Predicting Critical Properties and Acentric Factors of Fluids Using Multitask Machine Learning

Predicting Critical Properties and Acentric Factor of Fluids Using Multi-Task Machine Learning

Sayandeep Biswas,[†,‡] Yunsie Chung,[†,‡] Josephine Ramirez,[†] Haoyang Wu,[†] and William H. Green[*,†]

[†]*Department of Chemical Engineering, Massachusetts Institute of Technology, Cambridge, MA, 02139, U.S.A*

[‡]*The authors contribute equally to this paper.*

E-mail: whgreen@mit.edu

Abstract

Knowledge of critical properties, such as critical temperature, pressure, density, as well as acentric factor is essential to calculate thermo-physical properties of chemical compounds. Experiments to determine critical properties and acentric factor are expensive and time intensive; therefore, we developed a machine learning (ML) model that can predict these molecular properties given the SMILES representation of a chemical species. We explored directed message passing neural network (D-MPNN) and graph attention network as ML architecture choices. Additionally, we investigated featurization with additional atomic and molecular features, multi-task training, and pre-training using estimated data to optimize model performance. Our final model utilizes a D-MPNN layer to learn the molecular representation and is supplemented by Abraham parameters. A multi-task training scheme was used to train a single model to predict all the critical properties and acentric factor along with boiling point, melting point, enthalpy of vaporization, and enthalpy of fusion. The model was evaluated on both random and scaffold splits where it shows state-of-the-art accuracies. The extensive data set of critical properties and acentric factor contains 1144 chemical compounds and is made available in the public domain together with the source code that can be used for further exploration.

1 Introduction

An equation of state (EOS) is a thermodynamic expression relating pressure, volume, and temperature. It provides a means to calculate thermo-physical properties of a chemical compound that are regularly used for various industrial applications, such as sizing equipment and process design.[1–3] Various EOSs have been proposed over the years to better describe the behavior of species starting from the Van der Waals EOS to the recent GERG-2008.[4] A common link between EOSs is the corresponding states theory, which uses the critical point as a reference to compute reduced states. Therefore, knowledge of critical temperature (T_c), pressure (P_c), density (ρ_c), and the acentric factor (ω) is required for many widely used EOSs such as Peng-Robinson[5] and Soave-Redlich-Kwong.[6] Other than EOSs, various models use critical properties as inputs to predict physiochemical properties such as diffusion coefficients,[7–10] surface tension,[11,12] and solubilities.[13–16] In addition, they are used to predict Lennard-Jones parameters which are required to model transport and collisions in a reaction rate calculation.[10,17–19] It is clear that critical properties and acentric factor are widely used in a multitude of fields, and thus it is important to have accurate values for the same. However, obtaining these molecular properties through experiments is time-intensive and expensive, and therefore, the development of computational prediction tools is necessary.

A widely used prediction approach for thermodynamic properties consists of the group contribution (GC) methods. They have been particularly popular for critical property estimation with leading models by Joback and Reid,[20] Han and Peng,[21] Nannoolal et al.,[22] and the Gani research group.[23] More recently, GC models for critical properties were proposed by Mansour and Korichi,[24] and by Tahami et al.[25,26] While GC methods are relatively easy to implement and interpret, they typically suffer from lower accuracy and poorer generalizability compared to more modern deep learning approaches. Recently, we evaluated GC and machine learning (ML) approaches to predict solvation free energy and solvation enthalpy for solute-solvent systems and found machine learning to be superior.[27] This finding is in line with Fu et al. who also showed ML to be superior when predicting the lipophilicity of molecules.[28] Given the recent success of ML for molecular property prediction, it is a promising option for critical properties and acentric factor. However, ML for predicting these properties has been lightly explored in the existing literature.

Several ML models[29–31] have been developed to predict various subsets of critical properties and acentric factor, but a single ML model that can predict all four properties has not yet been reported. All ML and GC models in the literature have been exclusively tested on random test-training splits, which assumes that chemical property prediction is an interpolation problem. In recent years, multiple publications have convincingly shown that most chemical prediction tasks are extrapolations.[32–38] Thus, actual model errors are significantly larger than those observed when random splits are used. It is important to test models against more rigorous splits such as scaffold or substructure splits that better serve as a true performance metric.

Additionally, most ML models in the literature primarily employ heuristically chosen molecular descriptors to construct feature vectors as compared to learning the optimal molecular representation for a given task. Findings by Yang et al. show that learned molecular representation outperformed fixed molecular fingerprints on several public data sets such as QM7, QM8, QM9, ESOL, FreeSolv, Lipophilicity, BBBP, PDBbind-F, PCBA, BACE, Tox21, and ClinTox.[32] Feinberg et al. also found learned representations to be superior, especially for scaffold splits, as they better generalize to molecules outside the training set.[39] Existing ML models often employ molecular descriptors that are not readily available, such as boiling point and specific gravity used by Varamesh et al.,[29] and quantum chemical (QM) descriptors used by Banchero and Manna.[30] While QM descriptors are more generalizable, they are computationally expensive to calculate and their accuracy relies heavily upon the accuracy of chosen QM methods.

In addition to model architecture, a key challenge to obtaining good ML performance lies in the lack of good-quality training data. Most models, including this work, have less than 1000 training data for critical property prediction, which is relatively small for machine learning applications to cover diverse chemical space. Data scarcity is a common problem faced in ML, and multi-task learning can help mitigate the problem in some cases. During multi-task learning, a single ML model is trained to predict multiple targets simultaneously rather than predicting each target using a separate model. The underlying assumption of multi-task learning is that it can leverage knowledge shared by other related targets and learn more generalized representations. The advantage of a multi-task approach has been demonstrated in many applications including drug discovery and bioinformatics.[35,37,39–42] The ML model for critical property and acentric factor prediction is expected to particularly benefit from multi-task learning as correlations relating T_c, P_c, ρ_c, ω, with boiling point (T_b) have been established in literature.[43,44]

In this work, we construct multi-task models with a graph convolution neural network (GCNN) to predict the three critical properties (T_c, P_c, ρ_c) and acentric factor (ω). Directed message passing neural networks (D-MPNN)[32] and graph attention networks (GAT)[45] are explored as the GCNN layers, which can learn an optimized latent space molecular representation during the prediction tasks. An exhaustive study is conducted on target grouping to optimize the performance of multi-task models. This includes introducing four additional chemical properties as auxiliary prediction targets, namely boiling point (T_b), melting point (T_m), enthalpy of vaporization (ΔH_{vap}), and enthalpy of fusion (ΔH_{fus}), to compare whether the model performs better with these auxiliary targets that are potentially correlated with critical properties.[46,47] We further evaluate the effect of passing additional atomic and molecular features, including 2D RDKit descriptors,[48] Abraham parameters,[27] QM descriptors, and 3D geometries, on the model performance. The effect of pre-training the model with a larger, estimated data set prior to fine-tuning it with experimental data is also investigated. Optimal GCNN layer type, target groupings, additional features, and pre-training set for the three critical property and acentric factor predictions are identified through rigorous comparison using random and scaffold split test sets. Open-source access to the final ML model, source code, and data set used in this study is provided.

2 Methods – Data Set and Data Split

2.1 Data Set Summary

The experimental data of critical properties and acentric factor are collected for 916 compounds from several public sources.[49–54] Data for four auxiliary phase change properties (T_b, T_m, ΔH_{vap}, ΔH_{fus}) are also collected as a part of our data set. The auxiliary properties are often published together with the critical properties, and some of these properties have been previously seen to be correlated.[46,47] The Pearson correlation coefficient matrix presented in Figure S2 of the Supporting Information reveals that strong correlations exist between certain targets, especially among T_c, T_b, and ΔH_{vap}. These auxiliary property data are therefore included as prediction targets for multi-task models. The data are collected for the compounds containing H, C, N, O, S, P, F, Cl, Br, and I atoms. The compiled data are standardized by converting the compound names and CAS numbers to SMILES and InChI using PubChemPy,[55] CIRPy,[56] and RDKit.[48] If multiple chemical identifiers are given by the original data source (e.g. both compound name and CAS number are available), the InChI strings converted from all chemical identifiers are compared to ensure they agree with each other. Mean values are used when multiple data are available for the same compound. The summary of the collected data is presented in Table 1. The data set contains a total of 5539 compounds, primarily due to a large number of experimental data available for T_b and T_m compared to critical properties. More details regarding the data collection, data distribution, and data statistics can be found in Supporting Information Section S1.

Table 1: The number of data points (N Total), mean values, standard deviations (Std. Dev.), and minimum and maximum (Min, Max) values in the compiled data set. This excludes the data from the external test set.

Data Type	Symbol (Unit)	N Total	Mean	Std. Dev.	Min	Max
critical temperature	T_c (K)	888	588.87	126.86	33.13	983.00
critical pressure	P_c (bar)	782	37.93	19.08	6.00	220.32
critical density	ρ_c (mol/L)	818	3.231	2.160	0.485	17.874
acentric factor	ω (-)	524	0.359	0.175	-0.215	1.389
boiling point	T_b (K)	5188	467.68	110.91	20.30	988.15
melting point	T_m (K)	3138	259.81	77.09	14.00	700.15
enthalpy of vaporization at boiling point	ΔH_{vap} (kJ/mol)	367	34.52	11.08	0.92	71.01
enthalpy of fusion at melting point	ΔH_{fus} (kJ/mol)	815	15.04	12.96	0.12	105.04
Total number of compounds for T_c, P_c, ρ_c, and ω only					916	
Total number of compounds					5539	

Among all the data sources used in this work, Yaws' handbook[49] contains the most amount of data and includes both experimental and estimated data for various molecular properties. The majority of the estimated data obtained from Yaws are computed using the Joback method, but the source of some data is unclear. Adding the estimated data to the training set would cause data contamination as they are not true experimental values; yet, excluding data points is not always the optimal choice especially when experimental data are scarce. To circumvent this issue, the estimated data from Yaws are used to pre-train the ML model prior to fine-tuning the model with experimental data. The details of the pre-training process are outlined in Section 3.4, and the summary of the pre-training data set is provided in the Supporting Information Section S2.

We additionally collected 276 T_c, 185 P_c, and 89 ρ_c experimental data from the published work by Tsonopoulos, Ambrose, and their coworkers,[57–62] and used them as an external test set to evaluate a final ML model. These data were compiled with an extensive comparison of multiple data sources and uncertainty analysis in their original work and also contain more recent experimental measurements compared to Yaws' handbook. Therefore, this data set can serve as a more accurate test set for a final model assessment. The summary of the data statistics that include these additional data can be found in the Supporting Information Sections S14. The total number of compounds for critical properties and acentric factor data becomes 1144 after including these data.

2.2 Experimental Uncertainties in the Data Set

Most of the data are obtained from Yaw's handbook, but it does not report any experimental uncertainties. However, a subset of the molecules contained in our data set have been explored in detail by Ambrose et al.,[61–64] Tsonopoulos et al.,[58,65,66] Gude et al.,[67] Daubert et al.,[68] Kudchadker et al.,[57] and Marsh et al.[59,60] who have reported experimental errors for each measurement. It should be noted that these error values were not obtained via statistical analysis, but via consideration of the range of values reported in the literature for each compound. The error reported for each molecule depends on the number of reported experiments for a given molecule. Therefore, molecules that are difficult to measure due to reasons such as a lack of commercial availability at high purity and instability at their critical points have larger errors. Hence, the experimental uncertainty does not scale with the absolute value. In addition, the variance amongst the literature values reported by different experiments is nearly always greater than the random error reported in each experiment. Given this, it is likely that the error source is systematic and varies based on the specific apparatus which was used to generate the data. We recommend the following references [57,59,60,63–68] for detailed information on the errors.

