Bayesian Transfer Learning of Neural Network-Based Interatomic Force Models

Tim Rensmeyer[1], **Willi Großmann** [1], **Denis Kramer** [1], **Oliver Niggemann** [1]

[1]Helmut-Schmidt-University Hamburg
rensmeyt@hsu-hh.de, grossmaw@hsu-hh.de, d.kramer@hsu-hh.de, oliver.niggemann@hsu-hh.de

Abstract

Due to the computational complexity of evaluating interatomic forces from first principles, the creation of interatomic machine learning force fields has become a highly active field of research. However, the generation of training datasets of sufficient size and sample diversity itself comes with a computational burden, that can make this approach unpractical for modeling rare events or systems with a large configuration space. Transfer learning has been shown to achieve good accuracy for machine learning force fields with a much sparser sampling of configurations. Unfortunately, it does not provide solutions on how to efficiently sample different regions of configuration space in the first place and without such a method, the actual improvement in data efficiency remains much less obvious. In active learning, model uncertainty can be used to guide the sampling procedure towards regions of high uncertainty, resulting in a much more efficient sampling procedure. Therefore, it seems highly desirable to develop a framework for transfer learning that can also produce high-quality uncertainty quantification. In this paper, we demonstrate that Bayesian neural networks are suitable as such a framework by utilizing their prior density for transfer learning.

Introduction

Ever since the discovery of the laws of quantum mechanics almost a century ago, the prediction of molecular and material properties such as stress-strain relationships or catalytic activity from first principles has in theory been possible (Atkins and Friedman 2011; Gastegger and Marquetand 2020). However, in practice, this task remains challenging even to this day (Schütt et al. 2020; Giustino 2014). The major difficulty lies in the exponentially growing computational complexity of solving the underlying Schrödinger equation with an increasing number of electrons (Atkins and Friedman 2011). As a consequence, several alternative methods for property prediction have been developed, which are computationally tractable for larger systems at the cost of varying degrees of accuracy.

Some of the most popular of these methods are in roughly ascending order of accuracy and computational complexity Density Functional based Tight Binding (DFTB) methods (Spiegelman et al. 2020), Density Functional Theory (DFT) (Giustino 2014) and Coupled Cluster (CC) approaches (Bartlett and Musiał 2007). With these approximations, many properties, such as electronic structure, binding energies and interatomic forces can be predicted with a high degree of accuracy and a computational complexity that is feasible on a typical high-performance cluster for many tasks (Giustino 2014). While DFT has enabled the investigation of the properties for individual materials at quantum mechanical accuracy, high-throughput screening of materials or molecules for desired properties still remains very computationally demanding. Furthermore, Molecular Dynamics (MD) – the simulation of the time evolution of molecular systems and materials – remains challenging, as the forces on all atoms have to be calculated at each timestep. This severely limits the time horizon that can be achieved in practical amounts of time using DFT (Gastegger and Marquetand 2020).

Subsequently, the development of machine learning models that can predict interatomic forces has become an active field of research (Kocer, Ko, and Behler 2022). In particular neural network-based interatomic force models have made great strides in the past years and can achieve much higher accuracy than previous methods with just a few hundred well-sampled training configurations of a specific system(Klicpera, Becker, and Günnemann 2021; Unke et al. 2021; Schütt, Unke, and Gastegger 2021; Batzner et al. 2022; Haghighatlari et al. 2022; Qiao et al. 2022). However, in practice, the necessary amount of data that needs to be generated is much higher. First of all, to generate physically realistic and diverse atomic configurations, these training configurations usually have to be sampled from much larger MD trajectories containing thousands of configurations (Chmiela et al. 2017a; Christensen and von Lilienfeld 2020; Wen and Tadmor 2020). While some alternatives such as umbrella sampling exist (Torrie and Valleau 1977), that reduce some redundancy in the sampled trajectory by modifying the underlying dynamics, they are still based on simulating a dynamical trajectory that involves the costly calculation of forces. Due to this, there is still a large amount of redundancy due to configurations of subsequent time steps being very similar. This problem is compounded by the fact that a bigger system typically has a larger configuration space and hence requires longer trajectories to sample it sufficiently while the computational demand of calculating each time step is also increased. Consequently, when attempting to create a Machine Learning Force Field (MLFF)

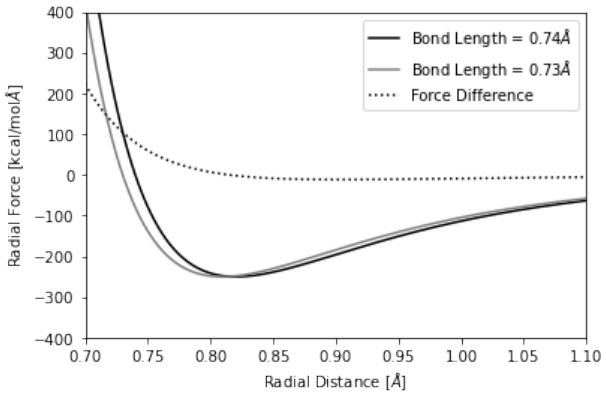

Figure 1: Lennard-Jones models of the radial force of a hydrogen molecule with correct bond energies but bond distances of 0.74 Å and 0.73 Å respectively. The dotted line indicates the difference in radial force values. Due to the rapidly growing radial forces under compression along the bond axis, an MD trajectory done with the model that has a larger bond distance will likely contain no configurations with a bond distance around 0.73 Å which is the most likely configuration of the other model.

for ever larger systems, one can quickly reach a limit due to the same reason the force field was needed in the first place: Dynamical trajectories at the desired accuracy can not be computed for a time horizon of adequate length. At first glance, computing the trajectory at a lower accuracy and computational complexity and then recalculating some subset of those configurations at the desired accuracy appears like a simple solution to increase sample diversity and reduce computational demand. Unfortunately, even though lower accuracy methods often predict qualitatively very similar force fields that result in almost identical chemistry, they often introduce some small biases such as slightly different bond lengths. For example, some DFT methods are known to have a bias towards bond lengths that are too small by up to one percent while the usual variation in bond length due to thermal fluctuations can be much lower than that(Giustino 2014). As a result, many typical configurations from a trajectory of one simulation method can become very rare in trajectories of the other method (Figure 1). Consequently, the MLFF would become less accurate for configurations it frequently encounters in its trajectories, since no similar configurations were contained in the training set.

