GRAPPA - A Hybrid Graph Neural Network for Predicting Pure Component Vapor Pressures

Marco Hoffmann, Hans Hasse, and Fabian Jirasek*

Laboratory of Engineering Thermodynamics, RPTU Kaiserslautern,

Erwin-Schrödinger-Str. 44, 67663 Kaiserslautern, Germany

E-mail: fabian.jirasek@rptu.de

Abstract

Although the pure component vapor pressure is one of the most important properties for designing chemical processes, no broadly applicable, sufficiently accurate, and open-source prediction method has been available. To overcome this, we have developed GRAPPA - a hybrid graph neural network for predicting vapor pressures of pure components. GRAPPA enables the prediction of the vapor pressure curve of basically any organic molecule, requiring only the molecular structure as input. The new model consists of three parts: A graph attention network for the message passing step, a pooling function that captures long-range interactions, and a prediction head that yields the component-specific parameters of the Antoine equation, from which the vapor pressure can readily and consistently be calculated for any temperature. We have trained and evaluated GRAPPA on experimental vapor pressure data of almost 25,000 pure components. We found excellent prediction accuracy for unseen components, outperforming state-of-the-art group contribution methods and other machine learning approaches in applicability and accuracy. The trained model and its code are fully disclosed, and GRAPPA is directly applicable via the interactive website `ml-prop.mv.rptu.de`.

Introduction

The vapor pressure p^{s} of pure components is a key property in designing and optimizing many chemical processes. However, since experimental data for the vapor pressure are, by far, not available for all relevant components and at all relevant temperatures, prediction methods for this property are paramount. We can thereby distinguish between two types of prediction problems:

1. The interpolation or extrapolation of the vapor pressure to unstudied temperatures for components for which some experimental data (at other temperatures) are available.

2. The prediction of the vapor pressure of components for which no data (at any temperature) are available.

The first, much easier, type of prediction problem is usually solved by employing semi-empirical correlations of the vapor pressure as a function of the temperature, of which the Antoine equation is a simple example:

$$\ln\left(p^{\mathrm{s}}/\mathrm{kPa}\right) = A - \frac{B}{C + T/\mathrm{K}} \tag{1}$$

The Antoine equation is a good compromise between simplicity and accuracy and therefore widely used - and there are databases containing the Antoine parameters for several thousand pure components in the literature.[1,2] However, for the largest share of the pure components from the chemical space, no experimental data for the vapor pressure are available, so fitting semi-empirical correlations like Eq. (1) is infeasible. In these cases, prediction methods that can generalize over components are needed. In the following, we give a brief overview of such methods from the literature. First, we discuss established categories, which we divide into corresponding states, group contribution, quantitative structure-property relations (QSPR), and equation of state (EoS) based methods. Then, we will discuss recent machine learning (ML) developments. We acknowledge some ambiguities and that some methods might fit

into multiple categories.

Corresponding states methods.[3–5] These methods follow the idea that the thermophysical properties of many components are similar if the properties are reduced using the components' critical properties. Consequently, corresponding states methods are based on the availability of critical data for the components of interest, specifically the critical pressure p_c and the critical temperature T_c. Some methods also require the acentric factor ω (representing one additional data point of the vapor pressure curve). Furthermore, these methods are often limited to components of similar chemical structure. For example, the method of Ambrose and Walton[4] only applies to alkanes and 1-alcohols. For these reasons, corresponding states methods for predicting vapor pressures usually have a relatively small scope, hampering their practical applicability.

Group contribution methods.[6–17] The idea behind group contribution methods (GCMs) is to decompose a component into pre-defined structural groups (molecular building blocks) and to model the properties of components as a function of their group composition. Usually, additivity of the contributions of the individual structural groups to the value of the target property of the component is assumed (occasionally, also group interactions are considered), and group-specific parameters are fitted to the training data. The target property can then, in principle, be derived for any component that can be built from the parameterized structural groups. Theoretically, this allows GCMs to predict vapor pressures only from molecular structures. However, in practice, most methods require one additional experimental data point, such as, e.g., the normal boiling temperature[10,13,14] or the critical temperature.[17] Only very few GCMs, e.g., those by Pankow and Asher[12] and Tu,[9] apply to a wide range of components since they do not require any experimental data. Further limitations arise from the chemical space covered by the structural groups considered in the GCMs.

QSPR and early machine learning methods.[18–20] The idea of quantitative structure-property relations (QSPR) methods[21] is to identify the molecular descriptors that are most

informative for the thermophysical property to be predicted and to find a suitable correlation between these descriptors and the target property. These molecular descriptors are associated with the chemical structure of the molecule. Some simple examples are the dipole moment, the molecular volume and information on the connectivity of the atoms in the molecule.[20] The QSPR approach has been used to develop methods for predicting vapor pressures in different ways, e.g., the descriptors have been used as input for a multilinear regression[20] or the training of neural networks (NNs).[18,19]

Equation of state based methods.[22–27] These methods derive the parameters of an EoS from the molecular structure (or certain descriptors of which) and use this EoS to calculate thermophysical properties, enabling the EoS to generalize over components. For example, Hsieh et al.[22] introduced a Peng-Robinson (PR) EoS parameterized using descriptors obtained from COSMO[28,29] calculations. With the resulting PR+COSMOSAC EoS,[22–24] pure component vapor pressures can be predicted. There are also some recent works[25–27] that leverage EoS for pure component vapor pressure prediction, which are, however, discussed in the next section, as the focus of these works lies on the implementation of modern ML methods. Generally, in the EoS-based methods, the vapor pressure needs to be calculated iteratively, which is in contrast to the other methods discussed above that allow for a direct calculation. This makes them less practicable for an application in process simulators.

Recent ML-based methods.[25–27,30,31] Recently, many methods based on machine learning have been published to predict the thermophysical properties of pure components and mixtures.[25–27,30–44] Among them, several methods focus on predicting pure component vapor pressures, which all incorporate thermodynamic knowledge in their architecture and are thus not purely data-driven but hybrid methods.[45] These methods can, e.g., be distinguished by the type of molecular embedding used as input for the prediction.

One option is to use extended connectivity fingerprints (ECFP)[46] as molecular embedding. For example, Habicht et al.[25] and Felton et al.[26] trained NNs to regressed PC(P)-SAFT EoS[47,48] parameters using ECFPs as input. The vapor pressures can then be calculated

from the EoS. Another possibility is to derive the molecular embedding directly from the SMILES[49] string of a molecule, using, e.g., a transformer-based language model. This approach was followed by Winter et al.,[27] who also predicted PC-SAFT parameters with their model; they did not only train on vapor pressure data but also on density data.

Another approach is to derive a molecular embedding using a graph neural network (GNN). These have gained increasing interest for molecular property prediction in the last years, based, among others, on the initial work by Duvenaud et al.[50] The underlying idea is that a molecule can intuitively be represented as a graph, with atoms as nodes and bonds as edges. Using a GNN, the node representations are iteratively updated [†] (using the so-called message passing) based on the surrounding nodes (and edges). Thus, each node is enriched with information on its chemical environment. The final graph embedding is then used to predict a target property. Because the parameters of the GNN are adjusted during the training process, the obtained embedding is explicitly tailored to the target property. This learnable embedding from GNNs significantly improves the "static" molecular fingerprints (such as ECFP). Given that GNNs take into account the chemical environment of the atoms in the molecule, they also have the potential to outperform classic GCMs, as the latter usually do not use the complete connectivity information of the structural groups in a component. For this reason, GNNs have already been adopted for the prediction of activity coefficients,[32,33,44] solubility,[34] normal boiling points,[35,52] critical temperatures[52] and several other thermophysical, safety-related, and environmental properties.[38]

Very recently, two works were published in which GNNs were used to predict vapor pressures.[30,31] These developments were carried out in parallel and independently from those on which we report here. The PUFFIN[30] framework is built on a graph convolutional network (GCN) that was pre-trained to predict normal boiling points. In a transfer-learning approach, the embeddings of this pre-trained GCN were then used to predict Antoine parameters, from which the vapor pressures can be calculated. For training, Santana et al.[30] used experimental

[†]In some architectures, the edge embeddings are updated instead, see Ref.[51] for an example.

data (at 298 K) and synthetic data (at four other temperatures) of 1851 pure components. Their results prove that their model can extrapolate to unseen temperatures. However, the ability to generalize to unseen components remains unclear, as their data set was not split component-wise and their model was not disclosed to enable testing the model accordingly.

Lin et al.[31] trained a directed message passing NN to synthetic vapor pressure data for more than 19,000 pure components, which were computed using the coefficients of the Wagner equation, the critical temperatures, and the critical pressures from the NIST-TRC data bank. The authors compared a direct prediction of vapor pressures with the prediction of parameters of embedded correlations for the vapor pressure curve. They report that, using a component-wise split, the Wagner equation yielded the best results on their test set, which is, however, to be expected when the Wagner equation is used for generating the data.

In this work, we propose **GRAPPA** - a **GRA**ph neural network for **P**redicting the **P**arameters of the **A**ntoine equation. In contrast to the two very recent works in which GNNs were used for vapor pressure prediction,[30,31] and to which we compare our model, we use a graph attention network (GAT)[53] for the message passing step, a pooling layer that can capture complex interactions through self-attention, and we train and evaluate exclusively on experimental data of 24,753 pure components. We hybridize our model by incorporating the Antoine equation in our prediction head, facilitating the straightforward implementation of GRAPPA into established process simulators, as it directly yields the Antoine parameters.

The paper is organized as follows: In the section "Data", the database, all data preprocessing steps, the data splitting, and the creation of the initial molecular graphs from SMILES are explained. The section "Methods" contains all model architecture and training information. In the "Results and Discussion" section, the model predictions are comprehensively evaluated and compared to literature methods. The links to the source code, the trained model, and our interactive website can be found in the section "Model Availability". Finally, in the "Conclusion", we summarize our findings and provide an outlook for future work.

