

Системы искусственного интеллекта

02.03.03 - Математическое обеспечение и администрирование информационных систем, направленность (профиль)
- разработка и администрирование информационных систем
09.03.03 - Прикладная информатика, направленность (профиль) - прикладная информатика в экономике

<http://vikchas.ru>

Тема 1. Введение в искусственный интеллект и машинное обучение Лекция 2 «Интеллектуальные агенты»

Часовских Виктор Петрович

д-р техн. наук, профессор кафедры ШИиКМ

ФГБОУ ВО «Уральский государственный экономический университет»

Екатеринбург 2023

Подход, основанный на использовании рационального агента

(Стюарт Рассел и Питер Норвиг)

Агентом считается все, что действует (слово *агент* произошло от из латинского слова *agere* — действовать).

Но предполагается, что **компьютерные агенты** обладают некоторыми другими атрибутами, которые отличают их от обычных “программ”, такими как способность функционировать под автономным управлением, воспринимать свою среду, существовать в течение продолжительного периода времени, адаптироваться к изменениям и обладать способностью взять на себя достижение целей, поставленных другими.

Рациональным агентом называется агент, который действует таким образом, чтобы можно было достичь наилучшего результата или, в условиях неопределенности, наилучшего ожидаемого результата.

Агентом является нечто воспринимающее и действующее в определенной среде.

Функция агента определяет действие, предпринимаемое агентом в ответ на любую последовательность актов восприятия.

Агентом является все, что может рассматриваться как воспринимающее свою, **среду** с помощью **датчиков** и воздействующее на эту среду с помощью **исполнительных механизмов**

Очевидно, что ответ на вопрос о том, возможно или невозможно создание искусственного интеллекта, зависит от того, как определено само понятие искусственного интеллекта.

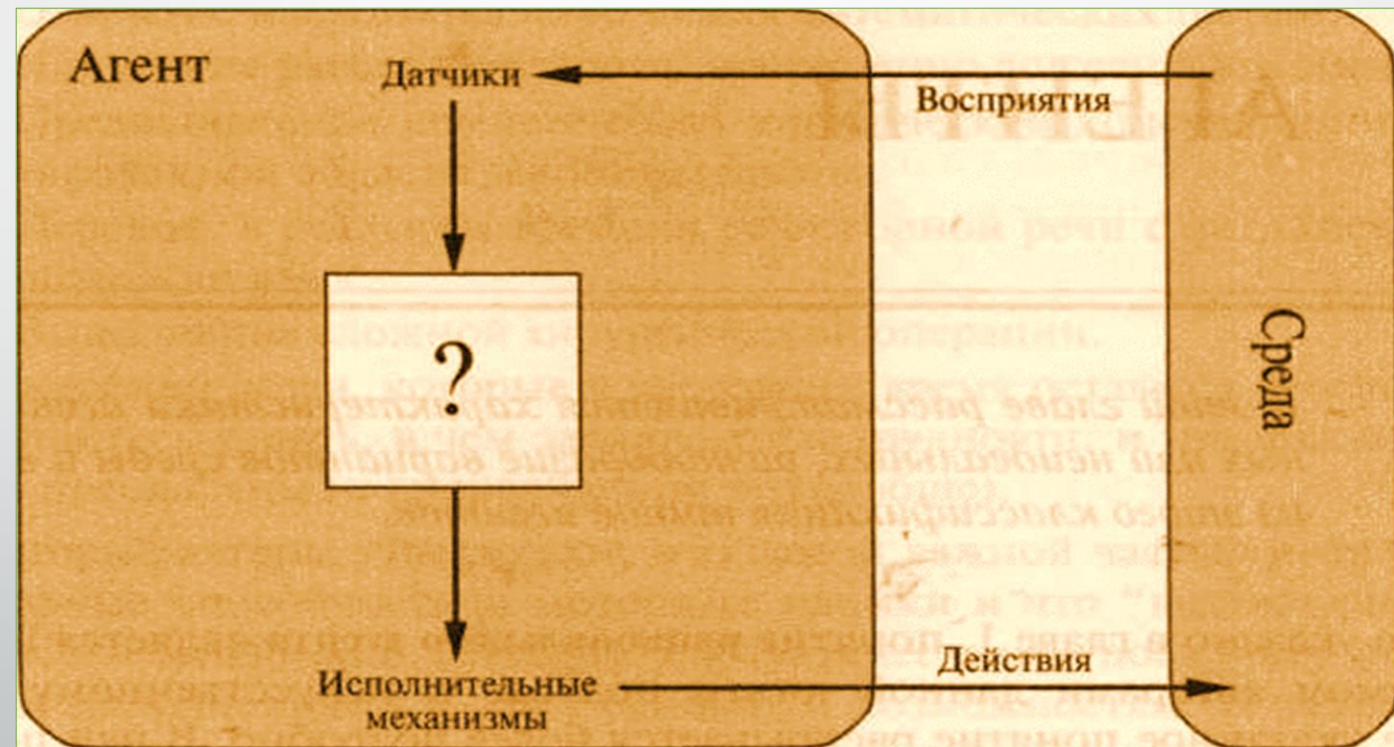
По существу создание искусственного интеллекта — это борьба за разработку наилучшей возможной программы агента в данной конкретной архитектуре.

