Deep Learning for Molecules and Materials

Andrew D White[1]

[1]Department of Chemical Engineering, University of Rochester, Rochester, NY

The textbook is available at https://dmol.pub. The source is maintained online on GitHub at https://github.com/whitead/dmol-book; to provide feedback, suggestions, or help improve it, please visit the GitHub repository and participate via the issue tracker.

This version dated June 28, 2022

Abstract Deep learning is becoming a standard tool in chemistry and materials science. Although there are learning materials available for deep learning, none cover the applications in chemistry and materials science or the peculiarities of working with molecules. The textbook described here provides a systematic and applied introduction to the latest research in deep learning in chemistry and materials science. It covers the math fundamentals, the requisite machine learning, the common neural network architectures used today, and the details necessary to be a practitioner of deep learning. The textbook is a living document and will be updated as the rapidly changing deep learning field evolves.

***For correspondence:**
andrew.white@rochester.edu (ADW)

1 Introduction

Deep learning is becoming a standard tool in chemistry and materials science. Deep learning is specifically about connecting some input data (features) and output data (labels) with a neural network function. Neural networks are differentiable and able to approximate any function. The classic example is connecting a molecule's structure and function. A recent example is dramatically accelerating quantum calculations to the point that you can achieve DFT level accuracy with a neural network[1]. What makes deep learning especially relevant is that it's a powerful tool for approximating previously intractable functions **and** its ability to generate new data.

In this textbook, we view deep learning as a set of tools that enables us to create models that were previously infeasible with classical machine learning. What sets deep learning apart from classic machine learning is feature engineering. Much of the data-driven work in the past required decisions about what features are important and how to compute them from molecules. These are called descriptors. Deep learning is typically trained end-to-end, meaning decisions about which features are important are no longer relevant and we can work directly with molecular structures.

Another reason deep learning is a standard method is its mature tools. Previously, training and using models in machine learning was tedious because it required deriving and implementing new equations for each model. Deep learning has removed this need and model changes can be done nearly effortlessly. Deep learning is not a new paradigm of science or a replacement for a chemist. It's a tool that is mature and now ready for application in molecules and materials.

2 Prerequisites and Background Knowledge

The target audience of this book is students with a programming and chemistry background that are interested in building competency in deep learning. For example, PhD students or advanced undergraduates in chemistry or materials science with some Python programming skills will benefit from this book. Sections A and B provide a pedagogical introduction to the principles of machine learning, but only covering topics necessary for deep learning. For example, topics like decision trees and SVMs are not covered because they are not criti-

Received: 14 November 2021
Accepted: 12 June 2022

https://doi.org/10.33011/livecoms.3.1.1499
Living J. Comp. Mol. Sci. 2022, 3(1), 1499

cal to understanding deep learning. Section C covers deep learning principles and details on specific architectures, like the important graph neural network and variational autoencoder. Other chapters, like "Deep Learning on Sequences", give a survey-level overview of a much larger area targeted towards chemistry and materials science. Finally Section D gives more complex examples on authentic deep learning problems from chemistry and materials science. Each section states at the top the required background knowledge, but Python programming ability is assumed throughout.

3 Scope

No textbook can be truly comprehensive and mine is no exception. The following areas are important but did not meet my opinionated threshold for inclusion: active learning, feature optimization and exploration, few-shot learning, self-supervised learning, generative adversarial networks, and Bayesian neural networks. This may change, but I did not think they are yet significant enough to be necessary for an introduction. I would like to emphasize that this textbook is about *deep learning* only, and so it should not cover broader AI topics like reinforcement learning, derivative free optimization (e.g., genetic algorithms), or machine learning topics like random forests or support vector machines. These are also important areas but I wanted to narrow the focus to deep learning, which is where I believe the most significant breakthroughs have occurred. The data in this book are skewed more towards chemistry than materials science. Part of the reason is that deep learning is more mature in chemistry.

Deep learning is always a little tied up in the implementation details – it's hard to grasp without seeing code. Thus, framework choice can be a part of the learning process. This book assumes familiarity with Python and `numpy` and we use exclusively Python. For the deep learning framework, we use `Jax`, `Tensorflow`, `Keras`, and `scikit-learn` for different purposes. `Jax` is easy to learn because it's essentially `numpy` with automatic differentiation and GPU/TPU-acceleration. In this book, we use `Jax` when it's important to understand the implementation details and connect the equations to the code. `Keras` is a high-level framework that has many common deep learning features implemented. It is used when we would like to work with more complex models and I'm trying to show a more complete model. Of course, you can use `Jax` for complete models and show detailed implementations in `Keras`. This is just my reasoning for the choice of framework. `scikit-learn` is an ML package and thus we'll see in the early chapters on ML. Finally, `Tensorflow` is the underlying library of `Keras` so if we want to implement new layers in `Keras` we do it through `Tensorflow`. `TensorflowProbability` is an extension to `Tensorflow` that supports random variables and probability distributions used in our generative models. The most important framework left out of this book is `PyTorch`, which has recently taken the lead to be the most popular framework in deep learning research (not necessarily industry). Ultimately, this book presents the equations and implementation details so that you will learn concepts that are independent of the framework. You should thus be able to quickly pick up `PyTorch`, `MXNet`, or whatever the next new framework might be. Interestingly, `PyTorch` and `Tensorflow` are both adding more `jax`-like (functional) features and `TensorflowProbability` is changing its underlying engine to be `jax`, so it these framework differences are decreasing over time.

One of the most common mistakes I see from students is that they try to learn deep learning via web searching questions and reading documentation. *This is a terrible way to learn deep learning.* There is quite a bit of information out there, but you will end up with a distorted and framework-specific understanding of deep learning. Remember, a high-ranking search result may be relevant and popular, but that doesn't mean it will help you learn. More importantly, learning through blogs and Stack overflow makes it so hard to grasp the mathematics and intuition. Web searching and hacking together code is definitely a part of deep learning (for better or worse), but you should do this once you have a firm grasp of the math and details of the model you want to implement.

The table of contents as of publication is shown below:

https://doi.org/10.33011/livecoms.3.1.1499
Living J. Comp. Mol. Sci. 2022, 3(1), 1499

17. Generative RNN in Browser

The book has been online since August 2020 and received feedback and contributions from many people since then. A complete list can be found in the book itself.

4 Conclusion

This textbook is *living* because the field of deep learning moves quickly and a traditional textbook would be out of date by the time it is printed. The textbook is current and will be updated as the field grows and evolves. I hope this textbook provides a valuable benefit to the computational chemistry community and can be a starting point for graduate education and research in deep learning.

Funding Information

Research reported in this work was supported by the National Institute of General Medical Sciences of the National Institutes of Health under award number R35GM137966. This material is based upon work supported by the National Science Foundation under Grant No. 1764415.

Author Information

ORCID:

Andrew D White: 0000-0002-6647-3965

References

[1] **Smith JS**, Isayev O, Roitberg AE. ANI-1: an extensible neural network potential with DFT accuracy at force field computational cost. Chemical science. 2017; 8(4):3192–3203.

https://doi.org/10.33011/livecoms.3.1.1499
Living J. Comp. Mol. Sci. 2022, 3(1), 1499