Data

Data Base and Curation

All vapor pressure p^s data for pure components were retrieved from the 2024 version of the Dortmund Data Bank (DDB).[1] We used only those components for which a canonical SMILES string could be generated from the DDB mol-files (describing the molecular structure) with the *rdkit*[54] package. Each data point consists of a SMILES representing the molecule, the temperature, and the corresponding experimental value for p^s . For the training process of the GNN, sufficient training data for each node type (i.e., the atom the node represents) is required. Because these data are not available for components that contain less common atoms (such as Ge, Fe, ...), we have a priori restricted the chemical space to define a clear applicability range of the model. The same holds for the temperature and pressure ranges. Thus, in the pre-processing, we first only extracted data points that fulfilled the following criteria:

- The component contains at least one carbon atom.
- The component only consists of the following atoms: C, N, O, Cl, S, F, Br, I, P.
- No atom in the component has formal charge or unpaired electrons.
- The temperature lies between 250 K and 600 K.
- The pressure lies between 1 and 10^7 Pa.

Data points marked as "poor quality" in the DDB were also excluded. Furthermore, to enable the model to differentiate between stereoisomers, only those data points were kept for which the isomerism was correctly represented in the SMILES. In the next step, components with conflicting data points (e.g., two contradicting vapor pressure curves from different sources) were manually removed. To additionally eliminate single outliers, the Antoine equation (cf. Eq. (1)) was fitted (using robust least squares) individually for all components with at

least five data points, and those data points with a relative deviation of p^s larger than 50 % from the Antoine fit were removed. For roughly a third of the components, there was only one available data point, and five or more data points were available for only about 17 % of the components. Consequently, removing faulty values or outliers was infeasible for a large share of the components. However, because it is crucial to cover the chemical space as well as possible, we decided to keep these components in the data set. After all pre-processing steps, the final data set contained 227,062 data points for 24,753 components. Fig. 1 shows the distribution of the final data set over pressure and temperature in two histograms.

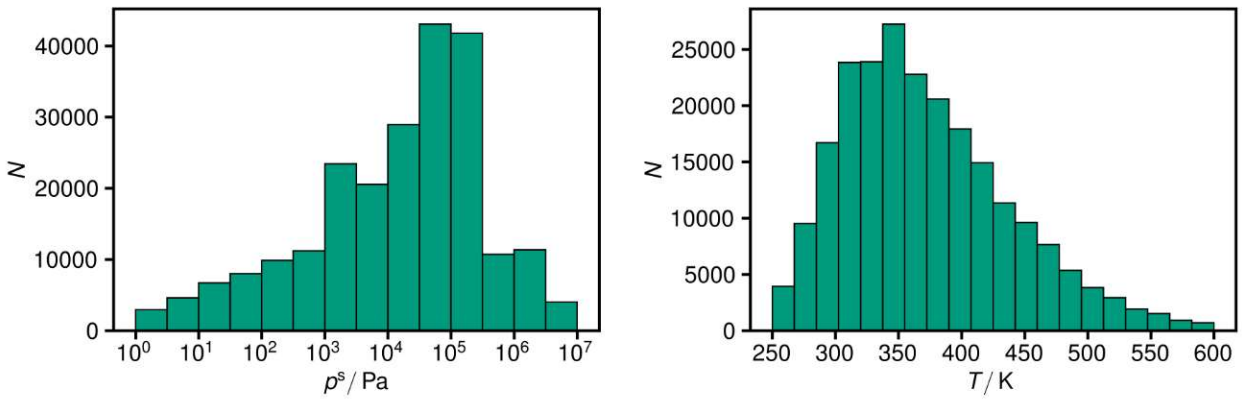

Figure 1: Histograms visualizing the distribution of the data points in the final data set after pre-processing. Left: Number of data points over the pressure. Right: Number of data points over the temperature.

Data Split

Our final data set was split component-wise: 80 % of the components were used as the training set, 10 % as the validation set, and the remaining 10 % as the test set. This split was done randomly, except for data for molecules with less than five carbon atoms, which were all put in the training set. The motivation for this procedure is that the extrapolation from large to small molecules is difficult for GNNs due to their message passing steps (cf. "Model Architecture"). However, while smaller molecules are more difficult to model, they are usually well-measured, so prediction methods for their properties are less necessary than

ones for more complex molecules, for which we test the model developed in this work. All data in the training set were used to fit the parameters of GRAPPA, and the validation set was used to optimize the hyperparameters and select the best model architecture. The test set was only used for the conclusive evaluation of GRAPPA. The final model disclosed with this work was trained to all available data.

Generation of the Initial Molecular Graphs

The SMILES of each molecule was converted to an initial molecular graph using *rdkit*.[54] In the molecular graph $\mathcal{G}$, the atoms (except for hydrogen, which is treated implicitly, cf. below) are represented as a set of nodes $\mathcal{V}$, and the bonds are represented as a set of bonds $\mathcal{E}$. All nodes and edges are represented by an individual feature vector encoding information about the corresponding atom or bond. The vector containing the features of node i is denoted as $\boldsymbol{x}_i$, and the features of the edge between nodes i and j are summarized in the vector $\boldsymbol{e}_{i,j}$. The node features in the initial molecular graph are the atom type, the number of bonds, the number of bonded hydrogen atoms, the hybridization, whether the atom is aromatic, and whether the atom is part of a ring. The edge features contain the bond type and whether the bond is conjugated, part of a ring, or part of a stereoisomer. All features with more than two classes were one-hot encoded, resulting in 24 node features and nine edge features. We have adapted some parts of the code of Sanchez-Medina et al.[32] to implement the molecular graph generation.

Methods

Model Architecture

The proposed model consists of three parts: First, a message passing between the nodes is applied to the initial molecular graph to enrich the graph embedding with information about the vicinities of the nodes. Second, the pooling layer derives a numerical embedding of fixed

length from the learned graph embedding. Finally, this numerical embedding serves as input for a FFNN that predicts component-specific vapor pressure curves in the form of Antoine parameters (subsequently called prediction head). The three parts are described in more detail in the following. A schematic overview of the model architecture is given in Fig. 2.

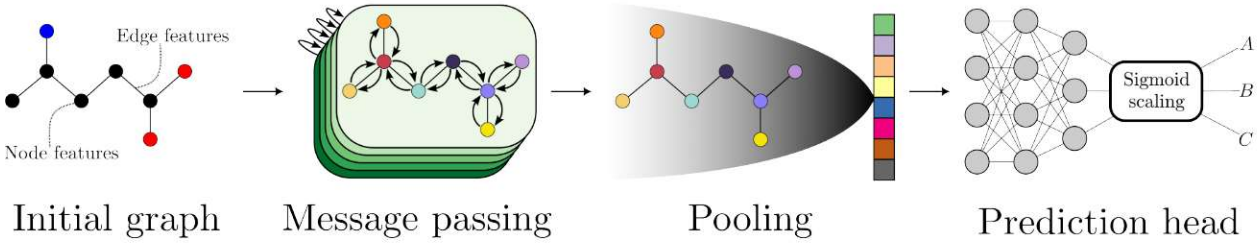

Figure 2: Schematic overview of the architecture of GRAPPA, with the initial molecular graph as the input and the three Antoine parameters (A,B,C) as the output. Details on the individual steps are given in the text.

Message passing. During the message passing, the graph embedding is updated, and information is exchanged between the nodes. From a chemical perspective, this enriches the node embeddings with information about their local environment and improves their expressivity. GRAPPA uses a graph attention network (GAT)[53] for message passing, more specifically, the revised implementation *GATv2Conv*[55] in *pytorch-geometric*.[56] The starting point of the message passing is the initial molecular graph to which multiple message passing layers are subsequently applied. In one message passing layer, the embeddings of all N nodes in the graph are updated as follows:

$$\boldsymbol{x}_i^{'} = \sum_{j \in \mathcal{N}(i) \cup i} \alpha_{i,j} \boldsymbol{\Theta}_{\mathcal{V}} \boldsymbol{x}_j \quad \text{for} \quad i = 1, ..., N \tag{2}$$

Here, $\boldsymbol{x}_i^{'}$ is the updated embedding of node i, calculated based on the embeddings of node i and all neighboring nodes $\mathcal{N}(i)$ from the previous layer, a trainable weight matrix $\boldsymbol{\Theta}_{\mathcal{V}}$, and the attention coefficient $\alpha_{i,j}$. In the first layer, the input embedding dimension (the length of the vector $\boldsymbol{x}_i$) is defined by the number of node features of the initial graph. For the output embedding of the first layer and all embeddings in the subsequent layers, we chose

the embedding dimension $d = 32$. The advantage of *GATv2Conv* over other message passing layers is that the attention coefficient $\alpha_{i,j}$ is learnable and weighs the incoming information from the neighboring nodes differently. Thus, the local substructures that have a high impact on p^{s} can be identified by inspecting the values of $\alpha_{i,j}$ after model training; an example is given in section "Examples for Predicted Vapor Pressure Curves and Evaluation of Attention Scores". The attention coefficient for two neighboring nodes i and j is computed with

$$\alpha_{i,j} = \frac{\exp\left(\boldsymbol{a}^{\top}\mathrm{LeakyReLU}\left(\boldsymbol{\Theta}_{\mathcal{V}}\boldsymbol{x}_i + \boldsymbol{\Theta}_{\mathcal{V}}\boldsymbol{x}_j + \boldsymbol{\Theta}_{\mathcal{E}}\boldsymbol{e}_{i,j}\right)\right)}{\sum_{k\in\mathcal{N}(i)\cup i}\exp\left(\boldsymbol{a}^{\top}\mathrm{LeakyReLU}\left(\boldsymbol{\Theta}_{\mathcal{V}}\boldsymbol{x}_i + \boldsymbol{\Theta}_{\mathcal{V}}\boldsymbol{x}_j + \boldsymbol{\Theta}_{\mathcal{E}}\boldsymbol{e}_{i,k}\right)\right)} \tag{3}$$

Here, $\boldsymbol{a}$ and $\boldsymbol{\Theta}_{\mathcal{E}}$ are trainable weight matrices, and *LeakyReLU* is the leaky rectified linear unit activation function. The embedding vector of the edge between nodes i and j is denoted as $\boldsymbol{e}_{i,j}$. We have used the multi-head attention option of the *GATv2Conv* module. The idea behind multi-head attention is to employ Eqs. (2) and (3) multiple times with different, independent parametrizations. Each of these attention heads thereby learns to focus on other substructures. In a multi-head attention with N_{MH} attention heads, the message passing layer thus produces N_{MH} distinct updated node embeddings $\boldsymbol{x}_i'$ for each node. The mean of the N_{MH} node embeddings is then calculated to retain the original embedding size.

