The Journal of Open Source Software

UnlockNN: Uncertainty quantification for neural network models of chemical systems

Alexander Moriarty [1], **Kazuki Morita** [1], **Keith T. Butler** [2], **and Aron Walsh** [1,3]

1 Department of Materials, Imperial College London, London, UK **2** SciML, STFC Scientific Computing Division, Rutherford Appleton Laboratories, UK **3** Department of Materials Science and Engineering, Yonsei University, Seoul, Korea

DOI: 10.21105/joss.03700

Software

- Review
- Repository
- Archive

Editor: Øystein Sørensen

Reviewers:

- @TahiriNadia
- @pmeier
- @Het-Shah

Submitted: 03 September 2021
Published: 05 July 2022

Summary

Machine learning models have gained interest from materials researchers for their ability to predict materials' properties accurately and faster than first-principles calculations based on physical laws, particularly for complex systems with many possible configurations (Butler et al., 2018; Ramprasad et al., 2017; Xue et al., 2016). In particular, neural networks (NNs) have been shown to achieve predictive accuracies within the threshold of "chemical accuracy" (Chen et al., 2019; Faber et al., 2017; Schütt et al., 2018).

However, most current implementations of NNs for materials property prediction lack *uncertainty quantification*, a measure of the confidence of a prediction. This is especially detrimental to a machine learning model, as its reliability is contingent upon the existence of "similar" materials in the training data set. To the end user, there is no easy way to tell whether this is the case.

Statement of need

UnlockNN provides an API to add uncertainty quantification to Keras-based models and comes packaged with a specific implementation for compatibility with MEGNet (Chen et al., 2019), which is a graph NN implementation for materials property prediction that has achieved state-of-the-art accuracy on many benchmark tasks (Dunn et al., 2020). The package is designed for materials and chemistry researchers to improve their models' reliability and identify the domain(s) of materials on which the models can perform well.

This uncertainty quantification is achieved by supplanting the output layer of the model with a variational Gaussian process (VGP) (Dillon et al., 2017; Hensman et al., 2013): a modification of a Gaussian process (GP) that allows for scalability to large data sets. Whilst a typical GP requires the entire training data set to be stored in memory and used for inference (an example of *instance-based* learning), the VGP infers a smaller set of inducing index points. The locations of these inducing index points are optimized during training to minimise the Kullback-Leibler divergence between the GP based on *all* training data and the VGP.

Once created, the probabilistic model must be trained in order to optimize the locations of the VGP's inducing index points and its kernel parameters. However, the number of training iterations required is typically only a small fraction of the training iterations needed to train the base NN it is modifying.

The primary interface for unlockNN is the `model` module, which contains an extensible `ProbNN` class for adding uncertainty quantification to arbitrary Keras models. It also contains a `MEGNetProbModel` class for adding uncertainty quantification to MEGNet, and a download suite for accessing example models and data. The models can be configured with different kernels for the VGP, which are implemented in a `kernel_layers` module.

Neural network-fed VGPs share a similar principle to the convolution-fed Gaussian processes formulated by Tran et al. (2020). UnlockNN also implements tools for calculating the performance metrics suggested by Tran et al. (2020), including sharpness and calibration error, via its `metrics` module.

Acknowledgements

This project was proposed by Keith Butler and has benefited hugely from his support, as well as the support of Aron Walsh and Kazuki Morita. Thanks also to the Royal Society for funding this project and the Science and Technology Facilities Council for providing access to their computing cluster, which was used extensively for testing during development.

References

Butler, K. T., Davies, D. W., Cartwright, H., Isayev, O., & Walsh, A. (2018). Machine learning for molecular and materials science. *Nature*, *559*(7715), 547–555. https://doi.org/10.1038/s41586-018-0337-2

Chen, C., Ye, W., Zuo, Y., Zheng, C., & Ong, S. P. (2019). Graph Networks as a Universal Machine Learning Framework for Molecules and Crystals. *Chemistry of Materials*, *31*(9), 3564–3572. https://doi.org/10.1021/acs.chemmater.9b01294

Dillon, J. V., Langmore, I., Tran, D., Brevdo, E., Vasudevan, S., Moore, D., Patton, B., Alemi, A., Hoffman, M., & Saurous, R. A. (2017, November 28). *TensorFlow Distributions*. http://arxiv.org/abs/1711.10604

Dunn, A., Wang, Q., Ganose, A., Dopp, D., & Jain, A. (2020). Benchmarking materials property prediction methods: The Matbench test set and Automatminer reference algorithm. *Npj Computational Materials*, *6*(1, 1), 1–10. https://doi.org/10.1038/s41524-020-00406-3

Faber, F. A., Hutchison, L., Huang, B., Gilmer, J., Schoenholz, S. S., Dahl, G. E., Vinyals, O., Kearnes, S., Riley, P. F., & von Lilienfeld, O. A. (2017). Prediction Errors of Molecular Machine Learning Models Lower than Hybrid DFT Error. *Journal of Chemical Theory and Computation*, *13*(11), 5255–5264. https://doi.org/10.1021/acs.jctc.7b00577

Hensman, J., Fusi, N., & Lawrence, N. D. (2013, September 26). *Gaussian Processes for Big Data*. http://arxiv.org/abs/1309.6835

Ramprasad, R., Batra, R., Pilania, G., Mannodi-Kanakkithodi, A., & Kim, C. (2017). Machine learning in materials informatics: Recent applications and prospects. *Npj Computational Materials*, *3*(1), 1–13. https://doi.org/10.1038/s41524-017-0056-5

Schütt, K. T., Sauceda, H. E., Kindermans, P.-J., Tkatchenko, A., & Müller, K.-R. (2018). SchNet – A deep learning architecture for molecules and materials. *The Journal of Chemical Physics*, *148*(24), 241722. https://doi.org/10.1063/1.5019779

Tran, K., Neiswanger, W., Yoon, J., Zhang, Q., Xing, E., & Ulissi, Z. W. (2020). Methods for comparing uncertainty quantifications for material property predictions. *Machine Learning: Science and Technology*, *1*(2), 025006. https://doi.org/10.1088/2632-2153/ab7e1a

Xue, D., Balachandran, P. V., Hogden, J., Theiler, J., Xue, D., & Lookman, T. (2016). Accelerated search for materials with targeted properties by adaptive design. *Nature Communications*, *7*(1), 11241. https://doi.org/10.1038/ncomms11241