2.3 Data Split for Model Comparison

We employ random and scaffold splits to evaluate model performance on interpolation and extrapolation tasks, respectively. To create random splits, the data are randomly split into 90% training/validation sets and 10% test sets using 5 folds. All compounds in the test set are chosen to have at least two carbon atoms since GC and ML models are typically unsuitable for smaller compounds. In the case of scaffold splits, we first manually select substructures on which we can evaluate out-of-range model performance. Molecules containing any of the chosen substructures are identified and separated into a test set using RDKit and SMARTS strings of substructures. The substructures are grouped such that each test set is approximately 10% of the data set. A total of three test sets (three folds) were prepared spanning 16 different substructures for the scaffold split. For both splits, the remaining 90% of the data set is randomly split to form an 80 % training and 10% validation set. The validation set is used to determine the epoch when training is stopped. We used a randomly split validation set for the scaffold split task for two major reasons. First, we have a relatively small data set with low substructure diversity. In this case, the further removal of entire substructures to form a validation set greatly reduces the information contained in the training set, leading to worse performance. Second, if the validation set contains specific substructures there is a chance that the model trains to overfit to those substructures. Hence, the randomly split validation set is used to determine early stopping for both the random and scaffold split models.

Within each fold, we generate an ensemble of 25 models that use different random seeds for parameter initializations and training/validation splits. Thus, the prediction for each test set is the average prediction from the ensemble of 25 models. The ensembling approach has been previously demonstrated to result in improved model accuracy,[32,35,69] in addition to serving as a measure of epistemic uncertainty which can be calculated from the sample standard deviation across all ensembles.[70] Optimal target groupings and additional features are chosen using only the first fold for both random and scaffold splits. Hyperparameters are optimized using only the training set of the first fold. The final model performance is then evaluated using all test folds (5 folds for the random split and 3 folds for the scaffold split) by computing the mean and the sample standard deviation across the folds.

We ensure that the molecules chosen for the test sets or molecules containing the substructures in the test set are removed from the pre-training set (estimated data) as well. It should be noted that the test sets consist only of experimental data. The list of the substructures chosen for the scaffold splits can be found in Supporting Information Section S3. All test and training sets are also provided as a part of the Supporting Information.

2.4 Evaluation Metric and Scaling Data

Root mean square error (RMSE) is used as a primary evaluation metric for model performance. Scaling is essential to prevent biasing the model towards targets that have larger numeric values. Therefore, all 8 targets are scaled (Z-scored) by subtracting the mean value and dividing by the standard deviation prior to any model training, and Z-scored RMSE is used for calculating the loss. Throughout our study, the Z-scored values are used to compare various ML design choices. For the final error report, the scaled prediction values are transformed back to the original unscaled values, and mean absolute error (MAE) and mean absolute percent error (MPE) are additionally computed.

3 Methods – Models

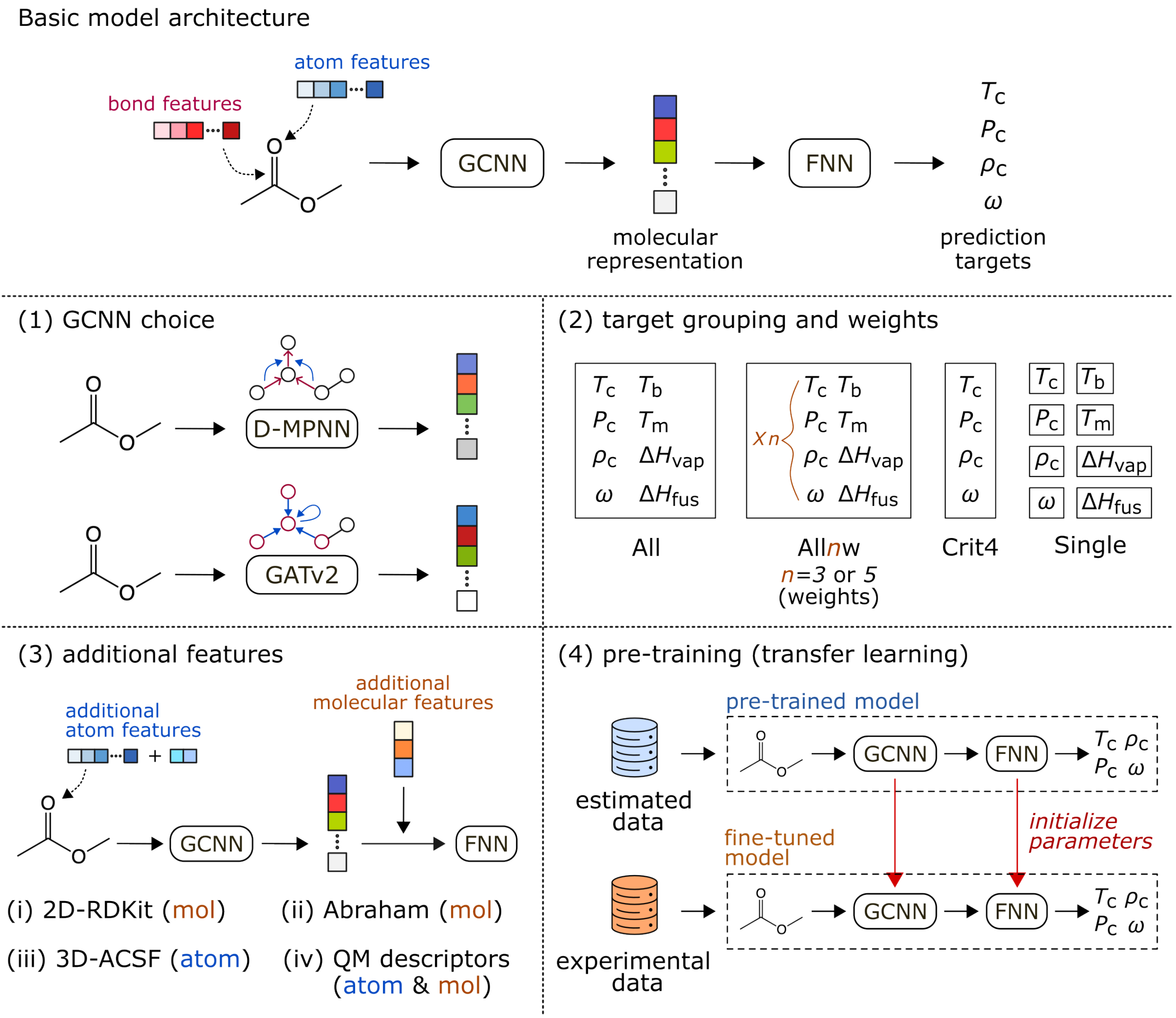


Figure 1: Overview of the ML model architecture and different design choices used in this work.

An overview of the ML model architectures and different design choices is illustrated in Figure 1. Four different design choices are considered: (1) the type of GCNN employed, (2) target groupings for multi-task models, (3) additional atomic and molecular features, and (4) transfer learning approach with the estimated data. Each of these design choices is described in the subsequent subsections.

3.1 Neural Network Architectures

The neural net architectures used in this work can be divided into two parts: an embedding layer, and an output layer. The embedding layer aims to learn the molecular representation from the atom and bond feature vectors derived from the SMILES string. Different architectures can be employed to learn molecular representation; in this work, we explore Directed Message Passing Neural Network (D-MPNN) and Graph Attention Networks (GATv2). This learnt representation is then passed through a fully connected feed forward neural network (FNN) that outputs the prediction targets. Both D-MPNN and GATv2 models take the SMILES string of a compound as an input and generate a graph structure of a molecule with a set of initial atom and bond features. The featurization process used to convert atoms and bonds into vectors has been described in detail by Yang et al.[32] A list of all atom and bond features is provided in Supporting Information Section S4. In addition to the vanilla representation, we explored additional atom and molecular features specific to the critical property and acentric factor prediction task as detailed in Section 3.3.

There are several hyperparameters associated with the neural network models used in this work. Hyperparameters are separately optimized for D-MPNN and GATv2 based networks using the software package Hyperopt[71] and Optuna,[72] respectively. Table S6 in Supporting Information summarizes the different sets of optimized hyperparameters that are used to train the models.

3.1.1 D-MPNN

Directed message passing neural network (D-MPNN) is a type of graph convolution model that uses hidden states and messages associated with directed bonds (edges) instead of the atom (nodes)-based message passing approach. The D-MPNN model is constructed using the open-source Python code Chemprop (`https://github.com/chemprop/chemprop`).[32] D-MPNN is selected because several studies have demonstrated its excellent performance for molecular property prediction.[27,32,73] For more details on the D-MPNN model, the reader is referred to the dedicated work by Yang et al.[32]

3.1.2 GATv2

Graph Attention Networks extend the base graph convolution model by leveraging self-attention layers. It works by performing a linear transform on the input nodes and uses the output of this transform to calculate attention coefficients. The update functions and the calculation of attention coefficients are described by Brody et al.[74] GATv2 used in this work employs a dynamic graph attention variant that is more expressive than the static attention mechanism of the traditional GATs. The GATv2 layer has an additional hyperparameter for multi-head attention, wherein several attention mechanisms run in parallel before they are concatenated and linearly transformed to the expected dimension. Multi-head attention allows the model to jointly attend to information from different representation sub-spaces.[75] The GATv2 layer is imported from PyTorch geometric.[74,76] The GATv2 model is constructed using the open-source Python code Mlprop (`https://github.com/Sayandeep00/chiral_gnn/tree/ml_prop`). Mlprop was developed using chiral_gnn[77] as a base. Several aspects of Mlprop such as featurization, ensembling, and use of additional molecular and atom features were inspired by Chemprop, but it boasts the added flexibility of switching between different GNN layers that can be imported from the PyTorch libraries.[32,77]

3.1.3 Baseline Models

There are a few challenges in performing a fair comparison between the models proposed in this work and other models. First, the data sets vary significantly in terms of size, diversity, and handling of computational/predicted data. None of the existing publications has provided means of recreating the exact data set used in their respective studies. Therefore, directly comparing reported errors across existing literature is an infeasible option. Additionally, we test our models on scaffold/substructure splits, which have not been considered in previous studies that predict critical properties. Hence, there are no reference values for our scaffold split results. In addition, several existing models in literature have problems such as minor data leakage caused by using a test set to determine early stopping,[30,31] unclear data set selection criteria,[24,25] and lack of information to enable retraining.[30] Therefore, it is necessary to obtain baseline models using our data set for a fair comparison. We compare our models with the group contribution methods developed by Joback[78] and by Nannoolal et al.,[22] and a radial basis function neural network with Morgan fingerprints based on the models from the literature.[30] The radial basis function layer is imported from the referenced GitHub repository[79] and implemented using the GATv2 framework.

3.2 Multi-task Learning and Target Weights

There are a total of 8 targets to predict, 4 of which are the critical properties and acentric factor (main targets) and the other 4 are phase change properties (auxiliary targets). Several studies indicate that multi-task learning with related targets can improve generalization and prediction accuracy.[41,42] Many empirical correlations imply the underlying link among T_c, T_b,

P_c, ρ_c and ω,[43,44] and therefore, multi-target training is expected to help model predictions. Several target groupings are explored to optimize model performance. The baseline model includes training 8 individual models for each of the molecular proprieties shown in Table 1. Multi-target training is performed on groups that include: 'Crit4' - all the critical properties and acentric factor $(T_c, P_c, \rho_c, \omega)$, and 'All' - all 8 targets. The data set for multi-target training is often sparse, and hence masking is applied to compute the loss. One issue with our multi-task approach is that the trained model may get biased towards a certain target with much more data points. In our case, there is significantly more data for melting and boiling points as shown in Table 1. We explore using different target weights for the loss function as a means to correct for this data imbalance. Weighted data sets include: 'All3w' and 'All5w' where the loss values computed for the three critical properties and acentric factor (Tc, Pc, ρ_c, ω) are weighed by 3 and 5 times greater, respectively, than the loss used for the other four auxiliary targets (T_b, T_m, ΔH_{vap}, ΔH_{fus}).

