# Практическая работа № 3

## «Обучение с подкреплением на основе модели»

по дисциплине «Проектирование интеллектуальных систем»

**Цели:** приобрести навыки обучения с подкреплением (ОСП) интеллектуальных агентов систем искусственного интеллекта с помощью модельных методов обучения, основанных на марковском процессе принятия решений (Markov Decision Process, MDP) и решении проблемы многорукого бандита (Multi-Armed Bandits).

**Задачи:**

1) Выполнить обучение интеллектуального(-ых) агента(-ов) автоматизированной интеллектуальной системы с помощью модельных (model-based, **см. Примечание 1**) алгоритмов ОСП (Reinforcement Learning, RL), выполнив следующие подзадачи:

– определить предметную область решаемой задачи (может совпадать с задачами из предыдущих практических работ, либо из курса РСППР), таковыми могут выступать популярное обучение агентов компьютерных игр (стабилизация стержня / палки / перевернутого маятника, прохождение агентом лабиринта, игры-платформера, гонок с обгоном и без, настольный теннис и т.д.), задачи оптимизации (поиск оптимумов, обучение нейронной сети, поиск решений уравнений или задач), обучение многоагентной системы (симуляция жизни или других процессов, многоагентная оптимизация, прогнозирование), машинное обучение с подкреплением (обучение алгоритмов вроде линейной регрессии, Байесовского классификатора, случайного леса и т.п. с помощью методов ОСП, **см. Примечание 2**);

– определить обучаемого(-ых) агента(-ов) (ими могут выступать нейронные сети, агенты многоагентной системы, алгоритмы машинного обучения, персонажи игры и т.д.), определить среду системы (обычно в ОСП среда моделируется марковским процессом принятия решений, однако при необходимости можно представить её и другой моделью, например, с

помощью пространства решений с поиском по дереву Монте-Карло (MCTS, который чаще используют вместе MDP), чаще всего используются модификации MDP вроде POMDP – partially observable Markov decision process, SC-MDP – ship-centric Markov decision process и т.п.), определить целевую функцию (или целевое состояние) системы и метрики качества модели (ошибка, точность);

– реализовать модельный метод обучения с подкреплением вроде системы агентов с «расширенным воображением (представлением)» (Imagination-Augmented Agents, I2A), системы на основе моделей с безмодельной тонкой настройкой (Model-Based Deep Reinforcement Learning with Model-Free Fine-Tuning, MBMF), моделей мира (World Models, WM), AlphaZero или других модельных методов (**опять см. Примечание 1**), для реализации алгоритмов можно использовать репозитории вроде facebookresearch/mbrl-lib и opendilab/awesome-model-based-RL.

2) В качестве дополнительного задания реализовать хотя бы один из следующих пунктов:

– обучение на основе алгоритмов решения многорукого бандита вроде методов «действие-ценность» (Action-Value Methods) и алгоритмов «градиентного бандита» (Gradient Bandit Algorithms, **см. Примечание 3**);

– сравнение работы систем (на основе 3-ёх главных критериев – время, память и качество работы), реализованных на основе обучения с учителем и без учителя, с системами, реализованных с ОСП (для этого необходимо отдельно обучить ту же самую модель классическими методами обучения, после чего отдельно обучить её же с помощью ОСП и сравнить их метрики качества и 3 главных критерия, описанные в начале этого пункта);

– создание гибридной системы (обучения с подкреплением несколькими методами или/и обучение нескольких моделей).

ПРИМЕЧАНИЕ:

1) Модельные методы ОСП основываются на глобальной или локальной моделях (где модель – лишь представление агентов о среде, позволяющее прогнозировать поведение среды, а не учится методом проб и ошибок), в которых оптимизируется траектория для наименьших затрат вместо максимизации вознаграждения (что актуально для безмодельных методов), для чего задаётся специальная функция стоимости (действий), в общем случае используют 2 функции: функцию вероятности перехода и функцию вознаграждения. Моделями могут выступать определённые законы (математики, физики и т.п.), правила (некоторой игры, среды, эксперимента), нейронные сети (чаще всего – рекуррентные и глубокие), пространство состояний или полноценные системы и процессы вроде популярных модели гауссовских процессов (и модификации «модель гауссовой смеси» – Gaussian Mixture Model, GMM) или линейных динамических систем. В общем случае модельные методы ОСП разделяют на 2 класса – обучающие (изучающие, learn the model) модель и учитывающие модель (given the model):

а) Первый класс моделей, включающий главным образом алгоритмы I2A, MBMF, MBVE и WM, основан на необходимости предварительного изучения модели, для чего необходимо сначала выбрать определённую базовую политику (где политика – отображение, которое выбирает действия на основе наблюдений от среды, как правило, политика представляет собой аппроксимирующую функцию с настраиваемыми параметрами, например, глубокую нейронную сеть), например случайную или любую образованную политику (random or educated policy), с последующим наблюдением за траекторией, затем на основе этих данных подбирается модель. Так, например, в отличие от большинства методов RL и планирования на основе моделей, которые предписывают, как следует использовать модель для разработки политики, I2A учится интерпретировать прогнозы изученной модели