При использовании такой формулировки создание искусственного интеллекта возможно по определению, поскольку для любой цифровой архитектуры, состоящей из k битов памяти, существует точно 2^k программ агентов, и для того чтобы найти наилучшую из них, достаточно просто последовательно проверить их все. Такой подход может оказаться неосуществимым при больших значениях k .

В настоящее время существует множество концепций и определений агента, моделей и агентных вычислений. Будем придерживаться истории развития и современного состояния агентов, проектов агентов Стюарта Рассела, Питера Норвига «Искусственный интеллект: современный подход», изд. ООО «Диалектика», «2021».

Агенты, варианты среды и связи между ними

Агентом является все, что может рассматриваться как воспринимающее свою среду с помощью датчиков и воздействующее на эту среду с помощью исполнительных механизмов.



Человек, рассматриваемый в роли **агента**, имеет глаза, другие органы чувств, а исполнительными механизмами для него служат руки, ноги, рот и другие части тела.

Робот, выполняющий функции **агента**, в качестве датчиков может иметь видеокамеры и инфракрасные дальномеры, а его исполнительными механизмами могут являться различные двигатели.

Программное обеспечение, выступающее в роли **агента**, в качестве входных сенсорных данных получает коды нажатия клавиш, содержимое файлов и сетевые пакеты, а его воздействие на среду выражается в том, что программное обеспечение выводит данные на экран, записывает файлы и передает сетевые пакеты.

Мы принимаем общее допущение, что каждый **агент** может воспринимать свои собственные действия (но не всегда их результаты).

Необходимо отметить, что понятие **агента** рассматривается как **инструмент** для анализа систем, а не как абсолютная классификация, согласно которой мир делится на агентов и не агентов.

Например, в качестве агента можно было бы рассматривать карманный калькулятор, который выбирает действие по отображению “4” после получения последовательности актов восприятия “2+2=”, но подобный анализ вряд ли поможет понять работу калькулятора.