3.3 Additional Features

The following additional features are examined for our models: 2D-RDKit (molecular), Abraham (molecular), 3D-ACSF (atomic), and QM descriptors (atomic & molecular). Additional atomic features are concatenated with a default feature vector of each atom before being passed to the embedding layer. On the other hand, molecular features are appended to the learned representation (output of the embedding layer) prior to the feed forward neural network. A detailed description of each feature is provided below.

3.3.1 RDKit

Additional 2D molecular features generated by RDKit[48] are explored. RDKit provides a total of 200 2D descriptors, which have been shown to improve the model performance for certain property predictions.[27,32] These are filtered down to the 14 most relevant features for our models based on variance threshold and random forest methods. A list of the 14 selected RDKit features is provided in Supporting Information Section S6. The RDKit features are normalized using the DescriptaStorus package.[80]

3.3.2 Abraham

Abraham solute parameters predicted using the SoluteML model by Chung et al.[27] are explored as additional molecular features. The Abraham parameters are chosen as one of the optional features for our models as a correlation by Li et al.[81] has revealed a direct relationship between T_c and the Abraham parameters. The Abraham parameters consist of five descriptors, E, S, A, B, and L, each of which is associated with a different physical property of a compound.[82] Normalized values are used for our ML models.

3.3.3 QM Descriptors

QM atomic, bond, and molecular descriptors are calculated using an improved version of the automated workflow previously developed by our group.[83] Compared to the previous version, the capability to calculate more atom and bond level descriptors and better support for charged molecules are added. In this work, 3D conformers of the molecules are generated from SMILES strings and then screened using the MMFF94s[84] force field in RDKit. The conformer with the lowest MMFF94s energy is further optimized at GFN2-xTB[85] level of theory followed by a frequency calculation at the same level and three DFT single point calculations (i.e. neutral, cation +1, anion -1) at B3LYP/def2-SVP[86] level of theory in Gaussian 16.[87] NBO 7.0[88] is then used to compute natural bond orbitals and associated descriptors. Results from the QM calculations are processed to obtain the desired descriptors using scripts in the automated workflow. To ensure the convergence of final optimized geometries, structural and vibrational frequency checks are implemented throughout the workflow.

While many different QM atomic, bond, and molecular features are computed, using all features is not desirable since irrelevant features can introduce noise or distract the models. Therefore, only two descriptors are selected as the final QM descriptors: Hirshfeld charge (atomic) and Mulliken total dipole moment (molecular). These two features are chosen because critical properties are associated with intermolecular forces, which are linked to the polarity and dipole moment of a compound. Out of the 5539 compounds in the data set, 85 compounds failed structural optimization at the GFN2-xTB level. These compounds are omitted from the training/validation and test sets of the ML models that use QM descriptors as additional features. A complete set of calculated QM descriptors with their values are provided as a part of the Supporting Information, and the automated workflow program used for the QM calculations is available through GitHub.[89]

3.3.4 3D-ACSF

Additional atomic features are generated using the converged 3D geometries obtained from the QM descriptor workflow described in the previous section. The 3D coordinates of each molecule are converted to atomic 3D features using atom-centered symmetry functions (3D-ACSF).[90] ACSFs use multiple many-body functions to encode a local environment near an atom within a molecule. Zhang et al. employed 3D-ACSF descriptors as initial atomic features for their GNN models to predict aqueous solvation free energy and showed the prediction improved compared to the baseline model that only used 2D featurization.[91] We adopt the same two-body symmetry functions as those used by Zhang et al. to generate 3D-ACSF atomic features. These functions give a total of 260 features for each atom within a molecule. Because the majority of the 3D-ACSF features are found to be zero for the compounds in our data set, only the 33 most relevant, non-zero features are selected as final 3D-ACSF atomic descriptors. Similar to the QM features, the 85 compounds that failed the geometry optimizations are omitted from the training and testing sets of the ML models with 3D-ACSF features.

3.4 Pre-training with Estimated Data (Transfer Learning)

Pre-training has been previously found to be beneficial when training ML models for the chemical space.[35,73,92] Therefore, we investigate the extent of improvement in model performance on experimental data when we pre-train using the estimated data. Pre-training allows the model to leverage the estimated values or the data from unclear sources while avoiding the risk of data contamination. In this transfer learning approach, the estimated data set from Yaws[49] is used to pre-train an ML model, and the optimized parameters from the pre-trained model are subsequently used to initialize the parameters for a new model that is fine-tuned with the experimental data set. None of the parameters are kept frozen in the fine-tuning step. The baseline models are trained only with the experimental data for comparison.

4 Results and Discussion

4.1 Results on Multi-Task Models

First, we investigate the impact of the multi-task learning approach. Figure 2 shows the scaled RMSE errors of each target for the D-MPNN models trained on different target groupings ("All", "Crit4", "Single"). Multi-task models ("All", "Crit4") have lower errors across most targets for both random and scaffold splits compared to the single-task models ("Single"). Therefore, the benefit of using multi-task training is evident. This observation is supported when investigating the Pearson correlation amongst the different targets as shown in Figure S2. We see that the targets have a strong correlation overall, and therefore, learning about one target helps the prediction on others. It also benefits from a more generalizable model that has a lower tendency to overfit on a specific task. Additionally, multi-task models benefit from seeing more diverse molecules as T_b and T_m data have a lot more compounds than the critical properties and acentric factor. Similar results are obtained for models using GATv2, where the multi-task training using "All" yields lower losses than "Crit4", which has fewer targets. The results of the GATv2 models are shown in Figure S4 of the Supporting Information.

We next investigate the effect of using different target weights in a loss function as a means of balancing the data sets. As mentioned previously, T_b and T_m have substantially more data points than the rest of the properties, and using an equal weight for all targets may cause the model to be predominantly trained on the targets with more data. Therefore, we perform a coarse search where the training loss arising from the three critical properties and acentric factor is scaled by a factor of 3 for "All3w" and 5 for "All5w". Figure 2 shows that

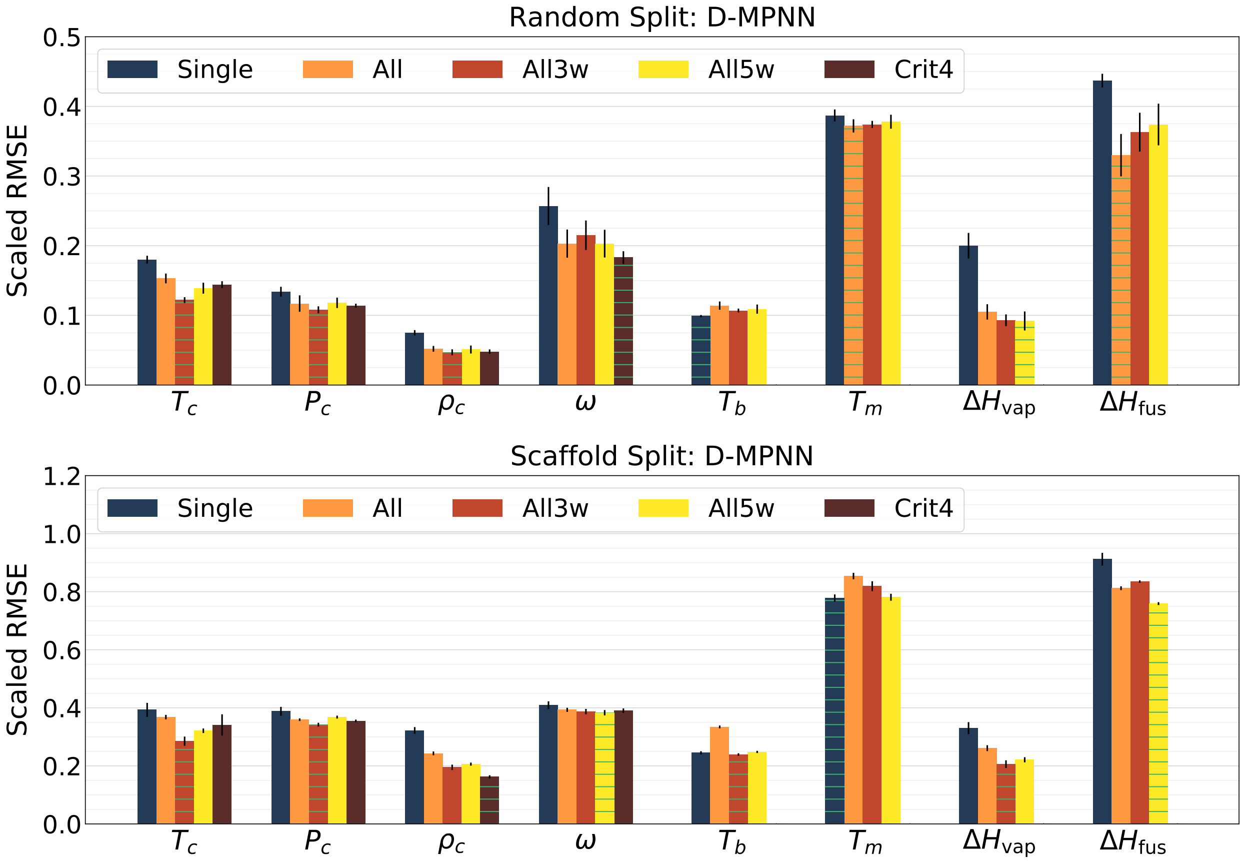


Figure 2: Comparison of the multi-task learning approach for the D-MPNN model. Scaled RMSE of each target is evaluated for different multi-task models using the random and scaffold split test sets 1. The best performing model is filled with horizontal lines for each target. The legend represents different target groupings and target weights used for each model.

the errors of the "All3w" across almost all critical targets has an overall better performance compared to "All" for both random and scaffold splits. Further indication of having a better model comes from the observation that "All3w" outperforms "All" even on most of the ancillary targets ($T_b, T_m, \Delta H_{\text{vap}}$) which were de-prioritized by adding weights. This result demonstrates that in order to maximize the benefits of multi-task learning, it is important to appropriately weigh the tasks such that each target is well represented in the loss function. The results of the GATv2 models (Figure S4 in the Supporting Information) however, do not have the same result as "All" is still the best performing target group.

Comparing the base case ("Single") to the optimized target grouping ("All3w"), we see the largest improvements for T_c and ρ_c. In case of T_c, it is likely that the large T_b and T_m data have greatly benefited the predictions as they are strongly correlated with T_c. The benefit for ρ_c seems to primarily come from co-training with other critical properties and acentric factor as we do not see much improvement when comparing "All3w" to "Crit4".

According to the Pearson correlation coefficients (Figure S2), ρ_c does not appear to have strong correlations with other properties, but it still greatly benefits from the multi-task approach. This is also seen for P_c, which benefits from multi-task learning despite lacking strong correlations. Nonetheless, the Pearson correlation coefficient is only a measure of linear relationship, and it is possible that some underlying non-linear relationships among the targets could be captured by our multi-task models. In case of ω, the multi-task models also perform better than the single-target model for the random split but negligible difference among the target groupings is observed for the scaffold split.

