ENERGETIC MATERIALS AND MACHINE LEARNING. REQUIREMENTS AND APPLICATION

Moritz Heil, Jan Langer
Fraunhofer ICT, Joseph-von-Fraunhofer-Str. 7, 76327 Pfinztal, Germany

Abstract

Machine Learning is a rapidly developing field, where data availability and data structure are key features that determine the quality and performance of models, which can be used to predict and optimize a variety of variables. In the case of energetic materials, these predictable variables can be material properties, for example. In the last few years, different methods and applications have been published about the use of machine learning to predict different properties of energetic materials, such as enthalpy of formation, sensitivity, or specific impulse.
In this work, the use case of machine learning to predict the enthalpy of formation based on the ICT Thermodynamics database is shown. Important aspects such as data fusion from various sources, dealing with substantial amounts of data and anomaly detection are discussed from the scientist's (i.e., the chemist's) view. Special attention is paid to the interface between the world of computer sciences and chemistry and how these two can and must interact with each other to obtain good and reliable results. Typical problems and basic knowledge to assess literature in this field will be presented.

Introduction

Development and synthesis of new energetic molecules is an ongoing task for researchers all over the world. Different key values (e.g., performance, sensitivity, toxicity and thermal stability) must be fulfilled or –even better – improved to create a promising new energetic material. Some of these quantities can be calculated in advance using thermodynamic codes. However, these codes rely on material properties such as density and heat of formation which can be calculated by quantum mechanical methods. These quantum mechanical methods are

expensive and time-consuming. This makes it difficult to use the combination of quantum mechanical and thermodynamic calculation as a screening tool for new energetic materials. The interest in using machine learning methods to calculate these data for new hypothetic molecules is large as it saves much time and allows synthesis to focus on potentially interesting molecules (*1*). Machine learning does not need explicitly known mathematical laws to calculate properties but can derive relationships on its own which makes it ideal for this task. Consequently, the so-called field of „quantitative structure property relation“ is developing rapidly (*2–4*). In this work we use the ICT Thermodynamics database to apply machine learning algorithms to predict the heat of formation of molecules by using the molecular structure as variable. Heat of formation has already been discussed in literature and the influence of different algorithms and variables on the outcome has been investigated (*5*). Elton et al. used a publicly available dataset with 109 substances as data for their work. The ICT Thermodynamics database is a much larger data set containing over 10000 substances. To make the machine learning approach to this topic easier to understand for non-computer scientists (i.e., mainly chemists in this field of work), we will also cover some of the more trivial aspects to create an understanding of the problems each side is facing.

Data acquisition and cleaning

The main resource of machine learning is data. The amount and quality of data determine the performance of the resulting models and are therefore the limiting factors. Large amounts of data are necessary to use the more sophisticated machine learning techniques. Large amounts of data also mean that no manual inspection or correction is possible due to the sheer amount of work that this step would require. Consequently, summary analysis and restriction to representative data is preferable over including all data points at the final cost of decreasing model performance.

The ICT Thermodynamics database is the basis of the ICT Thermodynamics Code and has been grown over the years to include energetic materials and non-energetic materials that are used as additives in energetic formulation. It is a Microsoft Access database with a total of 10498 entries. For each of these entries there are one or more identifiers such as a name (in German and/or English) or CAS (Chemical Abstracts Service) number, as well as a value for the heat of formation and density. By means of SQL (structured query language) queries, the content of the database was converted into a *.RData format, which enables a fast and

efficient further processing of the data using the programming language R. The existing SQL database's structure with nested tables is mapped by nested data frames (tibbles).
Before proceeding to model building with this extended data set, the existing data must be sifted and pre-processed. This prevents problems in the further processing of the data, as well as errors or unnecessary inaccuracies in the later model. Various questions must be answered, and the data adjusted accordingly. The following is a selection of questions that arose in our case:

- Which variables are relevant, which are not? Are there clear redundancies in the variables and how are they dealt with?
- Do data or the model need clear typing and if so, which data types come into question?
- How to deal with missing entries?
- Are there incorrect values, e.g., due to special characters or different formatting?
- Are there groups of data that are clearly underrepresented and therefore perhaps better discarded?

