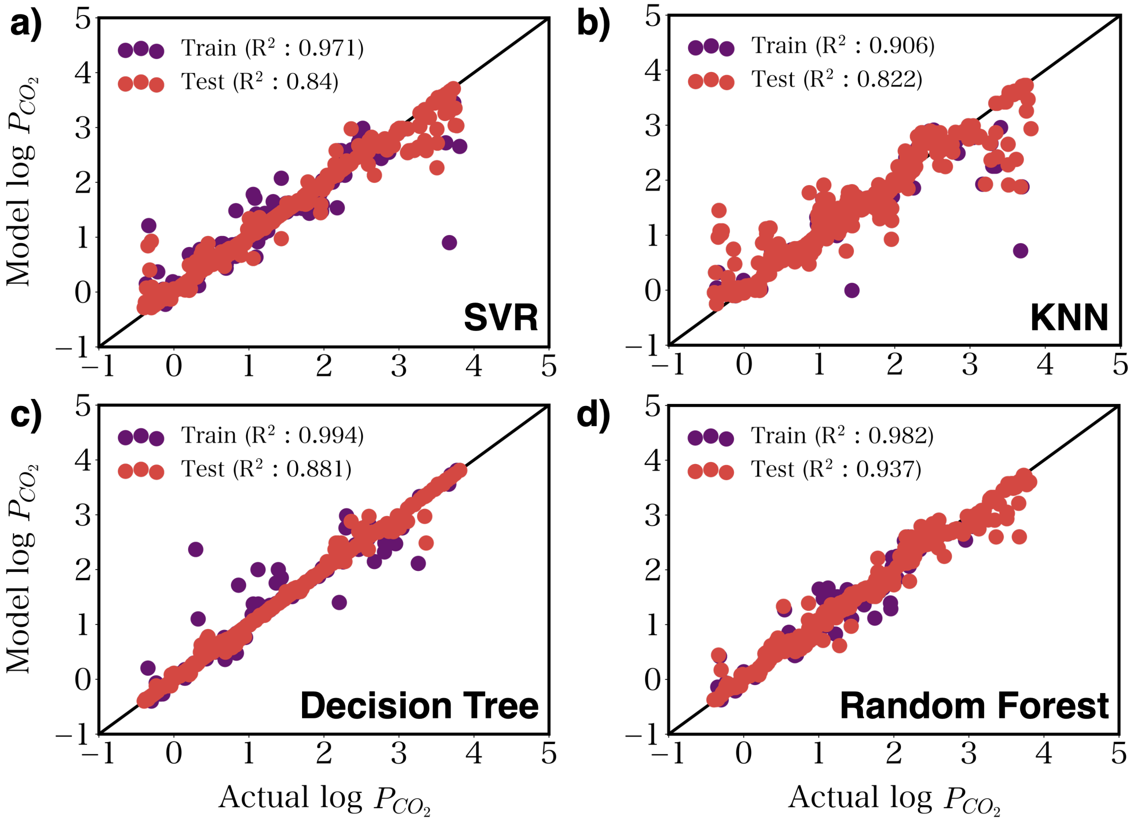

Figure 3: Comparison of CO_2 permeability model predictions using (a) SVR, (b) KNN regression, (c) Decision tree regression, and (d) Random forest regression models with ECFP fingerprints.

building blocks, or fragments, from our polymer library (*49*). A total of 79 unique fragments were extracted from 780 reference polymers. Figure 5 shows six functional groups that appear most frequently in our library. To initiate the GA process, 100 parent polymers consisting of 4 building blocks in their monomer unit were created in the first generation. The fragments were chosen randomly from our gene pool of the 79 chemical fragments. Then, 15 families with the smallest Tanimoto similarity score (*50*), with 3 parents in each family, were chosen to perform crossover and mutation operations to alter their sequence of chemical building blocks, resulting in 12 offspring polymers in each family. During crossover, two parents generate an offspring by combining one random segment from a parent with another random segment from the other