The problem of sampling relevant configurations is even more severe when trying to train a MLFF to explore chemical phenomena such as reaction mechanisms. Chemical reactions themself are almost never contained in MD trajectories due to their rarity. This is well illustrated simply by the fact that reaction rates are typically measured in change in concentration per second while MD simulations run at approximately femtosecond time steps. Subsequently, on the order of 10^{15} time steps would usually need to be simulated to have a reasonable chance of observing chemical reactions. A new paradigm for creating training datasets appears urgently needed in order to extend the capabilities of MLFFs to model larger systems and explore chemical phenomena or other rare events. An iterative approach utilizing transfer learning and active learning is a promising way forward to remedy some of the mentioned problems. Transfer learning has already been demonstrated to significantly reduce the necessary amount of data for creating accurate neural network models of atomic systems by pretraining the neural network on auxiliary datasets (Kolluru et al. 2022; Zaverkin et al. 2023; Smith et al. 2019) and then fine-tuning the pre-trained model on the actual training data. The auxiliary datasets could for example contain data generated from lower accuracy simulation methods that require orders of magnitude less computation or labeled configurations from existing datasets of different systems. The second point is particularly interesting in light of the recent efforts to create universal publicly available machine-learning force fields (Smith, Isayev, and Roitberg 2017; Chen and Ong 2022; Smith et al. 2018) that are applicable to a large variety of atomic systems out of the box but which do not reach the desired accuracy in many applications. Here tranfer learning might enable the fine-tuning of the universal force fields with only a sparse sampling of configuration space.
However, it is not obvious how to create such a sparse sampling that still contains samples from all physically relevant regions of configuration space without subsampling from a long MD trajectory. Hence, it seems unclear how much more efficient these approaches are in practice.

Uncertainty-based active learning methods have established themselves as a way to further improve data efficiency as well as accuracy during rare events (Vandermause et al. 2019; Podryabinkin et al. 2019; Podryabinkin and Shapeev 2017; Gubaev et al. 2019) by sampling configurations more efficiently. This can be achieved by selectively labeling samples from regions of configuration space where the MLFF still has a high uncertainty (Kulichenko et al. 2023). Thus, the combination of active learning and transfer learning appears very promising.
For example, on-the-fly learning is an elegant approach (Jinnouchi et al. 2020; Vandermause et al. 2019) where the pre-trained model might be used to drive the dynamics (e.g. MD simulation, transition state optimization, etc.) until a configuration is encountered that exceeds a certain uncertainty threshold. This configuration is then recalculated with classical simulation methods and used to update the model which then resumes the task until the next configuration above the uncertainty threshold is encountered. This approach has the additional benefit of being very easy to use for non-machine learning experts, like many computational chemists are, since essentially only the initial state of the system and the uncertainty threshold would have to be specified.

However, what is currently missing from the literature is a framework to systematically update a pre-trained neural network-based MLFF on new data while also being able to assess its uncertainty.

An appropriate framework needs to fulfill two separate conditions which we will use to assess suitability. First of

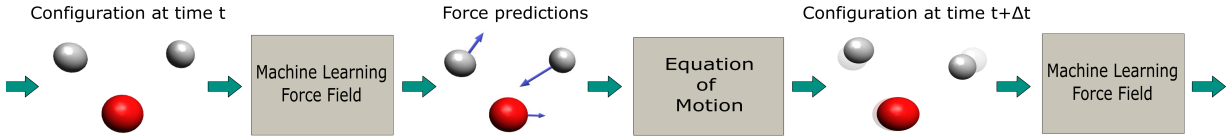

Figure 2: An illustration of a molecular dynamics workflow of a water molecule utilizing a machine learning force field.

all, it needs to produce an accurate force field with a sparser sampling of configuration space. Furthermore, it needs to be able to detect configurations with a high likelihood of a large error based on its predicted uncertainty in order to enable the sparse sampling of that configuration space. Bayesian Neural Networks (BNNs) have demonstrated a high quality of uncertainty quantification comparable to classical ensemble-based methods of uncertainty quantification (Kahle and Zipoli 2022). In addition, Bayesian methods offer a simple and systematic way of incorporating and data-based updating of preexisting knowledge by utilizing the Bayesian prior density and Bayes theorem (Shwartz-Ziv et al. 2022; Chen et al. 2019).
Due to these properties, the principal research question of this paper is whether BNN models of interatomic forces might be suitable as such a framework.

The main contributions of this work are the following:

- We develop a simple Bayesian framework for uncertainty-aware transfer learning of interatomic force fields, by harnessing a simple transfer learning prior and Monte Carlo Markov Chain (MCMC) sampling.
- We analyze this approach in three different transfer learning scenarios, DFTB to DFT, DFT to CC and the use of and transfer learning between different atomic systems at DFT level accuracy and demonstrate large improvements in accuracy and data efficiency for each scenario.
- We demonstrate that the quality of uncertainty quantification in all three scenarios is comparable to a Bayesian neural network model of similar accuracy which was trained from scratch.

Interatomic Force Modeling using Bayesian Neural Networks

Throughout the rest of this paper, bold case symbols designate vectors, n is the number of individual atoms of an atomic system, m is the size of the training set and $\boldsymbol{y}|\boldsymbol{x}$ denotes $\boldsymbol{y}$ conditioned on $\boldsymbol{x}$.

Machine-learned interatomic force fields for molecules aim to map an atomic configuration $\{(\boldsymbol{r}_1, z_1), ..., (\boldsymbol{r}_n, z_n)\}$, composed of the nuclear coordinates $\boldsymbol{r}_i \in \mathbb{R}^3$ and nuclear charges z_i, to the forces $\boldsymbol{F}_i$ acting on each nucleus $i \in \{1, ..., n\}$. The predicted forces can then for example be used in combination with Newton's equations of motion to model the time evolution of the molecules (Figure 2). Because generating large amounts of high-quality training data is typically infeasible due to the high computational demand of classical simulation methods, modern machine learning models have several forms of physical constraints built into them, in order to make them more data efficient (Unke et al. 2021; Schütt, Unke, and Gastegger 2021; Batzner et al. 2022; Qiao et al. 2022; Schütt et al. 2018; Chmiela et al. 2017b). One important constraint is energy conservation, meaning that the forces can be derived as the negative gradients of a potential energy surface $u(\boldsymbol{r}_1, ..., \boldsymbol{r}_n) \in \mathbb{R}$, i.e $\boldsymbol{F}_i = -\nabla_{\boldsymbol{r}_i} u(\boldsymbol{r}_1, ..., \boldsymbol{r}_n)$.
Further, the potential energy surface is known to be invariant under any transformation of the nuclear coordinates which leaves interatomic distances invariant.

