Predicting Single-Substance Phase Diagrams: A Kernel Approach on Graph Representations of Molecules

Yan Xiang[†a], Yu-Hang Tang[†b], Hongyi Liu[a], Guang Lin[*c], Huai Sun[*a]

[a]*School of Chemistry and Chemical Engineering, Shanghai Jiao Tong University, Shanghai 200240, China*

[b]*Lawrence Berkeley National Laboratory, Berkeley, California 94720, United States*

[c]*Department of Mathematics & School of Mechanical Engineering, Purdue University, West Lafayette, Indiana 47907, United States*

AUTHOR INFORMATION

* Corresponding Authors

† Authors contributed equally

Tel: +86 136 1180 2895

E-mail address: huaisun@sjtu.edu.cn, guanglin@purdue.edu

Abstract

This work presents a Gaussian process regression (GPR) model on top of a novel graph representation of chemical molecules that predicts thermodynamic properties of pure substances in single, double, and triple phases. A transferable molecular graph representation is proposed as the input for a marginalized graph kernel, which is the major component of the covariance function in our GPR models. Radial basis function kernels of temperature and pressure are also incorporated into the covariance function when necessary. We predicted three types of representative properties of pure substances in single, double, and triple phases, i.e., critical temperature, vapor-liquid equilibrium (VLE) density, and pressure-temperature density. The accuracy of the models is nearly identical to the precision of the experimental measurements. Moreover, the reliability of our predictions can be quantified on a per-sample basis using the posterior uncertainty of the GPR model. We compare our model against Morgan fingerprints and a graph neural network to further demonstrate the advantage of the proposed method.

1. Introduction

Thermodynamic properties of chemical substances are often prerequisites for chemical engineering designs.[1] They are traditionally measured by experimental means, which in many cases are costly, environmentally unfriendly, and sometimes hazardous. Consequently, computational methods have been developed to provide alternatives. For decades, the development of reliable quantitative property-structure relationship (QSPR) models, also known as quantitative activity-structure relationship (QSAR) models, has been an active research area.[2] In this class of approaches, molecular structures are commonly represented by a set of descriptors, while the correlations between the descriptors and the properties of interest are established by solving corresponding regression or classification problems. In recent years, QSPR has been modernized by machine learning (ML) in dealing with big data, with broad applications in organic chemistry,[3] drug discovery[4,5], and material design[1,6]. In both biological and material sciences, ML methods have been developed to predict thermodynamic or physical properties.[7-14] For example, several pieces of work were focused on the prediction of melting point.[7-10] Coley et al. predicted octanol solubility, aqueous solubility, and toxicity in addition to the melting point.[7] Gong et al. predicted multiple thermodynamic properties of alkanes using ML models trained on molecular dynamics simulation data.[11] Afzal et al. predicted liquid densities for a virtual library of organic molecules.[12] Zhu and Müller combined ML with the SAFT equation of state theory to predict multiple thermodynamic properties.[13] Various related works of ionic liquids have been well summarized in a recent review.[14]

Thermodynamic properties are sensitive to molecular structures. For

instance, a small difference in the connectivity or composition of two molecules may lead to significant differences in their thermodynamic properties. Most published ML predictions of thermodynamic properties are focused on a specific type of molecules for which specialized descriptors or fingerprints are adequate. However, for datasets of highly diversified molecules, it has been argued that using fixed-length vectors of descriptors is insufficient.[15]

Our goal is to predict the thermodynamic properties of highly diversified molecules using only their chemical formulas. The problem is unfeasible at first sight, as the thermodynamic properties are phenomena of the condensed-phase and are the result of the collective behavior of a huge number of molecules. From a statistical mechanics point of view, a molecular system's thermodynamic property is a result of ensemble average, which is the mean of the property that is a function of the microscopic state of the system. In the canonical ensemble, the value of a property *A* is given by:

$$\langle A \rangle = \frac{\int A(\mathbf{R}) e^{-\frac{U(\mathbf{R})}{k_B T}} d\mathbf{R}}{\int e^{-\frac{U(\mathbf{R})}{k_B T}} d\mathbf{R}}, \quad (1)$$

which indicates that the distribution of the configuration $e^{-\frac{U(\mathbf{R})}{k_B T}}$ and the property A is completely determined by the total potential energy function $U(\mathbf{R})$. Therefore, a *sufficient* input for predicting thermodynamic properties using ML must also implicitly encode the total potential energy function. Furthermore, taking the force field approach[16-29] in molecular simulations as a reference, we see that the Hamiltonian of a system can be approximated reasonably well by functions defined on the *atom types* and *molecular topologies*. Therefore, we believe that the thermodynamic properties can be predicted solely from the information in the chemical formula as long as the encoding accurately

represents the atom types and molecular topologies. Come to this point, it is apparent that a graph representation of molecules, where the edges of the graph represent bonds and the vertices of the graph represent atoms, can achieve this purpose.

The kernel trick, which implicitly projects data to a reproducing kernel Hilbert space (RKHS) for inner product operations, is a widely used approach to bridge a graph-based representation of the molecules to a large array of machine learning methods[30-33]. For molecular graphs, kernels based on various graph concepts and operations such as random walk,[34] shortest-path,[35,36] optimal assignment,[37] subgraph matching,[38] graph invariant,[39] hashing[40], and Wasserstein Weisfeiler-Lehman[41] have been proposed in the past.

The marginalized graph kernel (MGK), which evaluates the inner product between two graphs in a space of random walk paths of *labeled* nodes and edges, is a particularly interesting algorithm for comparing molecular graphs. The random walk interpretation of the kernel draws similarities to the diffusion of an electron along the chemical bonds of a molecule. More importantly, the use of sophisticated feature sets as labels to decorate graph nodes and edges can greatly enhance the kernel's ability to characterize the diverse structures of the molecules.

However, the application of the marginalized graph kernel in practice had traditionally been severely limited by computational cost and programming difficulty. The use of a Kronecker product formulation inside of the algorithm has given rise to an $O(n^6)$ complexity, where n is the number of nodes in the graphs, for solvers that are implemented in a naive fashion. Previous works were also constrained to use graphs with simple node and edge features due

to the difficulty in implementing sophisticated feature-level microkernels.[34,42] These problems have been further exacerbated in this era of heterogeneous computing as the general-purpose GPUs are still far from friendly in terms of programmability for domain scientists.

Recently, Tang et al. developed the GraphDot software package for the marginalized graph kernel to overcome the aforementioned challenges. The software builds on top of an advanced just-in-time code generation framework to create optimized GPU code for node and edge features and feature-level microkernels that domain scientists as users can easily create using a high-level Python API. A matrix-free sparse linear algebra backend then carries out the actual computation using only $O(n^2)$ time. This development enables the design and optimization of complex molecular graph representations and the associated marginalized graph kernel for production-scale datasets and models.

On a related front, since the pioneering work of Scarselli et al.,[43] graph neural networks (GNNs) have received considerable attention.[44] Graph convolutional network,[45-47] graph attention network[48,49], gated graph neural network[50,51], and message passing neural network[52,53] are examples of advanced GNN. In a recent review, Wieder et al. summarized 80 types of GNNs, which have been used to predict more than 20 molecular properties using 48 different data sets.[15] Despite many variants of GNN, the central idea is the same: the molecular representations are learned from the graph-structured data during the training process.