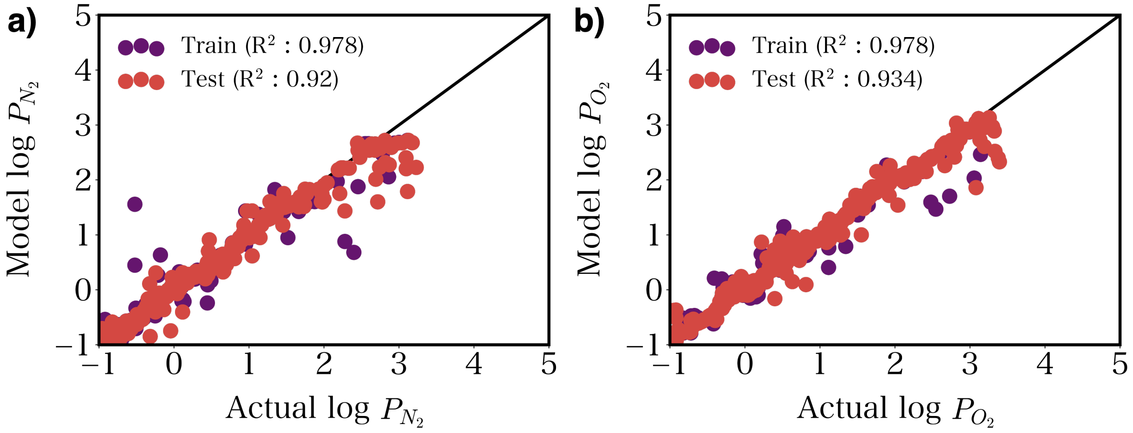

Figure 4: (a) N_2 and (b) O_2 permeability random forest regression model predictions with ECFP fingerprints.

parent. The segmentation point of a parent polymer was chosen according to a Gaussian distribution with a mean at the center of the sequence and standard deviation of 0.3 blocks. We also applied mutation operations on 60% of the genes to increase the chemical diversity, where we randomly selected the building block in the sequence and replaced it with a new building block randomly chosen from the list of the 79 blocks. In each GA iteration, the top performing offspring polymers with the highest fitness evaluation were retained as parents to create the next generation offspring polymers. We also assigned 10% migration rate between different families, whereby the highest-scoring polymers that were not selected as a parent migrated to a different family. An essential component in this evolutionary process is the polymer property estimation, which traditionally has been evaluated by experiments that are very time-consuming and expensive; here, we use our ML models for on-the-fly polymer property estimation.

We ran the GA for 100 generations and generated more than 16000 new polymer structures as shown in Figure 6. All the new polymers generated with the GA are reported in the SI, where we highlight the top performing 100 polymers. We optimized multiple parameters in the GA framework to guide the evolutionary process towards the targeted design area. First,

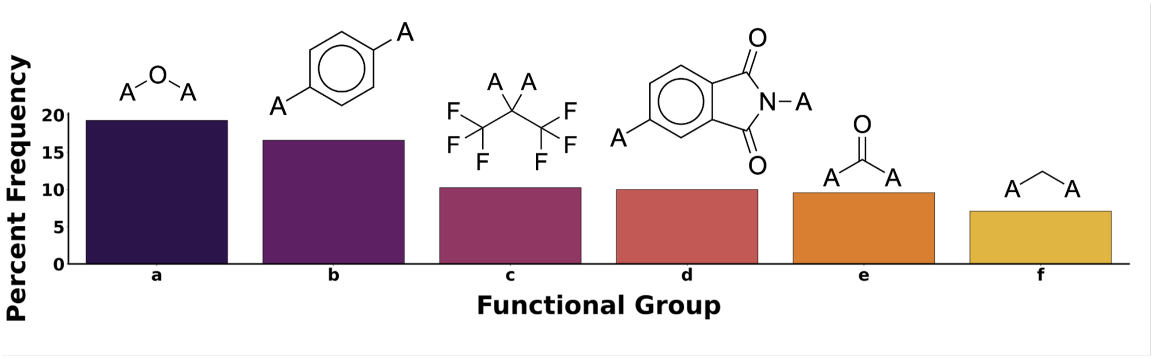

Figure 5: The six most chemical functional groups that appear in our library of experimental polymers using the BRICS algorithm. ”A” represents the binding sites.