Bayesian Neural Networks

Bayesian neural networks have demonstrated promising results for modeling uncertainties in neural network predictions and in particular in machine learning force fields (Kahle and Zipoli 2022; Kwon et al. 2018; Rensmeyer et al. 2023). The main difference between the Bayesian approach to neural networks and the regular approach is, that the trainable parameters of the neural network, e.g. its weights and biases, are modeled probabilistically. For simplicity of notation, we denote by $\boldsymbol{\theta}$ a vector containing all the trainable parameters of the neural network. For a given parameter vector $\boldsymbol{\theta}$ and input sample $\boldsymbol{x}$, the neural network predicts a probability density $p(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta})$ over the target variable $\boldsymbol{y}$. In the case of machine learning force fields, $\boldsymbol{x}$ will be an atomic configuration $\{(\boldsymbol{r}_1, z_1), ..., (\boldsymbol{r}_n, z_n)\}$ and $\boldsymbol{y}$ will be the atomic forces $\{\boldsymbol{F}_1, ..., \boldsymbol{F}_n\}$. The starting point of Bayesian methods is a prior density $p(\boldsymbol{\theta})$ over the parameters which expresses apriori knowledge about which sets of parameters are likely to result in a good model of the underlying data distribution. Given some training dataset $\mathcal{D} = \{(\boldsymbol{x}_1, \boldsymbol{y}_1), ..., (\boldsymbol{x}_m, \boldsymbol{y}_m)\}$ the prior density gets refined into the posterior density $p(\boldsymbol{\theta}|\mathcal{D})$ using Bayes' theorem:

$$p(\boldsymbol{\theta}|\mathcal{D}) = \frac{p(\mathcal{D}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathcal{D})}.$$

With the mild assumption on conditional independence

$$p(\boldsymbol{y}_1, ..., \boldsymbol{y}_m|\boldsymbol{x}_1, ..., \boldsymbol{x}_m, \boldsymbol{\theta}) = \Pi_{i=i}^m p(\boldsymbol{y}_i|\boldsymbol{x}_i, \boldsymbol{\theta})$$

this can be simplified to

$$p(\boldsymbol{\theta}|\mathcal{D}) = Z \cdot p(\boldsymbol{\theta}) \Pi_{i=i}^m p(\boldsymbol{y}_i|\boldsymbol{x}_i, \boldsymbol{\theta}),$$

where $Z = \frac{p(\boldsymbol{x}_1, ..., \boldsymbol{x}_m)}{p(\mathcal{D})}$ is a normalization constant. On a new input sample $\boldsymbol{x}$, the probability distribution of the target variable $\boldsymbol{y}$ can then be calculated via

$$p(\boldsymbol{y}|\boldsymbol{x}, \mathcal{D}) = \int p(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta}) p(\boldsymbol{\theta}|\mathcal{D}) d\boldsymbol{\theta} = \mathbb{E}_{p(\boldsymbol{\theta}|\mathcal{D})} \left[p(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta}) \right].$$

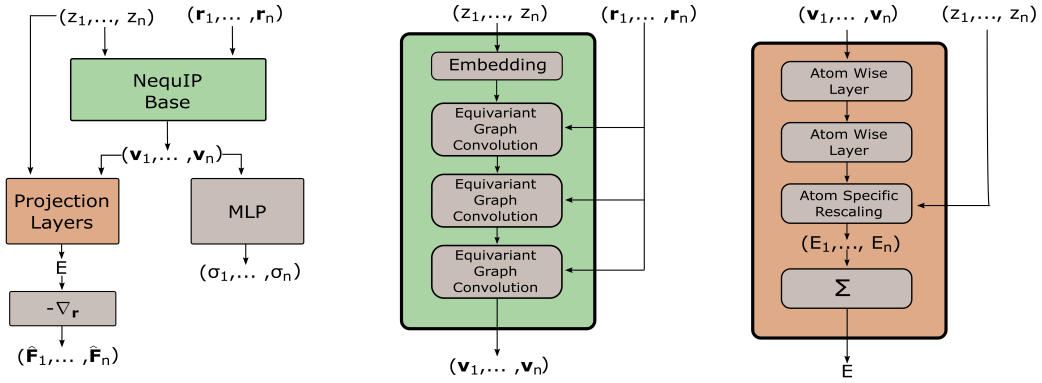

Figure 3: An illustration of the neural network architecture for predicting the means $\hat{\boldsymbol{F}}_i$ and standard deviations σ_i for all atoms. Except for the MLP this corresponds to the architecture of the original NequIP neural network model.

Because this integral is almost never analytically tractable for neural networks, a Monte Carlo estimate is typically used:

$$p(\boldsymbol{y}|\boldsymbol{x}, \mathcal{D}) \approx \frac{1}{k} \sum_{i=1}^{k} p(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta}_i), \quad \boldsymbol{\theta}_i \sim p(\boldsymbol{\theta}|\mathcal{D}),$$

where the parameter sets $\boldsymbol{\theta}_i$ are sampled from the posterior density using either approximate inference (Maddox et al. 2019; Blundell et al. 2015; Gal and Ghahramani 2016) or MCMC methods (Welling and Teh 2011; Chen, Fox, and Guestrin 2014; Ma, Chen, and Fox 2015). MCMC methods in particular have displayed good results in uncertainty quantification (Yao et al. 2019) due to their ability to sample different regions of the posterior. These methods work by simulating a stochastic process over the space of neural network parameters which converges in distribution to the posterior.

Relation to other Works

Even though Bayesian neural networks offer a very promising opportunity to systematically incorporate and update pre-existing knowledge via the prior density, we find that this approach is very underutilized in the literature and in fact, we could only find two instances in the literature, where this was attempted (Shwartz-Ziv et al. 2022; Chen et al. 2019). In (Chen et al. 2019) transfer learning of simulated to experimental data was done via a Bayesian neural network prior. A simple isotropic Gaussian prior with a mean derived during pretraining was used in that work, which we will also employ here.

A more sophisticated approach for constructing a transfer learning prior was introduced in (Shwartz-Ziv et al. 2022), where a rescaled local approximation of the posterior on the auxiliary dataset was used as a prior. However, in the applications considered here, this approach is not very practical. One reason for this is, that by definition this form of a prior assigns a low density to parameters that produce a low log-likelihood on the auxiliary dataset and hence discourages changes in the predictions on configurations included in the auxiliary dataset. This is a problem for transfer learning scenarios from lower accuracy simulations to higher accuracy ones, where those predictions necessarily need to change. As a consequence, we found no improvement in data efficiency in these scenarios.

Further, while such conceptual issues don't exist for the transfer learning scenarios from auxiliary datasets of different compounds generated with the same simulation method as the target dataset, there instead exists a practical problem. Oftentimes such a scenario will involve finetuning a publicly available universal machine-learned force field, which was trained via a classical neural network optimizer and hence, that method for constructing a prior will simply not be applicable.

