УДК 004.89

ФОРМАЛЬНЫЕ МЕТОДЫ ПРОВЕРКИ КОРРЕКТНОСТИ БИБЛИОТЕК МАШИННОГО ОБУЧЕНИЯ

Мельников Е.Н., студент гр. 054, II курс
Томский государственный университет, г. Томск

В условиях стремительного роста масштабов и сложности систем машинного обучения, а также их применения в ответственных и критически важных областях, таких как медицина, финансовый сектор и автономные системы, обеспечение корректности программных компонентов становится задачей первостепенной важности. Традиционные методы тестирования и отладки часто оказываются недостаточными для выявления тонких ошибок или некорректного поведения в сложных алгоритмах обучения и инференса. В этом контексте формальные методы проверки корректности представляют собой эффективный инструмент, позволяющий обеспечить доказательство соответствия реализации библиотек машинного обучения их спецификациям. В настоящей статье рассматриваются основные подходы и вызовы, связанные с применением формальных методов к разработке и верификации ML-библиотек [1].

Формальная проверка корректности подразумевает построение математически строгой модели поведения программного компонента и доказательство того, что реализация удовлетворяет заданным свойствам, выраженным в виде логических формул, инвариантов или контрактов. Для библиотек машинного обучения это означает формализацию алгоритмов оптимизации, функций активации, вычисления градиентов, а также процедур обработки данных и управления памятью. Применение таких методов позволяет обнаружить ошибки, недоступные традиционному тестированию, включая ситуации с неявными предположениями, некорректной обработкой граничных условий и параллельными вычислениями.

Среди наиболее распространённых инструментов формальной верификации выделяются системы автоматического доказательства теорем (например, Coq, Isabelle, Lean), а также методы model checking и статического анализа. Интеграция этих инструментов с процессом разработки библиотек машинного обучения требует адаптации к особенностям вычислительных графов, большим объёмам данных и стохастической природе алгоритмов обучения. Для этого используются подходы модульного верифицирования, когда отдельные компоненты библиотеки проверяются изолированно, а затем

их свойства композиционно объединяются для оценки корректности всей системы [2].

Особое внимание уделяется формализации алгоритмов дифференцируемого программирования и градиентного спуска, поскольку именно эти методы лежат в основе большинства современных моделей. Верификация корректности вычисления производных и обновления параметров позволяет гарантировать стабильность и предсказуемость обучения, а также исключить критические ошибки, способные привести к некорректным моделям или сбоям. Кроме того, проверка формальных свойств памяти и потокобезопасности обеспечивает надёжность в условиях параллельных и распределённых вычислений [3].

Не менее важным аспектом является верификация интерфейсов и взаимодействия с внешними библиотеками и аппаратными ускорителями. Для поддержания корректности в сложных экосистемах ML-библиотек необходимо разрабатывать формальные спецификации API и протоколов обмена данными, а также проводить их согласованную верификацию с учётом особенностей аппаратной архитектуры и драйверов [4].

Несмотря на высокую полезность, применение формальных методов сталкивается с рядом вызовов, включая высокую трудоёмкость и необходимость значительных ресурсов на построение формальных моделей и доказательств. Для снижения этих барьеров разрабатываются автоматизированные и полуавтоматические инструменты, а также методологии, сочетающие формальную верификацию с традиционным тестированием и динамическим анализом.

В заключение, формальные методы проверки корректности библиотек машинного обучения представляют собой критически важный элемент инженерии надёжных и безопасных ИИ-систем. Их применение способствует повышению качества кода, снижению рисков неожиданных ошибок и формированию доверия к результатам обучения и инференса. В перспективе развитие формальных методов и их интеграция в жизненный цикл разработки ML-библиотек станет фундаментом для создания сертифицируемых, масштабируемых и этически устойчивых систем искусственного интеллекта [5].

Список литературы:

1. Hoare, C. A. R. An Axiomatic Basis for Computer Programming / C. A. R. Hoare // Communications of the ACM. – 1969. – Vol. 12, № 10. – P. 576–580. – DOI 10.1145/363235.363259.

2. Clarke, E. M. Model Checking / E. M. Clarke, O. Grumberg, D. A. Peled. – Cambridge : MIT Press, 1999. – 314 p.

3. Katz, G. Reluplex: An Efficient SMT Solver for Verifying Deep Neural Networks / G. Katz, C. Barrett, D. Dill [et al.] // Computer Aided Verification. – Cham : Springer, 2017. – P. 97–117. – DOI 10.1007/978-3-319-63387-9_5.

4. Gehr, T. AI2: Safety and Robustness Certification of Neural Networks with Abstract Interpretation / T. Gehr, M. Mirman, D. Drachsler-Cohen [et al.] // IEEE Symposium on Security and Privacy. – 2018. – P. 3–18. – DOI 10.1109/SP.2018.00058.

5. Singh, G. An Abstract Domain for Certifying Neural Networks / G. Singh, T. Gehr, M. Mirman [et al.] // Proceedings of the ACM on Programming Languages. – 2019. – Vol. 3, POPL. – P. 1–30. – DOI 10.1145/3290354.