SchNetPack 3.0: A Neural Network Toolbox for Predictive and Generative Atomistic ML

N.W.A. Gebauer[1,2], S.S.P. Hessmann[1,2], J. Lederer[1,2], S. Gugler[1,2], K. Kahouli[1,2], K.-R. Müller[1,2,3,4]
[1]Machine Learning Group, Technische Universität Berlin, Berlin, Germany
[2]Berlin Institute for the Foundations of Learning and Data (BIFOLD), Berlin, Germany
[3]Department of Artificial Intelligence, Korea University, Seoul, Korea
[4]Max-Planck Institute for Informatics, Saarbrücken, Germany
niklas.gebauer@tu-berlin.de

The field of machine learning for atomistic systems has rapidly advanced in the past decade. SchNetPack [1,2] has established as a versatile software package that addresses both the requirements of method development and the application of atomistic machine learning. It is based on PyTorch and contains basic building blocks of atomistic neural networks, manages their training, and provides simple access to common benchmark datasets.

Besides enabling to predict potential energy surfaces and other quantum-chemical properties of molecules and materials, the package also provides a PyTorch implementation of molecular dynamics and interfaces to ASE [3] and LAMMPS [4]. An optional integration with PyTorch Lightning and the Hydra configuration framework powers a flexible command-line interface. Due to its modular design, SchNetPack is easily extendable with a custom code and ready for complex training tasks, as demonstrated in the schnetpack_gschnet extension for autoregressive generation of 3D molecules [5].

The package is under constant development and here we provide an update on its current status and the next updates. In recent years, the discovery of novel structures has become increasingly popular, with two main pathways emerging: active learning and generative models. With version 3.0, we adapt SchNetPack to these needs by providing basic tools for active learning, such as ensemble calculators and uncertainty estimation [6], as well as diffusion-based generative modeling capabilities. Alongside standard diffusion models, this includes MoreRed [7] for molecular relaxation, the novel generative pseudo-force field introduced at this workshop, and an accompanying suite of sampling algorithms to choose from. Additionally, we plan to roll-out pre-trained models for both property prediction and generative modeling tasks.

References

[1] Schütt, K. T., Kessel, P., Gastegger, M., Nicoli, K. A., Tkatchenko, A., & Müller, K. R. (2018). SchNetPack: A deep learning toolbox for atomistic systems. *Journal of chemical theory and computation*, *15*(1), 448-455.

[2] Schütt, K. T., Hessmann, S. S., Gebauer, N. W., Lederer, J., & Gastegger, M. (2023). SchNetPack 2.0: A neural network toolbox for atomistic machine learning. *The Journal of Chemical Physics*, *158*(14).

[3] Hjorth Larsen, A., Jørgen Mortensen, J., Blomqvist, J., Castelli, I. E., Christensen, R., Dułak, M., ... & Jacobsen, K. W. (2017). The atomic simulation environment—a Python library for working with atoms. *Journal of Physics: Condensed Matter*, *29*(27), 273002.

[4] Thompson, A. P., Aktulga, H. M., Berger, R., Bolintineanu, D. S., Brown, W. M., Crozier, P. S., ... & Plimpton, S. J. (2022). LAMMPS-a flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales. *Computer physics communications*, *271*, 108171.

[5] Gebauer, N. W., Gastegger, M., Hessmann, S. S., Müller, K. R., & Schütt, K. T. (2022). Inverse design of 3d molecular structures with conditional generative neural networks. *Nature communications*, *13*(1), 973.

[6] Hessmann, S. S., Schütt, K. T., Gebauer, N. W., Gastegger, M., Oguchi, T., & Yamashita, T. (2025). Accelerating crystal structure search through active learning with neural networks for rapid relaxations. *npj Computational Materials*, *11*(1), 44.

[7] Kahouli, K., Hessmann, S. S. P., Müller, K. R., Nakajima, S., Gugler, S., & Gebauer, N. W. A. (2024). Molecular relaxation by reverse diffusion with time step prediction. *Machine Learning: Science and Technology*, *5*(3), 035038.

Figures

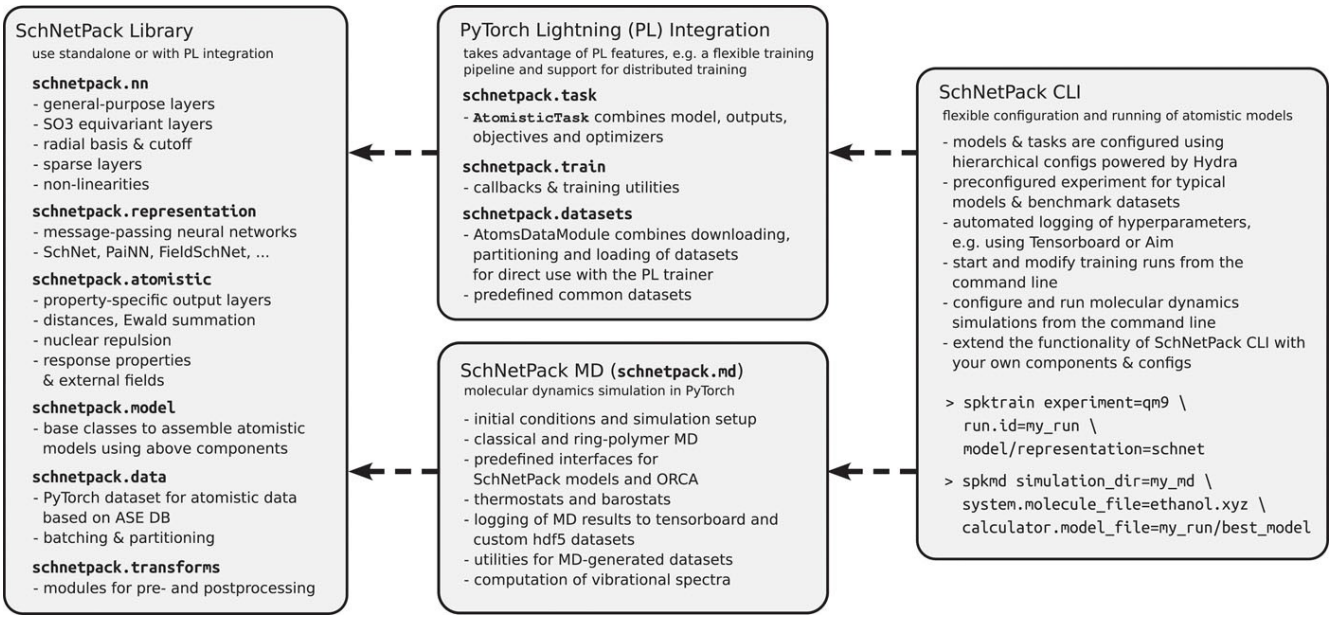

Figure 1. Overview of the four major components of the current SchNetPack 2.0 toolbox: the atomistic neural network library, PyTorch Lightning integration, command-line interface, and molecular dynamics code. The arrows indicate dependencies between the components, i.e., components that can be used independently of components on their right.