**Лекция №9. Обучение с подкреплением**

**Что такое стимулированное обучение?**

***Обучение с подкреплением (RL)*** – это метод машинного обучения (ML), который обучает программное обеспечение принимать решения для достижения наиболее оптимальных результатов. Такое обучение основано на имитации процесса обучения ***методом проб и ошибок***, который люди используют для достижения своих целей. Действия программного обеспечения, направленные на достижение цели, усиливаются, а действия, отвлекающие от цели, игнорируются.

Алгоритмы RL используют парадигму вознаграждения и санкций при обработке данных. Они учатся на основе отзывов о каждом действии и самостоятельно находят наилучшие способы обработки для достижения конечных результатов.

Большинство общеизвестных алгоритмов машинного обучения делает прогнозы. Но им не нужно «размышлять» над данными, поступающими за длительные периоды времени, или взаимодействовать с окружающим миром. Например, ***классификатор изображений***, получив изображение объекта, делает прогноз, отвечающий на вопрос о том, что это может быть за объект, например — яблоко или апельсин (рис. 1). Подобный классификатор изображений ***пассивен***. Он не взаимодействует с миром, ему не нужно принимать во внимание то, как изображение, предоставленное ему ранее, может повлиять на текущий прогноз. Задача классификации изображений относится к категории задач, где применяется обучение с учителем. ***Но это — не обучение с подкреплением****.*



Рисунок 1 – Классификация изображений

RL-алгоритмы, в отличие от пассивных прогнозных моделей, вроде классификаторов изображений, и взаимодействуют с внешним миром, и нуждаются в анализе данных за некие периоды времени. Это помогает таким моделям в достижении их целей. ***Цель RL-алгоритма*** заключается в максимизации выгод за заданный период времени. На интуитивном уровне такие алгоритмы похожи на классические психологические механизмы выработки условных реакций. Например, собак обычно обучают новым трюкам, поощряя их лакомством в том случае, если они правильно выполняют команды. Собака стремится максимизировать число получаемых угощений и изучает новые навыки, для того чтобы максимизировать своё вознаграждение.

**Основные понятия**

При обучении с подкреплением необходимо ознакомиться с несколькими ключевыми понятиями (рис. 2):

***Агент*** – это алгоритм МО (или автономная система).

***Среда (окружение)*** – это адаптивное проблемное пространство с такими атрибутами, как переменные, граничные значения, правила и допустимые действия.

***Действие*** – это шаг, который агент RL предпринимает для навигации по среде.

***Состояние*** – это среда в данный момент времени.

***Вознаграждение*** – это положительное, отрицательное или нулевое значение (другими словами, награда либо наказание) за выполнение действия.

***Совокупное вознаграждение*** – это сумма всех вознаграждений или конечное значение.



Рисунок 2 – Основные понятия обучения с подкреплением

**Основы алгоритма**

Обучение с подкреплением основано на процессе принятия решений по Маркову – математическом моделировании принятия решений с использованием дискретных временных шагов. На каждом этапе агент выполняет новое действие, которое приводит к изменению состояния окружающей среды. Точно так же текущее состояние приписывается последовательности предыдущих действий.

Путем проб и ошибок при перемещении по среде агент создает набор правил или политик типа ***«если-то»***. Эти правила помогают решить, какие действия предпринять дальше для получения ***оптимального совокупного вознаграждения***.

**Робот для сбора яблок**

Работа RL-алгоритмов очень похожа на механизм обучения собак новым фокусам. Но, вместо угощения, такие алгоритмы получают ***численные вознаграждения***. Например, предположим, что мы поместили робота в некое место, и нам нужно, чтобы он максимизировал бы количество собранных им яблок. Мы прибавляем единицу к вознаграждению робота каждый раз, когда он подбирает яблоко. По истечении фиксированного отрезка времени мы подсчитываем яблоки, которые собрал робот, и определяем его итоговую награду. RL-алгоритмы — это математический аппарат, цель которого — максимизация общей награды. Подобный алгоритм, обладая некоторым набором предположений относительно окружающей среды, гарантирует то, что робот соберёт ***максимально возможное количество яблок***, которые имеются в этой среде.

На примере робота для сбора яблок мы можем проиллюстрировать и ***фундаментальную проблему RL-алгоритмов***: задачу исследования окружающей среды (exploration problem). Предположим — робот находится в комнате, в которой имеется небольшое количество яблок (рис. 3). Но есть и другая комната, к которой ведёт длинный коридор. Там есть целая яблоня. Именно эта комната — наилучшее место для робота, но, чтобы туда добраться, ему нужно пройти по длинному коридору, не получая во время перехода никаких наград. RL-алгоритмы самостоятельно находят оптимальные решения предлагаемых им задач, но их обучение может занять много времени в том случае, если исследование окружающей среды оказывается непростым делом.