Pooling function. The pooling function takes the final graph embedding $\boldsymbol{X} \in \mathbb{R}^{N\times d}$ and produces a fixed-size numerical embedding $\boldsymbol{h} \in \mathbb{R}^{d}$ for each molecule. This step is necessary as the molecules (and therefore the graphs) differ in size, but the prediction head requires an embedding of a pre-defined size as input. In our work, we tested two pooling variants. The first one is standard *sum* pooling where the fixed-size embedding $\boldsymbol{h}$ is calculated as sum of all individual node embeddings $\boldsymbol{x}_i$ in the graph:

$$\boldsymbol{h} = \sum_{i=1}^{N}\boldsymbol{x}_i \tag{4}$$

Additionally, we have tested an attention-based pooling method, similar to Refs.,[57,58] which is built upon the self-attention algorithm.[59] In the first step, using the trainable weight

matrices $\boldsymbol{W}_\mathrm{q} \in \mathbb{R}^{d\times k}, \boldsymbol{W}_\mathrm{k} \in \mathbb{R}^{d\times k}$ and $\boldsymbol{W}_\mathrm{v} \in \mathbb{R}^{d\times d}$, the query matrix $\boldsymbol{Q}$, the key matrix $\boldsymbol{K}$, and the value matrix $\boldsymbol{V}$ are built from the node embedding matrix $\boldsymbol{X}$:

$$\boldsymbol{Q} = \boldsymbol{X} \cdot \boldsymbol{W}_\mathrm{q}, \quad \boldsymbol{K} = \boldsymbol{X} \cdot \boldsymbol{W}_\mathrm{k}, \quad \boldsymbol{V} = \boldsymbol{X} \cdot \boldsymbol{W}_\mathrm{v} \quad \text{with} \quad \boldsymbol{Q}, \boldsymbol{K} \in \mathbb{R}^{N\times k}; \quad \boldsymbol{V} \in \mathbb{R}^{N\times d} \tag{5}$$

For simplicity, we have chosen the key dimension k equal to the embedding dimension ($k = d = 32$). The three matrices are used to calculate the context matrix

$$\boldsymbol{Z} = \mathrm{softmax}\left(\frac{\boldsymbol{Q}\boldsymbol{K}^\top}{\sqrt{d}}\right) \cdot \boldsymbol{V} \quad \text{with} \quad \boldsymbol{Z} \in \mathbb{R}^{N\times d}. \tag{6}$$

The first term softmax(...) is a matrix of dimension $\mathbb{R}^{N\times N}$ and represents the attention weights, i.e. values for the interaction of all binary node pairs i and j in the graph. The multiplication with the value matrix $\boldsymbol{V}$ results in the context matrix $\boldsymbol{Z}$, which has the same dimensions as the graph embedding matrix $\boldsymbol{X}$ after the message passing, but contains information on the interaction between all nodes in the graph. Finally, the molecule's fixed-size embedding $\boldsymbol{h}$ is obtained by summing up the context vectors $\boldsymbol{z}_i$ of all nodes in the graph, analogous to Eq. (4). This pooling approach allows the model to capture interactions between all atoms in the molecules, which is especially relevant for large molecules with multiple functional groups farther apart than the message passing distance. It can also be interpreted as a final, global message passing step, followed by *sum pooling*. We refer to this approach as *interaction pooling* in the following.

Prediction head. The prediction head predicts the Antoine parameters using the numerical embedding $\boldsymbol{h}$ from the pooling function and the number of hydrogen donors and acceptors in the molecule, which were obtained through *rdkit*.[54] These are concatenated to the embedding $\boldsymbol{h}$ because preliminary studies showed that this increases the prediction accuracy. The prediction head was realized by a feed-forward neural network (FFNN) using hidden layers with 16 neurons. We employed batch normalization and the ELU activation function between all layers. The last layer of the FFNN yields three outputs for the three

Antoine parameters. To obtain the final parameters A, B, and C (cf. Eq. (1)), they are scaled using sigmoid functions. We found this approach to increase training stability and decrease the number of epochs to convergence. The parameter ranges for the scaling were defined as follows:

$$A \in [5, 20], \quad B \in [1500, 6000], \quad C \in [-300, 0]. \tag{7}$$

These were determined by fitting the Antoine equation for all components with at least ten data points in the training set. The parameter ranges in Eq. (7) were chosen slightly larger than those obtained by the fit to allow for extrapolation to unseen components with different behavior than those in the training set. By limiting the range of the B parameter to positive values, we have enforced the correct slope of the vapor pressure curve. The final prediction for the vapor pressure is then calculated using Eq. (1) and the temperature in Kelvin.

Model Training

The model is implemented using *pytorch*[60] and *pytorch-geometric*,[56] and the message passing layer, the pooling function, and the prediction head (including the Antoine equation) are connected in the computational graph and can therefore be trained in an end-to-end manner. The model training was divided into two parts: First, we performed a pre-training with the mean squared error (MSE, cf. Eq.(8)) as the loss function for 100 epochs. Then, we trained for 200 epochs using the Huber loss with a threshold of 0.5. The Huber loss transitions from the MSE loss to the mean absolute error (MAE, cf. Eq.(8) loss at the threshold, which prevents outliers from having a too large impact on the loss and, therefore, on the adjustment of the parameters. In both trainings, we used the *AdamW* optimizer.[61] We used the *OneCycleLR*[62] learning rate scheduler for the pre-training and the *ReduceLROnPlateau* scheduler for the main training. To prevent overfitting, an early stopping strategy was applied, whereby the best model on the validation set based on the median absolute percentage deviation (MAPE_i,

cf. "Training and Evaluation Metrics") was chosen.

We performed a grid search over the number of GAT layers, the number of GAT attention heads, and the number of hidden layers in the prediction head. Furthermore, we tested standard *sum* pooling and the *interaction pooling* approach presented above. The best model in this grid search had four GAT layers with two GAT attention heads each, three hidden layers in the prediction head, and used *interaction pooling*. All hyperparameters and the ranges covered in the grid search are given in Tab. S1 of the Supporting Information. The final model contains 15,319 trainable parameters.

Training and Evaluation Metrics

Different error metrics were used for the training and evaluation of GRAPPA. The mean absolute error (MAE) and mean squared error (MSE)

$$\begin{aligned} \mathrm{MAE} &= \frac{1}{M}\sum_{i=1}^{M} \left| \ln\left(p^{\mathrm{s}}_{\mathrm{pred},i}/\mathrm{kPa}\right) - \ln\left(p^{\mathrm{s}}_{\mathrm{exp},i}/\mathrm{kPa}\right) \right| \\ \mathrm{MSE} &= \frac{1}{M}\sum_{i=1}^{M} \left(\ln\left(p^{\mathrm{s}}_{\mathrm{pred},i}/\mathrm{kPa}\right) - \ln\left(p^{\mathrm{s}}_{\mathrm{exp},i}/\mathrm{kPa}\right)\right)^2 \end{aligned} \tag{8}$$

were calculated based on the natural logarithm of the vapor pressures in kPa. Here, M denotes the number of data points in the considered data set. Because it is a more practical error metric, we also use absolute percentage error (APE) of each data point i defined as

$$\mathrm{APE}_i = \left| \frac{p^{\mathrm{s}}_{\mathrm{pred},i} - p^{\mathrm{s}}_{\mathrm{exp},i}}{p^{\mathrm{s}}_{\mathrm{exp},i}} \right| \cdot 100\,\%. \tag{9}$$

Additionally, we calculated the component-wise absolute percentage error APE_C by averaging the APE_i over the K data points of the considered component C:

$$\mathrm{APE}_C = \frac{1}{K}\sum_{i=1}^{K} \mathrm{APE}_i \tag{10}$$

To measure the prediction accuracy over a considered data set, besides MAE and MSE, the medians of the APE_i and APE_C were used, as the median is more robust towards outliers in the data than the mean; they are labeled $MAPE_i$ and $MAPE_C$, respectively, in the following. Because the $MAPE_C$ is less biased towards well-studied components, it is used as the primary evaluation metric.

Results and Discussion

Performance of GRAPPA for Vapor Pressure Prediction

Tab. 1 summarizes all GRAPPA evaluation metrics on the train, validation, and test set.

Table 1: Error scores of GRAPPA for modeling p^s from the training (train), validation (valid), and test data set. The $MAPE_C$ is given for all components in the respective set, for components with at least two data points ($K \geq 2$) in the set only, and for components with at least five data points ($K \geq 5$) in the set only.

Data set	MAE	MSE	$MAPE_i$	$MAPE_C$	$MAPE_C$ ($K \geq 2$)	$MAPE_C$ ($K \geq 5$)
Train	0.127	0.148	4.39	22.77 %	13.65 %	9.60 %
Valid	0.233	0.480	8.53	26.90 %	17.45 %	12.35 %
Test	0.201	0.301	6.33	26.70 %	16.42 %	12.54 %

The scores are of similar magnitude for all data sets, demonstrating that the model did not overfit the training data and can generalize well to unseen components. The table lists the $MAPE_C$ for all components and, separately, only for those with at least two or at least five experimental data points. The $MAPE_C$ for components with at least two data points is almost half that for all components for all data sets, suggesting that the model predictions for the single-measured components deviate stronger from the experimental values. As discussed above (cf. "Data Split"), no checks regarding faulty data or outliers could be carried out for components with less than five data points. Because these larger deviations for single-measured components appear not only in the test data set but also in the training and validation data set, we can assume that many of these single data points carry significant measurement errors and are poorly suited for model evaluation. Because these data

potentially skew the MAPE_C, further evaluation is carried out only on the components for which at least two experimental data points are available. More information on the influence of the number of experimental data points on the prediction accuracy in the test set is given in Fig. S6 of the Supporting Information.

