УДК 004.85
doi: 10.15622/rcai.2025.051

ПЕРЕХОД С ОФФЛАЙН НА ОНЛАЙН ОБУЧЕНИЕ С ПОДКРЕПЛЕНИЕМ ТРАНСФОРМЕРНЫХ МОДЕЛЕЙ ДЛЯ РОБОТОТЕХНИЧЕСКОЙ МАНИПУЛЯЦИИ

Д.И. Поярков (*poiarkov.di@phystech.edu*)[A,B]
А.В. Староверов (*alstar8@yandex.ru*)[B,C]
А.И. Панов (*panov.ai@mipt.ru*)[A,B,C]

[A] Московский физико-технический институт, Долгопрудный
[B] Федеральный исследовательский центр «Информатика и управление» РАН, Москва
[C] AIRI, Москва

Трансформерные модели показывают выдающиеся результаты в роботизированной манипуляции, используя для этого обучение на больших оффлайн наборах данных, но нуждаются в онлайн-дообучении для повышения надёжности стратегии. В этом исследовании сравниваются способы совместить онлайн и оффлайн методы для повышения эффективности обучения. Эксперименты показывают, что оффлайн-предобучение при помощи RL достигает целевой производительности на 32% быстрее, что подчеркивает его потенциал для снижения размера выборки при сохранении преимуществ трансформерных моделей для роботизированного управления. Исходный код доступен по адресу https://gitlab.com/cogmod_mr/isaaclab_mod.

Ключевые слова: машинное обучение, искусственный интеллект, трансформер, оффлайн-обучение, онлайн-обучение, обучение с подкреплением.

Введение

Недавние достижения в роботизированной манипуляции в значительной степени обусловлены архитектурами на основе трансформеров, обученных с помощью клонирования поведения (BC) или оффлайн-обучения с подкреплением (RL) на больших демонстрационных наборах данных [Kim et al., 2025], [Chebotar et al., 2023]. Хотя эти подходы показали многообещающие результаты с показателями успеха 70-90% (включая имею-

щиеся в наборе данных задачи) [Staroverov et al., 2023], [Li et al., 2025], многие реальные приложения требуют более высокой надежности и устойчивости.

Многие RL-алгоритмы, в частности, Proximal Policy Optimization (PPO) [Schulman et al., 2017], способны достигать высоких результатов в задачах управления роботами. Однако, эти методы обычно требуют большого количества взаимодействий со средой для сходимости, что делает их вычислительно затратными и потенциально непрактичными для реальных приложений робототехники. Это поднимает важный вопрос: можем ли мы использовать эффективное по выборке оффлайн-предобучения для ускорения сходимости RL-алгоритмов?

Комбинация оффлайн- и онлайн-обучения успешно применялось в различных областях [Ramrakhya et al., 2023], [Baker et al., 2022], включая большие языковые модели [Minaee et al., 2024]. Однако применение этого подхода к роботизированной манипуляции с трансформерными моделями является особой проблемой. Трансформеры, хотя и очень эффективны во многих приложениях, имеют ряд ограничений в работе с частично наблюдаемыми Марковскими процессами (POMDP) [Lu et al., 2024], и их совместимость с онлайн-RL остается открытым вопросом.

В этой работе мы исследуем эффективность различных стратегий оффлайн-предобучения в ускорении сходимости PPO для роботизированной манипуляции при помощи трансформеров. В частности, мы сравниваем два подхода к предобучению: традиционное клонирование поведения (BC) и оффлайн-RL с использованием алгоритма ArCHer [Zhou et al., 2024]. Наши результаты показывают, что правильное предобучение может значительно сократить количество взаимодействий со средой, необходимых для достижения PPO высокой производительности, при этом предобучение ArCHer обеспечивает сходимость до 90% процентов успеха за 32% меньше итераций по сравнению с обучением с нуля.

