

NNMD: программный пакет на Python для обучения нейронных сетей Бехлера—Парринелло

А.С. Будников, А.В. Романов

Воронежский государственный университет

Применение нейронных сетей для исследования свойств различных материалов или молекул является распространенной практикой в задачах молекулярно-динамического моделирования, например, в материаловедении и биохимии. В частности, нейронные сети позволяют определить специальную модели межатомных потенциалов – машинно-обучаемые потенциалы, или нейронную сеть Бехлера—Парринелло [1], которые могут быть обучены на экспериментальных данных без серьезных временных затрат на построение таких моделей, а затем реиспользованы возможностью дообучения под новые данные.

Нейронная сеть Бехлера—Парринелло (Behler—Parrinello Neural Network) является нейронной сетью высокой размерности, т.е. она состоит из подсетей, называемых атомными НС (AtomicNN), которые обучаются совместно. Как дескрипторы используются функции Бехлера—Парринелло, которые являются функциями от межатомного расстояния, что позволяет запоминать для каждого атома системы его окружение в пределах сферы с радиусом отсечки.

Для атомов одного химического элемента подсеть будет одинаковой, поэтому количество подсетей определяется количеством химических элементов в рассматриваемой системе. Нейронная сеть такой структуры способна вычислять энергию системы через алгоритм прямое распространения, силу через алгоритм обратного распространения ошибки, так как сила – градиент энергии. В качестве функции ошибки выступает взвешенная функция среднеквадратичной ошибки, где веса регулируют вклад ошибки по энергии и силам.

NNMD (Neural Networks for Molecular Dynamics) [2, 3] реализован с помощью языка программирования Python, а также фреймворка машинного обучения PyTorch, который также используется при вычислении значений атомно-центрированных симметричных функций Бехлера—Парринелло из-за возможности векторизации алгоритма вычисления дескрипторов. В пакете реализован интерфейс для калькулятора ASE под молекулярно-динамические эксперименты с возможностью использования предобученной сети при работе симулятора.

Программный пакет разбит на несколько модулей: io (input/output), nn (нейронные сети и связанные с ними классы), features (для вычисления дескрипторов), md (имплементация калькулятора для ASE). Благодаря модульной структуре пользователь может использовать только необходимые инструменты без необходимости взаимодействовать со всеми компонентами пакета, а также расширять и адаптировать код под конкретные нужды и задачи.

В дальнейшем планируется доработать проект таким образом, чтобы пакет смог отрабатывать весь цикл обучения и применения сетей в реальных экспериментах с возможностью масштабирования результатов. Также при невозможности повышения производительности и точности через операции PyTorch вычисления будут перенесены на язык C++ с интеграцией кода на нем в пакет посредством биндинга Python/C++, что может значительно упростить вычислительную часть задачи, а также потенциально ускорить за счет применения параллельных технологий, например, CUDA.

Литература

1. Behler J. Atom-centered symmetry functions for constructing high-dimensional neural network potentials // The Journal of Chemical Physics. 2011. Vol. 134, no. 3. DOI: 10.1063/1.3553717.
2. Проект NNMD. URL: <https://github.com/ded-otshel'nik/nnm-d> (дата обращения: 01.04.2025)
3. Будников А.С., Романов А.В. GPU-реализация машинно-обучаемых потенциалов для молекулярной динамики // Труды молодых ученых факультета компьютерных наук Воронежского государственного университета. Т. 4. Воронеж: 2024. С. 26–33.