Regardless of the technology, a crucial step in developing high-quality, meaningful models with machine learning methods is the learning process of a model. There are different approaches, which are selected depending on the available data and the desired output of the model. Roughly simplified, a distinction is made between supervised and unsupervised learning methods. There are further gradations in between and beyond, but we will not further consider them here. In supervised learning, data pairs of input variables and the corresponding output variables are shown to the model during the training process so that the model itself forms the relationship between these variables. Unsupervised learning, on the other hand, is used when output variables matching the input variables are missing in the data set in order to be able to train a model. Unsupervised learning is therefore often used to recognize correlations within the input data set, such as patterns, clusters, or cross-dependencies. For the technical implementation, there are nowadays many freely available frameworks for various programming languages. Overall, however, Python has clearly prevailed here due to its ease of use and the number of additional packages, such as TensorFlow or SciKitLearn. In the scientific environment, the programming language R is also increasingly used, which is characterized by a clean structurability of the data and, if necessary, also allows the direct integration of Python code and its powerful packages.
After importing the data from the ICT Thermodynamics database, the data structure contained two important pieces of information:

1. a substance identifier, such as German or English name or CAS number and
2. the heat of formation for this substance.

The next step to prepare the data for machine learning analysis is transforming the substance identifier into a machine interpretable unique identifier. A common identifier is the so-called SMILES (Simplified molecular input line entry system) code (*6*). This is a system that allows representation of the molecular structure of any molecule by a simple string of characters. However, ICT Thermodynamics database does not contain those SMILES codes. To obtain these, data from other sources must be fused. In this case, the PubChem database (https://pubchem.ncbi.nlm.nih.gov/) (*7*) serves as source for the SMILES codes. With the help of the identifiers, some of which had to be translated into English in the first place, for the substances, the existing entries were subsequently expanded by many additional variables via automated queries (REST) to the public Internet database PubChem. The additional information obtained from PubChem's automated data retrieval also includes a SMILES string for each substance.

Preprocessing

The SMILES codes can be processed further by a variety of means. There are several libraries for different languages available, such as RDKit or CDK which has been used here via the "rcdk" package (*8*). This package converts SMILES codes into Java objects, which can be in turn further analyzed by functions from this package. This way, it is possible to extract bonds or atoms of a molecule or to obtain specific information e.g., if the molecule is aromatic or not.

Of the 10498 original entries in the ICT Thermodynamics database, 5711 were successfully expanded with information from the PubChem database. Entries with an absolute value of the enthalpy of formation greater than 1000 kJ/mol were discarded in order to eliminate strongly underrepresented target values (Figure 1). In the next step, only CHNO molecules were kept, reducing the data set to 3908. From this data, a train-test-split with a ratio of 80:20 has been built. A train-test-split is common practice in machine learning to prevent overfitting a model to your data and to have a realistic estimate of your model performance. After splitting your data into a train and a test set, the model is built only with data from the train set; the test set is out aside until the very end of model training. During the training process the train set again is split up into a train and validation set. The specified model is then built on the train set and its performance is determined by the quality of the prediction of the validation set. This way,

parameter optimization such as activation functions for neural networks, variable weights or hyperparameter tuning can be performed and assessed. After the models are built, the true performance is determined by using the model with the previously unused test set from the start. This means that the true model performance does not rely on previously known data which improves information value.