In Fig. 3, a hexbin plot in terms of the MAPE_i on the test data is shown to visualize the dependence of the prediction accuracy on pressure and temperature. A significant decline in

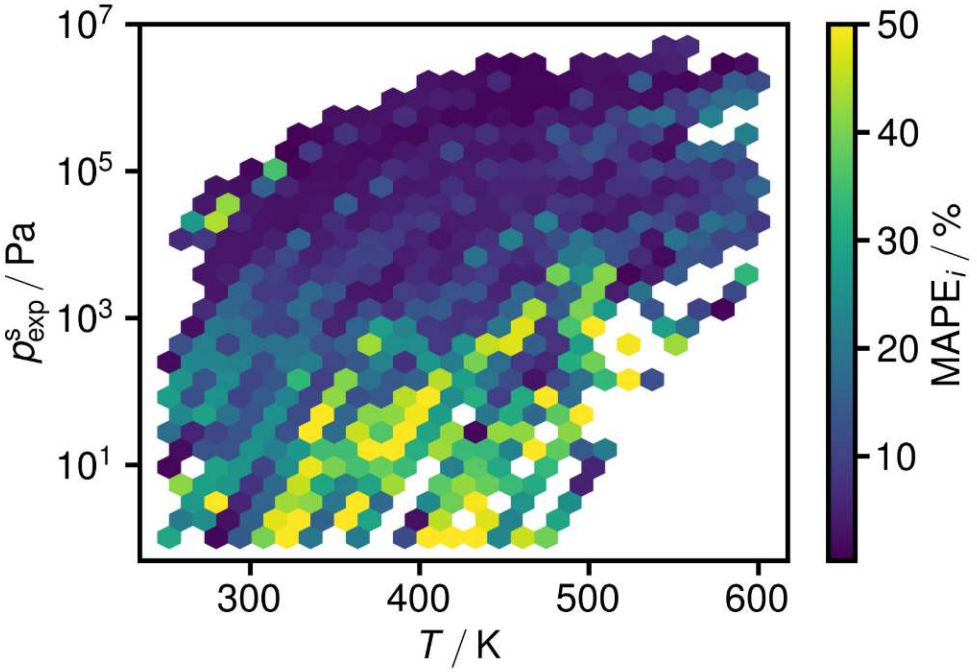

Figure 3: Hexbin plot to visualize the relation between the MAPE_i, the pressure, and the temperature on the test data set. Only results for components with at least two experimental data points are shown. Each hexagon is colored according to the MAPE_i of all data points in the covered area. Values larger than 50 % are clipped for readability.

prediction accuracy can be observed for very small pressures, approximately below 1 kPa. There are three reasons for this: First, and most importantly, the measurement uncertainties in the low-pressure regime are much larger than at higher pressures. Second, using a single set of Antoine parameters to describe the entire vapor pressure curve over a wide pressure range is known to cause issues in either the low- or the high-pressure regime. Third, the number of training data points in the low-pressure regime is smaller than in the medium- to the higher-pressure regime (cf. Fig. 1), leading to a generally lower prediction accuracy

for low pressures. Given these results, the prediction of GRAPPA for pressures below 1 kPa should be treated carefully. In contrast to pressure, temperature does not significantly impact prediction accuracy. The reader is referred to the Supporting Information for a deeper analysis of the pressure and temperature dependence of the prediction accuracy in Figs. S3 and S4.

Comparison to Literature Methods

We directly benchmarked GRAPPA against three group contribution methods from the literature for which the parametrization is available, namely the methods by Tu,[9] the SIMPOL method,[12] and the method by Nannoolal et al.[13] (incorporating the corresponding normal boiling point prediction method[63] from the same authors). These methods were chosen for comparison because they require only the molecular structure as input and are, therefore, comparable to GRAPPA in terms of general applicability. However, due to missing parameters for certain structural groups, the three group contribution methods are only applicable to a subset of the test data; more specifically, SIMPOL is only applicable to 53.8 % of the components from the test set (SIMPOL horizon), the method by Tu to 36.1 % (Tu horizon), and the method by Nannolal et al. to 94.9% (Nannolal horizon). GRAPPA was also evaluated separately on these subsets of our test set to enable a fair comparison. The results of this comparison in terms of APE_C on the test set from this work are shown in a boxplot in Fig. 4. It should be noted that this comparison is likely biased in favor of the literature methods, as it can be assumed that at least some of the data in our test set were used for training these methods, while this was not the case for GRAPPA. This is especially relevant for the method by Nannoolal et al., which was fitted to large amounts of data from the DDB, almost certainly including a large share of the components from our test set. Despite this bias in favor of the literature methods, GRAPPA significantly outperforms all three group contribution methods in both applicability and accuracy.

Of the three EoS-based prediction methods for vapor pressures discussed above, only

Felton et al.[26] disclosed their model. However, because they only included molecules with less than 15 atoms in their training data and our test data set contains mainly larger molecules, we refrained from a quantitative comparison on our test data set. On their test data set, they state an average error of 40 % for the prediction of vapor pressures, which is much higher than the error scores of GRAPPA. Habicht et al.[25] do not give an explicit number for their prediction accuracy for the vapor pressure but state that in most cases, the average absolute relative deviation (AARD, analogous to our APE_C) is below 25 %. The best scores of the EoS-based models are reported by the authors of SPT-PC-SAFT[27] with an $MAPE_C$ of 8.7 %. However, their data set has been extensively cleaned and contains only components with at least five data points for the vapor pressure and at least three data points for the density, which were also used for training their model.

Finally, we compare the results of GRAPPA to the two recent works that also use GNNs for the prediction of pure component vapor pressures. The trained PUFFIN[30] model is not disclosed, preventing a comparison on the same test data. The authors report an MSE of 0.1609 on the base-10 logarithmic vapor pressures on their test set. Converted to the natural logarithmic vapor pressures, this corresponds to an MSE of 0.853, which is significantly higher than the MSE of GRAPPA on our test set, which is 0.301.[‡] Lin et al.[31] disclosed their trained model, which we evaluated on our test set, thereby ignoring that some of our test data might have been part of their training set (cf. discussion on the favorable bias above). We used their best model, more precisely, the published ensemble of ten models with the embedded Wagner equation, and provided the canonical SMILES and the temperature as input. The results are also included in the boxplot in Fig. 4 (left). Despite comparing a single GRAPPA model with an ensemble of their models (of which each has 20 times more parameters than GRAPPA), GRAPPA significantly outperforms the model of Lin et al. Besides demonstrating the high prediction accuracy of GRAPPA, these results also indicate that the Wagner equation, as used in the model of Lin et al., while being indisputably the

[‡]The MAE and MSE of GRAPPA reported here were calculated on the entire test data set, including the components with only a single experimental data point, of which we assume many to be erroneous.

more accurate correlation compared to the Antoine equation when fitted, simply because of the increased flexibility due to the higher number of parameters, is not necessarily the better choice in a predictive scenario.

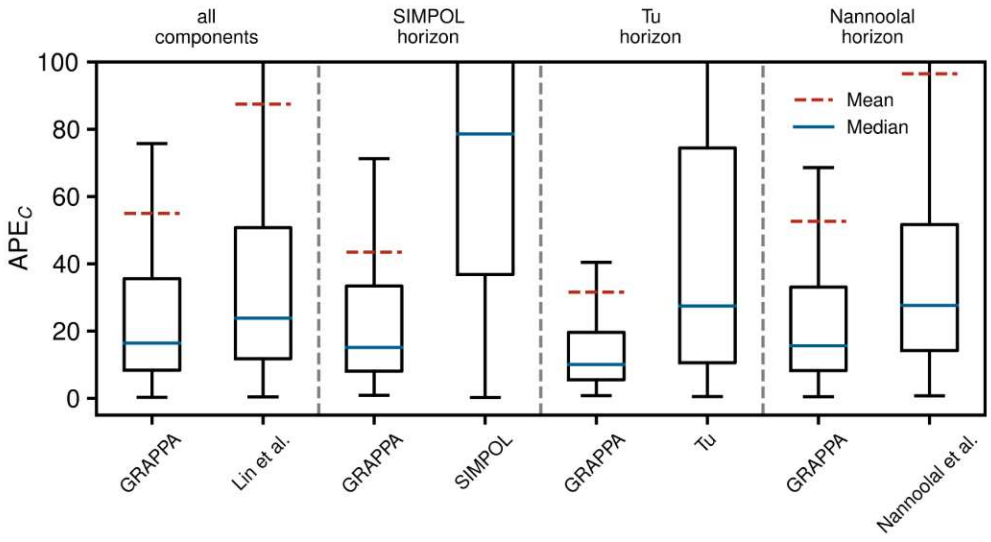

Figure 4: Boxplot comparing the prediction accuracy of GRAPPA with four literature methods on the test set of GRAPPA for components with at least two experimental data points. The boxes represent the interquartile range, and the whiskers are 1.5 times the interquartile range. Because some literature methods are limited in their applicability, GRAPPA was also evaluated only on components in their scope: the SIMPOL[12] horizon contains 53.8 % of our test components, that of the method by Tu[9] 36.1 %, and that of the method by Nannoolal et al.[13] 94.9%. The method by Lin et al.[31] is applicable to our entire test set.

Boiling Point Prediction

The Antoine equation can easily be rearranged to allow predictions for the boiling temperature at a given pressure. Hence, the Antoine parameters predicted with GRAPPA can directly be used for predicting boiling temperatures for any component whose molecular structure is known and that fulfills the criteria defined for the pre-processing (cf. section "Data Preparation"). Because the normal boiling point T_b is of great technical relevance, we evaluate the prediction accuracy of GRAPPA for T_b in the following.

For this purpose, we have first collected the experimental data points from our test set with a pressure between 99 kPa and 102 kPa (we, again, restrict the study to components

for which at least two data points, irrespective of the pressure, were in our test set). In cases with more than one data point in that range, we calculated the mean pressure and temperature. Using the Antoine parameters obtained with GRAPPA and the (mean) pressures, we predicted T_b for each component.

A comparison of the GRAPPA predictions and the experimental data for the normal boiling point is shown in a parity plot in the left panel of Fig. 5. The mean absolute error of T_b is 4.76 K and the mean relative error is 1.05 %. For comparison, we also calculated the normal boiling points of the same components with the GCM of Nannoolal et al.[63] and show the corresponding parity plots in the right panel of Fig. 5. Only for twelve components, the method by Nannoolal et al. could not be applied. The mean absolute error is 7.88 K and the mean relative error is 1.73 % and hence, clearly larger than for GRAPPA, despite the bias in favor of the method of Nannooal, which has presumably been trained on most of these data points, cf. discussion above.