In this work, we present an ML framework, abbreviated as GPR-nMGK-tMGR, that can learn and predict the thermodynamics properties of pure substances. In Section 2, we explain the framework in detail, including

transferrable molecular graph representation (tMGR), normalized MGK (nMGK), and hybrid kernel. In Section 3, we employ the framework to predict three typical data on a single-component phase diagram: the critical temperature (T_c) of three-phase point, vapor-liquid-equilibrium densities of liquid (VLE-ρ^l) and vapor (VLE-ρ^v) phases, and pressure-temperature-density (PT-ρ) of liquid of one-phase region. We also applied GNN and Morgan fingerprints on the T_c data set to illustrate the advantage of the framework.

2. Methods

Figure 1 illustrates the workflow developed in this work. The molecules are divided into a training set (black) and a test set (grey), the pairwise similarity matrices between and within each data set are computed by nMGK or hybrid kernel. The self-similarity matrix $K(\mathbf{X}, \mathbf{X})$ and the target values of the training set are used to construct a GPR model. The cross-similarity matrix $K(\mathbf{X}^*, \mathbf{X})$ of the training set and test set is used for properties prediction and uncertainty estimation.

In the workflow, the key component is the construction of an nMGK as shown in the top panel of the figure. The input molecules, as identified by InChI or SMILES strings, are converted into undirected graphs. The vertices and edges of the graph are labeled with features that describe the local environments of the atoms and bonds. The similarity between graphs is computed by the nMGK. Finally, the nMGK is coupled optionally with radial basis function (RBF) kernels on temperature and pressure by tensor product formulation to make the hybrid kernels.

2.1. Graph Kernel

2.1.1. Transferable Molecular Graphs Representation (tMGR)

Table 1 contains a list of the features of the atoms and bonds in our transferrable molecular graph representation, which essentially encapsulates the information for defining atom types and molecular topology. The *atomic number* is the most essential property of an atom. Two features are used to describe the cyclic environment: *Ring count* indicates the number of different rings in which the atom exists, *ring list* is a vector concatenating the ring sizes. The steric feature is *chirality*. Additional information about the local environment of the atom is given. The *Morgan substructure* with a radius of 3 (see Figure S2) is used for this purpose. Moreover, we found that the inclusion of propagated features, i.e. the details of the atoms on immediate and, are beneficial. A vector that concatenates atomic numbers up to the fourth layer denoted as *atomic number (list, 1-4)*, the number of the hydrogen atoms of the first layer *hydrogen (count, 1)*, and the number of heavy atoms up to the second layer *heavy atom (count, 1-2)*, are used. *Bond order* is the most basic information for a chemical bond. Besides, the steric features *E-Z (double bond)* and *E-Z (ring bond)* are used. The optimal combination of the features as a feature set is obtained by trial and error, while insignificant features, such as aromaticity, formal charge, and hybridization, are abandoned in the process. All of the features required can be extracted from the InChI[54] or SMILES[55,56] string using the cheminformatics toolkits RDKit.[57] As an example, α-hexachlorocyclohexane is given in Figure S4. The table also lists the hyperparameters δ and C, which will be explained below.

2.1.2. Normalized Marginalized Graph Kernel (nMGK).

The MGK defines a positive definite similarity function between a pair of graphs *G* and *G'*. It computes the overall similarity as the expectation of path similarities samples from a simultaneous random walk process:

$$K(G,G') = \sum_{\ell=1}^{\infty}\sum_{\mathbf{h}}\sum_{\mathbf{h}'}\begin{bmatrix} p_{\mathrm{s}}(h_1)p'_{\mathrm{s}'}(h'_1)\kappa_{\mathrm{v}}\left(v_{h_1}, v'_{h'_1}\right)p_{\mathrm{q}}(h_\ell)p'_{\mathrm{q}}(h'_\ell)\times \\ \left(\prod_{i=2}^{\ell}p_{\mathrm{t}}(h_i|h_{i-1})\right)\left(\prod_{j=2}^{\ell}p'_{\mathrm{t}}(h'_i|h'_{i-1})\right)\times \\ \left(\prod_{k=2}^{\ell}\kappa_{\mathrm{v}}\left(v_{h_k}, v'_{h'_k}\right)\kappa_{\mathrm{e}}\left(e_{h_k h_{k-1}}, e'_{h'_k h'_{k-1}}\right)\right) \end{bmatrix}, \quad (2)$$

where the summation iterates over random walk paths **h** on *G* and **h**' on *G'* of length l; microkernels $\kappa_v(\cdot,\ \cdot)$ and $\kappa_e(\cdot,\ \cdot)$ define the vertex-wise similarities and edge-wise similarities respectively; and p_s, p_q, p_t are the starting probability, stopping probability, and transition probability, of the random walk process, respectively. A uniform starting probability $p_s = 1.0$ and a uniform stop probability $p_q = 0.01$ are used in this work. The transition probability p_t is $1/n$ where *n* is the number of vertices adjacent to the current vertex. Eq 2 is reformulated into a generalized Kronecker product linear system and solved efficiently using general-purpose GPUs without explicit random-walk sampling.[58] More details can be found in the work of Tang and de Jong[59] and Kashima et al.[34]

The microkernels are multiplicative compositions of elementary similarity kernels between individual features.

$$\kappa_{\mathrm{v}}(v,v') = \prod_j \mu_j\left(\phi_j(v), \phi_j(v')\right), \quad (3)$$

$$\kappa_{\mathrm{e}}(e,e') = \prod_j \mu_j\left(\phi_j(e), \phi_j(e')\right), \quad (4)$$

where μ_j is the elementary kernel for the *j*-th feature ϕ_j.

Two types of elementary kernels, the Kronecker delta kernel $\delta(\cdot,\cdot)$ for features with a fixed length, and the sequence convolution kernel $C(\cdot,\cdot)$ for

features with variable length, are used. The Kronecker delta kernel is defined as:

$$\delta(\phi_1, \phi_2) = \begin{cases} 1 & , \phi_1 = \phi_2 \\ h \in (0,1), \text{ otherwise} \end{cases} \quad (5)$$

The sequence convolution kernel is computed as:

$$C(l_1, l_2) = \frac{f(l_1, l_2)}{\sqrt{f(l_1, l_1) f(l_2, l_2)}}, \quad (6)$$

where

$$f(l_1, l_2) = \sum_{\phi_1 \in l_1} \sum_{\phi_2 \in l_2} \delta(\phi_1, \phi_2) . \quad (7)$$

Here, l_1, l_2 are lengths of the two features. The Kronecker delta kernel has a learnable hyperparameter *h*, whose optimal values are given in Table 1. As shown in the table, a generic value of 0.9 is used for almost all features except that for the atomic number on the current vertex.

The MGK inherently contains the molecular size information as $K(G, G')$ is proportional to the product of the number of vertices in G and G'[45]. This makes it suitable for predicting properties that scale with molecular size such as atomization energy. However, thermodynamic properties generally do not scale up with molecular size. Therefore, we propose a weighted normalization:

$$\bar{K}(G, G') = F \frac{K(G, G')}{\sqrt{K(G, G) K(G', G')}} \exp\left[- \frac{\left(K(G, G) - K(G', G') \right)^2}{\lambda^2} \right], \quad (8)$$

where *F* is a hyperparameter that does not affect the predicted value but determines the magnitude of the predictive uncertainty. The hyperparameter λ is set to be 10^4.