Рисунок 3 – Задача исследования окружающей среды

**Каковы примеры использования обучения с подкреплением?**

Обучение с подкреплением (RL) можно применять к широкому спектру реальных случаев использования. Далее представлены несколько примеров.

***Персонализация маркетинга***

В таких приложениях, как системы рекомендаций, RL может настраивать предложения для отдельных пользователей в зависимости от их взаимодействия (рис. 4). Например, приложение может показывать пользователю рекламу на основе демографической информации. При каждом рекламном взаимодействии приложение узнает, какую рекламу следует показывать пользователю для оптимизации сбыта товаров. Количество возможных продуктов, которое может порекомендовать система, весьма велико. Поэтому нужен эффективный алгоритм, который не только подберёт то, что сильнее всего понравится пользователю, но и обеспечит то, что выбор, который он делает, приведёт, в долгосрочной перспективе, к улучшению показателей, важных для компании. Среди примеров использования RL в мощных рекомендательных системах можно отметить систему, рекомендующую товары на Alibaba, и систему рекомендаций видео на YouTube.



Рисунок 4 – Принцип работы системы рекомендаций

***Финансовые прогнозы***

Динамика финансовых рынков сложна, статистические свойства со временем меняются. Алгоритмы RL могут оптимизировать долгосрочную прибыль, учитывая транзакционные издержки и адаптируясь к рыночным изменениям.

Например, алгоритм наблюдает за правилами и закономерностями фондового рынка, прежде чем тестировать действия и регистрировать соответствующие вознаграждения. Алгоритм динамически создает значение функции и разрабатывает стратегию максимального увеличения прибыли.

***Управление беспилотными транспортными средствами***

В наши дни обучение с подкреплением используется в Tesla для планирования путей беспилотных транспортных средств и для управления этими транспортными средствами. Один из примеров — применение RL для планирования пути при самостоятельной парковке автомобиля. Количество возможных путей, которые может исследовать автомобиль, это показатель, для которого характерен экспоненциальный рост. Традиционные методы планирования пути медленны и неэффективны. На мероприятии AI Day компания Tesla показала алгоритм, направленный на планирование наиболее быстрых и эффективных путей движения автомобиля.

***Охлаждение дата-центров***

Ещё одно полезное применение RL — охлаждение дата-центров в Google. Дата-центры потребляют много энергии, а компьютеры, которые в них содержатся, рассеивают эту энергию в виде тепла. Поддержание дата-центра в работоспособном состоянии требует применения серьёзных систем охлаждения. Функционирование этих систем обходится недёшево. Благодаря RL компания Google смогла развернуть системы охлаждения, которые включаются лишь по необходимости, что позволяет исключить ситуации, когда такие системы работают, а оборудование в охлаждении не нуждается.

**Каковы преимущества обучения с подкреплением?**

Обучение с подкреплением (RL) имеет много преимуществ. Ниже описаны три наиболее распространенные из них.

***Высокая эффективность в сложной среде***

Алгоритмы RL можно использовать в сложных средах со множеством правил и зависимостей. В той же среде человек может быть не в состоянии определить наилучший путь, даже обладая превосходными знаниями о среде. Алгоритмы RL быстро адаптируются к постоянно меняющимся средам и находят новые стратегии для оптимизации результатов.

***Минимизация человеческого вмешательства***

В стандартных алгоритмах машинного обучения люди должны маркировать пары данных, чтобы управлять алгоритмом. При использовании алгоритма RL в этом нет необходимости, поскольку он учится самостоятельно. В то же время алгоритм предлагает механизмы интеграции обратной связи с людьми, позволяющие создавать системы, адаптированные к предпочтениям, опыту и корректировкам человека.

***Оптимизация для достижения долгосрочных целей***

Поскольку обучение RL ориентировано на максимальное увеличение вознаграждения в долгосрочной перспективе, оно эффективно для сценариев, в которых действия влекут за собой длительные последствия. Такой тип особенно хорошо подходит для реальных ситуаций, когда обратная связь по каждому шагу не всегда доступна, поскольку позволяет извлечь уроки из отсроченных вознаграждений.

Например, решения о потреблении или хранении энергии могут иметь долгосрочные последствия. Обучение RL можно использовать для оптимизации энергоэффективности и затрат в долгосрочной перспективе. При наличии соответствующей архитектуры агенты RL также могут обобщать выученные стратегии для выполнения похожих, но не идентичных задач.