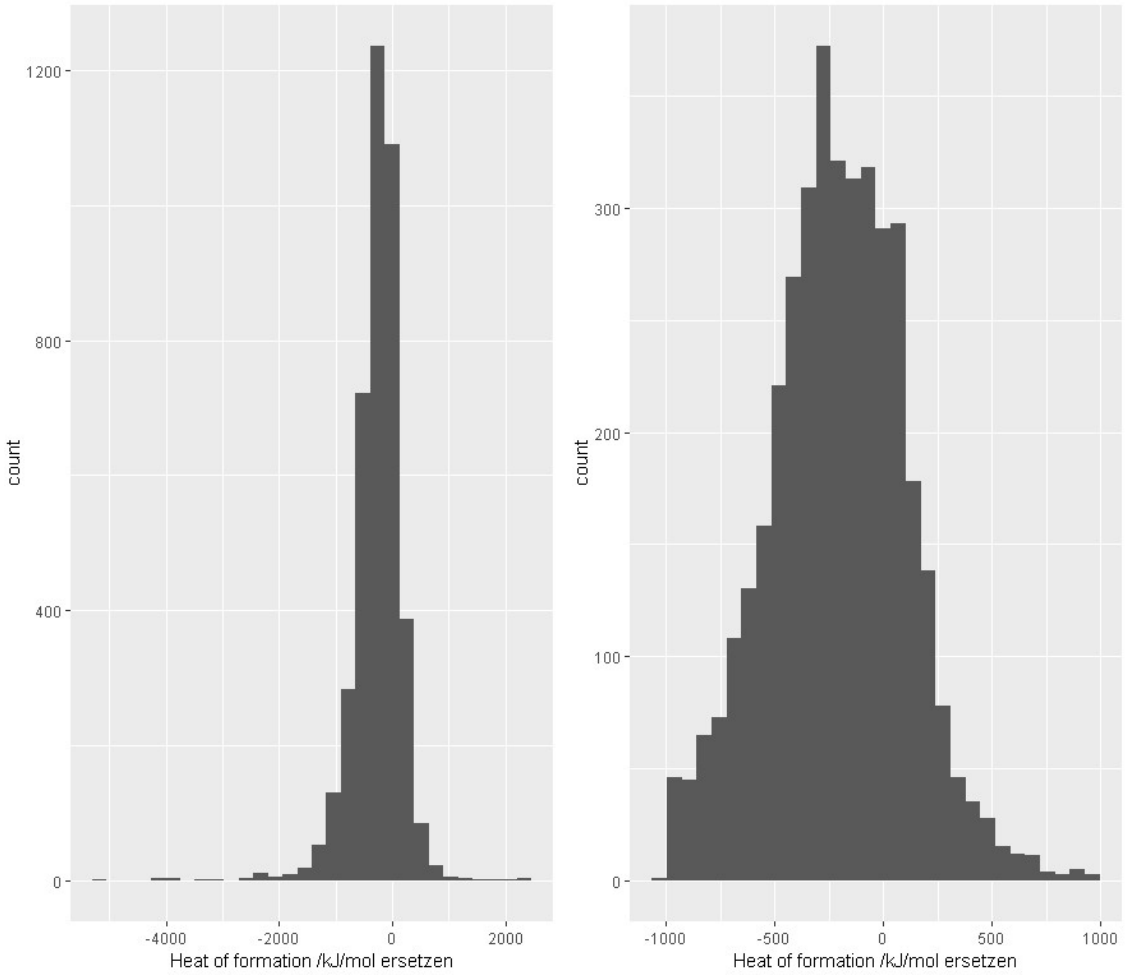

Figure 1: Histograms of data before and after setting the limit to 1000 kJ/mol.

In literature, the bonds have been reported to be a good set of variables for the prediction of heat of formation (*5*). In a first step, the SMILES codes were preprocessed using rcdk to extract the type and amount of different bonds (all permutations of bonds between C, H, N and O atoms with the respective bond order (single, double, triple). Additionally, aromaticity is included as well. This descriptor set (bonds and aromaticity) was used as input for a sequential deep neural network or multi-layer perceptron. It consists of seven densely connected layers, with an input size of 16, decreasing layer sizes for the hidden layers and a final output layer of size one. This neural network (NN) reached a mean absolute error (MAE, i.e., the mean value of the difference between the actual and the predicted heat of formation

over the entire test set) of about 140 kJ/mol. This result was unsatisfactory, and the descriptor set was extended by the number of the respective atoms that are present in the molecule. All further analysis was carried out using this extended dataset.

Results and discussion

This descriptor set has several advantages: It is quite easy to compile from a molecular structure because it could even be done by hand. It can also give some hints about the influence of the different variables and their significance, which makes it easier to design models with specific structural properties.