2.1.3. Hybrid Graph/Euclidean Kernels

Thermophysical properties may depend on state variables. The number of

variables is governed by the Gibbs phase law, $N_F = N_C - N_P + 2$. Here N_F is the number of thermodynamic variables, N_C is the number of components and N_P is the number of phases. For single-component system $N_C = 1$, $N_P = 0, 1, \text{or } 2$ depending on the number of phases. For the properties considered in this work, $N_F = 0$ for T_c, $N_F = 1$ that is the temperature for VLE-ρ, and $N_F = 2$ that are temperature and pressure for PT-ρ.

We used a hybrid kernel that combines molecular structure information with temperature and pressure using a tensor product formulation:

$$K\big((G,T,P),(G',T',P')\big) = K_G(G,G')K_T(T,T')K_P(P,P'). \quad (9)$$

Since the thermodynamic conditions can be conveniently encoded as fix-length feature vectors, any positive definiteness kernels on the Euclidean space can be potentially employed here. Specifically, we used the RBF kernels

$$K_{\mathrm{T}}(T,T') = \exp\left(-\frac{(T-T')^2}{{\lambda_T}^2}\right), \quad (10)$$

$$K_{\mathrm{P}}(P,P') = \exp\left(-\frac{(P-P')^2}{{\lambda_P}^2}\right), \quad (11)$$

The hyperparameters of the RBF kernels are listed in Table 2.

The reduced temperature $T_{\mathrm{red}} = T/T_c$ was used in the hybrid kernel for predicting VLE densities, because all the molecules behave similarly as the temperature close to T_c. For PT-ρ data, the absolute temperature in Kelvin and pressure in bar are used. The hyperparameters λ_T and λ_P have the same units.

2.2. Gaussian Process Regression.

GPR[31] is a non-parametric machine learning method with built-in uncertainty estimation capabilities. It is a Bayesian inference method that exploits the similarity between data points to make predictions. The probabilistic

nature of a GPR model allows it to make not only a point estimate about the *value* of the prediction target but also the associated *uncertainty* in the form of a posterior variance. It is instructive if we can relate the accuracy of predictions for unknown molecules to the posterior variance.

Given a training set T and associated property y_T, and the nMGK-tMGR described above, the GPR prediction y_* and posterior covariance matrix Σ_* of a test set * are given:

$$\mathbf{y}_* = \mathbf{K}_{T*}^{\mathrm{T}} \mathbf{K}_{TT}^{-1} \mathbf{y}_T, \tag{12}$$

$$\mathbf{\Sigma}_* = \mathbf{K}_{**} - \mathbf{K}_{T*}^{\mathrm{T}} \mathbf{K}_{TT}^{-1} \mathbf{K}_{T*}. \tag{13}$$

We calculated the mean absolute error (MAE) and R-squared values (R^2) of the test set to evaluate performance.

2.3. Data Sets

The reference data employed in this work are summarized in Table 3.T_c and VLE densities and some liquid PT-ρ data are taken from NIST Standard Reference Database (SRD) 103b[60] via Knovel.[61] The data covers highly diversified molecules consisting of elements H, B, C, N, O, F, Si, P, S, Cl, Br, and I. The data were filtered by removing molecules composed of a single heavy atom such as water, ammonia, and methane, data with relative uncertainty greater than 50%, and the low gas density data at temperature below $T_c/2$. The numbers of molecules are 15,422 for T_c, 14,417 for VLE-ρ^v, and 14,737 for VLE-ρ^l density. We found 400 molecules with various number of PT-ρ data points from SRD and augmented the dataset by adding simulation results of 7,580 molecules consisting of H, C, N, O elements.[11] The total numbers of data points are 15,422 for T_c, 1,207,668 for VLE-ρ^l, 1,181,358 for

VLE-ρ^v, and 578,617 for PT-ρ.

The data were randomly split in the ratio of 4:1 by molecule to the training sets and the test sets. To ensure the unbiased division of training and test sets, the process was repeated 10 times, and the results were averaged. Because the density data of the same molecule at different temperatures are highly correlated, the training sets for VLE-ρ^l, VLE-ρ^v and PT-ρ were further reduced by randomly selecting 40,000 data points for saving memory.

3. Results and Discussions

3.1. Predictions

Comparisons of predictions using GPR-nMGK-tMGR against the reference data are given in Figure 2. The properties compared are T_c (A), VLE-ρ^l (B), VLE-ρ^v (C), and PT-ρ(D). Despite the discrepancies of a small number of data points, overall, the prediction of the GPR-nMGK-tMGR model is satisfactory as justified by the R^2 values and the MAE values.

The reference data are associated with uncertainties data in the form of standard deviations.[60] Therefore, comparing the errors of the prediction with the uncertainties of reference data provides an assessment of the model performance. The comparisons between the errors obtained by GPR-nMGK-tMGR and the uncertainties in the reference data are shown in Figure 3. The prediction errors are comparable with the uncertainties provided by SRD for the three experimental data sets, which indicates that the model reaches a similar precision as the reference data.

3.2. Uncertainty

The main advantage of GPR is that it predicts not only a value but a probability distribution. Therefore, the confidence of predictions can be estimated by the posterior uncertainty.

One way to use the posterior uncertainty is to draw the MAE of the predictions as a function of the posterior uncertainties. As shown in Figure 4, the comparisons are presented for the four properties predicted using GPR-nMGK-tMGR. The black dashed line in each of the figures is the “ideal” MAE assuming that the errors obey Gaussian distribution. It can be seen from the figure that the MAE of GPR-nMGK-tMGR prediction is closed to the ideal MAE when the posterior uncertainty is about 2/3 of its maximum value, indicating that low posterior uncertainty is highly correlated to accurate prediction.

To show the distribution of predicted errors against posterior uncertainty, the data is divided into ten posterior uncertainty intervals and plotted as the violin strings in Figure 5. The maximum positive, maximum negative, median errors, and the distribution of error of the data that fall in one posterior uncertainty interval are collectively shown as a violin string along the y-axis. The percentage of data that fall in the interval, the MAE, and the R^2 of these data are listed below the string. The strings show that the prediction errors mostly obey the normal distribution, with an expectation of zero and covariance positively correlated to posterior uncertainty, except those strings associated with large posterior uncertainties. The most striking feature of this presentation is that it illustrates that there are very few predictions that exhibit large errors in the high confident range. Those are ‘outliers’ which will be discussed in the following content.

The quality of the predictive uncertainty can also be examined by

checking whether the confidence intervals cover the same percentage of the experimental values of the samples in the test set. In Figure 6, we plot the confidence interval versus the percentage of the experimental values of the samples in the test set covered by the confidence interval. As the hyperparameter *F* in eq.6 increases, the coverage curve moves towards the upper left, which means more samples are covered by the confidence interval. The optimal *F* is the one that results in the coverage curve closest to the diagonal.

3.3. Outliers

Although the predictions are generally satisfactory, a small number of outliers exist. We see the major cause of the outliers is the so-called “tottering” effect.[35,37,42,62,63] Using T_c as an example, we illustrate the problem by listing two sets of outlier molecules in Figure 7. In each set, the molecules have similar graphs but significantly different T_c values. In outlier set 1 one may speculate that the large variations in T_c is related to the presence of the diazenyl functional group (-N=N-), but outlier 2 does not show a common functional group responsible for the variations. A common feature of both sets is that all molecules have long alkyl chains.