Transfer learning of a pre-trained model for interatomic force fields has been investigated in (Kolluru et al. 2022; Zaverkin et al. 2023; Smith et al. 2019). However, the modeling of uncertainty has not been under consideration in those works.

Solution

We utilize a stochastic variant of the NequIP (Batzner et al. 2022) model introduced in (Rensmeyer et al. 2023). This model can be divided into three different components (Figure 3). The first one maps the input $x = \{(\boldsymbol{r}_1, z_1), ..., (\boldsymbol{r}_n, z_n)\}$ to latent variables $\{\boldsymbol{v}_1, ..., \boldsymbol{v}_n\}$ that are invariant under distance-preserving transformations of the atomic coordinates $\boldsymbol{r}_i$. The second is a Mulit-Layer Perceptron (MLP) which calculates a scalar standard deviation σ_i from each atomic feature vector $\boldsymbol{v}_i$. The third component calculates a scalar (virtual) energy contribution E_i for each atomic feature vector $\boldsymbol{v}_i$. A total potential energy is then calculated as $E = \sum_{i=1}^{n} E_i$ and from that the expected forces $\hat{\boldsymbol{F}}_i$ are calculated as the negative gradients of the energy with respect to the atomic coordinated $\boldsymbol{r}_i$.

From the predicted means and standard deviations, we model the distribution over the forces as

$$\boldsymbol{F}_1, ..., \boldsymbol{F}_n | (\boldsymbol{r}_1, z_1), ..., (\boldsymbol{r}_n, z_n), \boldsymbol{\theta} \sim \Pi_{i=1}^{n} N(\hat{\boldsymbol{F}}_i, \sigma_i^2 I),$$

where N denotes a normal distribution and I is the identity matrix.

To construct the prior, we pre-train this model on the auxiliary dataset to generate a set of neural network parame-

ters $\boldsymbol{\theta}_0$. We then set the transfer learning prior as $p_{TL} \sim N(\boldsymbol{\theta}_0, \sigma_{TL}^2 I)$ with a small standard deviation σ_{TL}, which is the approach used in (Chen et al. 2019). Importantly, in this formalism, a diverse ensemble of models can still be generated by sampling (approximately) independent parameter sets from the posterior. Further, it is not necessary to optimize several models from scratch as would be the case for deep ensembles but instead, we can sample all parameter sets from the same Markov chain, starting from the single parameter set $\boldsymbol{\theta}_0$. This is highly advantageous since the training of a single state-of-the-art neural network model from scratch can already take several days on a modern GPU. To sample the posterior density, we use the AMSGrad version of the Stochastic Gradient Hamiltonian Monte Carlo (SGHMC) algorithm (Chen, Fox, and Guestrin 2014) introduced in (Rensmeyer et al. 2023) (see Appendix A for details of the neural network architecture, sampling procedure and pre-training).

Empirical Evaluation

Three different benchmarks are used to evaluate our model, representing likely scenarios where transfer learning might be employed. The first test is a transfer learning scenario of finetuning a more general neural network model trained on a variety of different compounds to a specific molecule of interest not included in the pre-training dataset. More specifically we pre-train a neural network model on a dataset consisting of a variety of compounds of the MD17 (Chmiela et al. 2017a) and MD22 (Chmiela et al. 2023) datasets, which consist of MD trajectories of several molecules at DFT level accuracy and then finetune it on the paracetamol dataset of the MD17 dataset.

The second benchmark is a transfer learning scenario from DFTB level accuracy to DFT level accuracy. In particular, we generate a large dataset of different configurations of a stachyose molecule in DFTB for pre-training and then utilize the stachyose data from the MD22 dataset for the transfer learning task.

The final test scenario is a transfer learning task for reaching CC level accuracy on an ethanol molecule starting from a model pre-trained on the corresponding ethanol data from the MD17 dataset. The CC-level dataset used for this was introduced in (Bogojeski et al. 2019).

In all three test scenarios, a validation set of 10 configurations was used to recalibrate all predicted standard deviations on the test sets by multiplying them with a scalar factor that was chosen by matching the resulting mean predicted variance to the mean square error on the validation set. All experiments were done with 8 Monte Carlo samples generated from the same Markov chain, which has been identified as a good tradeoff between computational complexity and quality of uncertainty quantification in previous benchmarks (Rensmeyer et al. 2023). Details of all the datasets can be found in Appendix B. We set σ_{TL} as 0.2 in all experiments.

The Evaluation Metrics:

On all tasks, we evaluate the model's overall accuracy in terms of the Root Mean Square Error (RMSE) of the force components. We analyze the accuracy of the transfer learned model in dependence of the size of the training dataset and compare it to a model that is trained from scratch with a simple Gaussian mean field prior $p(\boldsymbol{\theta}) \sim N(\mathbf{0}, I)$.

One concern is, that a prior with a small standard deviation could lead to a suboptimally small predictive variance of the Monte Carlo samples from the posterior since parameter sets of all samples will tend to remain close to the mean of the prior. This might negatively impact the quality of uncertainty quantification at a given accuracy compared to a model generated from a wider uninformative prior e.g. $p(\boldsymbol{\theta}) \sim N(\mathbf{0}, I)$.

Hence we analyze the transfer learning models quality of uncertainty quantification in comparison to a model with a Gaussian mean field prior $p(\boldsymbol{\theta}) \sim N(\mathbf{0}, I)$. For this, we compare the Mean Log Likelihoods (MLLs) of the force components as a function of the RMSE for both models. To smooth each predicted distribution of the 8 Monte Carlo samples on this metric, we fit a normal distribution to the means and variances of each predicted distribution and use these smoothed distributions instead. Further, since the main goal of the uncertainty measure is the identification of configurations with a large error in the prediction, we evaluate the models in the task of detecting force components with a large prediction error based on the predicted uncertainty. More specifically, we analyze the corresponding AUC-ROC scores for detecting large errors via the predicted variance and plotting them as a function of the RMSE. On the ethanol and paracetamol dataset errors of more than 1kcal/molÅ were considered large, while on the more difficult stachyose dataset, the cutoff was set as 3kcal/molÅ because an error of 1kcal/molÅ could not be considered an outlier.

Results

As can be seen in Figure 4, very high accuracies were reached for the transfer learning model on the paracetamol dataset even for small training datasets in terms of the RMSE when compared to the model trained from scratch. Further, there appears to be no major decrease in the quality of uncertainty quantification at a given accuracy as measured by the MLLs and AUC-ROC scores and the plots are almost on top of each other where the RMSEs overlap. However, there might be a very small decrease in quality as indicated by Figure 4.