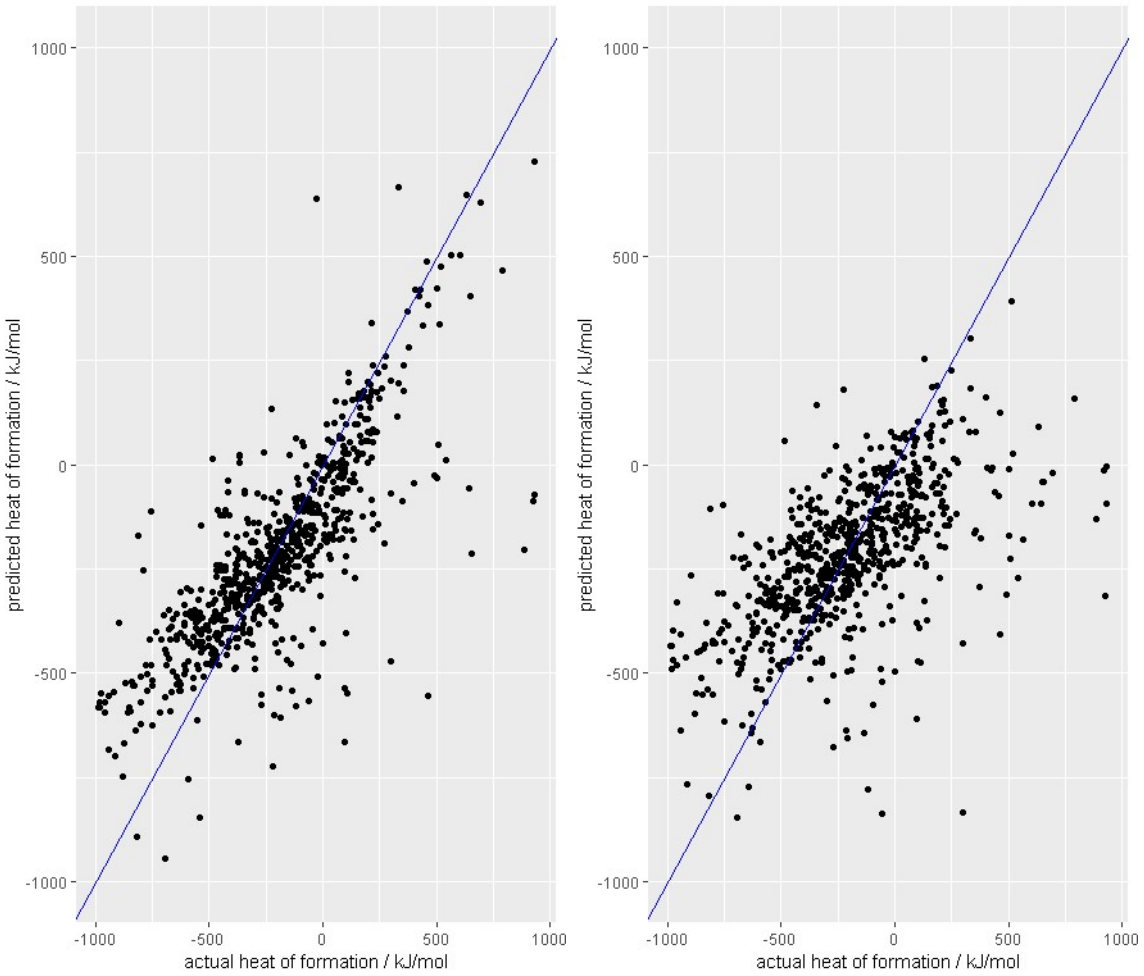

Figure 2: Parity plot of kernel ridge regression (left) and partial least squares regression (right). MAEs are 139 and 179 kJ/mol, respectively.

With this descriptor set, several algorithms were used to predict the heat of formation. For each algorithm, all variables were used without any forward or backward elimination. In this work we used standard linear regression with main affects and first order interaction, kernel ridge regression, partial least squares regression, random forest, and a neural network. Kernel

ridge regression showed the best performance in literature with a mean absolute error of 69 kJ/mol for heat of formation (*5*). **Fehler! Ungültiger Eigenverweis auf Textmarke.** shows the mean absolute error over the test set for the different algorithms. The errors are larger than those in literature, but those models were built on a smaller dataset.

Table 1: Performance of the different models. The performance is calculated using the test set.