From the thermodynamic point of view, alkyl groups exhibit weaker intermolecular interactions than the polarizable groups. Despite the relatively small part in the molecule, the polarizable functional groups have a strong impact on thermodynamic properties. However, the comparison of similarity using random walk on graphs does not discriminate the differences, therefore, the long alkyl chains dominate the similarity comparison, which leads to the

prediction outliers.

3.4. Comparison with Graph Neural Network

It is of interest to compare kernel approaches with GNN approaches as both are based on the graph representation of molecules. We applied GNN developed by Tsubaki et al.[64] (Supporting Information S1) on T_c data for comparison. The results are presented in Figure 8. The R_2 of 0.93 and MAE of 21.2 K are both slightly larger than that of our GPR-nMGK-tMGR prediction as shown in Figure 2A. It is however reasonable to assume that the performance of GNN could be improved with further optimizations. What can be concluded here is that similar quality of prediction can be obtained by using GPR-nMGK-tMGR and GNN.

3.5. Comparison with fix-length-vector fingerprint

Using Morgan fingerprint (Supporting Information S2), we constructed a GPR model to predict T_c using the same data set as discussed above. The comparison with reference data, as well as posterior analysis is dispatched in Figure 9. The R_2 of 0.77 and MAE of 46.5 shown in Figure 9A are much higher than that of GPR-nMGK-tMGR as shown in Figures 2A. Figure 9B, 9C and 9D illustrate the prediction accuracy are not clearly correlated with the posterior uncertainty.

4. Conclusions

A universal and transferable kernel machine for predicting thermodynamic properties of molecules was developed by combining a

transferable molecular graph representation with the marginalized graph kernel and Gaussian process regression. Using critical temperature, VLE density, and pressure-temperature-density as example targets, we have demonstrated the effectiveness of the proposed GPR-nMGK-tMGR model, which only relies on features that are created from atomic numbers, connectivity, and steric features. The presented molecular graph and associated hyperparameters are applicable for all the target properties and data sets considered in this study, suggesting that it is universal and potentially transferable for the prediction of even more types of properties. The computation is GPU-accelerated. Meanwhile, it allows incorrect predictions to be filtered out by posterior uncertainty to create an accurate ML database that will be several orders of magnitude larger than the existing experimental database.

Comparison against fix-length-vector fingerprint clearly shows the advantage of the graph kernel representation. In comparison with GNN, the prediction quality is comparable. However, the capability of posterior uncertain analysis of GPR is an advantage. In any graph-based model, the “tottering” effect is a bottleneck for further improvement of the prediction quality.

The application presented here on data of single-component phase diagram demonstrated the applicability of the proposed method. We believe the method is general and the graph representation is transferrable, our ongoing and future work will cover more thermodynamic properties important to the chemical industry.

Associated Content

The marginalized graph kernel is computed using GraphDot package at

https://github.com/yhtang/GraphDot. All codes used in this work can be found at https://github.com/Xiangyan93/Chem-Graph-Kernel-Machine.

Acknowledgments

This work was funded by the National Natural Science Foundation of China [Grant No. 21473112], [Grant No. 21403138], [Grant No. 21673138]. Guang Lin gratefully acknowledges the support from the National Science Foundation DMS-1555072 and U.S. Department of Energy (DOE) Office of Science Advanced Scientific Computing Research program DE-SC0021142.

References

(1) Agrawal, A.; Choudhary, A. Perspective: Materials informatics and big data: Realization of the "fourth paradigm" of science in materials science. *APL Mater.* **2016,** *4* (5), 053208.
(2) Tropsha, A. Best practices for QSAR model development, validation, and exploitation. *Mol. Inf.* **2010,** *29* (6-7), 476-488.
(3) Segler, M. H. S.; Preuss, M.; Waller, M. P. Planning chemical syntheses with deep neural networks and symbolic AI. *Nature.* **2018,** *555* (7698), 604-610.
(4) Chen, H.; Engkvist, O.; Wang, Y.; Olivecrona, M.; Blaschke, T. The rise of deep learning in drug discovery. *Drug Discov. Today.* **2018,** *23* (6), 1241-1250.
(5) Vamathevan, J.; Clark, D.; Czodrowski, P.; Dunham, I.; Ferran, E.; Lee, G.; Li, B.; Madabhushi, A.; Shah, P.; Spitzer, M.; Zhao, S. Applications of machine learning in drug discovery and development. *Nat. Rev. Drug Discov.* **2019,** *18* (6), 463-477.
(6) Butler, K. T.; Davies, D. W.; Cartwright, H.; Isayev, O.; Walsh, A. Machine learning for molecular and materials science. *Nature.* **2018,** *559* (7715), 547-555.
(7) Coley, C. W.; Barzilay, R.; Green, W. H.; Jaakkola, T. S.; Jensen, K. F. Convolutional Embedding of Attributed Molecular Graphs for Physical Property Prediction. *J. Chem. Inf. Model.* **2017,** *57* (8), 1757-1772.
(8) Tetko, I. V.; Sushko, Y.; Novotarskyi, S.; Patiny, L.; Kondratov, I.; Petrenko, A. E.; Charochkina, L.; Asiri, A. M. How accurately can we predict the melting points of drug-like compounds? *J. Chem. Inf. Model.* **2014,** *54* (12), 3320-9.
(9) Tetko, I. V.; D, M. L.; Williams, A. J. The development of models to predict melting and pyrolysis point data associated with several hundred thousand compounds mined from PATENTS. *J. Cheminf.* **2016,** *8*, 2.
(10) Sivaraman, G.; Jackson, N. E.; Sanchez-Lengeling, B.; Vázquez-Mayagoitia, Á.; Aspuru-Guzik, A.; Vishwanath, V.; de Pablo, J. J. A diversified machine learning strategy for predicting and understanding molecular melting points. **2019**.
(11) Gong, Z.; Wu, Y.; Wu, L.; Sun, H. Predicting Thermodynamic Properties of Alkanes by High-Throughput Force Field Simulation and Machine Learning. *J. Chem. Inf. Model.* **2018,** *58* (12), 2502-2516.
(12) Afzal, M. A. F.; Sonpal, A.; Haghighatlari, M.; Schultz, A. J.; Hachmann, J. A deep neural network model for packing density predictions and its application in the study of 1.5 million organic molecules. *Chem. Sci.* **2019,** *10* (36), 8374-8383.
(13) Zhu, K.; Muller, E. A. Generating a Machine-Learned Equation of State for Fluid Properties. *J. Phys. Chem. B.* **2020,** *124* (39), 8628-8639.
(14) Yusuf, F.; Olayiwola, T.; Afagwu, C. Application of Artificial Intelligence-based predictive methods in Ionic liquid studies: A review. *Fluid Phase Equilib.* **2021,** *531*.
(15) Wieder, O.; Kohlbacher, S.; Kuenemann, M.; Garon, A.; Ducrot, P.; Seidel, T.; Langer, T. A compact review of molecular property prediction with graph neural networks. *Drug Discovery Today: Technol.* **2020**.
(16) Brooks, B. R.; Bruccoleri, R. E.; Olafson, B. D.; States, D. J.; Swaminathan, S. a.; Karplus, M. CHARMM: a program for macromolecular energy, minimization, and dynamics calculations. *J. Comput. Chem.* **1983,** *4* (2), 187-217.
(17) Jorgenson, W.; Tirado-Rives, J. The OPLS potential functions for proteins. Energy minimizations for crystals of cyclic peptides and crambin. *J. Am. Chem. Soc.* **1988,** *110* (6), 1657-1666.
(18) Allinger, N. L.; Yuh, Y. H.; Lii, J. H. Molecular mechanics. The MM3 force field for hydrocarbons. 1. *J. Am. Chem. Soc.* **1989,** *111* (23), 8551-8566.
(19) Rappé, A. K.; Casewit, C. J.; Colwell, K.; Goddard III, W. A.; Skiff, W. M. UFF, a full periodic table force field for molecular mechanics and molecular dynamics simulations. *J. Am. Chem. Soc.* **1992,** *114* (25), 10024-10035.