среды для построения неявных планов произвольными способами, используя прогнозы в качестве дополнительного контекста в политике.

b) Второй класс моделей, к которым относятся, например, алгоритмы AlphaZero (в том числе и сами AlphaGo и AlphaStar), Dyna-Q (работает над созданием моделей функции перехода и функция вознаграждения, сам же по себе представляет собой модификацией безмодельного Q-метода, то есть является гибридом модельных и безмодельных методов), Exlt, POPLIN и M2AC уже имеют в себе модель, благодаря которой они и работают, при этом некоторые модели дают агентам описание всех возможностей и их вероятностей – их называют «моделями распределения» (distribution models), другие модели дают только одну из возможностей, выбранную в соответствии с вероятностями – их называют «моделями выборки» (sample models). Так в AlphaZero моделью является нейронная сеть, основанная на свёрточной и остаточной архитектурах, а в Dyna-Q моделью выступает архитектура Dyna, модернизирующую Q-метод и объединяющую обучение, планирование и реактивное (динамическое) выполнение.

ОСП выполняется в какой-либо среде (среда агентных систем, которая определяет результаты действий агентов, ограничения и значения их атрибутов, возможности и запреты взаимодействия и т.д.), как правило, описываемой с помощью марковского процесса принятия решений, однако важно понимать, что сам марковский процесс в модельных алгоритмах ОСП – это проблема (а не какой-то метод, алгоритм и т.п.), которую пытается решить ОСП, иными словами агенты и модель пытаются решить проблему самой среды (ориентироваться в ней для решения задачи), в которой они находятся.

2) В машинном обучении модели обучаются принятыми для них и уже достаточно устававшимися методами, так в линейной регрессии процесс обучение может выполняться, например, с помощью метода наименьших квадратов или среднеквадратичного отклонения, которые являются видом обучения с учителем, однако этот метод можно модернизировать или заменить

с помощью ОСП (здесь как раз можно использовать линейную систему в качестве модели), что актуально и для градиентного спуска в данной задаче, агентами в таком решении будут параметры регрессии, а их мотивацией для получения вознаграждения – стремление к усреднению расстояния между как можно большим количеством точек системы. Такой подход модернизации или замены классического обучения можно реализовать и в других алгоритмах машинного обучения, включая алгоритмы кластеризации и классификации (вроде k-means, c-means, k-nn и т.д.), прогнозирования и оптимизации.

3) Проблема многорукого бандита имеет следующую постановку: в определенной ситуации вам нужно выбрать одно действие из набора k возможных действий (для данного состояния), после каждого выбора вы получаете числовое вознаграждение, выбранное из стационарного распределения вероятностей (то есть истинное вознаграждение не меняется, просто какое именно вы получите сейчас вы и не знаете, потому что не знаете, что вам выпадет) в зависимости от выбранного вами действия, ваша цель – максимизировать ожидаемое общее вознаграждение за некоторый период времени. Примером проблемы многорукого бандита может быть выбор рекомендательной системой между рекомендацией уже привычного пользователю продукта и новым, который возможно ему понравится и увеличит сферу его интересов и потребления, а может и наоборот его отпугнуть, то есть, это выбор типа «эксплуатация против исследования»; другим примером этой проблемы могут выступать азартные игры (откуда и пришёл термин) и их подобие, например, процесс открытия популярных «лутбоксов»: у вас есть несколько ключей одного типа, каждый из них может открыть несколько разных ящиков с «лутом», при этом в некоторых из них возможен наибольший шанс выпадания нужного вам по типу или качеству предмета, но вы не знаете, в каких именно, задача заключается в поиска стратегии, в результате которой можно получить как можно больше нужных и/или лучших предметов. Методы решения проблемы «многорукого бандита»

(на профессиональном сленге иногда эти решения и называют самими «многорукими бандитами») разделяются обычно на 2 группы:

a) Методы «действие-ценность» (Action-Value Methods, AVM): сосредоточены на получении точных оценок стоимости каждого действия и использовании этих оценок для принятия решений о том, какое из них выбрать, состоят из двух частей: 1) оценка ценности действий; 2) выбор действия на основе оценки стоимости всех действий.

b) Методы «градиентного бандита» (Gradient Bandit Algorithms, GBA): в отличие от методов "действие-ценность", которые оценивают ценность действий, методы градиентного бандита не заботятся о ценности действия, а, скорее, учат предпочтение каждого действия перед другим, чем больше предпочтение, тем чаще совершается это действие, но предпочтение не интерпретируется с точки зрения вознаграждения – важно лишь относительное предпочтение одного действия над другим.

К описанным выше группам относятся следующие алгоритмы: алгоритм верхних доверительных границ (ВДГ, Upper Confidence Bounds, UCB) и его модификации – UCB1 и Байесовский ВДГ (Bayesian UCB), «ε-жадный алгоритм» (ε-Greedy Algorithm), стратегия (иногда – выборка) Томпсона.