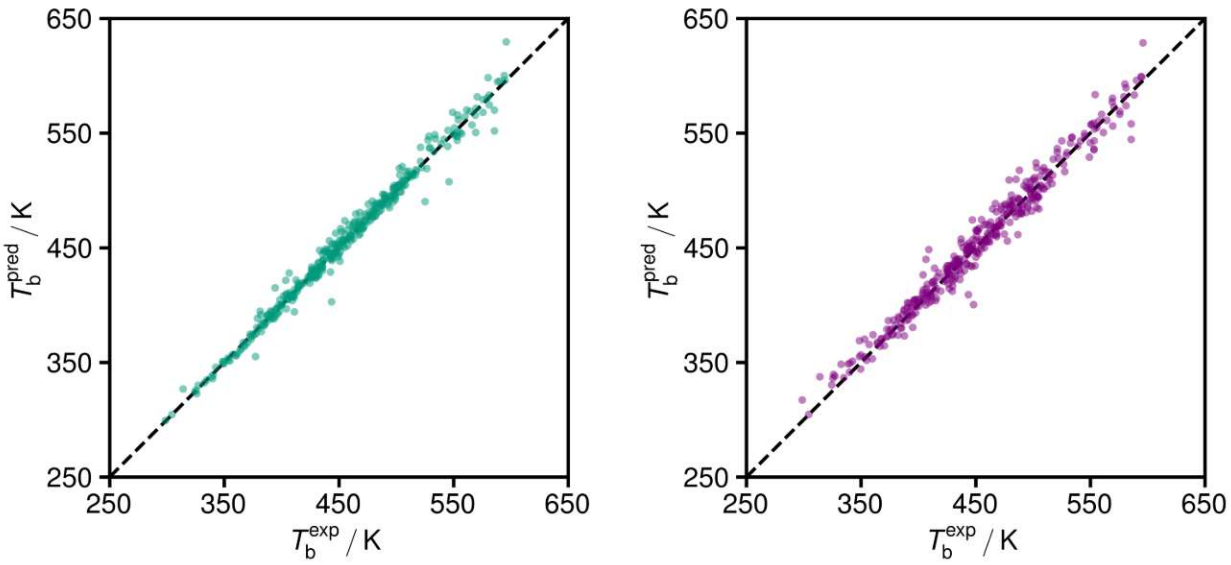

Figure 5: Parity plots showing predicted normal boiling points with GRAPPA (left) and the group contribution method by Nannoolal et al.[63] (right) over our experimental test data. Only data for components with at least two experimental data points in our test set are shown. For twelve of the components, a prediction using the method by Nannoolal et al. was infeasible due to problems with group fragmentation. The dashed line marks perfect predictions.

A recent publication[35] of a GNN that was explicitly trained for predicting normal boiling points achieves a similar performance as GRAPPA with a mean absolute error of 5.78 K and a mean relative error of 1.32 % on their test set, which is comparable in its size (385 components (theirs) vs. 367 components (ours)). These results prove that GRAPPA is also a powerful tool for predicting (normal) boiling temperatures, even though it has not been explicitly trained for this task.

Examples for Predicted Vapor Pressure Curves and Evaluation of Attention Scores

GRAPPA profits from the combination of its GNN architecture with the attention-based pooling function, which allows it to capture the effects of complex interactions in multi-functional components. In Fig. 6, we show vapor pressure curves for three such components as predicted with GRAPPA compared to experimental test data. The results show GRAPPA's

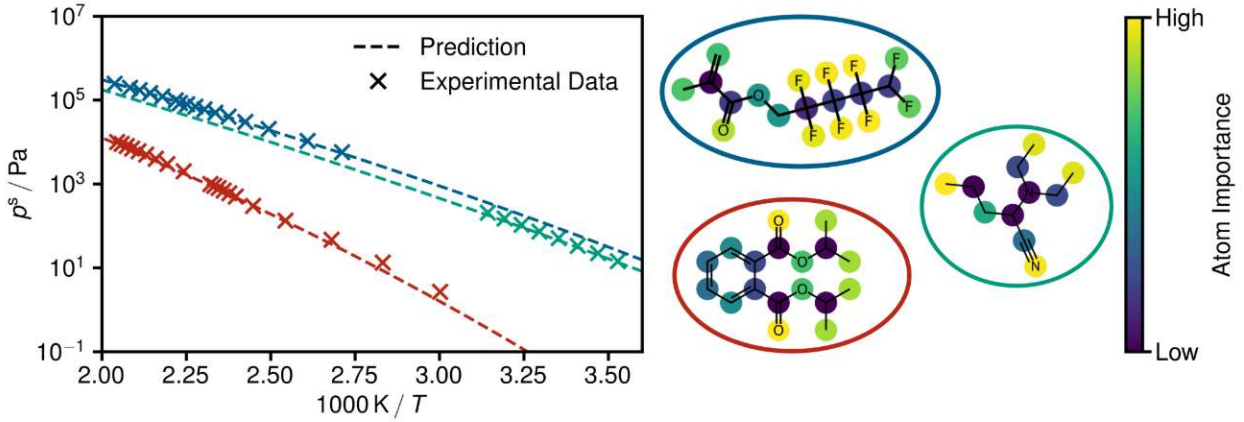

Figure 6: GRAPPA predictions and experimental data for the vapor pressure of three multi-functional components from our test set. The corresponding molecules are depicted on the right-hand side (CAS-RN from top to bottom: 355-93-1, 19340-91-1, 605-45-8). Each atom is colored based on its importance for the GRAPPA predictions, calculated by the mean of its outgoing attention coefficients in the last GAT layer. The scores are normalized for every molecule.

excellent predictive accuracy, even in challenging cases. Next to the plot, the molecules are visualized, and their atoms are colored based on the mean of their outgoing attention

coefficients in the last GAT layer, cf. Eq. (3). The color reflects each atom's importance in vapor pressure prediction. Notably, fluorine atoms, carbonyl groups, and open ends of alkane chains seem to play a prominent role in the prediction, which is in line with chemical intuition. The results show that the predictions of GRAPPA are not only accurate but also interpretable.

Conclusions

This work introduces GRAPPA - a broadly applicable and fully disclosed machine learning model for the prediction of pure component vapor pressures. GRAPPA is based on a graph neural network architecture and requires only the molecular structure as input. We have trained GRAPPA on more than 200,000 experimental data points of more than 20,000 pure components and evaluated it on a test set of unseen components, showing high prediction accuracy. GRAPPA predicts component-specific parameters of the Antoine equation, making the implementation in industrial applications, e.g., in process simulators, straightforward. Furthermore, GRAPPA can be used directly to predict boiling temperatures at a given pressure.

Despite the broad applicability of GRAPPA, the following limitations must be considered. As shown in the results, the prediction accuracy declines for pressures below 1 kPa, which may be due to the inability of the Antoine equation to describe the entire vapor pressure curve and the decline of measurement accuracy for low pressures. Therefore, GRAPPA should be used with care in such scenarios, e.g., for very large molecules. The chemical space in which GRAPPA can be used is limited to components that fulfill the criteria in our pre-processing. However, this covers most components of technical relevance in the chemical industry.

We have shown that GRAPPA performs significantly better than group contribution methods, which are the current state of the art for predicting vapor pressures or boiling

temperatures in industry. Furthermore, a direct comparison was made between GRAPPA and another GNN based prediction method. The results demonstrate GRAPPA's higher prediction accuracy despite its reduced complexity, both regarding the underlying equation describing the vapor pressure (Antoine equation vs. Wagner equation) and the number of parameters. Future modifications of GRAPPA might use information on the critical point and related properties (e.g., the enthalpy of vaporization) in the training to exploit thermodynamic relations. Additionally, the uncertainties in both the experimental data and the predictions could be taken into account to enhance confidence in the model.

Model Availability

Our github repository `github.com/marco-hoffmann/GRAPPA` contains the final model trained on all data. The repository includes an example code that explains how to calculate Antoine parameters, vapor pressures, and (normal) boiling temperatures with GRAPPA. Moreover, we have published an online prediction tool via our website `ml-prop.mv.rptu.de`, where the user can obtain predictions from GRAPPA without having to download or install the required code.

Acknowledgement

We gratefully acknowledge financial support by the Carl Zeiss Foundation in the frame of the project "Process Engineering 4.0" and by DFG in the frame of the Priority Program SPP2363 "Molecular Machine Learning" (grant number 497201843). Furthermore, FJ gratefully acknowledges financial support by DFG in the frame of the Emmy-Noether program (grant number 528649696). We would like to thank Jürgen Rarey for fruitful discussions regarding this work.

Supporting Information Available

We provide a Supporting Information (SI), which provides details on hyperparameters of the model and the grid search. Additionally, the SI includes a more detailed analysis of the prediction accuracy of GRAPPA.

References

(1) Dortmund Data Bank. www.ddbst.com, 2024.

(2) Yaws, C. L. *The Yaws handbook of vapor pressure*, second edition ed.; Gulf Professional Publishing is an imprint of Elsevier: Kidlington, Oxford, 2015.

(3) Lee, B. I.; Kesler, M. G. A generalized thermodynamic correlation based on three-parameter corresponding states. *AIChE Journal* **1975**, *21*, 510–527.

(4) Ambrose, D.; Walton, J. Vapour pressures up to their critical temperatures of normal alkanes and 1-alkanols. *Pure and Applied Chemistry* **1989**, *61*, 1395–1403.

(5) Riedel, L. Eine neue universelle Dampfdruckformel Untersuchungen über eine Erweiterung des Theorems der übereinstimmenden Zustände. Teil I. *Chemie Ingenieur Technik* **1954**, *26*, 83–89.

(6) Macknick, A. B.; Prausnitz, J. M. Vapor Pressures of Heavy Liquid Hydrocarbons by a Group-Contribution Method. *Industrial & Engineering Chemistry Fundamentals* **1979**, *18*, 348–351.

(7) Edwards, D. R.; Prausnitz, J. M. Estimation of vapor pressures of heavy liquid hydrocarbons containing nitrogen or sulfur by a group-contribution method. *Industrial & Engineering Chemistry Fundamentals* **1981**, *20*, 280–283.

(8) Burkhard, L. P. Estimation of vapor pressures for halogenated aromatic hydrocarbons by a group-contribution method. *Industrial & Engineering Chemistry Fundamentals* **1985**, *24*, 119–120.

(9) Tu, C.-H. Group-contribution method for the estimation of vapor pressures. *Fluid Phase Equilibria* **1994**, *99*, 105–120.

(10) Sawaya, T.; Mokbel, I.; Rauzy, E.; Saab, J.; Berro, C.; Jose, J. Experimental vapor pressures of alkyl and aryl sulfides - Prediction by a group contribution method. *Fluid Phase Equilibria* **2004**, *226*, 283–288.

(11) Asher, W. E.; Pankow, J. F. Vapor pressure prediction for alkenoic and aromatic organic compounds by a UNIFAC-based group contribution method. *Atmospheric Environment* **2006**, *40*, 3588–3600.