Model	**Mean absolute error / kJ/mol**
Kernel Ridge regression	139
Linear regression	137
Linear regression with interaction	112
Neural network	90
Partial least squares regression	179

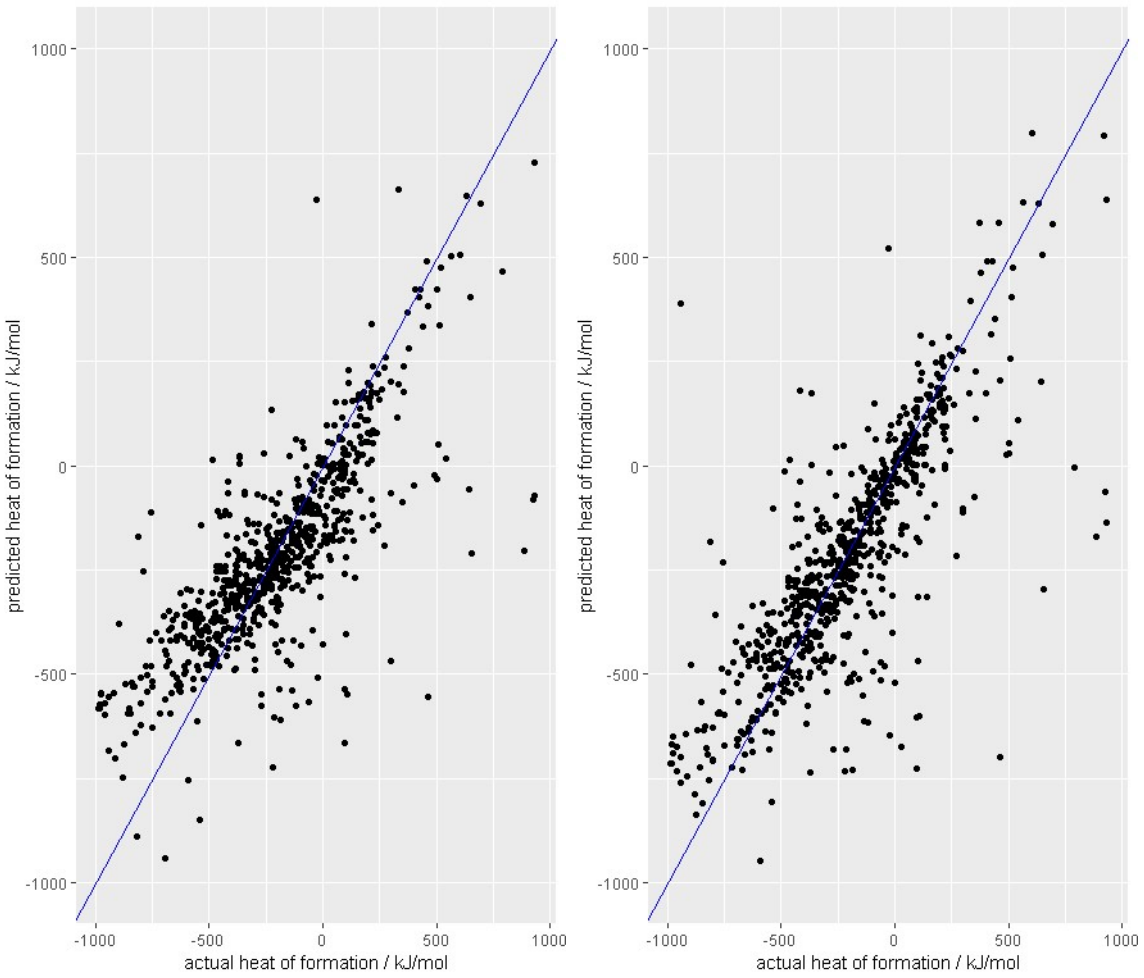

Figure 3: Parity plot of linear regression with main effects(left) and with first order interaction (right). MAEs are 139 and 112 kJ/mol, respectively.

A single metric (here the mean absolute error) is a common way to describe model performance in a very compact way that can be compared to other model performances. If possible, one should always make a residual analysis or a parity plot which plots the predicted values of the test set against the actual values of the test set. The parity plots are shown in Figure 2 to Figure 4. One can see that for the kernel ridge regression, the partial least squares regression and the linear regression with main effects (Figure 2 and Figure 3 on the left) show a tendency to overestimate the heat of formation low values and to underpredict for high values.