4.2 Results on Additional Features

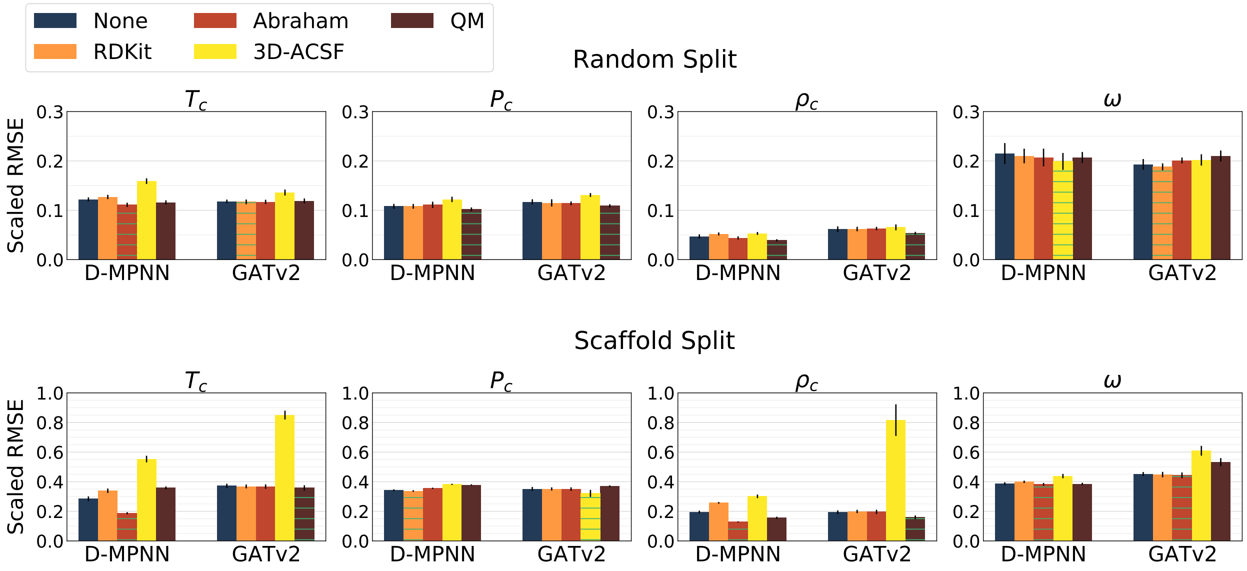


Figure 3: Scaled RMSE of each target for different additional features tested on the random and scaffold split test sets 1. The best performing model is filled with horizontal lines for each target. D-MPNN uses “All3w” and GATv2 uses “All”.

Several feature combinations were explored to further improve model performance as explained in Section 3.3. Figure 3 shows the results obtained from the feature study for both D-MPNN and GATv2. In case of D-MPNN, we see that the Abraham parameters give the best model performance. Its benefits are particularly evident when focusing on the scaffold split results, where we see a significant reduction of T_c and ρ_c errors. Neither Abraham nor the other features improve the predictions of P_c and ω, which imply that the optional features do not provide any additional information for these two targets.

For GATv2, we see that additional parameters do not help predictions. Upon initial investigation, we observed that the latent vector representations from the two ML models have relatively different magnitudes. For example, the latent vector of the GATv2 is $\mathcal{O}(10)$ while that of the D-MPNN is $\mathcal{O}(1)$ - based on the averaged first norm value across the compounds in the scaffold split test set. We initially hypothesized that the model tends to ignore the additional features that are an order of magnitude smaller than the other entries in the latent vector. Therefore, an additional model was tested where Abraham parameters were re-scaled to match the GATv2 latent representation scaling. Surprisingly, this model did not have a statistically significant improvement in the model performance. Several alternative hypotheses are therefore considered to explain the result. First, the latent representation of the GATv2 model may already capture the trends that are covered by the additional parameters explored in this work. Therefore, no new information is added when the additional features are concatenated to the latent representation, leading to no improvement in performance. This would suggest that successful feature selection depends on both the task and the model architecture. Another reason can be the difference in latent size that was used for the D-MPNN and GATv2 architectures. GATv2 employed a latent vector of length 1200, while D-MPNN uses a latent vector of length 300. The additional features have a much smaller length compared to the latent vector from the GATv2 and their effects may have consequently diminished. The choice of latent vector size is a hyperparameter, and ideally one would redo hyperparameter optimization for each set of additional features to obtain the best performance. However, given the high computational costs, we did not re-optimize the hyperparameters in this work.

Given that the critical properties are related to intermolecular interactions, we hypothesized that 3D structural information may improve the predictions by providing relevant information that cannot be described by the default 2D atom and bond features of our GCNNs. However, strikingly poor performance is observed on the scaffold split when the 3D-ACSF features are employed. The effect is more pronounced for GATv2, but the same trend is observed for D-MPNN. The 3D-ACSF features appear to introduce noise into our models, and thus it is possible that the 3D structure is not as relevant for the critical property prediction. Another potential reason is that the ML models are very sensitive to errors in the optimized 3D geometries. The sensitivity of a ML model to the input 3D structure is demonstrated by Spiekermann et al. in their work for reaction barrier height prediction.[35] Molecules can have many different structural conformations, and although a conformer search was performed in our work to find the lowest energy conformer, it is challenging to search the entire conformational space due to a high computational cost. Some important conformers may have been overlooked in our calculations, causing the 3D-ACSF features to perform poorly. It is also possible that the 3D-ACSF factors might require re-optimization of hyperparameters as it does change the featurization significantly. Moreover, an alternative method to encode the 3D information from the optimized geometries may be more suitable for the property predictions. The QM data set used to calculate the 3D-ACSF factors is provided open access and can serve as a good basis for further investigation in this matter.

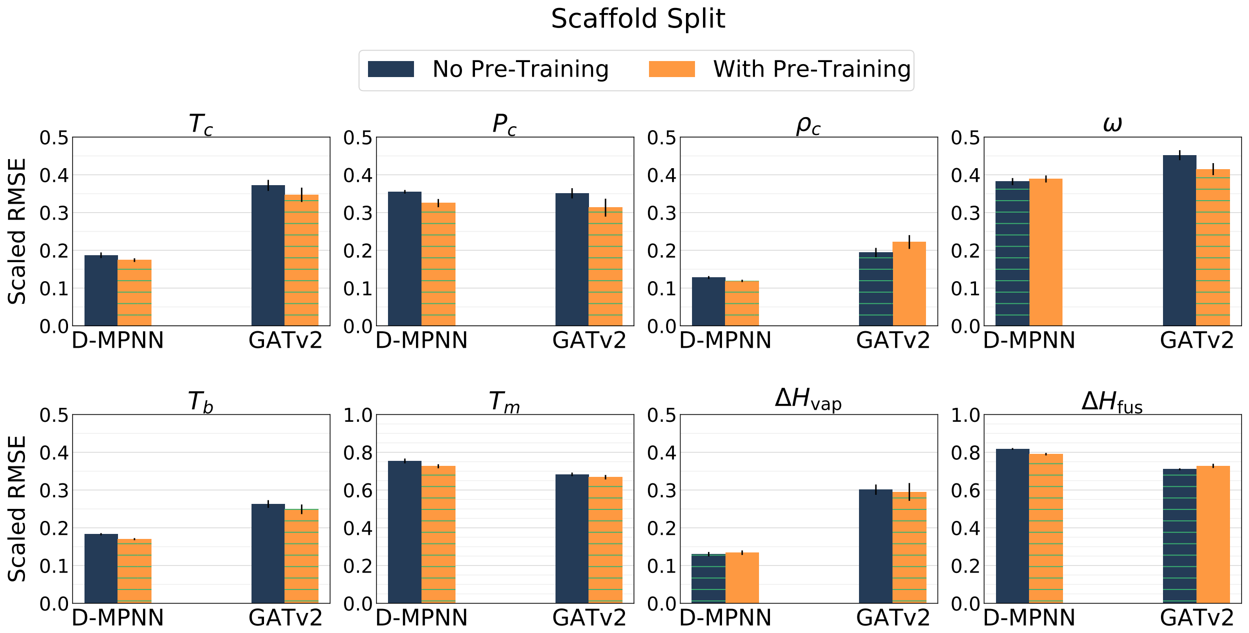


Figure 4: Scaled RMSE of each target for various models tested with and without pre-training. The errors are evaluated on the scaffold split test set 1. The best performing model is filled with horizontal lines for each target. D-MPNN uses "All3w" with Abraham features, and GATv2 uses "All" without additional features.

4.3 Results on Pre-Training

The effect of the pre-training on the scaffold and random splits are shown in Figure 4 and Figure S5 (see the Supporting Information), respectively. For the random split, we do not observe a statistically significant improvement in predictions when the models are pre-trained with the estimated data set. The minimal benefit for random split is a recurring observation that is previously seen during the additional feature study. However, the benefits of pre-training is evident for scaffold splits as the errors for both D-MPNN and GATv2 decrease when pre-training is used. Given its benefits, pre-training serves as a great way to use estimated data, or data from unclear sources (experimental or predicted). Particular to our study in which limited experimental data are available for the critical properties, existing group contribution methods such as Joback can be effectively used to create a pre-training data set. It is especially useful for GCNN models employed in this work since they are large models that aim to learn the molecular representation as compared to using fixed descriptors.

In this study, a relatively simple pre-training scheme was employed as we used the pre-trained model to initialize the parameters of a new, fine-tuned model. This already improves performance, but there are multiple approaches to further optimize the pre-training task. For example, different learning rates can be used during pre-training and fine-tuning, or certain layers can be kept frozen in the fine-tuning step. We believe there is great potential to effectively use pre-training for property prediction tasks where we often have pre-existing models along with a dearth of experimental data.

4.4 Final Models

Table 2 reports the errors from the best D-MPNN and GATv2 models for the three critical properties and acentric factor. The results for the other four targets (T_b, T_m, ΔH_{vap}, and ΔH_{fus}) can be found in the Supporting Information Table S8. The best D-MPNN network uses the target grouping "All3w" and the Abraham features, and is pre-trained with the estimated data set. The best GATv2 model uses the target grouping "All" without any additional features, and is also pre-trained with the estimated data set. For scaffold splits, D-MPNN outcompetes GATv2 on T_c, while performing similarly for P_c, ρ_c, and ω. In the case of the random splits, we see a better performance of D-MPNN for T_c and ρ_c, while the errors for other targets are similar. Overall, the D-MPNN model has superior performance as the mean errors are consistently lower for all target-split combinations. Based on the results obtained during the optimization of model choices (target grouping, additional features, pre-training), we see that performance on random splits does not change significantly with the design choices. On the other hand, scaffold split results are sensitive, thus serve as a better metric on which to evaluate and optimize the design choices.

Table 2: Error summary for the final D-MPNN and GATv2 models. The errors are computed on 5 different test sets for the random split, and 3 different test sets for the scaffold split. We report the mean and sample standard deviation (in parenthesis) of errors for both splits. The sample standard deviation of errors are computed over 5 different test sets for the random split, and are computed over 16 different substructures for the scaffold split. Both RMSE and MAE are reported in the same unit as each target.