On the stachyose dataset, again a clear improvement in accuracy at equal amounts of training samples is visible when compared to the baseline model (Figure 4). However, both models have higher RMSEs than their counterparts on the paracetamol dataset at equal amounts of training configurations. The MLLs of the transfer learning model appear to be slightly lower than for a model trained from scratch when controlled for accuracy. The same is true only to a much smaller degree for the AUC-ROC scores. Further analysis revealed, that the validation set was too small for the large configuration space of stachyose to properly recalibrate the uncertainties which led to an overestimation of the errors on the test set for the transfer learned models but not for the baseline models. This also

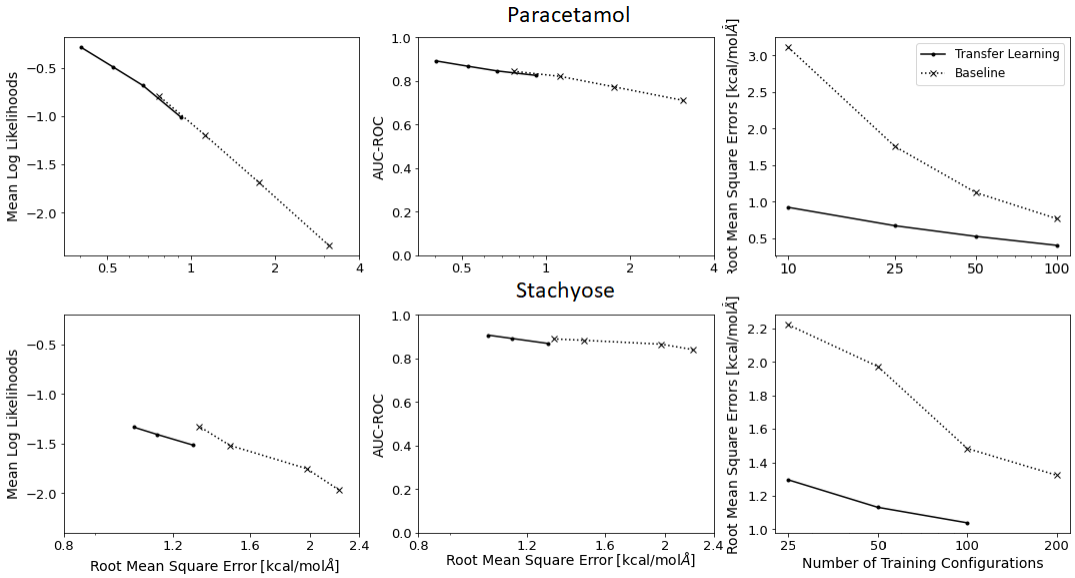

Figure 4: Results on the paracetamol and stachyose datasets. On the left are the mean log-likelihoods as a function of RMSE (all means and standard deviations in kcal/molÅ). In the middle are the AUC-ROC scores for uncertainty-based detection of force components with a high prediction error. On the right are the Root Mean Square Errors as a function of the number of training configurations

explains the absence of such reduced performance on the AUC-ROC scores which are invariant under recalibration of uncertainties. Accounting for the slightly wrong calibration by recalibrating the uncertainties on the test set instead of the validation set confirmed miscalibration as the main source of the gap in MLLs. In particular, the gap between the MLLs of the transfer learning models that are closest in RMSE reduced from 0.191 to 0.086.
The biggest improvement in accuracy, when compared to the baseline model, was found on the ethanol dataset (Figure 5), with an RMSE of less than 0.5kcal/molÅ with only 10 configurations. There appears to be no decrease in the quality of uncertainty quantification both in terms of MLLs as well as AUC-ROCs on this benchmark.

Interestingly, for all transfer learning scenarios, the error of the pre-trained model was quite large with mean absolute errors of 2.31kcal/molÅ on the paracetamol validation set, 4.34kcal/molÅ on the stachyose validation set and 5.12kcal/molÅ on the ethanol validation set. Further, all pre-trained models achieved a validation loss smaller than 0.15kcal/molÅ on their pre-training datasets strongly indicating, that DFTB, DFT and CC methods disagree quite substantially in their force predictions for a given configuration. However, as was already alluded to in the introduction, this can most likely be traced back to simple biases in the simulation methods, such as slightly different equilibrium bond lengths. These small biases in different simulation methods can lead to qualitatively very similar force fields that may disagree substantially on the forces of a given configuration. This would also explain, why transfer learning is very efficient in these cases, as the model mostly has to correct for those biases such as equilibrium bond lengths. Importantly, those force fields will lead to very similar predictions of physical and chemical properties despite their apparently large disagreement, while a machine-learned force field with a similar magnitude of error to one of those methods can not in general be expected to yield those properties as well and hence needs to be trained to a much higher accuracy.
One additional result that stands out is the relatively high RMSE of both the transfer learning and the baseline model on the stachyose dataset when compared to the other two test scenarios. However, there are two factors that make this dataset particularly challenging. First of all, stachyose is a larger molecule than paracetamol and ethanol which in addition contains many single sigma bonds that allow for rotational degrees of freedom along the bond axis. This results in a very large configuration space for stachyose molecules even relative to their size. The second factor that makes this benchmark more challenging for the transfer learning model is that unlike in the ethanol case, the higher accuracy dataset was not composed of configurations generated from an MD trajectory of the lower accuracy method but instead from a trajectory at DFT-level accuracy. As a result, the distribution of configurations in the DFT dataset will be different from the one from the DFTB dataset. Lastly, one important observation we made is that the transfer learning approach converges much faster than when training from scratch. While state-of-the-art models can take days to train from scratch, training and validation losses converged within minutes on the transfer learning tasks. In fact, the only reason we let the sampling algorithm run for as long as described in the Appendix is to make sure that no pathological overfitting takes place.

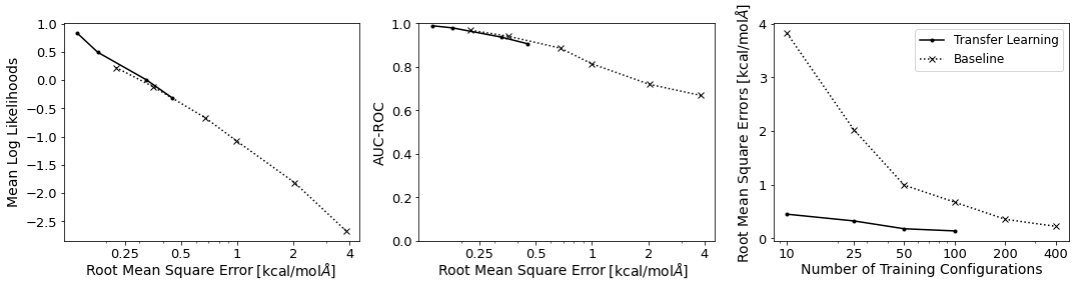

Figure 5: Results on the ethanol dataset. On the left are the mean log-likelihoods as a function of RMSE (all means and standard deviations in kcal/molÅ). In the middle are the AUC-ROC scores for uncertainty-based detection of force components with a high prediction error. On the right are the Root Mean Square Errors as a function of the number of training configurations