The linear regression with first order interaction does not show this behavior, at least not to this extent. Random forest and the NN show a good correlation between actual and predicted values with no skew. This could be because random forest and NN are particularly good at modelling non-linear correlations without explicit description. It is also noteworthy that the addition of the number of atoms to the descriptor set has significantly improved the performance of the model, from 140 kJ/mol to 90 kJ/mol for the NN. Additional enhancement of descriptors such as oxygen balance, which can also easily be calculated from the molecular formula might improve performance even more.

One should also note that no systematic hyperparameter tuning for the random forest (e.g., split rule or the number of variables to possibly split) or the neural network (e.g., permutation of all activation functions or number of hidden layers) has been performed. Hyperparameters are parameters for machine learning methods that are not related to the actual descriptors but to the numerical treatment by those methods during the training process. In other words, the same data with the same variables and outcomes will lead to different (i.e., better or worse) results if the hyperparameters for the model are chosen suitably or not. One problem with hyperparameter is that it is not possible to know in advance. Normally, one would have to establish a test grid of different hyperparameters and then run repeated training cycles and see how the different hyperparameter sets perform.

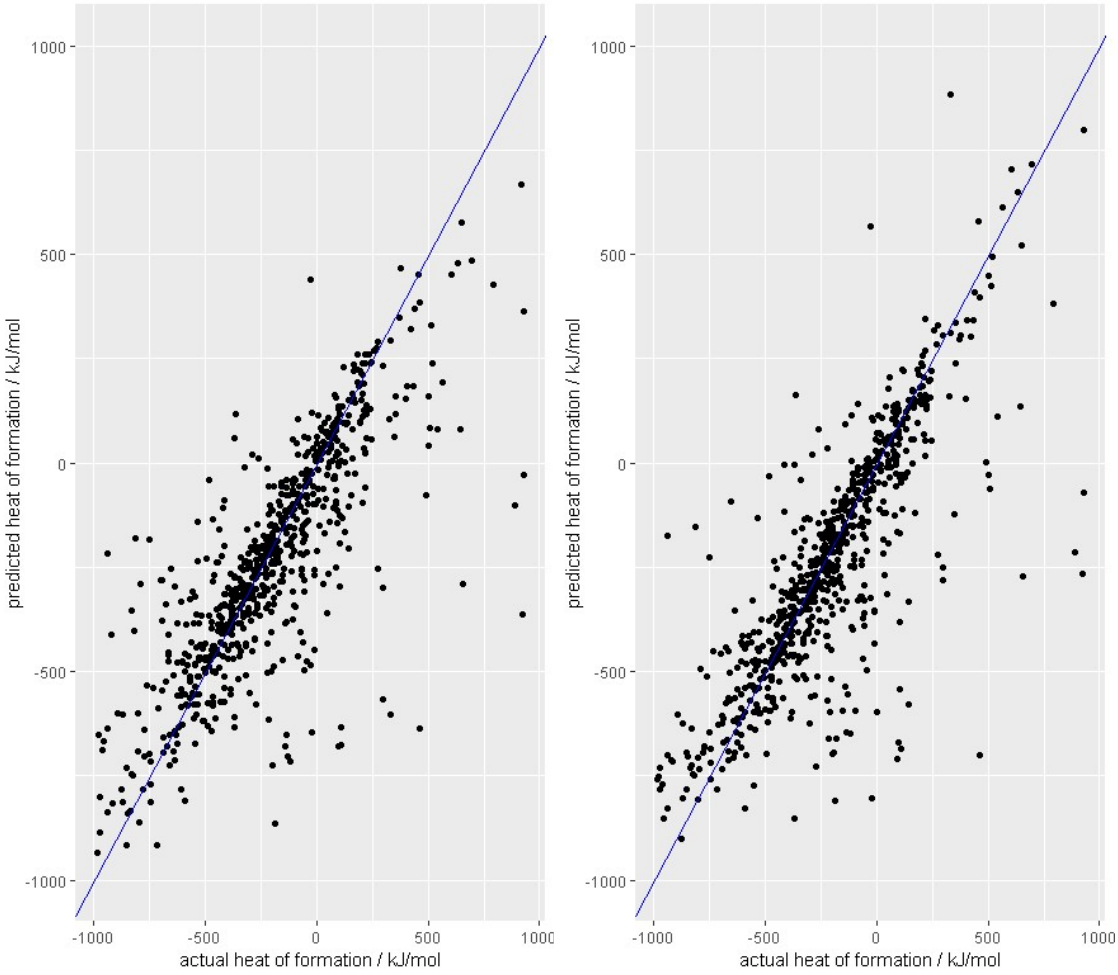

Figure 4: Parity plot of random forest (left) and neural network (right). MAEs are 109 and 90 kJ/mol, respectively.

The neural network has a lot of hyperparameters to tune and it is very time consuming to run a full test across all hyperparameter combinations. It is therefore useful to inspect the learning history (Figure 5) of the NN. In the upper part the loss function (here the MAE) is shown for both the training set and the test set. The epochs are the repeated cycles in which the NN is learning. The lower part shows the so-called learning rate which is how strong the network can change between runs. The value is being reduced step wise by a callback function as soon as the change in the constantly recalculated loss value falls below a predefined limit value over a certain number of epochs. This helps to ensure that the model does not enter an overfitting phase too early. From Figure 5, after 100 epochs the validation loss no longer significantly decreases. For hyperparameter tuning this would mean that a cycle of 100 epochs is good enough to get a realistic result for model performance instead of 200 used here, halving the optimization time.