we ran the GA with an ML model trained with two different fingerprints, ECFP and MACCS keys, and found that the fingerprint used in the ML does not influence the top performing polymers identified from the GA framework. Next, we tried running the GA for an additional 100 generations to see if running the GA for longer will result in better performing polymers, but found that the additional iterations did not result in any improved polymer structures. Finally, we tried multiple fitness functions to optimize the evolutionary trajectory. To optimize both CO_2/N_2 and CO_2/O_2 selectivity as well as CO_2 permeability, one needs a fitness function that includes all three metrics. However, the functional form of the fitness function is not clear *a priori*. We tried the fitness function $\log\left(P_{CO_2}\right) \times \alpha_{CO_2/N_2} \times \alpha_{CO_2/O_2}$ and showed that GAs using this fitness function failed to identify new polymer structures that are above both upper bounds. We found that the fitness function based on the actual P_{CO_2} permeability values $\left(P_{CO_2} \times \alpha_{CO_2/N_2} \times \alpha_{CO_2/O_2}\right)$ did result in several polymers that are above both the CO_2/N_2 and CO_2/O_2 upper bounds as shown in Figure 6. Because CO_2 permeability values are generally orders of magnitude larger than the selectivity values, this model favors the polymers that have higher permeability, thus biasing the GA towards better performing polymers.

We used the BRICS algorithm on the GA-generated polymers to understand which chemical building blocks are frequently observed in the top performing polymers. The top six frequently

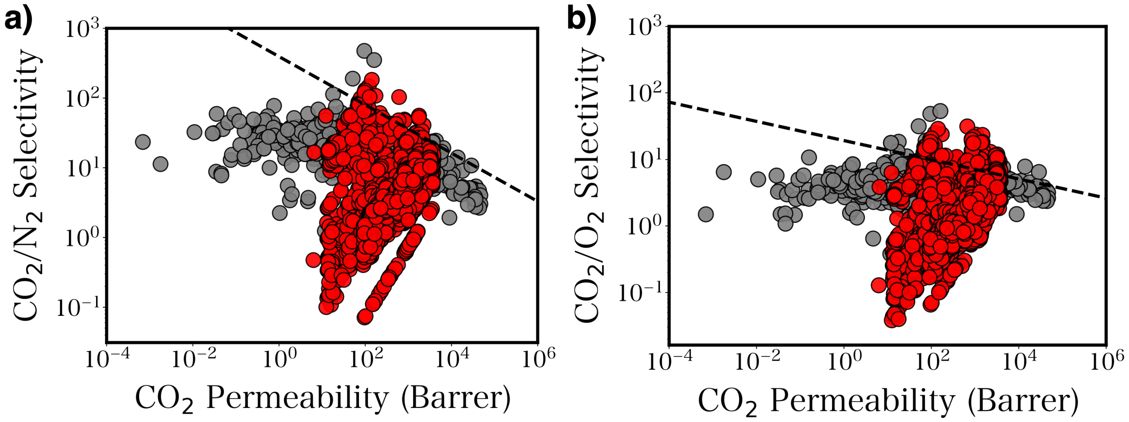

Figure 6: Robeson plot of selectivity versus permeability for (a) CO_2/N_2 and (b) CO_2/O_2 separations. The 2008 Robeson upper bounds are shown as dashed black lines (*6*). Colors represent experimental and GA generated polymers. Red color represents polymers generated with the GA. Each data point represents a single polymer.