Наш основной вклад:

- Систематическая оценка стратегий оффлайн-предобучения для роботизированной манипуляции на основе трансформеров, демонстрирующая, что предварительное обучение ArCHer обеспечивает на 32% более быструю сходимость к 90% успешности по сравнению с обучением с нуля.
- Унифицированная архитектура трансформерной модели, которая служит как актером, так и критиком, с использованием группировки действий и отдельными выходными сегментами для оценки стратегии, что обеспечивает эффективное обучение в задачах непрерывного управления.
- Структура для перехода между оффлайн- и онлайн-фазами обучения, которая поддерживает стабильность модели с помощью заморозки актора и плавного графика скорости обучения (learning rate).

- Реализация с открытым исходным кодом, полностью совместимая с фреймворком IsaacLab, что обеспечивает воспроизводимость и модифицируемость проведённых экспериментов (https://gitlab.com/cogmod_mr/isaaclab_mod).

1. Связанные работы

1.1. Робототехника с трансформерными моделями

Трансформерная архитектура [Vaswani et al., 2017] сегодня является основой для большинства систем робототехнической манипуляции. Недавние работы, такие как RT-2 [Brohan et al., 2023], OpenVLA [Kim et al., 2024] и Octo [Octo et al., 2024] продемонстрировали эффективность трансформерных архитектур в задачах манипуляции, достигая показателей успеха 70-90% (включая новые задачи и задачи в наборах данных). Эти системы в основном полагаются на клонирование поведения для обучения.

Недавняя теоретическая работа [Lu et al., 2024] вызвала обеспокоенность по поводу ограничений трансформерных моделей в решении POMDP.

1.2. Способы предобучения моделей для робототехнического управления

Текущие подходы к предобучению моделей роботизированной манипуляции можно разделить на две группы. Первая категория, включающая RT-1 [Brohan et al., 2023a], RT-2 and OpenVLA, фокусируется на работе “zero-shot”, используя клонирование поведения на разнообразных наборах данных [Collaboration et al., 2024], [Dalal et al., 2024]. Хотя эти методы эффективны для манипуляции общего назначения, они обычно достигают плато при 70-90% успеха.

Вторая категория, представленная PerAct [Shridhar et al., 2022] и ACT [Zhao et al., 2023], делает акцент на обучении, ориентированном на конкретные задачи, на ограниченном количестве демонстраций. Наша работа нацелена на объединение этих концепций, комбинируя оффлайн-предобучение с онлайн-RL.

Оффлайн-онлайн обучение в робототехнике. Недавние работы показали многообещающие результаты в объединении оффлайн- и онлайн-обучения для робототехнических задач [Ramrakhaya et al., 2023], [Baker et al., 2022]. Наш подход основывается на этих результатах, однако делает акцент на проблемы онлайн-RL с трансформерной архитектурой. Подобно ArCHer [Zhou et al., 2024], мы используем иерархический актор-критик, но адаптируем его для непрерывного управления в робототехнической манипуляции.

2. Метод

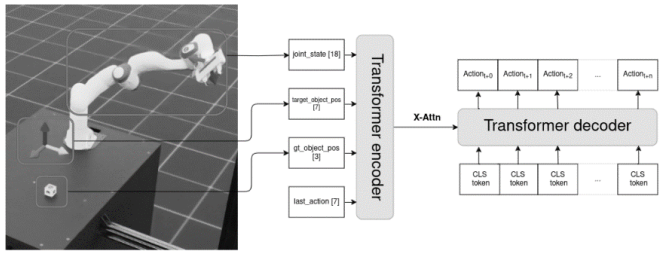

Рис. 1. Архитектура предлагаемой модели актора на основе трансформатора для непрерывного управления. Модель обрабатывает наблюдения за состоянием через энкодер и генерирует последовательность действий через декодер

Формулировка задачи

Мы формулируем задачу роботизированной манипуляции как марковский процесс принятия решений (MDP), определяемый набором , где представляет собой пространство состояний, – пространство действий, определяет динамику перехода, – функция вознаграждения, а – коэффициент дисконтирования.

Группировка действий (Action Chunking)

Вместо прогнозирования отдельных действий мы определяем группу действий длиной [Zhao et al., 2023] как последовательность , где каждое . Формально, для состояния в момент времени стратегия имеет на выходе:

,

где обозначает распределение вероятностей для действий.

Архитектура модели

Наша архитектура следует модели актор-критик, где оба компонента совместно используют общую трансформерную основу, но поддерживают отдельные выходные «головы».

