УДК 004.89

ОБЕСПЕЧЕНИЕ ВОСПРОИЗВОДИМОСТИ ВЫВОДОВ ПРЕДОБУЧЕННЫХ МОДЕЛЕЙ С ПОМОЩЬЮ ДОВЕРЕННЫХ СРЕД

Горошко П.Н., Оператор научной роты, III курс
Военная академия связи имени Маршала Советского Союза С. М. Буденного,
г. Санкт-Петербург

В эпоху повсеместного внедрения предобученных моделей машинного обучения в прикладные и критически важные сферы – от финансов и медицины до автоматизации юридических процедур и управления городскими инфраструктурами – проблема воспроизводимости выводов приобретает фундаментальное значение. Предобученные модели, такие как трансформеры, крупные сверточные нейросети и мультимодальные архитектуры, работают в условиях, где даже незначительные отклонения в окружении, версиях библиотек, конфигурации вычислительных систем или случайных параметрах могут приводить к расхождениям в предсказаниях. Для задач, в которых на основе вывода модели принимаются управленческие, диагностические или юридически значимые решения, такая нестабильность становится неприемлемой. Именно поэтому всё большее внимание уделяется обеспечению воспроизводимости выводов предобученных моделей с помощью доверенных сред исполнения [1].

Под воспроизводимостью вывода в контексте предобученных моделей понимается возможность гарантированно получить одинаковый результат на идентичном входе, при соблюдении заданных условий вычислений. Это не просто вопрос детерминизма на уровне псевдослучайных генераторов – он включает в себя целый спектр факторов: версионность зависимостей, точность математических операций, особенности графических процессоров (GPU/TPU), аппаратные оптимизации, использование сторонних API, и даже особенности компиляции ядра операционной системы. В случае предобученных моделей, особенно больших языковых моделей (LLM), даже малейшие сдвиги в seed, конфигурации слоя dropout или механизме генерации (например, sampling vs greedy decoding) могут приводить к существенно различающимся результатам. Визуальные и аудиальные модели также чувствительны к порядку предобработки, масштабированию входа, битовой глубине и форматам кодирования.

В реальных условиях ИИ-модель часто проходит через цепочку: предобучение (обычно за пределами организации), дообучение или адаптация к задаче, деплой в конкретную инфраструктуру и, наконец, эксплуатация с постоянным потоком данных. На каждом из этих этапов может произойти

рассогласование между ожидаемым и фактическим поведением модели, особенно если не задействованы механизмы контроля среды исполнения.

Поэтому решение задачи воспроизводимости требует создания доверенной среды, которая включает в себя комплекс технологических, криптографических и процедурных инструментов, фиксирующих и стабилизирующих все аспекты вычислений [2].

Один из наиболее распространённых подходов – использование контейнеризованных сред, таких как Docker, Podman или Singularity, которые позволяют замораживать все зависимости, включая ОС, библиотеки, драйверы и конфигурации фреймворков (TensorFlow, PyTorch, HuggingFace). Однако контейнеризация сама по себе не гарантирует воспроизводимость на уровне битовой точности вывода – для этого необходима дополнительно фиксация версий компиляторов и опций вычислений (например, параметров BLAS, MKL, CUDA), а также детерминизация источников случайности. Современные фреймворки начинают поддерживать режимы "deterministic execution", однако на практике они требуют жёсткой дисциплины на всех этапах деплоя и исполнения.

Следующий уровень – это системы доверенного выполнения (Trusted Execution Environments, TEE), например, Intel SGX, AMD SEV или Arm TrustZone, которые позволяют запускать модель в аппаратно защищённой зоне, изолированной от основной операционной системы. Это не только гарантирует безопасность и приватность, но и позволяет достоверно фиксировать поведение модели – включая контроль за внешними воздействиями. В сочетании с механизмами хеширования модели и входа это даёт возможность доказательно воспроизвести вывод, т.е. убедиться, что на данном входе при заданных параметрах модель выдала именно тот результат, который заявлен.