(20) Maple, J. R.; Hwang, M. J.; Stockfisch, T. P.; Dinur, U.; Waldman, M.; Ewig, C. S.; Hagler, A. T. Derivation of class II force fields. I. Methodology and quantum force field for the alkyl functional group and alkane molecules. *J. Comput. Chem.* **1994,** *15* (2), 162-182.
(21) Asensio, J. L.; Jimenez-Barbero, J. The use of the AMBER force field in conformational analysis of carbohydrate molecules: Determination of the solution conformation of methyl α-lactoside by NMR spectroscopy, assisted by molecular mechanics and dynamics calculations. *Biopolymers: Original Research on Biomolecules.* **1995,** *35* (1), 55-73.
(22) Halgren, T. A. Merck molecular force field. II. MMFF94 van der Waals and electrostatic parameters for intermolecular interactions. *J. Comput. Chem.* **1996,** *17* (5-6), 520-552.
(23) Brooks, B. R.; Bruccoleri, R. E.; Olafson, B. D.; States, D. J.; Swaminathan, S.; Karplus, M. CHARMM: A program for macromolecular energy, minimization, and dynamics calculations. *Journal of Computational Chemistry.* **1983,** *4* (2), 187-217.
(24) Jorgensen, W. L.; Tiradorives, J. THE OPLS POTENTIAL FUNCTIONS FOR PROTEINS - ENERGY MINIMIZATIONS FOR CRYSTALS OF CYCLIC-PEPTIDES AND CRAMBIN. *Journal of the American Chemical Society.* **1988,** *110* (6), 1657-1666.
(25) Allinger, N. L.; Yuh, Y. H.; Lii, J. H. Molecular mechanics. The MM3 force field for hydrocarbons. 1. *Journal of the American Chemical Society.* **1989,** *111* (23), 8551-8566.
(26) Rappe, A. K.; Casewit, C. J.; Colwell, K. S.; Goddard, W. A.; Skiff, W. M. UFF, a full periodic table force field for molecular mechanics and molecular dynamics simulations. *Journal of the American Chemical Society.* **1992,** *114* (25), 10024-10035.
(27) Maple, J. R.; Hwang, M. J.; Stockfisch, T. P.; Dinur, U.; Waldman, M.; Ewig, C. S.; Hagler, A. T. Derivation of class II force fields. I. Methodology and quantum force field for the alkyl functional group and alkane molecules. *Journal of Computational Chemistry.* **1994,** *15* (2), 162-182.
(28) Halgren, T. A. Merck molecular force field. II. MMFF94 van der Waals and electrostatic parameters for intermolecular interactions. *Journal of Computational Chemistry.* **1996,** *17* (5-6), 520-552.
(29) Asensio, J. L.; Jimenez-Barbero, J. The use of the AMBER force field in conformational analysis of carbohydrate molecules: Determination of the solution conformation of methyl α-lactoside by NMR spectroscopy, assisted by molecular mechanics and dynamics calculations. *Biopolymers:* **1995,** *35* (1), 55-73.
(30) Schölkopf, B.; Smola, A.; Müller, K.-R. In *Kernel principal component analysis*. International conference on artificial neural networks; Springer: 1997; pp 583-588.
(31) Williams, C. K.; Rasmussen, C. E. *Gaussian processes for machine learning*. MIT press: 2006.
(32) Vovk, V. Kernel ridge regression. In *Empirical inference*, Springer: 2013; pp 105-116.
(33) Cortes, C.; Vapnik, V. Support-vector networks. *Machine learning.* **1995,** *20* (3), 273-297.
(34) Kashima, H.; Tsuda, K.; Inokuchi, A. In *Marginalized kernels between labeled graphs*. Proceedings of the 20th International Conference on Machine Learning; 2003; pp 321-328.
(35) Borgwardt, K. M.; Kriegel, H.-P. In *Shortest-path kernels on graphs*. Fifth IEEE International Conference on Data Mining; IEEE: 2005; p 8 pp.
(36) Hermansson, L.; Johansson, F. D.; Watanabe, O. In *Generalized shortest path kernel on graphs*. International Conference on Discovery Science; Springer: 2015; pp 78-85.
(37) Fröhlich, H.; Wegner, J. K.; Sieker, F.; Zell, A. In *Optimal assignment kernels for attributed molecular graphs*. Proceedings of the 22nd International Conference on Machine Learning; 2005; pp 225-232.
(38) Kriege, N.; Mutzel, P. In *Subgraph matching kernels for attributed graphs*. Proceedings of the 29th International Coference on International Conference on Machine Learning; Edinburgh, Scotland, Omnipress: Edinburgh, Scotland, 2012; pp 291–298.
(39) Orsini, F.; Frasconi, P.; De Raedt, L. In *Graph invariant kernels*. Proceedings of the 24th International Joint Conference on Artificial Intelligence; IJCAI-INT JOINT CONF ARTIF INTELL: 2015; pp 3756-3762.
(40) Morris, C.; Kriege, N. M.; Kersting, K.; Mutzel, P. In *Faster kernels for graphs with continuous*