observed functional groups with the 100 fittest GA-generated polymers are shown in Figure 7. We identified a total of 464 chemical fragments within the fittest 100 polymers, and 18% of the fragments were pyridine functional groups. More than 70 polymers in the top performing GA-generated polymers have the pyridine functional group in their repeating unit. This is by far the most frequently observed functional group, which is then followed by benzoxazole with 3%. We observe benzoxazole functional group in 13 polymers within the 100 fittest GA-generated polymers. Similarly, benzene, phosphonamidic acid, naphthalene, and dibromobenzene functional groups are also observed in the top performing polymers generated with the GA. We show six example polymer structures that have high fitness function values in Figure 8. We note that these polymers have pyridine, benzoxazole, benzene, and phosphonamidic acid functional groups, which we identified as the most abundant functional groups with the BRICS algorithm. These polymers also include oxygen-, sulfur-, and nitrogen- containing motifs, similar to the three experimental polymers that are above the CO_2/N_2 upper bound. Oxygen- and nitrogen- containing motifs are reminiscent of imines and polyethers, which are known to be high performing polymer membranes. Interestingly, our initial analysis using the Polymer Genome software (*51–54*) suggests that most of the top 100 fittest polymers have high glass transition temperature—well above the standard operating conditions ($>$ 400 K). We note that the three polymers in the experimental data set that are above the CO_2/N_2 upper bound also have glass transition temperatures around 400 K. We speculate that the superior performance of these glassy amorphous polymers for gas separation may be due to their high fractional free volume and high number of microvoids (*55, 56*). It remains an open question whether or not these polymers are easily synthesizable and easy to implement as membranes given their complicated chemistry. Further computational and experimental studies will be required to better understand these polymers and their efficacy as membrane materials.

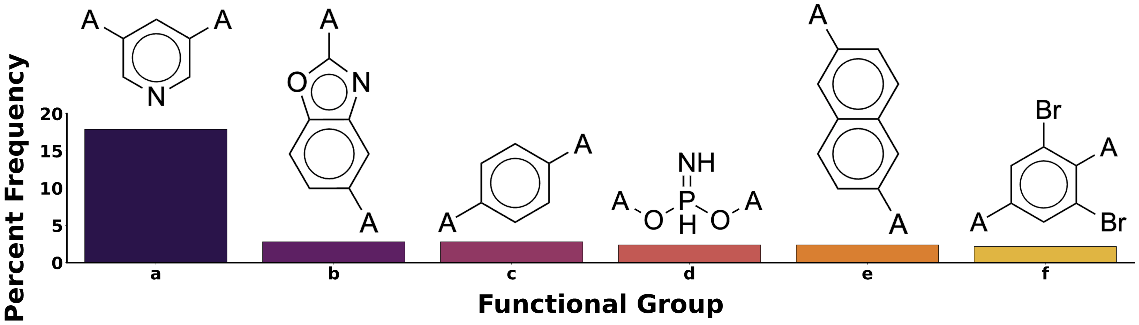

Figure 7: The six functional groups that appear most in the 100 fittest polymers generated with the GA using the BRICS algorithm. ”A” represents the binding sites.

Discussion

We constructed an ML-driven GA to tackle the inverse design problem of polymer membranes for CO_2 separation. We showed that the hashed-based ECFP4 yields lower predictive errors on the test sets than the substructure keys-based fingerprints. We presented different regression-based ML models, where random forest regression models resulted in the lowest RMSE and highest R^2 values for both the test and the training set. Although random forest regression