		D-MPNN			GATv2		
Target	Split	RMSE	MAE	MPE (%)	RMSE	MAE	MPE (%)
T_c (K)	Random	13.2 (2.6)	7.6 (0.8)	1.30 (0.13)	19.6 (5.2)	11.6 (1.7)	2.01 (0.31)
	Scaffold	20.1 (5.6)	16.2 (4.3)	2.44 (0.89)	37.2 (5.4)	28.1 (4.2)	4.95 (1.15)
P_c (bar)	Random	2.09 (0.17)	1.31 (0.09)	3.82 (0.31)	2.21 (0.27)	1.46 (0.10)	4.48 (0.46)
	Scaffold	4.46 (2.73)	3.21 (1.34)	8.96 (5.64)	7.69 (1.17)	4.13 (1.01)	9.89 (2.14)
ρ_c (mol/L)	Random	0.120 (0.031)	0.075 (0.009)	2.78 (0.32)	0.141 (0.022)	0.097 (0.011)	3.67 (0.45)
	Scaffold	0.247 (0.159)	0.177 (0.103)	6.93 (4.86)	0.543 (0.159)	0.357 (0.156)	10.56 (3.17)
ω (-)	Random	0.0511 (0.0245)	0.0277 (0.0059)	8.74 (1.38)	0.0486 (0.016)	0.0305 (0.005)	9.77 (1.39)
	Scaffold	0.0494 (0.0336)	0.0401 (0.0246)	12.21 (9.02)	0.0667 (0.0158)	0.048 (0.0119)	16.07 (1.42)

Principle component analysis (PCA) is performed on the latent representation for both

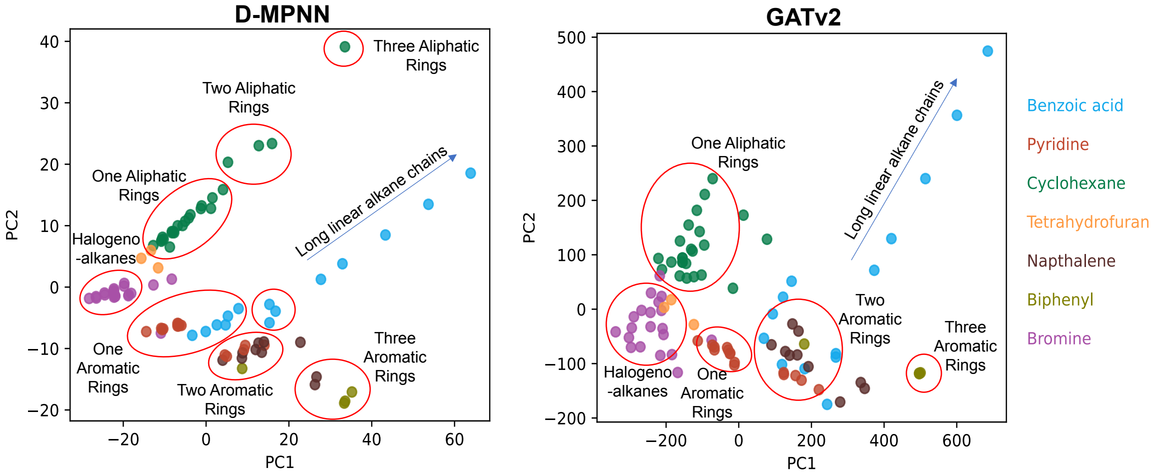


Figure 5: Principal component analysis was performed on the latent representation obtained for the D-MPNN and GATv2 models on scaffold split test set 1. 2D PCA analysis demonstrates the importance of molecular size, aliphatic and aromatic nature of the molecule towards critical properties, based on the clusters found in the reduced latent space. The first two PCA directions explain 86% and 85% of the total variance for D-MPNN and GATv2 respectively.

D-MPNN and GATv2 as shown in Figure 5 for the scaffold split test set 1. Visually, D-MPNN is able to form better clusters which can explain better predictions. Additionally, both models identify molecular weight, aliphatic, and aromatic nature of the molecule to be important factors when predicting critical properties. This is in agreement with basic chemical intuition.

To further understand the sources of error and better explain the difference in performance between the two architectures, we evaluate the error per substructure and error by the number of carbons for all targets. In Figure 6, we see higher errors for bromine and chlorine across all targets for both D-MPNN and GATv2 when compared to the other substructures. This result is expected as bromine and chlorine are out-of-range atoms, i.e. they do not appear in the training set. In case of T_c and ρ_c, D-MPNN performance is vastly superior to GATv2 for chlorine and bromine. However, this is not sufficient evidence to claim that D-MPNN performs better for out-of-range atoms given the similar performances for P_c and ω. Figure S6 in the Supporting Information shows that both D-MPNN and GATv2 prediction is worse for molecules with fewer carbon atoms on the scaffold split. This effect is most prominent when molecules have zero or one carbon for which the error is significantly higher across all targets. This observation can be explained when one considers the distribution of the available training data as shown in the Supporting Information Figure S1. The molecular weight distribution shows that there are only a few small molecules in the data set. Additionally, molecules containing no carbon atoms (inorganic molecules) share little commonality with organic molecules. Therefore, there is limited knowledge gained for the inorganic compounds when the training data are primarily comprised of organics. It can be seen that there are a few data points for large molecules, but the predictions for them are good. This result is best interpreted as the larger molecules being constructed by several smaller molecules in an additive manner- the principal idea of GC methods. Therefore, if the training data contain the small molecules that are the building blocks for the larger molecules, the ML model is able to provide good predictions. For most use cases, molecules of interest are primarily organic and have more than two carbon atoms, and therefore the error reported in Table 2 should be an overestimate. When comparing D-MPNN and GATv2 for the small molecules, we see that there is no clear winner across all targets as we see overlapping error bars. Therefore, neither architecture is particularly favored to generate predictions for smaller molecules with less than two carbons.

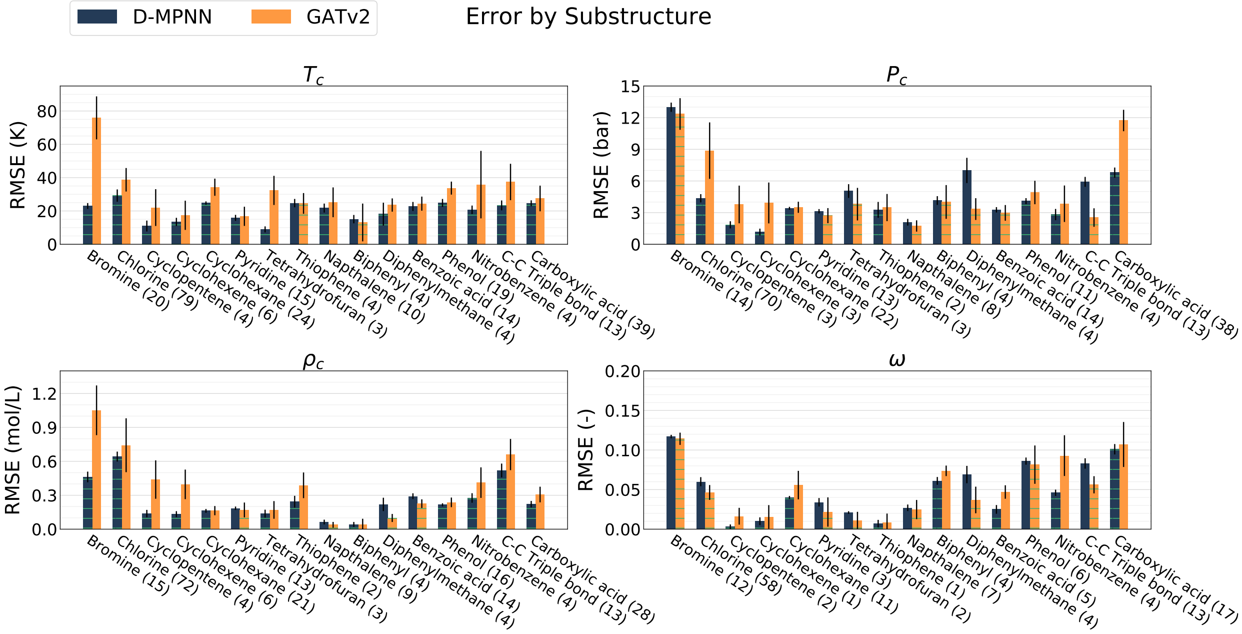


Figure 6: Final error comparison per scaffold. The numbers in the parenthesis refer to the total number of molecules in the test set for the particular scaffold.

The final D-MPNN and GATv2 models are compared with the baseline RBFNN model and the GC methods by Joback and by Nannoolal et al. in Supporting Information Figure S7. We see that the D-MPNN model outcompetes all other models and is superior to the RBFNN and GC methods for the majority of the targets. In the case of scaffold splits, we see lower errors for P_c and ρ_c from the Joback GC method and lower errors for ρ_c and T_b from the GC method by Nannoolal et al. However, unlike the ML models, these GC methods are trained on different data sets which are highly likely to have some overlap with our scaffold test sets (e.g. cyclohexane, pyridine, bromine, etc. See SI Figure S3.). Therefore, a direct comparison between the performance of ML and GC methods is not possible for the scaffold splits. The RBFNN model constructed using the Morgan fingerprint with a radius of 2 has the largest errors for nearly all targets in both random and scaffold splits. This demonstrates that the learned latent vector from the GCNN model provides a much better molecular representation than fixed fingerprints for property predictions.

Figure 7 shows the learning curve, a log-log plot showing the relationship between model performance and training data set size for both random and scaffold splits. For the random splits, RMSE for both T_c and ρ_c decrease linearly with increasing data set size, indicating that adding more data to the training set will improve performance on these targets. In the case of P_c, we see that beyond 100 data points in the training set, there is no significant reduction in RMSE indicating that the model has likely reached the aleatoric limit. This is further corroborated by the experimental errors reported in references [57,59,60,63–68] where errors for P_c are $\mathcal{O}(1)$. Therefore, further improvement for P_c would require the collection of experimental data with lower uncertainties. For ω we see that there is a decay in the slope of the learning curve but it has not flattened like P_c; thus, there might be performance improvements if more data are included but it is likely to be close to the aleatoric limit. The learning curve for scaffold splits, as expected, is noisier than random splits but displays a general trend of improving model performance with more training data. Additionally, the higher errors in extrapolative predictions are an indication of poor coverage of the chemical space. This lack of data coverage is further supported by the larger standard deviation in model performance observed for the different test sets compared to each ensemble. It is clear that increasing the size of the training data set will help model performance.

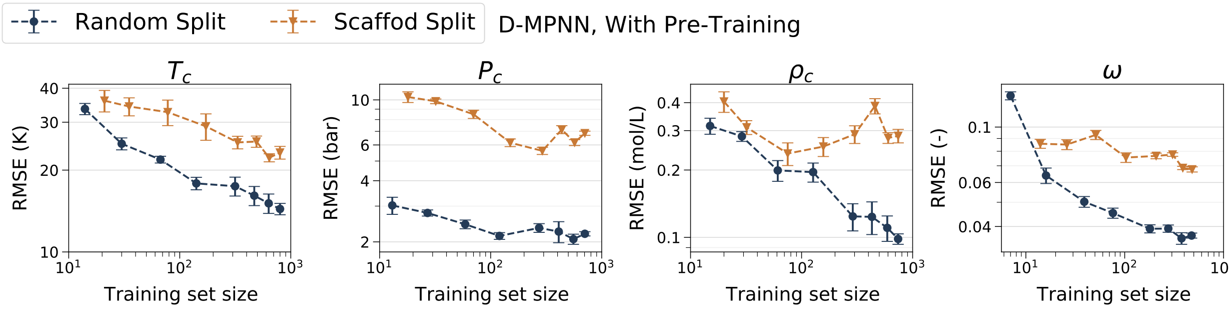


Figure 7: The RMSE of the D-MPNN models trained with 2, 5, 10, 20, 40, 60, 80, and 100 % of the training set. The models are trained with “All3w” target grouping, Abraham features, and pre-training.

However, it should be noted that improving data coverage is not just about including more data points but also ensuring that the additional data include molecules that are sufficiently different than the current data set. Collecting more experimental data would be ideal but experimental measurements are costly and time-consuming. Furthermore, there are limitations on the molecules whose critical parameters can be measured experimentally with sufficient accuracy, primarily due to practical challenges such as high reactivity and instability at the critical point, or inability to get pure samples which inhibit accurate experimental measurements. As an alternative approach, computational methods such as Gibbs-ensemble Monte Carlo (GEMC) simulations can be used to exponentially grow the data set in the future.[93,94] Transferable potentials for phase equilibria (TraPPE) is a common family of force fields used for this application. This family contains several force fields that were trained on specific molecular subsets.[95–103] GEMC is able to get high accuracy of prediction of critical parameters, often within experimental uncertainty limits, but it is computationally expensive and slow to perform. Therefore, it is not a replacement for fast-solving ML models but an effective method for increasing the size of the existing data set to further improve ML performance.

The parity plots of the best D-MPNN model on both random and scaffold splits are shown in Figure 8. The parity plots for the other four targets (T_b, T_m, ΔH_{vap}, and ΔH_{fus}) can be found in the Supporting Information Figure S8. It can be seen that the predictions closely lie on the parity line for the majority of the targets. Relatively larger deviations are observed for the acentric factor on the scaffold split. This is as expected as the acentric factor has the fewest number of data compared to the critical properties. Nonetheless, the scaffold split is designed to be more challenging, and our model is still able to provide reasonable predictions for the acentric factor. The result of the best D-MPNN model on the external test set that consists of the data from 57–62 is presented in the Supporting Information Table S9. The RMSE of the model predictions on T_c, P_c, and ρ_c are 28.6 K, 4.01 bar, and 0.182 mol/L, respectively. The R^2 (coefficient of determination) values on T_c, P_c, and ρ_c are 0.96, 0.88, and 0.96, respectively, which are similar to those of the scaffold split test set.