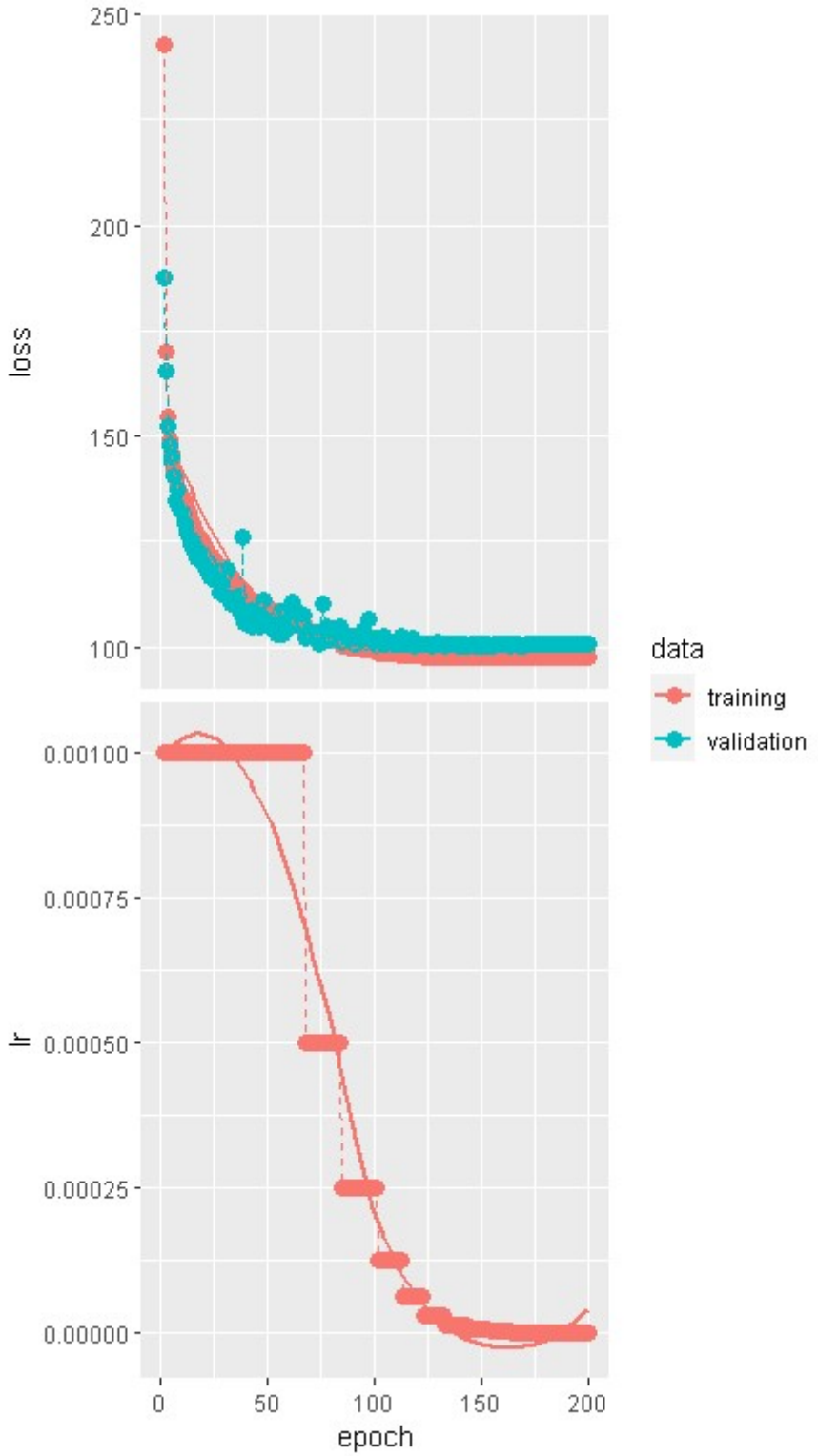

Figure 5: History of the evolution of the neural network. The epochs are the number of consecutive runs and show the learning curve of the network.

Summary

In this work the heat of formation of approx. 3000 molecules has been modelled with the use of machine learning methods based on the ICT Thermodynamic database. Information about data acquisition and cleaning as well as data fusion and feature engineering have been given. The heat of formation was predicted by using the bonds and atoms in the molecule with six different models. A neural network performed best with a mean absolute error of 90 kJ/mol.

References

1. Kang, P.; Liu, Z.; Abou-Rachid, H.; Guo, H. Machine-Learning Assisted Screening of Energetic Materials. *The journal of physical chemistry. A* **2020,** *124* (26), 5341–5351. DOI: 10.1021/acs.jpca.0c02647.
2. Mathieu, D.; Alaime, T. Impact sensitivities of energetic materials: Exploring the limitations of a model based only on structural formulas. *Journal of molecular graphics & modelling* **2015,** *62*, 81–86. DOI: 10.1016/j.jmgm.2015.09.001.