Для непрерывного управления мы моделируем каждое действие в последовательности с помощью гауссовой стратегия. Получая на выходе модели вектор h , мы вычисляем среднее значение и стандартное отклонение для каждого действия:

.

Архитектура критика

Нейросеть критика использует ту же трансформерную основу, но использует разные выходные головы для оценки V и Q-значения. Для заданного состояния и последовательности действий оценки значений вычисляются следующим образом:

,

,

где представляет собой первый токен энкодера, а – последний токен декодера.

Стабилизация обновлений. В течение первых 100 итераций PPO мы замораживаем сеть актора, позволяя критику обучиться на новых онлайн-данных. Такое количество было взято как достаточное для адаптации критика к онлайн-данным.

Структура обучения

Наша структура обучения состоит из двух фаз: оффлайн-предобучение и онлайн-дообучение.

Мы исследуем два метода оффлайн-предобучения: ArCHer и клонирование поведения (BC).

Предобучение ArCHer. Алгоритм ArCHer работает как на уровне последовательностей, так и на уровне отдельных токенов. Цели обучения Q-функции и функции значения (value-функции) следующие:

.

Чтобы решить проблемы оффлайн-обучения, используются цели IQL и AWR:

.

,

где – функция потерь на основе экспектилей.

BC-предобучение. Для BC минимизируется среднеквадратичную ошибку между предсказанными и продемонстрированными действиями:

где – предсказанное действие, а – использованное на шаге действие.

Онлайн-дообучение. Для онлайн-обучение используется PPO, который оптимизирует обрезанную суррогатную цель:

clip ,

где ———— – отношение вероятностей, а – оценка преимущества.

Полная функция цели онлайн-обучения объединяет потери стратегия и value-функции:

где – это потеря value-функции, – это значение энтропии, а – весовые коэффициенты.

Комбинированное обучение. Полная процедура обучения состоит из двух фаз:

1. Оффлайн-фаза: модель предобучается с помощью ArCHer или BC на демонстрационном наборе данных на фиксированном количестве шагов.
2. Онлайн-фаза: дообучение модели с использованием PPO со следующими изменениями:
    - Сеть актора заморожена на первые 100 итераций, чтобы позволить критику адаптироваться.
    - Коэффициент скорости обучения (learning rate) постепенно возвращается к целевому значению.

3. Описание и настройка задачи

Чтобы оценить наш подход, мы рассматриваем задачу роботизированной манипуляции, которая охватывает ключевые непрерывного управления. Рассматривается задача поднятия куба из фреймворка IsaacLab [Mittal et al., 2023].

3.1. Описание среды

Задача формулируется как процесс принятия решений Маркова (MDP) со следующими компонентами:

- **Пространство состояний** : Пространство состояний включает конфигурацию сочленений 7-DOF манипулятора Franka, куба в качестве объекта и плоского рабочего пространства.
- **Пространство действий** : Действия определяются как , где представляет собой приращения положения сочленений, а задаёт бинарное состояние захвата.
- **Начальное состояние** : Эпизоды начинаются с манипулятора в фиксированной конфигурации. Начальное положение куба случайным образом выбирается из рабочей области на поверхности стола.

- **Динамика перехода** : Окружающая среда развивается в соответствии с физическим моделированием в IsaacLab, которое обрабатывает динамику твердого тела и контактные взаимодействия между манипулятором, кубом и столом.

3.2. Формула вознаграждения

Функция вознаграждения r направлена на поощрение успешного выполнения задачи и обеспечение стабильного управления. Она состоит из нескольких слагаемых: за приближение к объекту (**reach**), поднятие объекта (**lift**), достижение цели (**goal**), точное размещение (**precise**), а также штрафов за высокую скорость сочленений (**vel**) и резкие действия (**action**).

3.3. Пространство наблюдения

Вектор наблюдения содержит:

,

где

: Положения суставов

: Скорости суставов

: Положение объекта (истинное значение)

: Положение цели

: Предыдущее действие

Это пространство наблюдения было выбрано, чтобы предоставить достаточно информации как для имитационного обучения, так и для обучения с подкреплением, сохраняя при этом относительно низкую размерность, что способствует эффективному обучению.