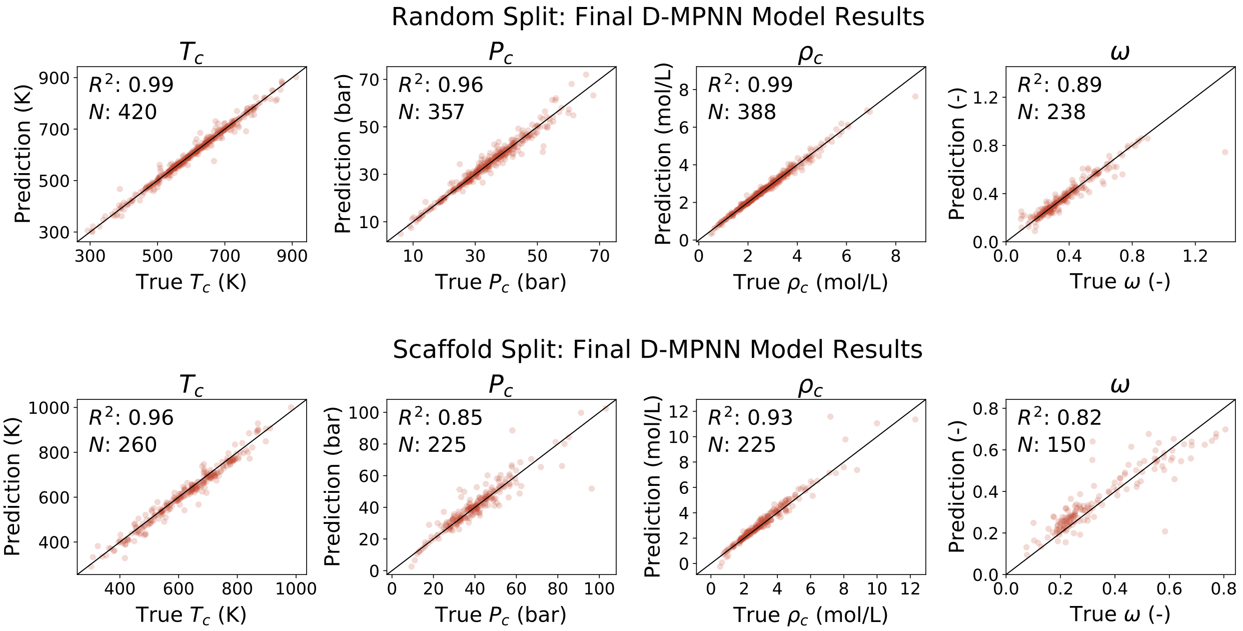


Figure 8: Final D-MPNN parity plots. R^2 and N represent the coefficient of determination and the total number of test data, respectively.

The parity plots clearly show some outliers (fail cases) across all targets, especially for scaffold splits. The outliers include molecules such as tetrabromomethane, tetrachloroethylene, hexachloroethane, nitrosyl chloride, chlorine, hydrogen bromide, and hydrogen chloride. This is expected as the bromine and chlorine were not present in the training set of the scaffold split, and each of these compounds contains a high fraction of out-of-range atoms. Another class that is poorly predicted includes small acids such as formic and acetic acid. Ambrose et al. performed the experimental measurements and report that both formic and acetic acid do not conform to general correlations.[104] It is suspected that the special behavior is due to dimerization effects. Finally, we also have fluorocyclohexane which had significant errors for both T_c and P_c. The poor performance was unexpected as cyclic rings were present in the training set, the molecule is relatively large, comprised majorly of carbon, and does not have any unique effects such as dimerization. The experimental values reported by NIST for T_c and P_c are 667.93 K and 51.7259 bar respectively, while our model predicts 567.04 K and 41.11 bar.[52] We considered other sources that report critical properties of fluorocyclohexane, and found that ThermoDataEngine (TDE) reports a critically evaluated value (generated through assessment of available experimental and predicted data) of 575±16 K and 37.7±24 bar for T_c and P_c respectively.[105] The TDE values are in good agreement with our predictions, leading us to believe that there were large experimental errors.

5 Conclusions

We developed a machine learning model to predict the critical temperature, pressure, density, and acentric factor starting from a SMILES representation of a chemical species. It was tested on both random and scaffold splits. To our knowledge, this is the first time that prediction errors have been reported for scaffold splits within the critical property literature. The model achieves state-of-the-art accuracies on both random and scaffold splits when compared to baseline models. We explored various design options to optimize model performance. D-MPNN and GATv2 were considered as graph convolutional layers that are used to learn the molecular representations. For the critical property and acentric factor prediction task, we obtain a better overall performance using D-MPNN than GATv2. Our results on additional feature study show that Abraham parameters help the model achieve better performance for critical property and acentric factor prediction. However, the effectiveness of additional parameters is sensitive to model architecture as the latent representation varies depending on the encoding layer architecture.

While investigating target groupings, our results indicate that a multi-task training scheme where a single model is trained to predict all the critical properties and acentric factor along auxiliary phase change properties is beneficial to model performance. The multi-task approach is particularly useful for this task as the critical properties are known to be physically correlated to the auxiliary properties. Additionally, for multi-task training, weighting targets is important to achieve a good performance, especially when we have an imbalanced data set i.e. certain targets have a lot more data points than others. We also find that pre-training the model using estimated data from group contribution methods such as Joback, followed by fine-tuning on experimental data helps reduce model errors further. Using estimated data for pre-training is particularly useful when there is a dearth of experimental data, as is the case with critical properties. The extensive critical property, acentric factor, and phase change property data set containing 5680 chemical compounds used for this work including the external test set, along with a complete set of QM descriptors for each compound is made available in the public domain (see Section 6). We also provide public access to our final model via source code.

A natural extension of this work is to investigate other QM descriptors that have been provided in the data set to further improve model performance. This work could also involve computing QM descriptors at different levels of theory to investigate the effect of QM accuracy on the ML model performance. It is also highly encouraged to perform Gibbs-ensemble Monte Carlo (GEMC) simulations to compute high-accuracy critical property data and increase the size of the data set, which is likely to improve model accuracy. Another potential direction would be to improve the uncertainty estimates that were computed in this work. This could include the use of methods discussed by Scalia et al, and Schwalbe-Koda et al.[106,107] Nevertheless, this work develops a state-of-the-art machine learning model for predicting critical properties and acentric factor that can be built on in future endeavors.

6 Data and Software Availability

All data sets, data splits, additional features, QM calculations, final model predictions, and final ML models are provided through Zenodo: `https://zenodo.org/record/8072892`. The data sets and models are open access and distributed under the terms and conditions of the Creative Commons Attribution (CC BY 4.0) license (`https://creativecommons.org/licenses/by/4.0/`). Citations should refer directly to this manuscript. The final D-MPNN models (Chemprop) that are distributed through Zenodo are pre-trained with all estimated data and fine-tuned with all experimental data including the external test set. The sample code on how to use the final models can be found at `https://github.com/yunsiechung/chemprop/tree/crit_prop`. The code used to train the GATv2 models can be found at `https://github.com/Sayandeep00/chiral_gnn/tree/ml_prop`.

7 Supporting Information

- PDF: Details of data collection, data distribution, scaffolds, default atom and bond features used for the ML models, selected RDKit features, ML hyperparameters, results of the GATv2 models, comparison of the final models with the baseline models, pre-training results on the random split, final model error by a number of carbon atoms, and final D-MPNN results on the other four targets (Tb, Tm, ΔH_{vap}, ΔH_{fus})

- Zip: Data sets, data splits, additional features, and model predictions.

8 Acknowledgements

The authors acknowledge the Machine Learning for Pharmaceutical Discovery and Synthesis Consortium (MLPDS) and Defense Advanced Research Projects Agency - Accelerated Molecular Discovery (DARPA-AMD) for funding. The authors also acknowledge MIT SuperCloud Lincoln Laboratory Supercomputing Center[108] for providing the computational resources used in this work. The authors would also like to thank Dr. Angiras Menon and Kevin Spiekermann for their helpful reviews and discussions of this study. The authors also thank Jonathan Zheng for his help with data compilation. We also appreciate and thank Prof. Jüergen Rarey for his assistance in obtaining the predictions from Nannoolal et al. group contribution method used as a baseline in this work.

References

(1) Walker, P. J.; Yew, H.-W.; Riedemann, A. Clapeyron.jl: An Extensible, Open-Source Fluid Thermodynamics Toolkit. *Ind. Eng. Chem. Res.* **2022**, *61*, 7130–7153.

(2) Walker, P. J. Toward Advanced, Predictive Mixing Rules in SAFT Equations of State. *Ind. Eng. Chem. Res.* **2022**, *61*, 18165–18175.

(3) Tronci, S.; Garau, D.; Stateva, R. P.; Cholakov, G.; Wakeham, W. A.; Errico, M. Analysis of hybrid separation schemes for levulinic acid separation by process intensification and assessment of thermophysical properties impact. *Sep. Purif. Technol.* **2023**, *310*, 123166.

(4) Kunz, O.; Wagner, W. The GERG-2008 Wide-Range Equation of State for Natural Gases and Other Mixtures: An Expansion of GERG-2004. *J. Chem. Eng. Data* **2012**, *57*, 3032–3091.

(5) Peng, D.-y.; Robinson, D. B. A rigorous method for predicting the critical properties of multicomponent systems from an equation of state. *AICHE J.* **1977**, *23*, 137–144.

(6) Soave, G. Equilibrium constants from a modified Redlich-Kwong equation of state. *Chem. Eng. Sci.* **1972**, *27*, 1197–1203.

(7) Fiorentino, E.-A.; Wortham, H.; Sartelet, K. Combining homogeneous and heterogeneous chemistry to model inorganic compound concentrations in indoor environments: the H 2 I model (v1. 0). *Geosci. Model Dev.* **2021**, *14*, 2747–2780.

(8) Ravindran, P.; Davis, E.; Ray, A. Diffusivities of low-volatility species in light gases. *AICHE J.* **1979**, *25*, 966–975.

(9) Chen, N. H.; Othmer, D. F. New generalized equation for gas diffusion coefficient. *J. Chem. Eng. Data* **1962**, *7*, 37–41.

(10) Wang, H.; Frenklach, M. Transport properties of polycyclic aromatic hydrocarbons for flame modeling. *Combust. Flame* **1994**, *96*, 163–170.

(11) Randová, A.; Bartovská, L. Group contribution method: Surface tension of linear and branched alkanes. *Fluid Phase Equilib.* **2016**, *429*, 166–176.

(12) Dobbelaere, M. R.; Ureel, Y.; Vermeire, F. H.; Tomme, L.; Stevens, C. V.; Van Geem, K. M. Machine learning for physicochemical property prediction of complex hydrocarbon mixtures. *Ind. Eng. Chem. Res.* **2022**, *61*, 8581–8594.

(13) Chung, Y.; Gillis, R. J.; Green, W. H. Temperature-dependent vapor–liquid equilibria and solvation free energy estimation from minimal data. *AICHE J.* **2020**, *66*, e16976.

(14) Vermeire, F. H.; Chung, Y.; Green, W. H. Predicting Solubility Limits of Organic Solutes for a Wide Range of Solvents and Temperatures. *J. Am. Chem. Soc.* **2022**, *144*, 10785–10797.

(15) Wang, H.-W.; Hsieh, C.-M. Prediction of solid solute solubility in supercritical carbon dioxide from PSRK EOS with only input of molecular structure. *J. Supercrit. Fluids* **2022**, *180*, 105446.