Conclusion and Outlooks

Despite the simplicity of the chosen transfer learning prior, the outcomes demonstrate a clear improvement in accuracy and data efficiency on all three tasks. While technically the computational demand of generating the lower accuracy data has to be taken into account for the DFTB to DFT and DFT to CC transfer learning scenarios, this is not very relevant in practice since the higher accuracy simulation methods are orders of magnitude slower. The predicted uncertainties of the transfer learning model do not appear to have a substantially diminished quality despite the relatively small standard deviation of the transfer learning prior. One exception to this are the somewhat lower MLLs on the stachyose benchmark due to miscalibration. Particularly encouraging are the high AUC-ROC scores, which are close to the optimal value of one during all experiments. This strongly indicates that the transfer-learned models can reliably detect configurations with a high likelihood of a large error. Consequently, the stated conditions for a suitable framework to combine active and transfer learning are clearly met by the transfer learning algorithm investigated here. This opens up new possibilities to extend the application of MLFFs to larger systems and new scenarios such as rare events.

The biggest remaining obstacle to this is developing active learning algorithms suitable for different application scenarios e.g. MD simulations or transition state optimization and integrating them with the transfer learning algorithm discussed here. Additionally, the introduced framework offers an opportunity to make transfer-learned models more trustworthy by evaluating their uncertainty and recomputing configurations with high uncertainty on the fly with classical simulation methods.
In summary, the results in this paper point towards Bayesian transfer learning of machine-learned force fields as a viable option for trustworthy and data-efficient molecular modeling. Further, the quantification of predictive uncertainty could potentially be used in the future in iterative active learning approaches to enhance data efficiency even more and to ensure high accuracy even for rare configurations by sampling the configuration space more efficiently.

Acknowledgements

This research as part of the projects CouplteIT! and KIBIZ is funded by dtec.bw – Digitalization and Technology Research Center of the Bundeswehr which we gratefully acknowledge. dtec.bw is funded by the European Union – NextGenerationEU.

References

Atkins, P.; and Friedman, R. 2011. *Molecular Quantum Mechanics*. OUP Oxford. ISBN 9780199541423.

Bartlett, R. J.; and Musiał, M. 2007. Coupled-cluster theory in quantum chemistry. *Rev. Mod. Phys.*, 79: 291–352.

Batzner, S.; Musaelian, A.; Sun, L.; Geiger, M.; Mailoa, J. P.; Kornbluth, M.; Molinari, N.; Smidt, T. E.; and Kozinsky, B. 2022. E(3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials. *Nature Communications*, 13.

Blundell, C.; Cornebise, J.; Kavukcuoglu, K.; and Wierstra, D. 2015. Weight Uncertainty in Neural Network. In *International Conference on Machine Learning*.

Bogojeski, M.; Vogt-Maranto, L.; Tuckerman, M. E.; Müller, K.-R.; and Burke, K. 2019. Quantum chemical accuracy from density functional approximations via machine learning. *Nature Communications*, 11.

Chen, C.; and Ong, S. P. 2022. A universal graph deep learning interatomic potential for the periodic table. *Nature Computational Science*, 2(11): 718–728.

Chen, C. H.; Parashar, P.; Akbar, C.; Fu, S. M.; Syu, M.-Y.; and Lin, A. 2019. Physics-Prior Bayesian Neural Networks in Semiconductor Processing. *IEEE Access*, 7: 130168–130179.

Chen, T.; Fox, E. B.; and Guestrin, C. 2014. Stochastic Gradient Hamiltonian Monte Carlo. In *International Conference on Machine Learning*, 1683–1691.

Chmiela, S.; Tkatchenko, A.; Sauceda, H. E.; Poltavsky, I.; Schütt, K. T.; and Müller, K.-R. 2017a. Machine learning of accurate energy-conserving molecular force fields. *Science Advances*, 3(5): e1603015.

Chmiela, S.; Tkatchenko, A.; Sauceda, H. E.; Poltavsky, I.; Schütt, K. T.; and Müller, K.-R. 2017b. Machine learning of accurate energy-conserving molecular force fields. *Science Advances*, 3(5): e1603015.

Chmiela, S.; Vassilev-Galindo, V.; Unke, O. T.; Kabylda, A.; Sauceda, H. E.; Tkatchenko, A.; and Müller, K.-R. 2023. Accurate global machine learning force fields for molecules with hundreds of atoms. *Science Advances*, 9(2): eadf0873.

Christensen, A. S.; and von Lilienfeld, O. A. 2020. On the role of gradients for machine learning of molecular energies and forces. *Machine Learning: Science and Technology*, 1.

Elstner, M.; Porezag, D.; Jungnickel, G.; Elsner, J.; Haugk, M.; Frauenheim, T.; Suhai, S.; and Seifert, G. 1998. Self-consistent-charge density-functional tight-binding method for simulations of complex materials properties. *Phys. Rev. B*, 58: 7260–7268.

Gal, Y.; and Ghahramani, Z. 2016. Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning. In *International Conference on Machine Learning*, 1050–1059.

Gastegger, M.; and Marquetand, P. 2020. Molecular Dynamics with Neural Network Potentials. In *Machine Learning Meets Quantum Physics*, 233–252. Springer.

Gaus, M.; Cui, Q.; and Elstner, M. 2011. DFTB3: Extension of the Self-Consistent-Charge Density-Functional Tight-Binding Method (SCC-DFTB). *Journal of Chemical Theory and Computation*, 7(4): 931–948.

Gaus, M.; Lu, X.; Elstner, M.; and Cui, Q. 2014. Parameterization of DFTB3/3OB for Sulfur and Phosphorus for Chemical and Biological Applications. *Journal of Chemical Theory and Computation*, 10(4): 1518–1537. PMID: 24803865.

Giustino, F. 2014. *Materials Modelling Using Density Functional Theory: Properties and Predictions*. Oxford University Press. ISBN 9780199662449.

Gubaev, K.; Podryabinkin, E. V.; Hart, G. L.; and Shapeev, A. V. 2019. Accelerating high-throughput searches for new alloys with active learning of interatomic potentials. *Computational Materials Science*, 156: 148–156.

Haghighatlari, M.; Li, J.; Guan, X.; Zhang, O.; Das, A.; Stein, C. J.; Heidar-Zadeh, F.; Liu, M.; Head-Gordon, M.; Bertels, L.; Hao, H.; Leven, I.; and Head-Gordon, T. 2022. NewtonNet: a Newtonian message passing network for deep learning of interatomic potentials and forces. *Digital discovery*, 1(3): 333–343.