3.4. Настройка оффлайн-обучения

3.6.1 Набор данных экспертных демонстраций.

Для оффлайн-предобучения мы собрали набор данных , содержащий приблизительно 10 000 экспертных демонстраций. Каждая демонстрация представляет собой набор, содержащий состояние, действие, вознаграждения и метку о завершении эпизода:

,

где шагов. Демонстрации были сгенерированы с использованием обученной модели, с траекториями, усеченными до 40 шагов, чтобы избежать проблем, связанных с горизонтом планирования для value-функции во время оффлайн-обучения.

4. Эксперименты

Наши эксперименты нацелены на сравнение трех стратегий обучения: (1) чистое онлайн-обучение с PPO (базовый эксперимент), (2) BC-предобучение и PPO-дообучение и (3) предобучение иерархическим RL с последующим PPO-дообучением.

4.1. Экспериментальная установка

Все эксперименты используют одинаковый трансформерный актор-критик, как описано в разделе 3. Для справедливого сравнения мы поддерживаем согласованные гиперпараметры во всех конфигурациях обучения:

- Оффлайн-предобучение: 12 000 шагов модели со скоростью обучения 6e-5.
- Онлайн-обучение: 5 000 итераций со скоростью обучения актера 4e-5 и скоростью обучения критика 2e-3.
- Количество шагов для вычисления процента успеха: 1 000 сред × 5 эпизодов (всего 5 000 эпизодов).

Для экспериментов с предварительно обученными моделями мы замораживаем актор на первые 100 итераций онлайн-обучения.

4.2. Исследование длины выходной последовательности

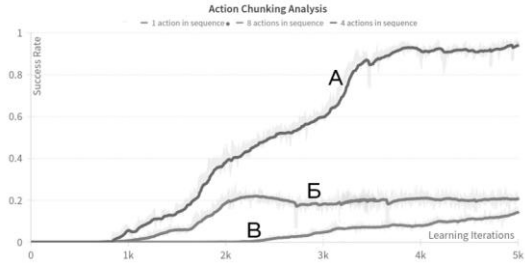

Рис. 2. Обучение моделей с разной длиной последовательности действий: 1 действие (Б), 4 действия (А), 8 действий (В)

Сперва мы исследуем влияние длины последовательности действий на эффективность обучения. Мы оцениваем длину последовательности из 1, 4 и 8 действий, используя базовый эксперимент с PPO. Результаты, показанные на рис. 2, показывают, что оптимальным числом действий для выбранной задачи является 4. Эта длина последовательности используется для всех последующих экспериментов.

4.3. Подходы к обучению

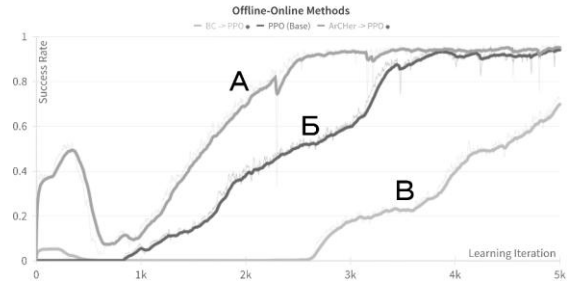

Рис. 3. Двухэтапные методы и базовый метод обучения.
PPO с нуля (Б), BC + дообучение PPO (В), Оффлайн-RL + дообучение PPO (А)

Мы оцениваем три различных подхода к обучению. Все они имеют общую конфигурацию для инициализации сети, общие гиперпараметры и настройки симулятора.

PPO с нуля (базовый). Модель инициализируется случайным образом и обучается с использованием PPO. Процент успеха для этого эксперимента показан на рис. 3.

BC + дообучение PPO. Модель предварительно обучена с использованием клонирования поведения на демонстрационном наборе данных, за которым следует PPO-дообучение. Критик инициализируется случайным образом в начале дообучения.

Оффлайн-RL + дообучение PPO. Модель предварительно обучена с использованием иерархического оффлайн-RL на демонстрационном наборе данных и дообучена при помощи PPO. И актор, и критик используют предобученные веса своих моделей.