(16) Adenekan, K.; Hutton-Prager, B. Modeling the solubility of Alkyl Ketene Dimer in supercritical carbon dioxide: Peng-Robinson, group contribution methods, and effect of critical density on solubility predictions. *Fluid Phase Equilib.* **2020**, *507*, 112415.

(17) Tee, L. S.; Gotoh, S.; Stewart, W. E. Molecular parameters for normal fluids. Lennard-Jones 12-6 Potential. *Ind. Eng. Chem. Res.* **1966**, *5*, 356–363.

(18) Wang, H.; Liu, B.; Xie, C.; Li, Y.; Cui, J.; Xing, L.; Wang, Z. Thermal decomposition of isopentanol: A theoretical calculation and kinetic modeling analysis. *Combust. Flame* **2022**, *245*, 112320.

(19) Liu, M.; Grinberg Dana, A.; Johnson, M. S.; Goldman, M. J.; Jocher, A.; Payne, A. M.; Grambow, C. A.; Han, K.; Yee, N. W.; Mazeau, E. J.; Blondal, K.; West, R. H.; Goldsmith, C. F.; Green, W. H. Reaction mechanism generator v3. 0: advances in automatic mechanism generation. *J. Chem. Inf. Model.* **2021**, *61*, 2686–2696.

(20) Joback, K. G.; Reid, R. C. Estimation of Pure-Component Properties from Group-Contributions. *Chem. Eng. Commun.* **1987**, *57*, 233–243.

(21) Han, B.; Peng, D. A group-contribution correlation for predicting the acentric factors of organic compounds. *Can. J. Chem. Eng.* **1993**, *71*, 332–334.

(22) Nannoolal, Y.; Rarey, J.; Ramjugernath, D. Estimation of pure component properties: Part 2. Estimation of critical property data by group contribution. *Fluid Phase Equilibria* **2007**, *252*, 1–27.

(23) Hukkerikar, A. S.; Sarup, B.; Ten Kate, A.; Abildskov, J.; Sin, G.; Gani, R. Group-contribution + (GC +) based estimation of properties of pure components: Improved property estimation and uncertainty analysis. *Fluid Phase Equilib.* **2012**, *321*, 25–43.

(24) Mansour, K.; Korichi, M. *Comput. Aided Chem. Eng.*; Elsevier, 2016; Vol. 38; pp 1237–1242.

(25) Tahami, S.; Movagharnejad, K.; Ghasemitabar, H. Estimation of the critical constants of organic compounds via a new group contribution method. *Fluid Phase Equilib.* **2019**, *494*, 45–60.

(26) Tahami, S.; Ghasemitabar, H.; Movagharnejad, K. Estimation of the acentric factor of organic compounds via a new group contribution method. *Fluid Phase Equilib.* **2019**, *499*, 112246.

(27) Chung, Y.; Vermeire, F. H.; Wu, H.; Walker, P. J.; Abraham, M. H.; Green, W. H. Group Contribution and Machine Learning Approaches to Predict Abraham Solute Parameters, Solvation Free Energy, and Solvation Enthalpy. *J. Chem. Inf. Model.* **2022**, *62*, 433–446.

(28) Fu, L.; Liu, L.; Yang, Z.-J.; Li, P.; Ding, J.-J.; Yun, Y.-H.; Lu, A.-P.; Hou, T.-J.; Cao, D.-S. Systematic Modeling of log D7.4 Based on Ensemble Machine Learning, Group Contribution, and Matched Molecular Pair Analysis. *J. Chem. Inf. Model.* **2020**, *60*, 63–76.

(29) Varamesh, A.; Hemmati-Sarapardeh, A.; Moraveji, M. K.; Mohammadi, A. H. Generalized models for predicting the critical properties of pure chemical compounds. *J. Mol. Liq.* **2017**, *240*, 777–793.

(30) Banchero, M.; Manna, L. Comparison between multi-linear- and radial-basis-function-neural-network-based QSPR Models for the prediction of the critical temperature, critical pressure and acentric factor of organic compounds. *Molecules* **2018**, *23*, 1–13.

(31) Su, Y.; Wang, Z.; Jin, S.; Shen, W.; Ren, J.; Eden, M. R. An architecture of deep learning in QSPR modeling for the prediction of critical properties using molecular signatures. *AICHE J.* **2019**, *65*, 1–11.

(32) Yang, K.; Swanson, K.; Jin, W.; Coley, C.; Eiden, P.; Gao, H.; Guzman-Perez, A.; Hopper, T.; Kelley, B.; Mathea, M.; Palmer, A.; Settels, V.; Jaakkola, T.; Jensen, K.; Barzilay, R. Analyzing Learned Molecular Representations for Property Prediction. *J. Chem. Inf. Model.* **2019**, *59*, 3370–3388.

(33) Yu-Tung Wang, A.; Murdock, R. J.; Kauwe, S. K.; Oliynyk, A. O.; Gurlo, A.; Brgoch, J.; Persson, K. A.; Sparks, T. D. Machine Learning for Materials Scientists: An Introductory Guide toward Best Practices. *Chem. Mater.* **2020**, *2020*, 50.

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

(35) Spiekermann, K. A.; Pattanaik, L.; Green, W. H. Fast predictions of reaction barrier heights: toward coupled-cluster accuracy. *J. Phys. Chem. A* **2022**, *126*, 3976–3986.

(36) Greenman, K. P.; Green, W. H.; Gómez-Bombarelli, R. Multi-fidelity prediction of molecular optical peaks with deep learning. *Chem. Sci.* **2022**, *13*, 1152–1162.

(37) Heid, E.; Green, W. H. Machine learning of reaction properties via learned representations of the condensed graph of reaction. *J. Chem. Inf. Model.* **2021**, *62*, 2101–2110.

(38) Stuyver, T.; Coley, C. W. Quantum chemistry-augmented neural networks for reactivity prediction: Performance, generalizability, and explainability. *J. Chem. Phys.* **2022**, *156*, 084104.

(39) Feinberg, E. N.; Joshi, E.; Pande, V. S.; Cheng, A. C. Improvement in ADMET prediction with multitask deep featurization. *J. Med. Chem.* **2020**, *63*, 8835–8848.

(40) Ramsundar, B.; Liu, B.; Wu, Z.; Verras, A.; Tudor, M.; Sheridan, R. P.; Pande, V. Is Multitask Deep Learning Practical for Pharma? *J. Chem. Inf. Model* **2017**, *57*, 27.

(41) Thung, K. H.; Wee, C. Y. A brief review on multi-task learning. *Multimed. Tools. Appl.* **2018**, *77*, 29705–29725.

(42) Zhang, Y.; Yang, Q. A survey on multi-task learning. *IEEE Trans. Knowl. Data Eng.* **2021**, *34*, 5586–5609.

(43) Lacerda, D. B.; Scardini, R. B.; Vinhal, A. P. C. M.; Pires, A. P.; Priimenko, V. I. *Recent Insights in Petroleum Science and Engineering*; InTech, 2018.

(44) Ghomisheh, Z.; Sobati, M. A.; Gorji, A. E. New empirical correlations for the prediction of critical properties and acentric factor of S-containing compounds. *J. Sulphur Chem.* **2022**, *43*, 327–351.

(45) Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lio, P.; Bengio, Y. *arXiv preprint arXiv:1710.10903,* **2017**.

(46) Marschner, R. F.; Beverly, J. B. The simple relation between critical temperature and boiling point. *J. Chem. Educ.* **1956**, *33*, 604.

(47) Varshni, Y. P. Critical temperatures of organic compounds from their boiling points. *Phys. Chem. Liq.* **2009**, *47*, 383–398.

(48) Landrum, G. RDKit: Open-Source Cheminformatics. 2006; `http://www.rdkit.org/`.

(49) Yaws, C. L. *Thermophysical Properties of Chemicals and Hydrocarbons*; William Andrew, 2009.

(50) Green, D. W.; Perry, R. H. *Perry's Chemical Engineers' Handbook (Section 2 - Physical and Chemical Data)*, 8th ed.; McGraw-Hill's AccessEngineering; McGraw-Hill: New York, 2000.

(51) Kleiber, M.; Joh, R.; Span, R. D3 Properties of Pure Fluid Substances: Datasheet from VDI-Buch ·Volume : "VDI Heat Atlas" in SpringerMaterials (https://doi.org/10.1007/978-3-540-77877-6). `https://materials.springer.com/lb/docs/sm_nlb_978-3-540-77877-6_18`.

(52) Linstrom, P. J.; Mallard, W. G.; Eds., NIST Chemistry WebBook, NIST Standard Reference Database Number 69. `https://doi.org/10.18434/T4D303`.

(53) Rumble, J. R.; ed., *CRC Handbook of Chemistry and Physics, 102nd Edition (Internet Version 2021)*; CRC Press/Taylor & Francis: Boca Raton, FL.

(54) Joback, K. G. A unified approach to physical property estimation using multivariate statistical techniques. Ph.D. thesis, Massachusetts Institute of Technology, 1984.

(55) Swain, M. PubChemPy. 2013; `https://pypi.org/project/PubChemPy/1.0/`.

(56) Swain, M. CIRpy. 2016; `https://pypi.org/project/CIRpy/`.

(57) Kudchadker, A. P.; Ambrose, D.; Tsonopoulos, C. Vapor-liquid critical properties of elements and compounds. 7. Oxygen compounds other than alkanols and cycloalkanols. *J. Chem. Eng. Data* **2001**, *46*, 457–479.

(58) Tsonopoulos, C.; Ambrose, D. Vapor-liquid critical properties of elements and compounds. 8. Organic sulfur, silicon, and tin compounds (C+ H+ S, Si, and Sn). *J. Chem. Eng. Data* **2001**, *46*, 480–485.

(59) Marsh, K. N.; Young, C. L.; Morton, D. W.; Ambrose, D.; Tsonopoulos, C. Vapor-liquid critical properties of elements and compounds. 9. Organic compounds containing nitrogen. *J. Chem. Eng. Data* **2006**, *51*, 305–314.

(60) Marsh, K. N.; Abramson, A.; Ambrose, D.; Morton, D. W.; Nikitin, E.; Tsonopoulos, C.; Young, C. L. Vapor-liquid critical properties of elements and compounds. 10. Organic compounds containing halogens. *J. Chem. Eng. Data* **2007**, *52*, 1509–1538.

(61) Ambrose, D.; Tsonopoulos, C.; Nikitin, E. D. Vapor-Liquid Critical Properties of Elements and Compounds. 11. Organic Compounds Containing B+ O; Halogens+ N,+ O,+ O+ S,+ S,+ Si; N+ O; and O+ S,+ Si. *J. Chem. Eng. Data* **2009**, *54*, 669–689.

(62) Ambrose, D.; Tsonopoulos, C.; Nikitin, E. D.; Morton, D. W.; Marsh, K. N. Vapor-liquid critical properties of elements and compounds. 12. Review of recent data for hydrocarbons and non-hydrocarbons. *J. Chem. Eng. Data* **2015**, *60*, 3444–3482.

(63) Ambrose, D.; Young, C. L. Vapor-liquid critical properties of elements and compounds. 1. An introductory survey. *J. Chem. Eng. Data* **1995**, *40*, 345–357.

(64) Ambrose, D.; Tsonopoulos, C. Vapor-liquid critical properties of elements and compounds. 2. Normal alkanes. *J. Chem. Eng. Data* **1995**, *40*, 531–546.

(65) Tsonopoulos, C.; Ambrose, D. Vapor-liquid critical properties of elements and compounds. 3. Aromatic hydrocarbons. *J. Chem. Eng. Data* **1995**, *40*, 547–558.

(66) Tsonopoulos, C.; Ambrose, D. Vapor-liquid critical properties of elements and compounds. 6. Unsaturated aliphatic hydrocarbons. *J. Chem. Eng. Data* **1996**, *41*, 645–656.