attributes via hashing. 2016 IEEE 16th International Conference on Data Mining; IEEE: 2016; pp 1095-1100.
(41) Togninalli, M.; Ghisu, E.; Llinares-López, F.; Rieck, B.; Borgwardt, K. In *Wasserstein weisfeiler-lehman graph kernels*. Advances in Neural Information Processing Systems; 2019; pp 6439-6449.
(42) Mahé, P.; Ueda, N.; Akutsu, T.; Perret, J.-L.; Vert, J.-P. In *Extensions of marginalized graph kernels*. Proceedings of the 21st International Conference on Machine Learning; 2004; p 70.
(43) Scarselli, F.; Gori, M.; Tsoi, A. C.; Hagenbuchner, M.; Monfardini, G. The graph neural network model. *IEEE Trans Neural Netw.* **2009,** *20* (1), 61-80.
(44) Zhou, J.; Cui, G.; Zhang, Z.; Yang, C.; Liu, Z.; Wang, L.; Li, C.; Sun, M. Graph neural networks: A review of methods and applications. *arXiv preprint arXiv:1812.08434.* **2018**.
(45) Kipf, T. N.; Welling, M. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907.* **2016**.
(46) Zhuang, C.; Ma, Q. In *Dual graph convolutional networks for graph-based semi-supervised classification*. Proceedings of the 2018 World Wide Web Conference; 2018; pp 499-508.
(47) Duvenaudt, D.; Maclaurin, D.; Aguilera-Iparraguirre, J.; Gomez-Bombarelli, R.; Hirzel, T.; Aspuru-Guzik, A.; Adams, R. P. In *Convolutional Networks on Graphs for Learning Molecular Fingerprints*. Advances in Neural Information Processing Systems; 2015.
(48) Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lio, P.; Bengio, Y. Graph attention networks. *arXiv preprint arXiv:1710.10903.* **2017**.
(49) Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; Polosukhin, I. In *Attention is all you need*. Advances in Neural Information Processing Systems; 2017; pp 5998-6008.
(50) Li, Y.; Tarlow, D.; Brockschmidt, M.; Zemel, R. Gated graph sequence neural networks. *arXiv preprint arXiv:1511.05493.* **2015**.
(51) Tai, K. S.; Socher, R.; Manning, C. D. Improved semantic representations from tree-structured long short-term memory networks. *arXiv preprint arXiv:1503.00075.* **2015**.
(52) Gilmer, J.; Schoenholz, S. S.; Riley, P. F.; Vinyals, O.; Dahl, G. E. Neural message passing for quantum chemistry. *arXiv preprint arXiv:1704.01212.* **2017**.
(53) Withnall, M.; Lindelöf, E.; Engkvist, O.; Chen, H. Building attention and edge message passing neural networks for bioactivity and physical–chemical property prediction. *J. Cheminf.* **2020,** *12* (1).
(54) Heller, S. R.; McNaught, A.; Pletnev, I.; Stein, S.; Tchekhovskoi, D. InChI, the IUPAC international chemical identifier. *J. Cheminf.* **2015,** *7* (1), 23.
(55) Weininger, D. SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules. *J. Chem. Inf. Comput. Sci.* **1988,** *28* (1), 31-36.
(56) Weininger, D.; Weininger, A.; Weininger, J. L. SMILES. 2. Algorithm for generation of unique SMILES notation. *J. Chem. Inf. Comput. Sci.* **1989,** *29* (2), 97-101.
(57) Landrum, G. RDKit: Open-Source Cheminformatics Software. http://www.rdkit.org/ (accessed Sep 1).
(58) Tang, Y.-H.; Selvitopi, O.; Popovici, D. T.; Buluç, A. In *A high-throughput solver for marginalized graph kernels on GPU*. 2020 IEEE International Parallel and Distributed Processing Symposium; IEEE: 2020; pp 728-738.
(59) Tang, Y. H.; de Jong, W. A. Prediction of atomization energy using graph kernel and active learning. *J. Chem. Phys.* **2019,** *150* (4), 044107.
(60) Diky, V.; Muzny, C. D.; Smolyanitsky, A. Y.; Bazyleva, A.; Chirico, R. D.; Magee, J. W.; Paulechka, Y.; Kazakov, A. F.; Townsend, S. A.; Lemmon, E. W. *ThermoData Engine (TDE) Version 10 (Pure Compounds, Binary Mixtures, Ternary Mixtures, and Chemical Reactions): NIST Standard Reference Database 103b*; National Institute of Standards and Technology: 2015.
(61) Knovel Data Analysis Beta: NIST ThermoDynamics Pure Compounds. https://app.knovel.com/web/poc/ms/discovery.html (accessed Sep 1).
(62) Kriege, N. M.; Johansson, F. D.; Morris, C. A survey on graph kernels. *Appl. Sci.* **2020,** *5* (1).
(63) Mahé, P.; Ueda, N.; Akutsu, T.; Perret, J.-L.; Vert, J.-P. Graph kernels for molecular structure–

activity relationship analysis with support vector machines. *J. Chem. Inf. Model.* **2005,** *45* (4), 939-951.
(64) Tsubaki, M.; Tomii, K.; Sese, J. Compound-protein interaction prediction with end-to-end learning of neural networks for graphs and sequences. *Bioinformatics.* **2019,** *35* (2), 309-318.

Table 1. Atom and Bond Features and their Hyperparameters (h) in Micro-Kernel Functions

features	elementary kernel	*h*
atom		
atomic number	δ	0.75
ring size (list)	C	0.9
ring (count)	δ	0.9
chirality	δ	0.9
Morgan substructure (r=3)	δ	0.9
atomic number (list, 1)	C	0.9
atomic number (list, 2)	C	0.9
atomic number (list, 3)	C	0.9
atomic number (list, 4)	C	0.9
hydrogen (count, 1)	δ	0.9
heavy atom (count, 1)	δ	0.9
heavy atom (count, 2)	δ	0.9
bond		
bond order	δ	0.9
E-Z (double bond)	δ	0.9
E-Z (ring bond)	δ	0.9

Table 2. Hyperparameters for Thermodynamic Variables

property	F^a	T_{red}	$T(\mathrm{K})$	$P(\mathrm{bar})$
T_c	120	/	/	/
$\mathrm{VLE}-\rho^l(T)$	180	0.1	/	/
$\mathrm{VLE}-\rho^v(T)$	30	0.02	/	/
$\mathrm{PT}-\rho(T,P)$	30	/	100	500

$^a F$ in eq 6.

Table 3. Summary of the Data Set Used in This Work

source	property	N_{mol}	N_T	N_P	N_{data}
Expt.	T_c	15422	-	-	15422
	$\mathrm{VLE}-\rho^l(T)$	14744	~80	1	1207667
	$\mathrm{VLE}-\rho^v(T)$	14414	~80	1	1181357
	$\mathrm{PT}-\rho(T,P)$	400	var	var	155116
Simu.	$\mathrm{PT}-\rho(T,P)$	7580	8	7	423501