5. Результаты

5.1. Производительность методов двухэтапного обучения

Эксперименты показывают, что переход между оффлайн- и онлайн-обучением создаёт видимый «провал» в производительности моделей в начале обучения. При этом, BC-предобучение приводит к ухудшению производительности в целом.

Иерархический оффлайн-RL тоже изначально испытывает падение процента успеха, но, используя знания, полученные во время предварительного обучения, восстанавливается и в итоге достигает 90% успеха примерно на **32% быстрее**, чем базовый метод.

Экспериментальные результаты наших установок обучения приведены в табл. 1. Для индикации производительности для каждой установки мы взяли момент, в который модель достигла успешности 90%, который мы обозначили как итерацию успеха 90%.

Таблица 1

Медианный процент успеха для оффлайн → онлайн методов, выраженный в номере итерации, на которой достигнут 90% процент успеха (меньше – лучше)

Метод	90% SI
Random → PPO	3700
BC → PPO	>5000
Offline RL → PPO	**2500**

5.2. Производительность методов оффлайн-обучения

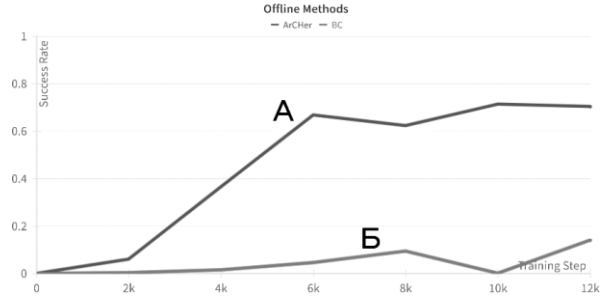

Рис. 4. Производительность оффлайн-предобучения: ArCHer (А) и BC (Б)

Производительность выбранных оффлайн алгоритмов с точки зрения показана в табл. 2 и демонстрирует прогресс по этапам обучения. Оффлайн обучение ArCHer показывает превосходную производительность по сравнению с простым BC.

Таблица 2

Успех оффлайн-методов после 12000 шагов обучения. Параметр сэмплинга показывает, применяется ли значение стандартного отклонения, взятое из модели, для рандомизации ее выходных данных

Метод	SR (с сэмплингом)	SR (без сэмплинга)
BC	14%	14%
ArCHer	70%	78%

Заключение

В работе исследуется объединение оффлайн-предобучения и онлайн-дообучения трансформеров для роботизированной манипуляции. Показано, что предварительное оффлайн-обучение с использованием RL ускоряет достижение целевой производительности на 32% по сравнению с обу-

чением с нуля, сохраняя высокую итоговую эффективность. Это подчеркивает значимость оффлайн-этапа для повышения эффективности онлайн-обучения.

В то же время выявлены сложности при переходе между фазами: простое клонирование поведения снижает качество обучения, что указывает на важность правильного задания функции цели. Полученные результаты особенно актуальны для робототехники, где критичны эффективность по выборке и надежность. Перспективным направлением является разработка более продуманных стратегий перехода и расширение подхода на более сложные задачи.

Список литературы

[Baker et al., 2022] Baker B., Akkaya I., Zhokhov P., Huizinga J., Tang J., Ecoffet A., Houghton B., Sampedro R., Clune J. Video pretraining (vpt): Learning to act by watching unlabeled online videos // 36th Conference on Neural Information Processing Systems (NeurIPS 2022). – 2022. – doi: 2206.11795.

[Brohan et al., 2023a] Brohan A. [et al.]. RT-2: Vision-Language-Action Models Transfer Web Knowledge to Robotic Control // Proceedings of The 7th Conference on Robot Learning. – PMLR, 2023. – doi: 10.48550/arXiv.2307.15818.

[Brohan et al., 2023b] Brohan A. [et al.] . RT-1: Robotics Transformer for Real-World Control at Scale // Robotics: Science and Systems XIX. Daegu, Republic of Korea, 2023. – doi: 10.15607/RSS.2023.XIX.025.