Hourahine, B.; Aradi, B.; Blum, V.; Bonafé, F.; Buccheri, A.; Camacho, C.; Cevallos, C.; Deshaye, M. Y.; Dumitrică, T.; Dominguez, A.; Ehlert, S.; Elstner, M.; van der Heide, T.; Hermann, J.; Irle, S.; Kranz, J. J.; Köhler, C.; Kowalczyk, T.; Kubař, T.; Lee, I. S.; Lutsker, V.; Maurer, R. J.; Min, S. K.; Mitchell, I.; Negre, C.; Niehaus, T. A.; Niklasson, A. M. N.; Page, A. J.; Pecchia, A.; Penazzi, G.; Persson, M. P.; Řezáč, J.; Sánchez, C. G.; Sternberg, M.; Stöhr, M.; Stuckenberg, F.; Tkatchenko, A.; Yu, V. W.-z.; and Frauenheim, T. 2020. DFTB+, a software package for efficient approximate density functional theory based atomistic simulations. *The Journal of Chemical Physics*, 152(12): 124101.

Jinnouchi, R.; Miwa, K.; Karsai, F.; Kresse, G.; and Asahi, R. 2020. On-the-Fly Active Learning of Interatomic Potentials for Large-Scale Atomistic Simulations. *The Journal of Physical Chemistry Letters*, 11: 6946–6955.

Kahle, L.; and Zipoli, F. 2022. Quality of uncertainty estimates from neural network potential ensembles. *Phys. Rev. E*, 105: 015311.

Klicpera, J.; Becker, F.; and Günnemann, S. 2021. GemNet: Universal Directional Graph Neural Networks for Molecules. In *Advances in Neural Information Processing Systems*.

Kocer, E.; Ko, T. W.; and Behler, J. 2022. Neural Network Potentials: A Concise Overview of Methods. *Annual Review of Physical Chemistry*, 73(1): 163–186. PMID: 34982580.

Kolluru, A.; Shoghi, N.; Shuaibi, M.; Goyal, S.; Das, A.; Zitnick, C. L.; and Ulissi, Z. 2022. Transfer learning using attentions across atomic systems with graph neural networks (TAAG). *The Journal of Chemical Physics*, 156(18). 184702.

Kulichenko, M.; Barros, K.; Lubbers, N.; Li, Y. W.; Messerly, R.; Tretiak, S.; Smith, J.; and Nebgen, B. 2023. Uncertainty-driven dynamics for active learning of interatomic potentials. *Nature Computational Science*, 3.

Kwon, Y.; Won, J.-H.; Kim, B. J.; and Paik, M. C. 2018. Uncertainty quantification using Bayesian neural networks in classification: Application to ischemic stroke lesion segmentation. In *Medical Imaging with Deep Learning*.

Ma, Y.-A.; Chen, T.; and Fox, E. 2015. A Complete Recipe for Stochastic Gradient MCMC. In *Advances in Neural Information Processing Systems*, volume 28.

Maddox, W. J.; Izmailov, P.; Garipov, T.; Vetrov, D. P.; and Wilson, A. G. 2019. A Simple Baseline for Bayesian Uncertainty in Deep Learning. In *Advances in Neural Information Processing Systems*, volume 32.

Martyna, G. J.; Tuckerman, M. E.; Tobias, D. J.; and Klein, M. L. 1996. Explicit reversible integrators for extended systems dynamics. *Molecular Physics*, 87(5): 1117–1157.

Podryabinkin, E. V.; and Shapeev, A. V. 2017. Active learning of linearly parametrized interatomic potentials. *Computational Materials Science*, 140: 171–180.

Podryabinkin, E. V.; Tikhonov, E. V.; Shapeev, A. V.; and Oganov, A. R. 2019. Accelerating crystal structure prediction by machine-learning interatomic potentials with active learning. *Phys. Rev. B*, 99: 064114.

Qiao, Z.; Christensen, A. S.; Welborn, M.; Manby, F. R.; Anandkumar, A.; and Miller, T. F. 2022. Informing geometric deep learning with electronic interactions to accelerate quantum chemistry. *Proceedings of the National Academy of Sciences*, 119(31): e2205221119.

Rensmeyer, T.; Craig, B.; Kramer, D.; and Niggemann, O. 2023. High Accuracy Uncertainty-Aware Interatomic Force Modeling with Equivariant Bayesian Neural Networks. *arXiv*.

Schütt, K.; Chmiela, S.; von Lilienfeld, A.; Tkatchenko, A.; Tsuda, K.; and Müller, K.-R. 2020. *Machine Learning Meets Quantum Physics*. ISBN 978-3-030-40244-0.

Schütt, K. T.; Sauceda, H. E.; Kindermans, P.-J.; Tkatchenko, A.; and Müller, K.-R. 2018. SchNet – A deep learning architecture for molecules and materials. *The Journal of Chemical Physics*, 148(24): 241722.

Schütt, K. T.; Unke, O. T.; and Gastegger, M. 2021. Equivariant message passing for the prediction of tensorial properties and molecular spectra. In *International Conference on Machine Learning*.

Shwartz-Ziv, R.; Goldblum, M.; Souri, H.; Kapoor, S.; Zhu, C.; LeCun, Y.; and Wilson, A. G. 2022. Pre-Train Your Loss: Easy Bayesian Transfer Learning with Informative Prior. In *First Workshop on Pre-training: Perspectives, Pitfalls, and Paths Forward at ICML*.

Smith, J. S.; Isayev, O.; and Roitberg, A. E. 2017. ANI-1: an extensible neural network potential with DFT accuracy at force field computational cost. *Chem. Sci.*, 8: 3192–3203.

Smith, J. S.; Nebgen, B.; Lubbers, N.; Isayev, O.; and Roitberg, A. E. 2018. Less is more: Sampling chemical space with active learning. *The Journal of Chemical Physics*, 148(24). 241733.

Smith, J. S.; Nebgen, B. T.; Zubatyuk, R. I.; Lubbers, N.; Devereux, C.; Barros, K.; Tretiak, S.; Isayev, O.; and Roitberg, A. E. 2019. Approaching coupled cluster accuracy with a general-purpose neural network potential through transfer learning. *Nature Communications*, 10.

Spiegelman, F.; Tarrat, N.; Cuny, J.; Dontot, L.; Posenitskiy, E.; Martí, C.; Simon, A.; and Rapacioli, M. 2020. Density-functional tight-binding: basic concepts and applications to molecules and clusters. *Advances in Physics: X*, 5(1): 1710252. PMID: 33154977.