3. Hansen, K.; Biegler, F.; Ramakrishnan, R.; Pronobis, W.; Lilienfeld, O. A. von; Müller, K.-R.; Tkatchenko, A. Machine Learning Predictions of Molecular Properties: Accurate Many-Body Potentials and Nonlocality in Chemical Space. *The journal of physical chemistry letters* **2015,** *6* (12), 2326–2331. DOI: 10.1021/acs.jpclett.5b00831.
4. Chun, S.; Roy, S.; Nguyen, Y. T.; Choi, J. B.; Udaykumar, H. S.; Baek, S. S. Deep learning for synthetic microstructure generation in a materials-by-design framework for heterogeneous energetic materials. *Scientific reports* **2020,** *10* (1), 13307. DOI: 10.1038/s41598-020-70149-0.
5. Elton, D. C.; Boukouvalas, Z.; Butrico, M. S.; Fuge, M. D.; Chung, P. W. Applying machine learning techniques to predict the properties of energetic materials. *Scientific reports* **2018,** *8* (1), 9059. DOI: 10.1038/s41598-018-27344-x.
6. 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. DOI: 10.1021/ci00062a008.
7. Kim, S.; Chen, J.; Cheng, T.; Gindulyte, A.; He, J.; He, S.; Li, Q.; Shoemaker, B. A.; Thiessen, P. A.; Yu, B.; Zaslavsky, L.; Zhang, J.; Bolton, E. E. PubChem 2023 update. *Nucleic Acids Res* **2023,** *51* (D1), D1373-D1380. DOI: 10.1093/nar/gkac956.
8. Guha, R. Chemical Informatics Functionality in R. *J. Stat. Soft.* **2007,** *18* (5), 1–16. DOI: 10.18637/jss.v018.i05.