Интересное направление развивается в области блокчейн-инфраструктур, где результаты выполнения моделей, особенно в публичных сервисах, могут быть зафиксированы в распределённом реестре, что исключает возможность их подмены или некорректного воспроизведения. Хотя такие методы пока ограничены в производительности, они позволяют строить репутационные механизмы доверия к выводу модели, особенно в B2B-сервисах и автоматизированных платформах принятия решений.

Дополнительную роль играют механизмы аудита и логирования, в том числе с применением стандартизированных форматов (например, ML Metadata, Model Cards, Data Sheets for Datasets), которые позволяют фиксировать всё окружение модели – от источника данных до конкретной версии API и состояния seed. Такая документация становится необходимым условием регуляторной совместимости и сертификации в юрисдикциях, где вывод ИИ может быть оспорен в правовом контексте.

На концептуальном уровне доверенная среда исполнения должна обладать следующими свойствами:

- Детерминированность: при одинаковом входе и окружении – всегда один и тот же результат.
- Трассируемость: каждый компонент исполнения должен быть идентифицирован и версионирован.
- Защищённость: модель и данные не могут быть подменены или модифицированы незаметно.
- Подотчётность: вся история взаимодействия и вывода логируется в воспроизводимом формате.
- Проверяемость: возможно внешнее верифицированное подтверждение корректности вывода.

Инфраструктура воспроизводимости также может быть интегрирована в платформы MLOps, в которых определяются пайплайны построения, тестирования и деплоя моделей. Здесь доверенные среды помогают не только верифицировать поведение модели, но и запускать её тесты в идентичных условиях, автоматически сравнивая отклонения от эталонного вывода. Это критично для CI/CD-циклов в чувствительных задачах – от банковских скорингов до онкологической диагностики [3].

Особенно перспективным направлением является воспроизводимость как сервис (Reproducibility-as-a-Service), где внешняя инфраструктура обеспечивает зафиксированную среду выполнения модели и предоставляет API, через которое можно воспроизводить вывод. Это позволяет организациям делегировать инфраструктурную сложность, сохраняя при этом уверенность в корректности и сопоставимости результатов [4].

Таким образом, обеспечение воспроизводимости выводов предобученных моделей с помощью доверенных сред – это не просто вопрос инженерного удобства, а необходимое условие для зрелого, ответственного и регламентируемого применения искусственного интеллекта [5].

В условиях растущей правовой ответственности, требований к прозрачности ИИ и всё более сложных архитектур моделей, только наличие технических и процедурных гарантий того, что вывод модели стабилен и проверяем, может служить основанием для её использования в реальных задачах, затрагивающих интересы людей, организаций и государств.

Список литературы:

1. Henderson, P. Deep Reinforcement Learning that Matters / P. Henderson, R. Islam, P. Bachman [et al.] // Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence. – Palo Alto : AAAI Press, 2018. – P. 3207–3214.

2. Pineau, J. Improving Reproducibility in Machine Learning Research / J. Pineau, P. Vincent-Lamarre, K. Sinha [et al.] // Journal of Machine Learning Research. – 2021. – Vol. 22, № 164. – P. 1–20.

3. Tatman, R. A Practical Taxonomy of Reproducibility for Machine Learning Research / R. Tatman, J. VanderPlas, S. Dane // ICML Workshop on Reproducibility in Machine Learning. – 2018.

4. Gundersen, O. E. State of the Art : Reproducibility in Artificial Intelligence / O. E. Gundersen, S. Kjensmo // Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence. – Palo Alto : AAAI Press, 2018. – P. 1644–1651.

5. Boettiger, C. An Introduction to Docker for Reproducible Research / C. Boettiger // ACM SIGOPS Operating Systems Review. – 2015. – Vol. 49, № 1. – P. 71–79. – DOI 10.1145/2723872.2723882.