(12) Pankow, J. F.; Asher, W. E. SIMPOL.1: a simple group contribution method for predicting vapor pressures and enthalpies of vaporization of multifunctional organic compounds. *Atmospheric Chemistry and Physics* **2008**, *8*, 2773–2796.

(13) Nannoolal, Y.; Rarey, J.; Ramjugernath, D. Estimation of pure component properties: Part 3. Estimation of the vapor pressure of non-electrolyte organic compounds via group contributions and group interactions. *Fluid Phase Equilibria* **2008**, *269*, 117–133.

(14) Moller, B.; Rarey, J.; Ramjugernath, D. Estimation of the vapour pressure of non-electrolyte organic compounds via group contributions and group interactions. *Journal of Molecular Liquids* **2008**, *143*, 52–63.

(15) Ceriani, R.; Gani, R.; Liu, Y. Prediction of vapor pressure and heats of vaporization of edible oil/fat compounds by group contribution. *Fluid Phase Equilibria* **2013**, *337*, 53–59.

(16) Rezakazemi, M.; Marjani, A.; Shirazian, S. Development of a Group Contribution Method Based on UNIFAC Groups for the Estimation of Vapor Pressures of Pure Hydrocarbon Compounds. *Chemical Engineering & Technology* **2013**, *36*, 483–491.

(17) Wang, T.-Y.; Meng, X.-Z.; Jia, M.; Song, X.-C. Predicting the vapor pressure of fatty acid esters in biodiesel by group contribution method. *Fuel Processing Technology* **2015**, *131*, 223–229.

(18) Beck, B.; Breindl, A.; Clark, T. QM/NN QSPR Models with Error Estimation: Vapor Pressure and LogP. *Journal of Chemical Information and Computer Sciences* **2000**, *40*, 1046–1051.

(19) Gharagheizi, F.; Eslamimanesh, A.; Ilani-Kashkouli, P.; Mohammadi, A. H.; Richon, D. QSPR molecular approach for representation/prediction of very large vapor pressure dataset. *Chemical Engineering Science* **2012**, *76*, 99–107.

(20) Katritzky, A. R.; Slavov, S. H.; Dobchev, D. A.; Karelson, M. Rapid QSPR model development technique for prediction of vapor pressure of organic compounds. *Computers & Chemical Engineering* **2007**, *31*, 1123–1130.

(21) Katritzky, A. R.; Lobanov, V. S.; Karelson, M. QSPR: the correlation and quantitative prediction of chemical and physical properties from structure. *Chemical Society Reviews* **1995**, *24*, 279.

(22) Hsieh, C.; Lin, S. Determination of cubic equation of state parameters for pure fluids from first principle solvation calculations. *AIChE Journal* **2008**, *54*, 2174–2181.

(23) Wang, L.-H.; Hsieh, C.-M.; Lin, S.-T. Improved Prediction of Vapor Pressure for Pure Liquids and Solids from the PR+COSMOSAC Equation of State. *Industrial & Engineering Chemistry Research* **2015**, *54*, 10115–10125.

(24) Liang, H.-H.; Li, J.-Y.; Wang, L.-H.; Lin, S.-T.; Hsieh, C.-M. Improvement to PR+COSMOSAC EOS for Predicting the Vapor Pressure of Nonelectrolyte Organic Solids and Liquids. *Industrial & Engineering Chemistry Research* **2019**, *58*, 5030–5040.

(25) Habicht, J.; Brandenbusch, C.; Sadowski, G. Predicting PC-SAFT pure-component parameters by machine learning using a molecular fingerprint as key input. *Fluid Phase Equilibria* **2023**, *565*, 113657.

(26) Felton, K. C.; Raßpe-Lange, L.; Rittig, J. G.; Leonhard, K.; Mitsos, A.; Meyer-Kirschner, J.; Knösche, C.; Lapkin, A. A. ML-SAFT: A machine learning framework for PCP-SAFT parameter prediction. *Chemical Engineering Journal* **2024**, *492*, 151999.

(27) Winter, B.; Rehner, P.; Esper, T.; Schilling, J.; Bardow, A. Understanding the language of molecules: Predicting pure component parameters for the PC-SAFT equation of state from SMILES. arXiv preprint, 2023; 2309.12404.

(28) Klamt, A.; Eckert, F. COSMO-RS: a novel and efficient method for the a priori prediction of thermophysical data of liquids. *Fluid Phase Equilibria* **2000**, *172*, 43–72.

(29) Bell, I. H.; Mickoleit, E.; Hsieh, C.-M.; Lin, S.-T.; Vrabec, J.; Breitkopf, C.; Jäger, A. A Benchmark Open-Source Implementation of COSMO-SAC. *Journal of Chemical Theory and Computation* **2020**, *16*, 2635–2646.

(30) Santana, V. V.; Rebello, C. M.; Queiroz, L. P.; Ribeiro, A. M.; Shardt, N.; Nogueira, I. B. PUFFIN: A path-unifying feed-forward interfaced network for vapor pressure prediction. *Chemical Engineering Science* **2024**, *286*, 119623.

(31) Lin, Y.-H.; Liang, H.-H.; Lin, S.-T.; Li, Y.-P. Advancing vapor pressure prediction: A machine learning approach with directed message passing neural networks. *Journal of the Taiwan Institute of Chemical Engineers* **2024**, 105926.

(32) Sanchez Medina, E. I.; Linke, S.; Stoll, M.; Sundmacher, K. Graph neural networks for the prediction of infinite dilution activity coefficients. *Digital Discovery* **2022**, *1*, 216–225.

(33) Sanchez Medina, E. I.; Linke, S.; Stoll, M.; Sundmacher, K. Gibbs–Helmholtz graph neural network: capturing the temperature dependency of activity coefficients at infinite dilution. *Digital Discovery* **2023**, *2*, 781–798.

(34) Ahmad, W.; Tayara, H.; Chong, K. T. Attention-Based Graph Neural Network for Molecular Solubility Prediction. *ACS Omega* **2023**, *8*, 3236–3244.

(35) Qu, C.; Kearsley, A. J.; Schneider, B. I.; Keyrouz, W.; Allison, T. C. Graph convolutional neural network applied to the prediction of normal boiling point. *Journal of Molecular Graphics and Modelling* **2022**, *112*, 108149.

(36) Aouichaoui, A. R.; Cogliati, A.; Abildskov, J.; Sin, G. *33rd European Symposium on Computer Aided Process Engineering*; Elsevier, 2023; pp 575–581.

(37) Aouichaoui, A. R. N.; Fan, F.; Mansouri, S. S.; Abildskov, J.; Sin, G. Combining Group-Contribution Concept and Graph Neural Networks Toward Interpretable Molecular Property Models. *Journal of Chemical Information and Modeling* **2023**, *63*, 725–744.

(38) Aouichaoui, A. R.; Fan, F.; Abildskov, J.; Sin, G. Application of interpretable group-embedded graph neural networks for pure compound properties. *Computers & Chemical Engineering* **2023**, *176*, 108291.

(39) Hayer, N.; Hasse, H.; Jirasek, F. Prediction of Temperature-Dependent Henry's Law Constants by Matrix Completion. *The Journal of Physical Chemistry B* **2024**, *129*, 409–416.

(40) Hayer, N.; Wendel, T.; Mandt, S.; Hasse, H.; Jirasek, F. Advancing thermodynamic

group-contribution methods by machine learning: UNIFAC 2.0. *Chemical Engineering Journal* **2025**, *504*, 158667.

(41) Specht, T.; Nagda, M.; Fellenz, S.; Mandt, S.; Hasse, H.; Jirasek, F. HANNA: hard-constraint neural network for consistent activity coefficient prediction. *Chemical Science* **2024**, *15*, 19777–19786.

(42) Damay, J.; Jirasek, F.; Kloft, M.; Bortz, M.; Hasse, H. Predicting Activity Coefficients at Infinite Dilution for Varying Temperatures by Matrix Completion. *Industrial & Engineering Chemistry Research* **2021**, *60*, 14564–14578.

(43) Rittig, J. G.; Felton, K. C.; Lapkin, A. A.; Mitsos, A. Gibbs-Duhem-Informed Neural Networks for Binary Activity Coefficient Prediction. *Digital Discovery* **2023**, *2*, 1752–1767.

(44) Rittig, J. G.; Ben Hicham, K.; Schweidtmann, A. M.; Dahmen, M.; Mitsos, A. Graph neural networks for temperature-dependent activity coefficient prediction of solutes in ionic liquids. *Computers & Chemical Engineering* **2023**, *171*, 108153.

(45) Jirasek, F.; Hasse, H. Combining Machine Learning with Physical Knowledge in Thermodynamic Modeling of Fluid Mixtures. *Annual Review of Chemical and Biomolecular Engineering* **2023**, *14*, 31–51.

(46) Rogers, D.; Hahn, M. Extended-Connectivity Fingerprints. *Journal of Chemical Information and Modeling* **2010**, *50*, 742–754.

(47) Gross, J.; Sadowski, G. Perturbed-Chain SAFT: An Equation of State Based on a Perturbation Theory for Chain Molecules. *Industrial & Engineering Chemistry Research* **2001**, *40*, 1244–1260.

(48) Gross, J.; Vrabec, J. An equation-of-state contribution for polar components: Dipolar molecules. *AIChE Journal* **2005**, *52*, 1194–1204.

(49) Weininger, D. SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules. *Journal of Chemical Information and Computer Sciences* **1988**, *28*, 31–36.

(50) Duvenaud, D.; Maclaurin, D.; Aguilera-Iparraguirre, J.; Gómez-Bombarelli, R.; Hirzel, T.; Aspuru-Guzik, A.; Adams, R. P. Convolutional Networks on Graphs for Learning Molecular Fingerprints. arXiv preprint, 2015; 1509.09292.

(51) 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. *Journal of Chemical Information and Modeling* **2019**, *59*, 3370–3388.

(52) Wang, G.; Hu, P. Prediction of normal boiling point and critical temperature of refrigerants by graph neural network and transfer learning. *International Journal of Refrigeration* **2023**, *151*, 97–104.

(53) Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Liò, P.; Bengio, Y. Graph Attention Networks. arXiv preprint, 2018; 1710.10903.

(54) RDKit: Open-source cheminformatics. `http://www.rdkit.org`, Version: 2023.03.1.