(67) Gude, M.; Teja, A. S. Vapor-liquid critical properties of elements and compounds. 4. Aliphatic alkanols. *J. Chem. Eng. Data* **1995**, *40*, 1025–1036.

(68) Daubert, T. E. Vapor-liquid critical properties of elements and compounds. 5. Branched alkanes and cycloalkanes. *J. Chem. Eng. Data* **1996**, *41*, 365–372.

(69) Dietterich, T. G. Ensemble Methods in Machine Learning. Multiple Classifier Systems. MCS 2000. Lecture Notes in Computer Science, vol 1857. Springer, Berlin, Heidelberg. 2000; pp 1–15, https://link.springer.com/chapter/10.1007/3-540-45014-9_1.pdf.

(70) McGill, C.; Forsuelo, M.; Guan, Y.; Green, W. H. Predicting infrared spectra with message passing neural networks. *J. Chem. Inf. Model.* **2021**, *61*, 2594–2609.

(71) Bergstra, J.; Yamins, D.; Cox, D. Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures. *ICML.* 2013; pp 115–123.

(72) Akiba, T.; Sano, S.; Yanase, T.; Ohta, T.; Koyama, M. Optuna: A Next-generation Hyperparameter Optimization Framework. *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.* 2019.

(73) Vermeire, F. H.; Green, W. H. Transfer learning for solvation free energies: From quantum chemistry to experiments. *Chem. Eng. J* **2021**, *418*, 129307.

(74) Brody, S.; Alon, U.; Yahav, E. *arXiv preprint arXiv:2105.14491,* **2021**.

(75) Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Advances in neural information processing systems* **2017**, *30*.

(76) Fey, M.; Lenssen, J. E. *arXiv preprint arXiv:1903.02428,* **2019**.

(77) Pattanaik, L.; Ganea, O.-E.; Coley, I.; Jensen, K. F.; Green, W. H.; Coley, C. W. *arXiv preprint arXiv:2012.00094,* **2020**.

(78) Shi, C.; B. Borchardt, T. JRgui: A Python Program of Joback and Reid Method. *ACS Omega* **2017**, *2*, 8682–8688.

(79) Flynn, H. PyTorch-Radial-Basis-Function-Layer. `https://github.com/JeremyLinux/PyTorch-Radial-Basis-Function-Layer`, 2021.

(80) Kelley, B. DescriptaStorus. `https://github.com/bp-kelley/descriptastorus#readme`, 2022.

(81) Li, Z.; Zuo, L.; Wu, W.; Chen, L. The New Method for Correlation and Prediction of Thermophysical Properties of Fluids. Critical Temperature. *J. Chem. Eng. Data* **2017**, *62*, 3723–3731.

(82) Abraham, M. H.; Acree, W. E. Correlation and prediction of partition coefficients between the gas phase and water, and the solvents dodecane and undecane. *New J. Chem.* **2004**, *28*, 1538–1543.

(83) Guan, Y.; Coley, C. W.; Wu, H.; Ranasinghe, D.; Heid, E.; Struble, T. J.; Pattanaik, L.; Green, W. H.; Jensen, K. F. Regio-selectivity prediction with a machine-learned reaction representation and on-the-fly quantum mechanical descriptors. *Chem. Sci.* **2021**, *12*, 2198–2208.

(84) Halgren, T. A. Merck molecular force field. I. Basis, form, scope, parameterization, and performance of MMFF94. *J. Comput. Chem.* **1996**, *17*, 490–519.

(85) Bannwarth, C.; Ehlert, S.; Grimme, S. GFN2-xTB—An accurate and broadly parametrized self-consistent tight-binding quantum chemical method with multipole electrostatics and density-dependent dispersion contributions. *J. Chem. Theory Comput.* **2019**, *15*, 1652–1671.

(86) Stephens, P. J.; Devlin, F. J.; Chabalowski, C. F.; Frisch, M. J. Ab Initio Calculation of Vibrational Absorption and Circular Dichroism Spectra Using Density Functional Force Fields. *J. Phys. Chem* **1994**, *98*, 11623–11627.

(87) Frisch, M. J.; Trucks, G. W.; Schlegel, H. B.; Scuseria, G. E.; Robb, M. A.; Cheeseman, J. R.; Scalmani, G.; Barone, V.; Petersson, G. A.; Nakatsuji, H.; Li, X.; Caricato, M.; Marenich, A. V.; Bloino, J.; Janesko, B. G.; Gomperts, R.; Mennucci, B.; Hratchian, H. P.; Ortiz, J. V.; Izmaylov, A. F.; Sonnenberg, J. L.; Williams-Young, D.; Ding, F.; Lipparini, F.; Egidi, F.; Goings, J.; Peng, B.; Petrone, A.; Henderson, T.; Ranasinghe, D.; Zakrzewski, V. G.; Gao, J.; Rega, N.; Zheng, G.; Liang, W.; Hada, M.; Ehara, M.; Toyota, K.; Fukuda, R.; Hasegawa, J.; Ishida, M.; Nakajima, T.; Honda, Y.; Kitao, O.; Nakai, H.; Vreven, T.; Throssell, K.; Montgomery Jr., J. A.; Peralta, J. E.; Ogliaro, F.; Bearpark, M. J.; Heyd, J. J.; Brothers, E. N.; Kudin, K. N.; Staroverov, V. N.; Keith, T. A.; Kobayashi, R.; Normand, J.; Raghavachari, K.; Rendell, A. P.; Burant, J. C.; Iyengar, S. S.; Tomasi, J.; Cossi, M.; Millam, J. M.; Klene, M.; Adamo, C.; Cammi, R.; Ochterski, J. W.; Martin, R. L.; Morokuma, K.; Farkas, O.; Foresman, J. B.; Fox, D. J. Gaussian 16 Revision C.01. 2016.

(88) Glendening, E. D.; Badenhoop, J. K.; Reed, A. E.; Carpenter, J. E.; Bohmann, J. A.; Morales, C. M.; Karafiloglou, P.; Landis, C. R.; Weinhold, F. NBO 7.0. Theoretical Chemistry Institute, University of Wisconsin, Madison, 2018.

(89) Wu, H. QM descriptors calculation. `https://github.com/oscarwumit/QM_descriptors_calculation/tree/ions`, 2022.

(90) Behler, J. Atom-centered symmetry functions for constructing high-dimensional neural network potentials. *J. Chem. Phys.* **2011**, *134*, 074106.

(91) 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.

(92) Pan, S. J.; Yang, Q. A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.* **2010**, *22*, 1345–1359.

(93) Panagiotopoulos, A. Z. Direct determination of phase coexistence properties of fluids by Monte Carlo simulation in a new ensemble. *Mol. Phys.* **1987**, *61*, 813–826.

(94) Wilding, N. B. Critical-point and coexistence-curve properties of the Lennard-Jones fluid: A finite-size scaling study. *Phys. Rev. E* **1995**, *52*, 602.

(95) Martin, M. G.; Siepmann, J. I. Transferable potentials for phase equilibria. 1. United-atom description of n-alkanes. *J. Phys. Chem. B* **1998**, *102*, 2569–2577.

(96) Martin, M. G.; Siepmann, J. I. Novel configurational-bias Monte Carlo method for branched molecules. Transferable potentials for phase equilibria. 2. United-atom description of branched alkanes. *J. Phys. Chem. B* **1999**, *103*, 4508–4517.

(97) Chen, B.; Siepmann, J. I. Transferable potentials for phase equilibria. 3. Explicit-hydrogen description of normal alkanes. *J. Phys. Chem. B* **1999**, *103*, 5370–5379.

(98) Wick, C. D.; Martin, M. G.; Siepmann, J. I. Transferable potentials for phase equilibria. 4. United-atom description of linear and branched alkenes and alkylbenzenes. *J. Phys. Chem. B* **2000**, *104*, 8008–8016.

(99) Chen, B.; Potoff, J. J.; Siepmann, J. I. Monte Carlo calculations for alcohols and their mixtures with alkanes. Transferable potentials for phase equilibria. 5. United-atom description of primary, secondary, and tertiary alcohols. *J. Phys. Chem. B* **2001**, *105*, 3093–3104.

(100) Wick, C. D.; Stubbs, J. M.; Rai, N.; Siepmann, J. I. Transferable potentials for phase equilibria. 7. Primary, secondary, and tertiary amines, nitroalkanes and nitrobenzene, nitriles, amides, pyridine, and pyrimidine. *J. Phys. Chem. B* **2005**, *109*, 18974–18982.

(101) Lubna, N.; Kamath, G.; Potoff, J. J.; Rai, N.; Siepmann, J. I. Transferable potentials for phase equilibria. 8. United-atom description for thiols, sulfides, disulfides, and thiophene. *J. Phys. Chem. B* **2005**, *109*, 24100–24107.

(102) Rai, N.; Siepmann, J. I. Transferable potentials for phase equilibria. 9. Explicit hydrogen description of benzene and five-membered and six-membered heterocyclic aromatic compounds. *J. Phys. Chem. B* **2007**, *111*, 10790–10799.

(103) Rai, N.; Siepmann, J. I. Transferable potentials for phase equilibria. 10. Explicit-hydrogen description of substituted benzenes and polycyclic aromatic compounds. *J. Phys. Chem. B* **2013**, *117*, 273–288.

(104) Ambrose, D.; Ghiassee, N. Vapour pressures and critical temperatures and critical pressures of some alkanoic acids: C1 to C10. *J. Chem. Thermodyn.* **1987**, *19*, 505–519.

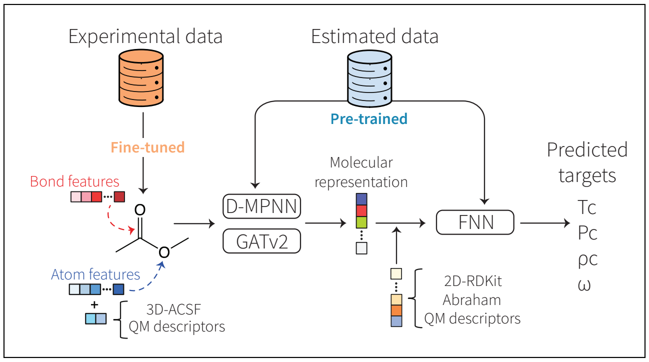
(105) Frenkel, M.; Chirico, R. D.; Diky, V.; Yan, X.; Dong, Q.; Muzny, C. ThermoData Engine (TDE): software implementation of the dynamic data evaluation concept. *J. Chem. Inf. Model.* **2005**, *45*, 816–838.

(106) Scalia, G.; Grambow, C. A.; Pernici, B.; Li, Y.-P.; Green, W. H. Evaluating scalable uncertainty estimation methods for deep learning-based molecular property prediction. *J. Chem. Inf. Model.* **2020**, *60*, 2697–2717.

(107) Schwalbe-Koda, D.; Tan, A. R.; Gómez-Bombarelli, R. Differentiable sampling of molecular geometries with uncertainty-based adversarial attacks. *Nat. Commun.* **2021**, *12*, 1–12.

(108) Reuther, A.; Kepner, J.; Byun, C.; Samsi, S.; Arcand, W.; Bestor, D.; Bergeron, B.; Gadepally, V.; Houle, M.; Hubbell, M.; Jones, M.; Klein, A.; Milechin, L.; Mullen, J.; Prout, A.; Rosa, A.; Yee, C.; Michaleas, P. Interactive supercomputing on 40,000 cores for machine learning and data analysis. 2018 IEEE High Performance extreme Computing Conference (HPEC). 2018; pp 1–6.

TOC Graphic


Experimental data
Estimated data
Pre-trained
Fine-tuned
Bond features
Molecular representation
Predicted targets
D-MPNN
GATv2
FNN
Tc
Pc
ρc
ω
Atom features
+
3D-ACSF
QM descriptors
2D-RDKit
Abraham
QM descriptors
O
O