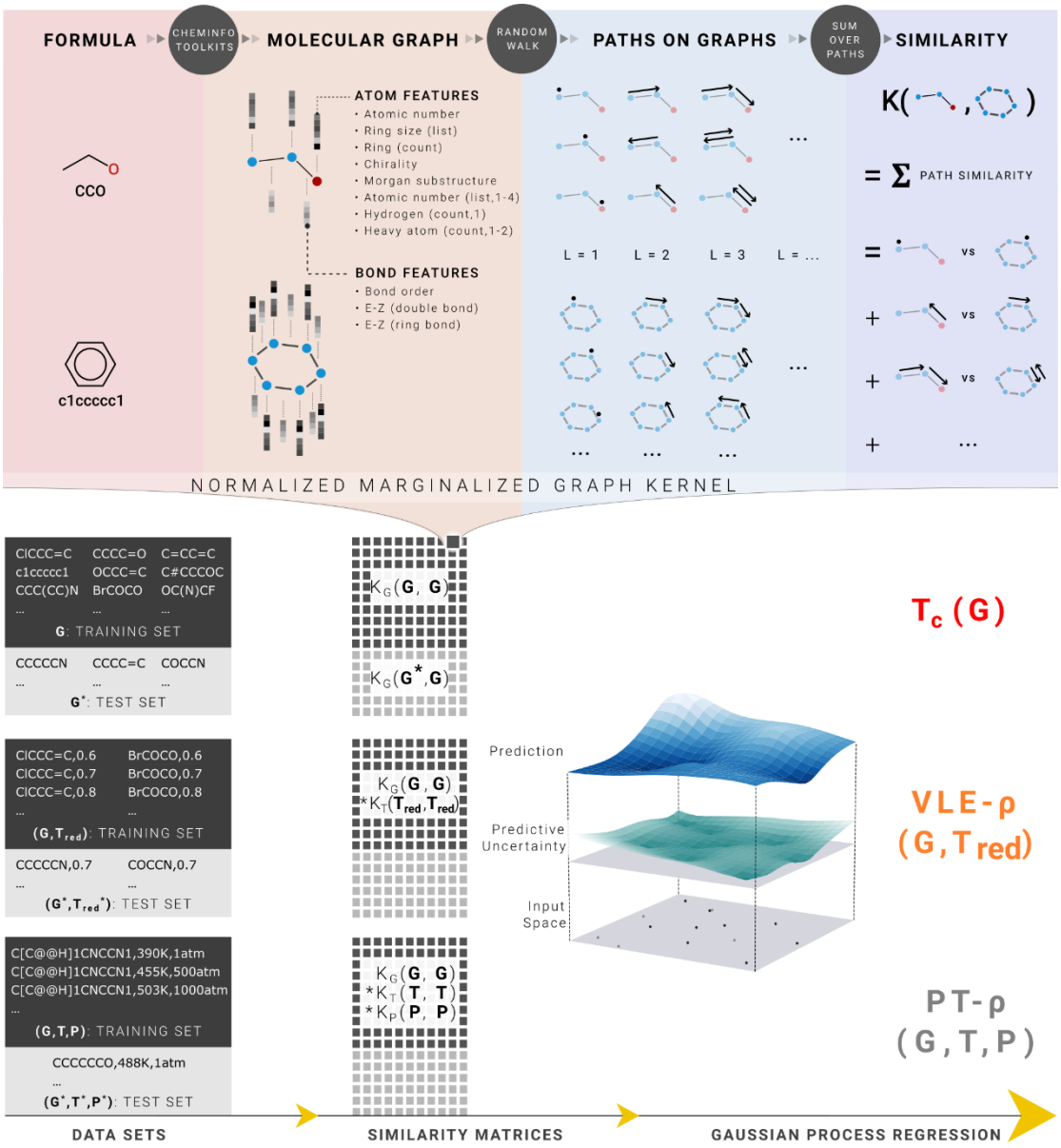

Figure 1. An overview of the machine learning pipeline proposed in this work. Upper: Molecules are first converted into labeled graphs, where a series of features are invented to describe the local environment of the atoms and bonds. Then the similarity between two graphs is computed by normalized marginalized graph kernel as the expectation of the similarity between all possible simultaneous random walk paths generated on each graph. Lower: Molecular graph, temperature, and pressure are respectively represented as G, T, P. Given the training set (**G**, (**G**, **T**) or (**G**, **T**, **P**)) and property-unknown test set (**G***, (**G***, **T***) or (**G***, **T***, **P***)) to be predicted, the pairwise similarity matrices

between and within each data set are computed. The training set self-similarity matrix and target values are used to construct a GPR model. The training-prediction cross-similarity matrix and prediction self-similarity matrix are used for properties prediction and uncertainty estimation.

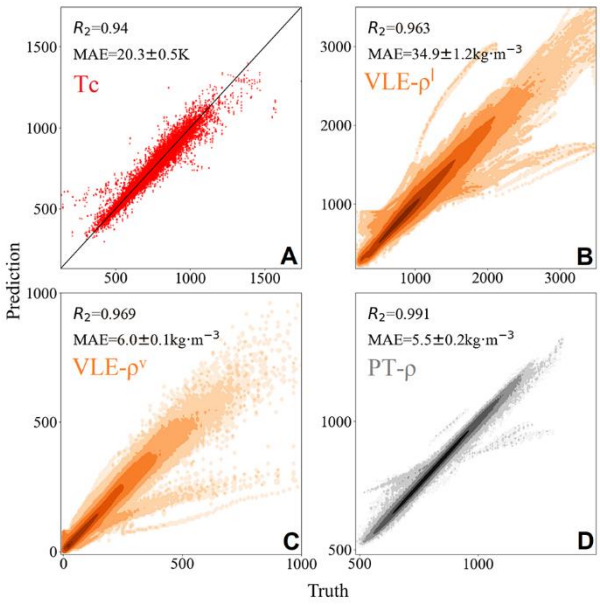

Figure 2. Relationship between the prediction and the truth value using GPR-nMGK-tMGR. (A) Critical temperature. (B) VLE liquid density. (C) VLE gas density. (D) Liquid density.

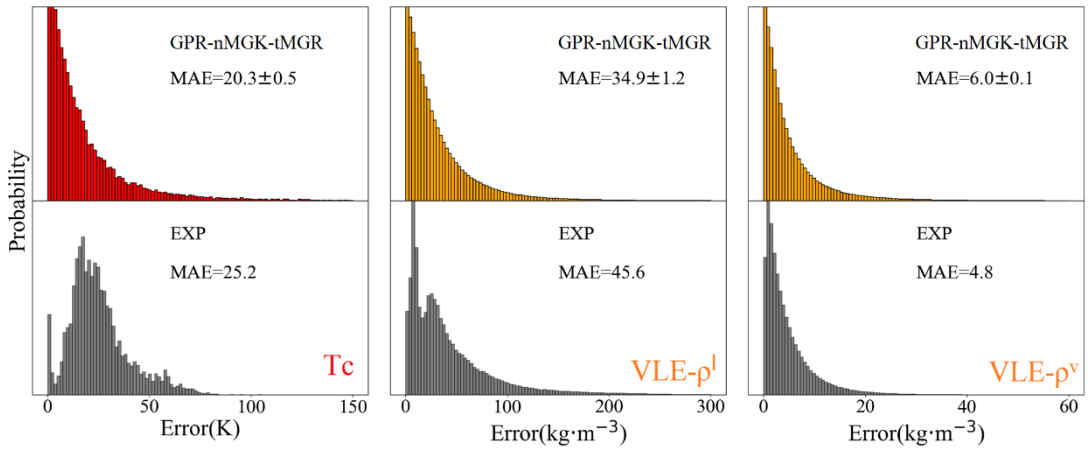

Figure 3. The probability histograms of the error of the predictions and the uncertainty provided by the database. (left) Critical temperature, (middle) VLE liquid density, and (right) VLE gas density.

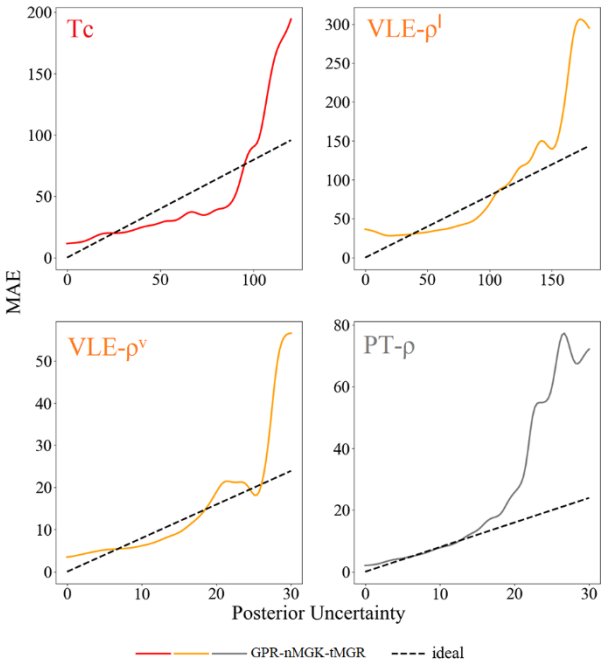

Figure 4. Relationship between the MAE and the posterior uncertainty of GPR-nMGK-tMGR (solid). $y = \sqrt{\frac{\pi}{2}}$ is the MAE of ideal Gaussian distribution. The unit of T_c is K and the unit of density is $\mathrm{kg} \cdot \mathrm{m}^{-3}$.