(55) Brody, S.; Alon, U.; Yahav, E. How Attentive are Graph Attention Networks? International Conference on Learning Representations. 2022.

(56) Fey, M.; Lenssen, J. E. Fast Graph Representation Learning with PyTorch Geometric. 2019; `https://github.com/pyg-team/pytorch_geometric`.

(57) Buterez, D.; Janet, J. P.; Kiddle, S. J.; Oglic, D.; Liò, P. Modelling local and general quantum mechanical properties with attention-based pooling. *Communications Chemistry* **2023**, *6*.

(58) Baek, J.; Kang, M.; Hwang, S. J. Accurate Learning of Graph Representations with Graph Multiset Pooling. arXiv preprint, 2021; 2102.11533.

(59) Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; Polosukhin, I. Attention Is All You Need. arXiv preprint, 2017; 1706.03762.

(60) Paszke, A. et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library. arXiv preprint, 2019; 1912.01703.

(61) Loshchilov, I.; Hutter, F. Decoupled Weight Decay Regularization. arXiv preprint, 2017; 1711.05101.

(62) Smith, L. N.; Topin, N. Super-Convergence: Very Fast Training of Neural Networks Using Large Learning Rates. arXiv preprint, 2017; 1708.07120.

(63) Nannoolal, Y.; Rarey, J.; Ramjugernath, D.; Cordes, W. Estimation of pure component properties: Part 1. Estimation of the normal boiling point of non-electrolyte organic compounds via group contributions and group interactions. *Fluid Phase Equilibria* **2004**, *226*, 45–63.

Supporting Information for GRAPPA - A Hybrid Graph Neural Network for Predicting Pure Component Vapor Pressures

Marco Hoffmann, Hans Hasse, and Fabian Jirasek*

Laboratory of Engineering Thermodynamics, RPTU Kaiserslautern, Erwin-Schrödinger-Str. 44, 67663 Kaiserslautern, Germany

E-mail: fabian.jirasek@rptu.de

Details on Model Optimization

Tab. S1 lists the hyperparameters of the GRAPPA architecture and the training process. The hyperparameters for which sets are specified were systematically varied during the grid search, and the bold values indicate the final choices selected based on the validation scores. The remaining parameters were not varied but set as specified.

Detailed Results of the Vapor Pressure Prediction

The left panel of Fig. S1 shows a parity plot of the vapor pressure predictions of GRAPPA over the experimental data. In the right panel of Fig. S1, the histogram shows the number of components predicted with a certain APE_C with GRAPPA and the results of the method by Lin et al.[1] for comparison. In Fig. S2, parity plots for the vapor pressure predictions of the four studied literature methods over the experimental data are shown. The pressure and

Table S1: Overview of the hyperparameters of GRAPPA. Hyperparameters for which sets are specified were varied during the grid search, whereby the bold values indicate the final choices selected based on the validation scores.

Hyperparameter	**Values**
Number of GNN layers	{2, 3, **4**, 5}
Number of GNN heads	{1, **2**, 3, 4, 5}
GNN convolutional dimension	32
Number of hidden layers	{1, 2, **3**}
Neurons in the hidden layers	16
Pooling method	{*sum*, ***interaction***}
OneCycleLR `max_lr`	0.001
ReduceLROnPlateau `factor`	0.5
ReduceLROnPlateau `patience`	5
Batch size	512
Huber loss `delta`	0.5
Number of epochs in warm-up training	100
Number of epochs in main training	100

temperature dependence of the prediction accuracy is further investigated in the boxplots in Fig. S3 and Fig. S4. The relationship between molecular weight and prediction accuracy is visualized in the boxplot in Fig. S5. Fig. S6 shows how the prediction accuracy correlates with the number of experimental data points for a specific component in the test set.

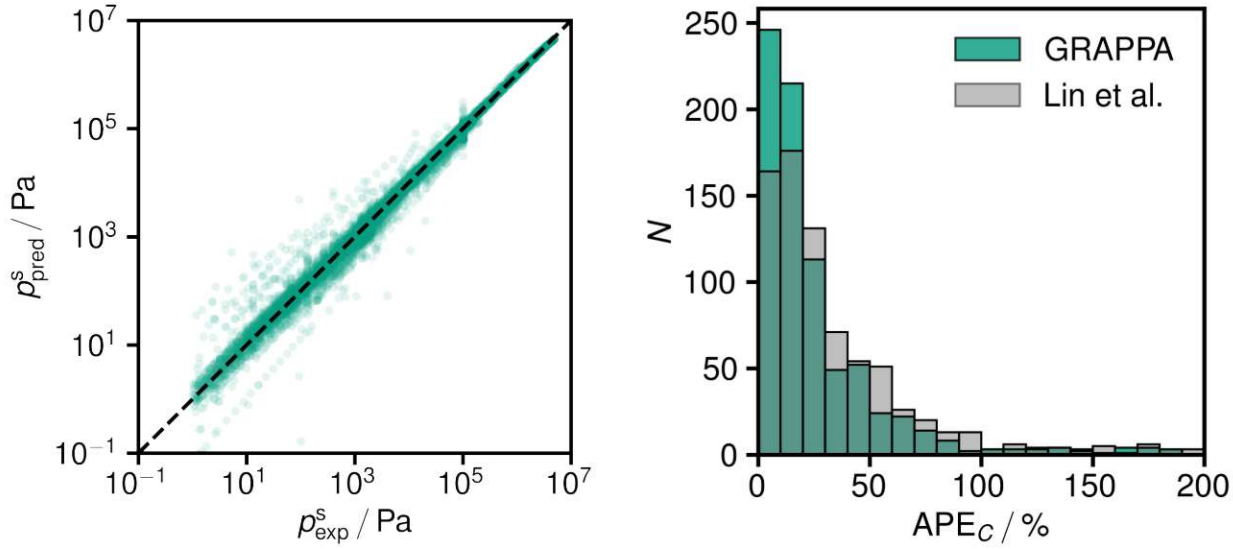

Figure S1: Left: Parity plot showing the predicted vapor pressure for our test set over the experimental values for GRAPPA. The dashed line marks perfect predictions. Right: Histogram showing the number of components over the APE_C for GRAPPA and the method by Lin et al.[1] Only results for components with at least two data points in the test set are shown in both plots. The interval displayed in the histogram covers 96.3 % and 94.5 % of the considered components for GRAPPA and the method by Lin et al., respectively.

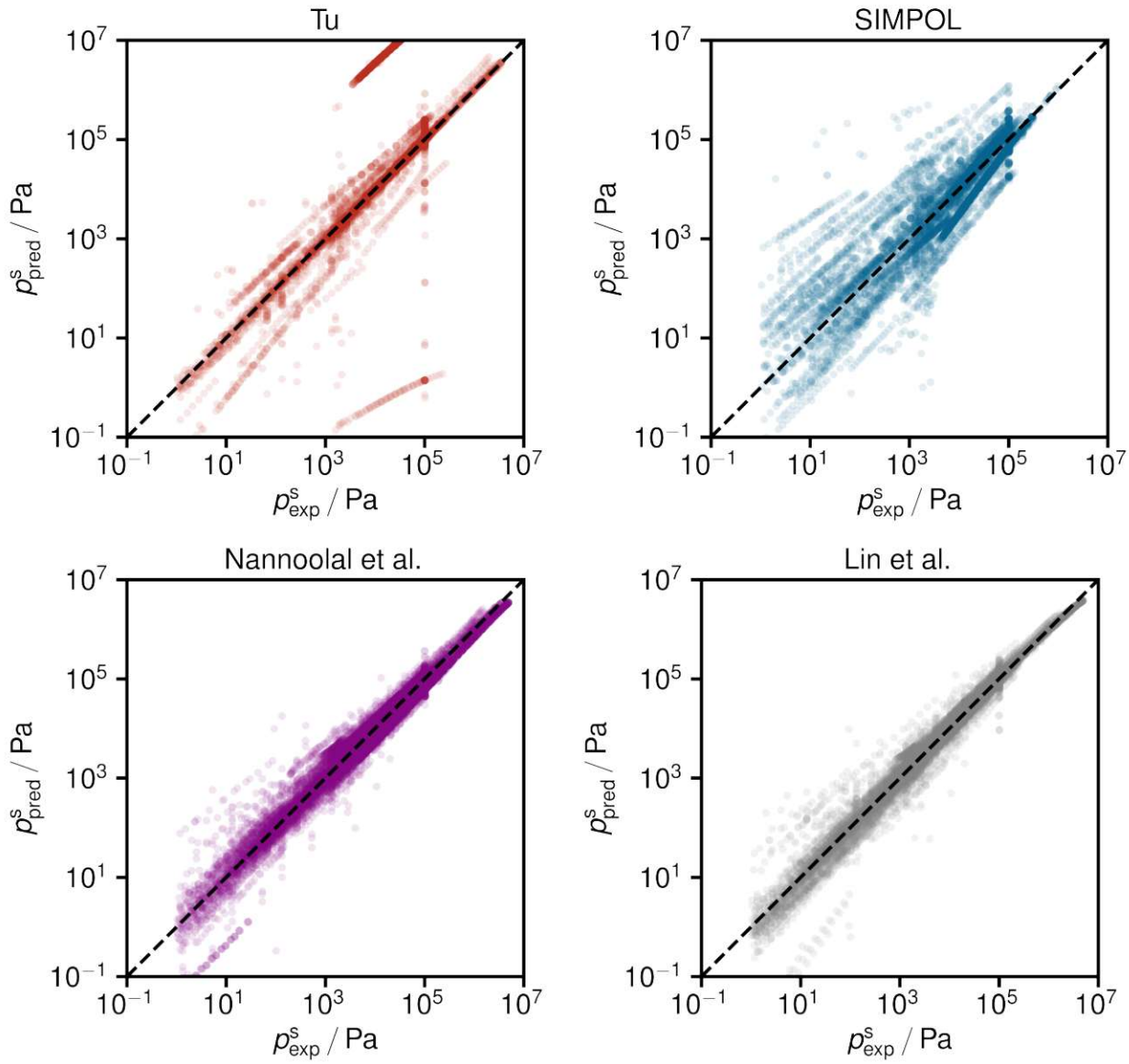

Figure S2: Parity plots showing the predicted vapor pressure for our test set over the experimental values for the method by Tu,[2] SIMPOL,[3] the method by Nannoolal et al.,[4] and the method by Lin et al.[1] Only results for components with at least two data points in the test set are shown. The dashed lines mark perfect predictions. Because of limitations of the literature models, predictions were only feasible for 36.1 % (Tu), 53.8 % (SIMPOL), and 94.9 % (Nannoolal et al.) of the components. The method by Lin et al.[1] is applicable to our entire test set.