Torrie, G.; and Valleau, J. 1977. Nonphysical sampling distributions in Monte Carlo free-energy estimation: Umbrella sampling. *Journal of Computational Physics*, 23(2): 187–199.

Unke, O. T.; Chmiela, S.; Gastegger, M.; Schütt, K. T.; Sauceda, H. E.; and Müller, K.-R. 2021. SpookyNet: Learning force fields with electronic degrees of freedom and nonlocal effects. *Nature Communications*, 12.

Vandermause, J.; Torrisi, S. B.; Batzner, S. L.; Xie, Y.; Sun, L.; Kolpak, A. M.; and Kozinsky, B. 2019. On-the-fly active learning of interpretable Bayesian force fields for atomistic rare events. *npj Computational Materials*, 6: 1–11.

Welling, M.; and Teh, Y. W. 2011. Bayesian Learning via Stochastic Gradient Langevin Dynamics. In *International Conference on Machine Learning*.

Wen, M.; and Tadmor, E. B. 2020. Uncertainty quantification in molecular simulations with dropout neural network potentials. *npj Computational Mathematics*, 6: 124.

Yao, J.; Pan, W.; Ghosh, S.; and Doshi-Velez, F. 2019. Quality of Uncertainty Quantification for Bayesian Neural Network Inference. In *International Conference on Machine Learning: Workshop on Uncertainty & Robustness in Deep Learning (ICML)*.

Zaverkin, V.; Holzmüller, D.; Bonfirraro, L.; and Kästner, J. 2023. Transfer learning for chemically accurate interatomic neural network potentials. *Phys. Chem. Chem. Phys.*, 25: 5383–5396.

Implementation Details

The Neural Network Architecture

For the base neural network architecture we use a NequIP model with four interaction blocks, a latent dimension of 64 and even and odd parity features up to and including angular momentum number l=2. The standard deviations σ_i are predicted by a three-layer MLP with input dimension 64, latent dimensions 32 and 16 and output dimension 1. SiLU activation functions are used for the latent layers and the output activation function is the exponential function. We do not normalize the force values and instead rescale the predicted means by the root mean square of the forces in the training dataset. The predicted standard deviations are not rescaled.

Generating Samples from the Posterior

For sampling the Bayesian posterior, we use the SGHMC algorithm (Chen, Fox, and Guestrin 2014) with the adaptive mass term introduced in (Rensmeyer et al. 2023). For the ethanol and paracetamol test cases, the step size γ is exponentially decreased from 10^{-2} and $0.3 \cdot 10^{-2}$ to 10^{-5} during the first 10^6 steps for the baseline model and transfer learning model respectively. At the end of this phase, the first model is sampled. Afterward, the cyclical learning rate schedule used in (Rensmeyer et al. 2023):

$$\gamma_i = \frac{\gamma_0}{2}\left(\cos\left(\pi + \frac{i \cdot \pi}{K}\right) + 1\right)$$

with $\gamma_0 = 0.001$ and cycle length $K = 50000$
is employed to sample the subsequent models from the same Markov chain at the end of each cycle.
The same procedure is also utilized for the baseline model on the stachyose test case, however, the initial convergence phase is shortened to $0.5 \cdot 10^6$ steps for the transfer learning model, as the other two test cases had revealed a quicker convergence for the transfer learning models. For the paracetamol and ethanol cases, a batch size of 30 is used and for the stachyose case, it is set as 15. After the first 90 percent of the initial convergence phase, the mass term is kept constant to ensure close convergence to the posterior.

Pretraining the Models

To pre-train a model, we converge it to a local maximum of the log-posterior on the pre-training dataset with a Gaussian mean field prior $p(\boldsymbol{\theta}) \sim N(\mathbf{0}, I)$. Almost the same sampling algorithm and hyperparameters are used as in the sampling of posterior of the corresponding baseline model. The only differences are, that the injected noise is downscaled by a factor of 0.1 and only the first model is sampled. The injected noise was not set to zero, because we found that a small amount of injected noise actually speeds up convergence, especially at the beginning of the optimization.

The Datasets

The Ethanol Transfer Learning Datasets

To pre-train the model 5000 randomly sampled configurations from the MD17 ethanol dataset are used. This dataset consists of over 500000 configurations generated from a molecular dynamics trajectory calculated at DFT level accuracy. We use the training and test datasets of ethanol at CCSD(T) level accuracy introduced in (Bogojeski et al. 2019) for the transfer learning task. We use the last 10 configurations of the training set as validation data. The actual training data consisted of the first $m \in \mathbb{N}$ configurations of the training dataset for varying values of m.

The Paracetamol Transfer Learning Datasets

The pretraining dataset consists of randomly sampled configurations from the aspirin, benzene, malonaldehyde, toluene, salicylic acid, naphthalene, ethanol, uracil and azobenzene from the MD17 dataset as well as the AT-AT DNA base pair, stachyose, Ac-Ala3-NHMe and docosahexaenoic acid datasets from the MD22 dataset. The first 100000 configurations from each MD17 dataset and all configurations from the MD22 datasets were used to form a pool of configurations from which 100000 are randomly drawn as the pretraining dataset.
For the actual training set $m \in \mathbb{N}$ configurations are randomly sampled from the MD17 paracetamol dataset for varying values of m. 10 additional configurations are randomly sampled as a validation set. The rest of the 106490 configurations are used as a test set.

The Stachyose Transfer Learning Datasets

The pretraining dataset was generated from a long molecular dynamics trajectory of a stachyose molecule in DFTB+(Hourahine et al. 2020). The initial geometry was generated from a structural relaxation with a convergence criterium of $10^{-3} H/\text{Å}$ for the maximal force component. The MD trajectory was simulated at 1 femtosecond time steps with a Nose Hoover thermostat (Martyna et al. 1996) at 600° Kelvin with a coupling strength of 3200 cm^{-1}. The simulation ran for 10^6 time steps using the velocity verlet driver with one configuration sampled every ten time steps yielding a dataset of 100000 configurations. For both

the geometry optimization as well as the MD simulation a Hamiltonian with self-consistent charges (Elstner et al. 1998) and third-order corrections (Gaus, Cui, and Elstner 2011) was used in correspondence with the 3ob-3-1 Slater Coster files (Gaus et al. 2014). For all atoms, s- and p-orbitals were used in the Hamiltonian.

For the actual training set $m \in \mathbb{N}$ configurations are randomly sampled from the first 10000 configurations of the MD22 stachyose dataset for varying values of m. 10 additional configurations are randomly sampled from the configurations 10100 to 10900 as a validation set. Configurations 11000 up to 27000 are used as a test set.