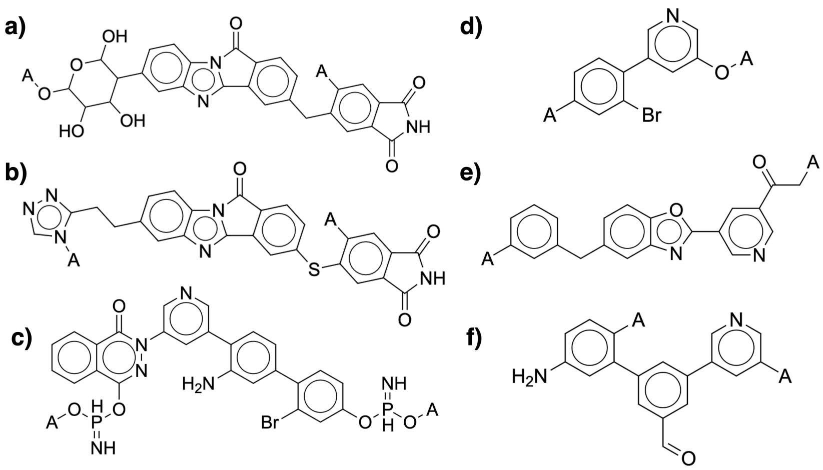

Figure 8: Example of polymer repeating units generated with the GA. "A" represents the binding sites.

models can successfully predict both the gas permeability and the selectivity, we used models trained on the gas permeability, since these models have better predictive power than the models trained on the selectivity. After obtaining the ML models to predict the performance of any polymer membrane candidate, we implemented a data-driven inverse design algorithm to efficiently explore the polymer material space. In theory, one can use any inverse design algorithm for such a problem, but we have implemented a GA since it has been successfully used for other polymer applications. We created the gene pool using the BRICS algorithm on the experimental data and obtained 79 unique genes to initiate the GA process with 100 parent polymers that have 4 genes in their monomer unit. Fitness function is a key driving parameter in the GA, and as such, we used $P_{CO_2} \times \alpha_{CO_2/N_2} \times \alpha_{CO_2/O_2}$ to determine the fitness of the polymers. We performed crossover and mutation functions for 100 generations to create more than 16000 polymers during the GA process. Among these 16000 polymers, we were able to identify more than 20 new polymers that are above both CO_2/N_2 and CO_2/O_2 upper bounds. While validation of the new polymer structures identified in this work requires future molecular dynamics simulations and experimental measurements, this work helps identify the strengths and the weaknesses of combining ML models and GAs as discussed below.

Three key points emerged from our analysis on comparing popular fingerprints. First, hashed-based fingerprints result in lower predictive errors on the test sets than the substructure keys-based fingerprints. However, hashed-based fingerprints have one main disadvantage compared to the substructure keys-based fingerprints, which is not having a one-to-one correspondence between the fingerprint vector and the chemical structure. This is not the case with the substructure keys-based fingerprints, where each bit corresponds to a predetermined substructure. This disadvantage does not affect our framework since we do not go back-and-forth between the fingerprint and the chemical structure, and only use the fingerprinting when evaluating the fitness function in the GA framework. Not performing crossover and mutation functions on the fingerprints makes it possible to overcome this main disadvantage associated with hashed-based fingerprints. Next, we note that the top performing polymer candidates identified within this framework does not depend on which fingerprint is used in the GA. The relative strength of the polymers with each other are similar with the two different fingerprints. The main difference with using different fingerprints is the absolute value of the fitness function, since the ML models trained on the substructure keys-based fingerprints tend to underestimate the gas permeability. Finally, using the proper descriptor for a given application is still an open question in the polymer informatics field, but we have shown that most often it does not affect the final result. However, we emphasize that more sophisticated descriptors, like physical descriptor vectors that include bulk properties of the polymer membranes, may make a difference in the top performing polymers identified from the GA (*57*). For example, we believe the glass transition temperature is a key property for polymer membranes, and including this information in the fingerprint can lead to better performing polymer membranes within our framework. The main bottleneck in switching from popular fingerprints to physical descriptors is gathering consistent measurement data from hundreds of different experimental papers. The only way to overcome this bottleneck is to create our own data sets using computational simulations so that we can consistently calculate the physical properties of interest for each polymer.