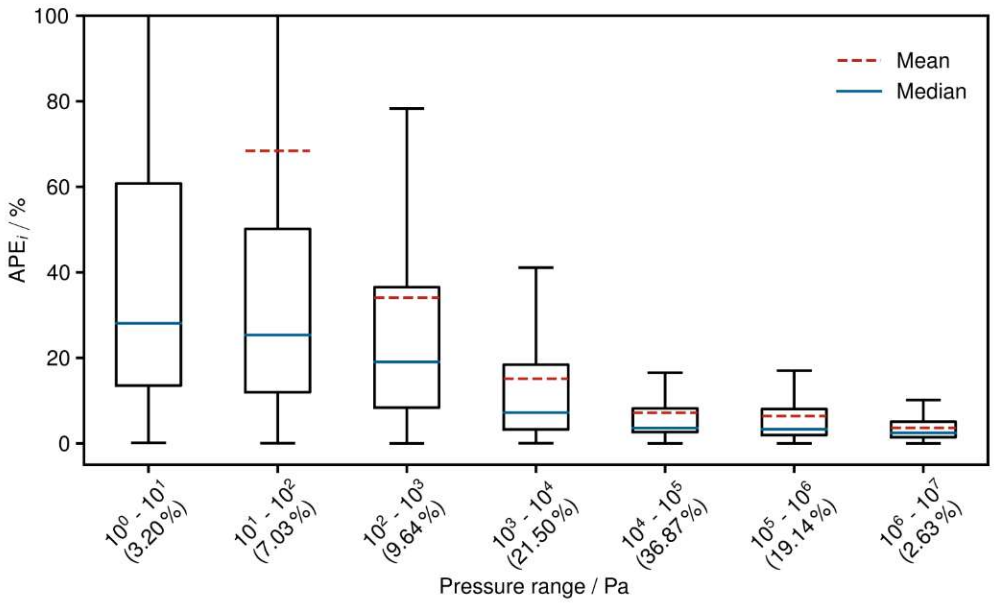

Figure S3: Boxplot visualizing the prediction accuracy of the developed GRAPPA model in terms of APE_i for different pressure intervals on the test set for components with at least two data points. The boxes represent the interquartile range, and the whiskers are 1.5 times the interquartile range. The numbers in the brackets denote the percentage of data points falling into the respective pressure range.

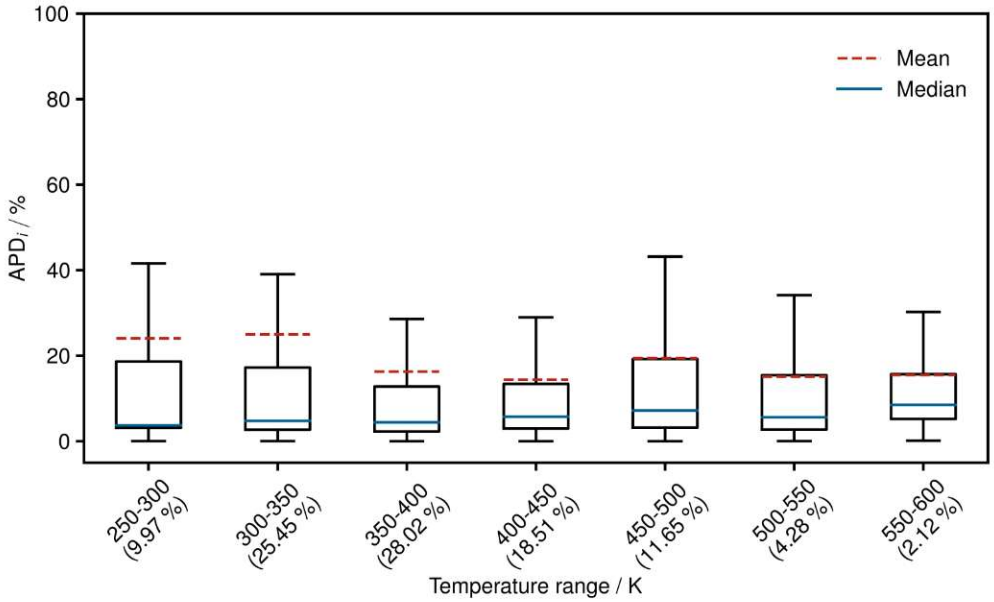

Figure S4: Boxplot visualizing the prediction accuracy of the developed GRAPPA model in terms of APE_i for different temperature intervals on the test set for components with at least two data points. The boxes represent the interquartile range, and the whiskers are 1.5 times the interquartile range. The numbers in the brackets denote the percentage of data points falling into the respective temperature range.

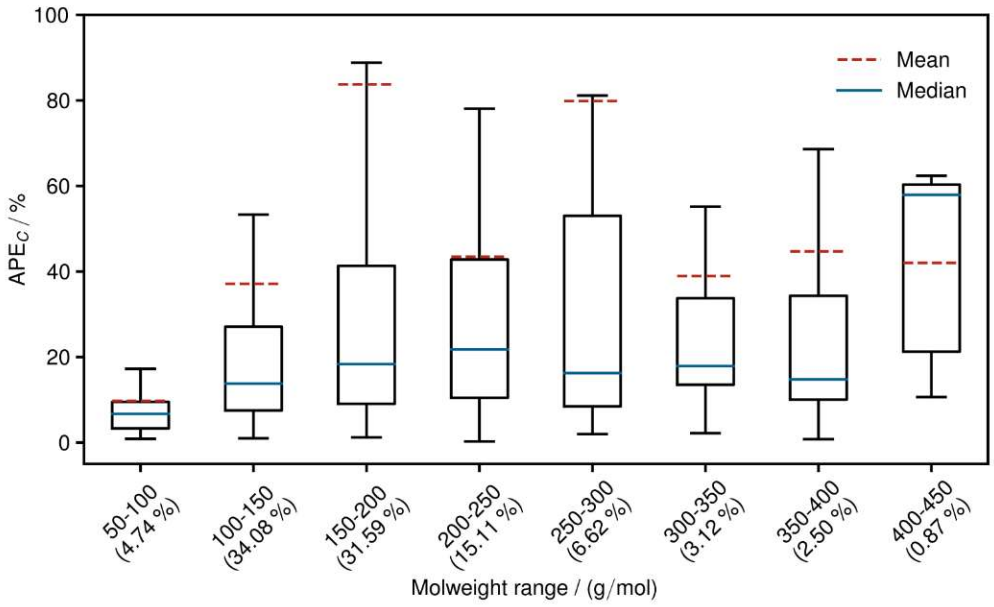

Figure S5: Boxplot visualizing the prediction accuracy of the developed GRAPPA model in terms of APE_C for different mol weight intervals on the test set for components with at least two data points. The boxes represent the interquartile range, and the whiskers are 1.5 times the interquartile range. The numbers in the brackets denote the percentage of data points falling into the respective mol weight range.

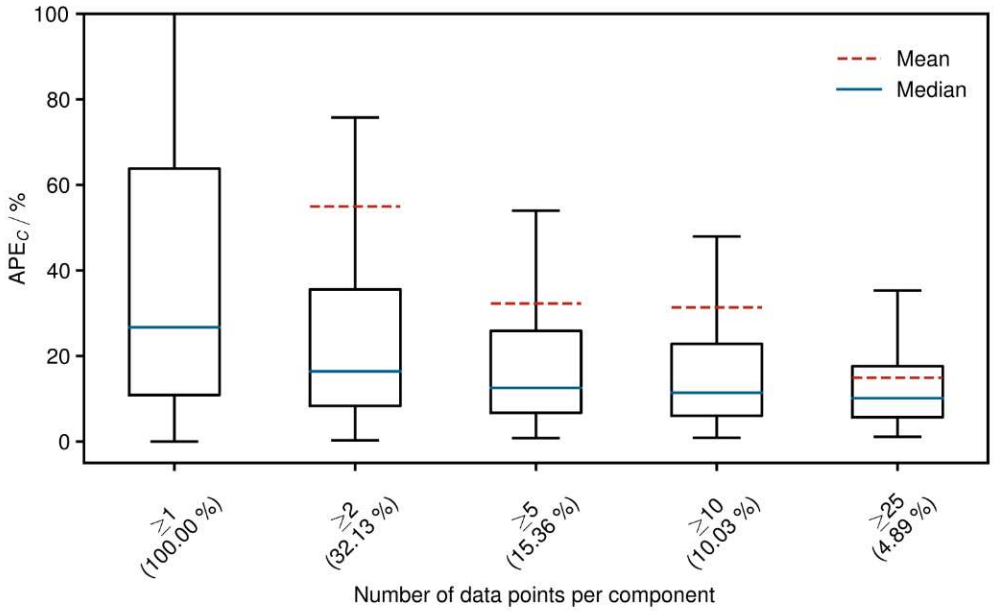

Figure S6: Boxplot visualizing the prediction accuracy of the developed GRAPPA model in terms of APE_C for different minimum numbers of data points per component in the test set. The boxes represent the interquartile range, and the whiskers are 1.5 times the interquartile range. The numbers in the brackets denote the percentage of components for which the minimum amount of data points are available.

References

(1) Lin, Y.-H.; Liang, H.-H.; Lin, S.-T.; Li, Y.-P. Advancing vapor pressure prediction: A machine learning approach with directed message passing neural networks. *Journal of the Taiwan Institute of Chemical Engineers* **2024**, 105926.

(2) Tu, C.-H. Group-contribution method for the estimation of vapor pressures. *Fluid Phase Equilibria* **1994**, *99*, 105–120.

(3) Pankow, J. F.; Asher, W. E. SIMPOL.1: a simple group contribution method for predicting vapor pressures and enthalpies of vaporization of multifunctional organic compounds. *Atmospheric Chemistry and Physics* **2008**, *8*, 2773–2796.

(4) Nannoolal, Y.; Rarey, J.; Ramjugernath, D. Estimation of pure component properties: Part 3. Estimation of the vapor pressure of non-electrolyte organic compounds via group contributions and group interactions. *Fluid Phase Equilibria* **2008**, *269*, 117–133.