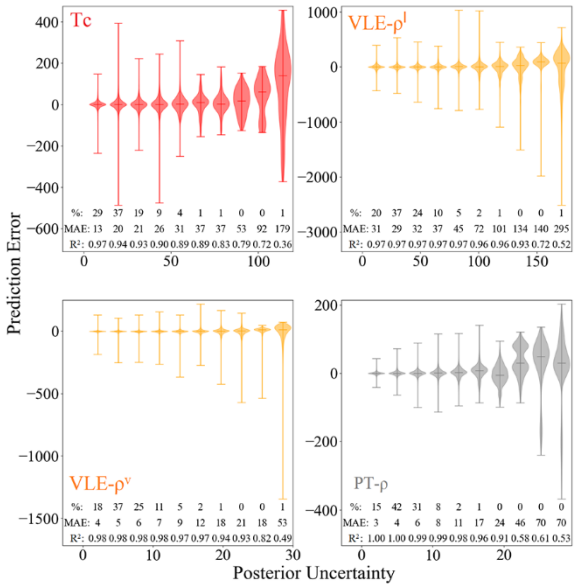

Figure 5. Relationship between predicted error and posterior uncertainty of GPR-nMGK-tMGR. Each string of the violin plot shows the distribution of predicted error within the corresponding posterior uncertainty interval. The unit of T_c is K and the unit of density is $\mathrm{kg} \cdot \mathrm{m}^{-3}$.

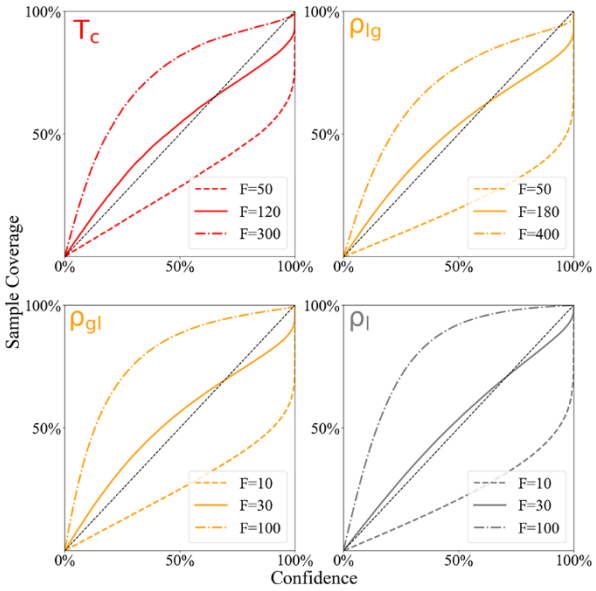

Figure 6. The proportional percentage of the target property is contained by the predictive confidence interval of GPR-nMGK-tMGR. The results of different hyperparameters *F* in eq 6 are compared.

.

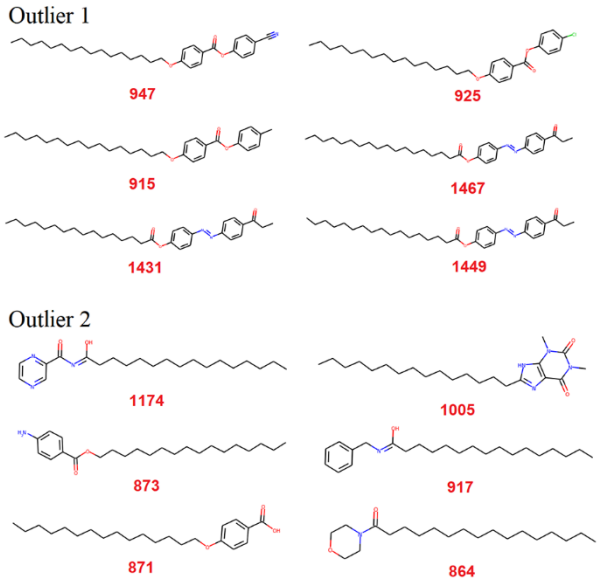

Figure 7. Examples of two groups of outliers in the critical temperature data set, where the molecules are similar to each other but their critical temperatures differ greatly.

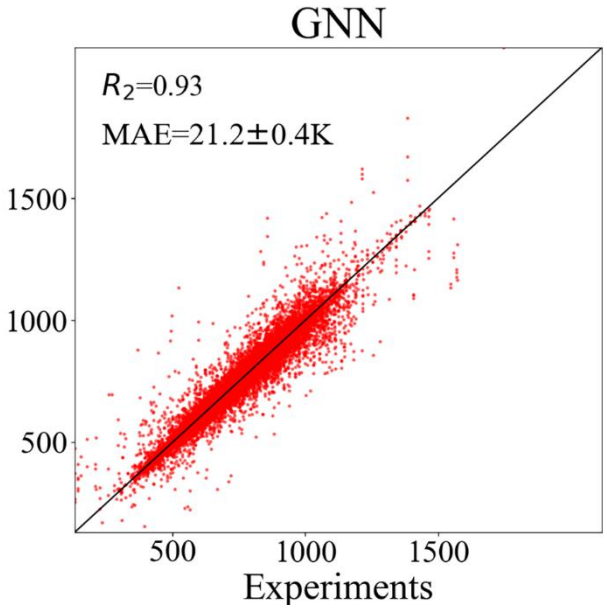

Figure 8. Relationship between the prediction and the truth value using GNN.

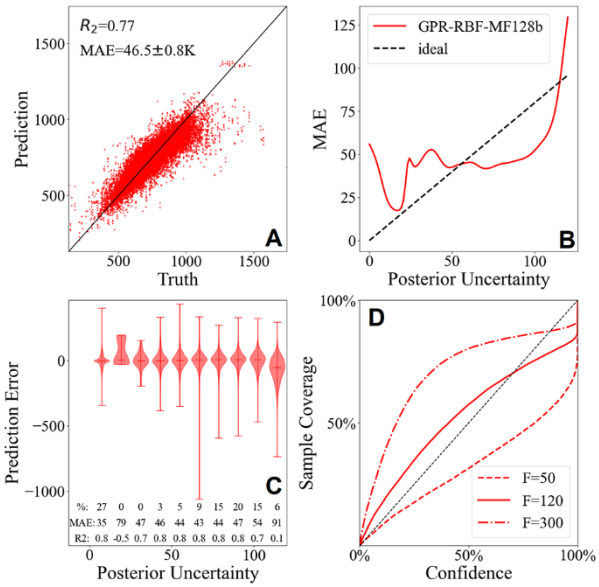

Figure 9. The results of GPR-RBF-MF128b. (A) Relationship between the prediction and the truth value. (B) Relationship between the MAE and the posterior uncertainty. (C) Relationship between predicted error and posterior uncertainty. (D) The proportional percentage of the critical temperature is contained by the predictive confidence interval.

Table of Contents (TOC) Image