We have demonstrated that a random forest regression model performs best when predicting the gas permeability and selectivity of polymer membranes. Because there is a significant difference in the R^2 values and the RMSEs, we use random forest regression models in the entire framework. However, random forest algorithms, like essentially all data-driven methods, are intrinsically interpolative (*10*). They are only suited to optimize properties within the bounds of the data the model was trained on. Models can still generalize, and interpolate "between molecules" in some abstract design space (*58*), but they will not make accurate predictions outside of this space. Thus, the performance of the new polymer structures identified in this framework depends on the initial data set and the range of selectivity and permeability values covered. One way to address this challenge is by using computer simulations combined with e.g. an active learning loop to curate a polymer library will enable us to expand the number of data points included in the ML framework. It is important to note that regardless of the data-driven approach, ML models will always have a strong dependency on the initial data that they are trained on. The only way to surpass this main limitation is to move towards active learning algorithms where we give the algorithm the ability to "learn" and draw inferences from its experience to accelerate the evolutionary process (*59, 60*). New molecules generated as part of the GA procedure could be screened by an uncertainty-quantifying algorithm, and when confidence is low, new simulations can be run to acquire new ground truth data, which can then be used to retrain the model. This is only possible if we use computer simulations (or a very high-throughput, autonomous experiment) to curate the data since we will need to make on-the-fly property estimations with an active learning framework.

Finally, we identify two significant points for coupling GAs with ML models to design new polymer membranes. First, and most importantly, the fitness function drives the evolutionary process of the GA. It is therefore of utmost importance to select the correct fitness function to direct the GA towards the targeted design area. It is customary to train ML models on the logarithmic permeability values since it narrows the range of the data, hence resulting in more accurate models. Thus we tried the fitness function $\log\left(P_{CO_2}\right) \times \alpha_{CO_2/N_2} \times \alpha_{CO_2/O_2}$, aiming to maximize both the gas permeability and selectivity throughout the evolutionary process. However, this fitness function was not able to push the GA towards the targeted design area. Even though ML models trained on logarithmic permeability have slightly higher predictive power, the small numerical value of the logarithmic permeability diminishes the importance of the permeability contribution to the fitness function. On the other hand, with a fitness function that includes the absolute value of the permeability $\left(P_{CO_2} \times \alpha_{CO_2/N_2} \times \alpha_{CO_2/O_2}\right)$, we were able to push the evolutionary process toward the targeted design area and identified more than 20 new polymers that are above both CO_2/N_2 and CO_2/O_2 upper bounds. We attribute the superior performance of this fitness function to the fact that the absolute value of the permeability is usually two orders of magnitude higher than selectivity values. This analysis also shows improving the selectivity with the GA is much harder than the permeability since the fitness function becomes insensitive to the selectivity values when we include the gas permeability. In the future this can be avoided by normalizing the parameters where the normalization would negate this effect. Next we emphasize that, our GA was able to converge within 100 generations, since running the algorithm for an additional 100 generations did not result in any superior polymer membranes. With 100 generations and 4 initial building blocks in the first generation, a total of 17571 new unique polymer structures were created. With 79 unique genes, the number of sequences that can be generated by the GA is at least 79^4 (since longer sequences are generated throughout the evolutionary process). This suggests that the GA converges very fast, only exploring less than 1% of the possible polymer material space. We decided to use 4 genes with the initial generation because we have a relatively small gene pool. Using more building blocks with the initial generation could have created more complicated structures throughout the evolutionary process. This can be further explored when we have a larger gene pool.

Our approach demonstrates successful implementation of an ML-driven GA to design polymer membranes for CO_2 separation, but more importantly, this framework can be used to design polymer structure for any application (e.g. ion separation membranes and polymer electrolytes for batteries), where there is a constrained optimization problem. The main limitation of the current framework arises from its dependence on the initial experimental data. Curating the data with computer simulations is a possible way to overcome this limitation. With better control over the initial data set we will be in a position to explore more sophisticated descriptors and switch to an active learning framework where we make on-the-fly property estimations. Computational ML-driven inverse design of polymer membranes is a promising platform that can be further tailored to consider functions that incorporate the sustainability and synthetic viability of the polymers, in addition to gas selectivity and permeability, which are not yet widely considered in computational studies.