[Chebotar et al., 2023] Chebotar Y. [et al.]. Q-Transformer: Scalable Offline Reinforcement Learning via Autoregressive Q-Functions // Proceedings of The 7th Conference on Robot Learning. – PMLR, 2023. – doi: 10.48550/arXiv.2309.10150.

[Collaboration et al., 2024] Collaboration E. [et al.]. Open X-Embodiment: Robotic Learning Datasets and RT-X Models // Proceedings of the 2024 IEEE International Conference on Robotics and Automation (ICRA). – IEEE, 2024. – doi: 10.1109/ICRA57147.2024.10611477.

[Dalal et al., 2024] Dalal M. [et al.]. Neural MP: A Generalist Neural Motion Planner // arXiv. – 2024. – doi: 10.48550/arXiv.2409.05864.

[Kim et al., 2025] Kim M. [et al.]. OpenVLA: An Open-Source Vision-Language-Action Model // Proceedings of The 8th Conference on Robot Learning. – PMLR, 2025. – doi: 10.48550/arXiv.2406.09246.

[Li et al., 2025] Li X. [et al.]. Evaluating Real-World Robot Manipulation Policies in Simulation // Proceedings of The 8th Conference on Robot Learning. – PMLR, 2025. – doi: 10.48550/arXiv.2405.05941.

[Lu et al., 2024] Lu C. [et al.]. Rethinking Transformers in Solving POMDPs // Proceedings of the 41st International Conference on Machine Learning. – 2024. – doi: 10.48550/arXiv.2405.17358.

[Minaee et al., 2024] Minaee S. [et al.]. Large Language Models: A Survey // arXiv:2402.06196. – 2024. – doi: 10.48550/arXiv.2402.06196.

[Mittal et al., 2023] Mittal M. [et al.]. Orbit: A Unified Simulation Framework for Interactive Robot Learning Environments // IEEE Robotics and Automation Letters. – 2023. – doi: 10.1109/LRA.2023.3270034.

[Ramrakhya et al., 2023] Ramrakhya R., Batra D., Wijmans E., Das A. PIRLNav: Pretraining with Imitation and RL Finetuning for ObjectNav // Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). – 2023.

[Schulman et al., 2017] Schulman J., Wolski F., Dhariwal P., Radford A., Klimov O. Proximal Policy Optimization Algorithms // arXiv:1707.06347. – 2017. – doi: 10.48550/arXiv.1707.06347.

[Shridhar et al., 2022] Shridhar M., Manuelli L., Fox D. Perceiver-Actor: A Multi-Task Transformer for Robotic Manipulation // Proceedings of the Conference on Robot Learning (CoRL). – 2022. – doi: 10.48550/arXiv.2209.05451.

[Staroverov et al., 2023] Staroverov A., Gorodetsky A.S., Krishtopik A.S., Yudin D.A., Kovalev A.K., Panov A.I. Fine-tuning Multimodal Transformer Models for Generating Actions in Virtual and Real Environments // IEEE Access. – 2023. – Vol. 11. – P. 130548-130559. – doi: 10.1109/ACCESS.2023.3334791.

[Octo Team et al., 2024] Octo Model Team, D. Ghosh, H. Walke, K. Pertsch [и др.]. Octo: An Open-Source Generalist Robot Policy // Proceedings of Robotics: Science and Systems (RSS). – 2024. – URL: https://arxiv.org/abs/2405.12213.

[Vaswani et al., 2017] Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A.N., Kaiser Ł., Polosukhin I. Attention Is All You Need // Advances in Neural Information Processing Systems. – 2017. – Vol. 30. – P. 5998-6008.

[Zhao et al., 2025] Zhao T.Z., Tompson J., Driess D., Florence P., Ghasemipour S.K.S., Finn C., Wahid A. ALOHA Unleashed: A Simple Recipe for Robot Dexterity // Proc. of the 8th Conf. on Robot Learning (CoRL). PMLR 270, 2025. – P. 1910-1924.

[Zhou et al., 2024] Zhou Y., Zanette A., Pan J., Levine S., Kumar A. ArCHer: Training Language Model Agents via Hierarchical Multi-Turn RL // Proceedings of the 41st International Conference on Machine Learning (ICML 2024). – PMLR, 2024.