Acknowledgments

This work was supported in part by Resnick Sustainability Institute (YB, ZGW). Additional support was provided by Hong Kong Quantum AI Lab, AIR@InnoHK of Hong Kong Government (ZGW). DP acknowledges support by the Caltech Amgen Scholars Program. This material is also based upon work supported by the U.S. Department of Energy, Office of Science, Office of Basic Energy Sciences, under Award Numbers FWP PS-030 and DE-SC-0012704 (MRC, SKK). At Columbia financial support for this work was provided by the U.S. Department of Energy under Grants DE-SC0021272 (TS).

Supplementary Material

We provide the initial polymer library we have used to train our ML models with each polymer represented as a SMILES string as well as the GA generated polymers with permeability for three gasses (CO_2, N_2, and O_2).

References

1. D. F. Sanders, *et al.*, *Polymer* **54**, 4729 (2013).
2. R. W. Baker, K. Lokhandwala, *Ind. Eng. Chem. Res.* **47**, 2109 (2008).
3. R. W. Baker, B. T. Low, *Macromolecules* **47**, 6999 (2014).
4. D. S. Sholl, R. P. Lively, *Nature* **532**, 435 (2016).
5. L. M. Robeson, *J. Membr. Sci.* **62**, 165 (1991).
6. L. M. Robeson, *J. Membr. Sci.* **320**, 390 (2008).
7. H. B. Park, *et al.*, *Science* **318**, 254 (2007).

8. M. D. Guiver, Y. M. Lee, *Science* **339**, 284 (2013).

9. N. Du, *et al.*, *Nat. Mater.* **10**, 372 (2011).

10. M. R. Carbone, *MRS Bulletin* **47**, 968–974 (2022).

11. V. M. Alves, *et al.*, *Sci. Adv.* **5**, eaav9784 (2019).

12. L. Tao, G. Chen, Y. Li, *Patterns* **2**, 100225 (2021).

13. L. Tao, V. Varshney, Y. Li, *J. Chem. Inf. Model* **61**, 5395 (2021).

14. Q. Xu, J. Jiang, *ACS Appl. Polym. Mater.* **2**, 3576 (2020).

15. M. Wang, Q. Xu, H. Tang, J. Jiang, *ACS Appl. Mater. Interfaces* **14**, 8427 (2022).

16. M. Wang, J. Jiang, *ACS Appl. Mater. Interfaces* **14**, 31203 (2022).

17. M. Rahimi, S. M. Moosavi, B. Smit, T. A. Hatton, *Cell Rep. Phys. Sci.* **2**, 100396 (2021).

18. Y. Liu, O. C. Esan, Z. Pan, L. An, *Energy and AI* **3**, 100049 (2021).

19. T. K. Patra, *ACS Polym. Au* **2**, 8 (2021).

20. A. Tayyebi, A. S. Alshami, X. Yu, E. Kolodka, *J. Membr. Sci. Letters* p. 100033 (2022).

21. K. Sattari, Y. Xie, J. Lin, *Soft Matter* (2021).

22. R. S. K. Valappil, N. Ghasem, M. Al-Marzouqi, *J. Ind. Eng. Chem.* **98**, 103 (2021).

23. S. Gupta, L. Li, *JOM* pp. 1–15 (2022).

24. L. Chen, *et al.*, *Mater. Sci. Eng. R Rep.* **144**, 100595 (2021).

25. Y. Amamoto, *Polym. J.* pp. 1–11 (2022).

26. Q. Xu, J. Jiang, *Mol. Syst. Des. Eng.* (2022).

27. J. W. Barnett, *et al.*, *Sci. Adv.* **6**, eaaz4301 (2020).

28. Q. Yuan, *et al.*, *J. Membr. Sci.* **627**, 119207 (2021).

29. J. Yang, L. Tao, J. He, J. R. McCutcheon, Y. Li, *Sci. Adv.* **8**, eabn9545 (2022).

30. B. Srinivasan, *et al.*, *Proc. Natl. Acad. Sci.* **110**, 18431 (2013).

31. C. Kim, R. Batra, L. Chen, H. Tran, R. Ramprasad, *Comput. Mater. Sci.* **186**, 110067 (2021).

32. S. La Cognata, *et al.*, *Eur. J. Chem.* **28**, e202201631 (2022).

33. Y. Zu, *et al.*, *Microporous and Mesoporous Mater.* **334**, 111779 (2022).

34. H. Lin, B. D. Freeman, *Macromolecules* **39**, 3568 (2006).

35. J. Liu, X. Hou, H. B. Park, H. Lin, *Eur. J. Chem.l* **22**, 15980 (2016).

36. T. Tran, Y. Fu, D.-e. Jiang, H. Lin, *Macromolecules* **55**, 9860 (2022).

37. G. Polotskaya, S. Agranova, T. Antonova, G. Elyashevich, *J. Appl. Polym. Sci.* **66**, 1439 (1997).

38. Y. Li, M. Ding, J. Xu, *J. Appl. Polym. Sci.* **63**, 1821 (1997).

39. D. Weininger, *J. Chem. Inf. Model.* **28**, 31 (1988).

40. M. Krenn, *et al.*, *Patterns* **3**, 100588 (2022).

41. D. Rogers, R. D. Brown, M. Hahn, *J. Biomol. Screen.* **10**, 682 (2005).

42. D. Rogers, M. Hahn, *J. Chem. Inf. Model.* **50**, 742 (2010).

43. M. S. Keys, *CA, USA* (2011).

44. J. L. Durant, B. A. Leland, D. R. Henry, J. G. Nourse, *J. Chem. Inf. Model.* **42**, 1273 (2002).

45. A. Cereto-Massagué, *et al.*, *Methods* **71**, 58 (2015).

46. M. Sastry, J. F. Lowrie, S. L. Dixon, W. Sherman, *J. Chem. Inf. Model* **50**, 771 (2010).

47. H. L. Morgan, *J. Chem. Doc.* **5**, 107 (1965).

48. X. Yu, *Fibers Polym.* **11**, 757 (2010).

49. J. Degen, C. Wegscheid-Gerlach, A. Zaliani, M. Rarey, *ChemMedChem* **3**, 1503 (2008).

50. P. Baldi, R. Nasr, *Journal of chemical information and modeling* **50**, 1205 (2010).

51. C. Kim, A. Chandrasekaran, T. D. Huan, D. Das, R. Ramprasad, *J. Phys. Chem. C* . **122**, 17575 (2018).

52. C. Kuenneth, *et al.*, *Patterns* **2**, 100238 (2021).

53. H. Doan Tran, *et al.*, *J. Appl. Phys.* **128**, 171104 (2020).

54. A. Chandrasekaran, C. Kim, S. Venkatram, R. Ramprasad, *Macromolecules* **53**, 4764 (2020).

55. R. Mahajan, R. Burns, M. Schaeffer, W. J. Koros, *J. Appl. Polym. Sci.* **86**, 881 (2002).

56. R. Recio, *et al.*, *J. Appl. Polym. Sci.* **107**, 1039 (2008).

57. R. A. Patel, C. H. Borca, M. A. Webb, *Mol. Syst. Des. Eng.* **7**, 661 (2022).

58. R. Gómez-Bombarelli, *et al.*, *ACS central science* **4**, 268 (2018).

59. T. D. Loeffler, S. Banik, T. K. Patra, M. Sternberg, S. K. Sankaranarayanan, *J. Phys. Commun.* **5**, 031001 (2021).

60. T. K. Patra, V. Meenakshisundaram, J.-H. Hung, D. S. Simmons, *ACS Comb. Sci.* **19**, 96 (2017).