КОНЦЕПЦИЯ РАЦИОНАЛЬНОСТИ

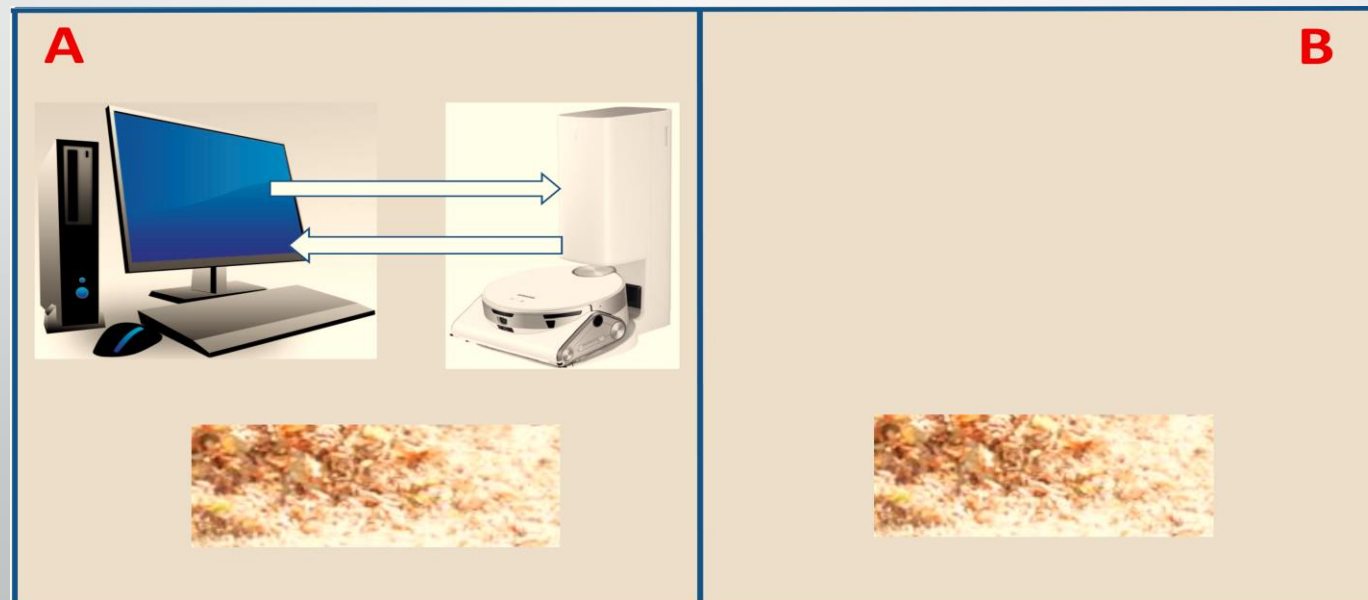
Рациональным агентом является такой агент, который выполняет правильные действия; выражаясь более формально, таковым является агент, в котором каждая запись в таблице для функции агента заполнена правильно.

Очевидно, что выполнение правильных действий лучше, чем осуществление неправильных действий, но что подразумевается под выражением “выполнение правильных действий”?

В первом приближении можно сказать, что правильным действием является такое действие, которое обеспечивает наиболее успешное функционирование агента. Поэтому требуется определенный способ измерения успеха. Критерии успеха, наряду с описанием среды, а также датчиков и исполнительных механизмов агента, предоставляют полную спецификацию задачи, с которой сталкивается агент. Имея эти компоненты, мы можем определить более точно, что подразумевается под, словом “рациональный”.

Рассмотрим пример уборка мусора пылесосом. Есть всего два помещения, представленных квадратами А и В. Каждый квадрат может быть чистым или грязным, а агент может перемещаться в левый или правый квадрат и выполнять в нем уборку.

В различных версиях мира пылесоса могут действовать разные правила в отношении того, что может воспринимать агент, всегда ли его действия будут успешны и т.д.



Функции простого агента в мире пылесоса оформим в следующей таблице:

Последовательность восприятия	Действие
[А, Чисто]	Вправо
[А, Грязно]	Убрать
[В, Чисто]	Влево
[В, Грязно]	Убрать
[А, Чисто], [А, Чисто]	Вправо
[А, Чисто], [А, Грязно]	Убрать
⋮	⋮
[А, Чисто], [А, Чисто], [А, Чисто]	Вправо
[А, Чисто], [А, Чисто], [А, Грязно]	Убрать
⋮	⋮

Агент убирает текущий квадрат, если в нем грязно, а в противном случае перемещается в другой квадрат. Обратите внимание, что эта таблица будет иметь бесконечно большой размер, если нет ограничений на длину допустимых последовательностей восприятия

Рациональность

В любой конкретный момент времени оценка рациональности действий агента зависит от четырех перечисленных ниже факторов.

- › Показатели производительности, которые определяют критерии успеха.
- › Знания агента о среде, приобретенные ранее.
- › Действия, которые могут быть выполнены агентом.
- › Последовательность актов восприятия агента, которые произошли до настоящего времени.

С учетом этих факторов можно сформулировать следующее **определение рационального агента**: «Для каждой возможной последовательности актов восприятия рациональный агент должен выбрать действие, которое, как ожидается, максимизирует его показатели производительности, с учетом фактов, предоставленных данной последовательностью актов восприятия и всех встроенных ¹¹ знаний, которыми обладает агент.

Свойства окружающей среды

Прежде чем приступить к созданию рациональных агентов рассмотрим понятие **проблемной среды**, по сути представляющей собой "проблему", для которой рациональный агент служит "решением".

Начнем с демонстрации того, как определить среду задачи, и проиллюстрируем этот процесс на ряде примеров. Затем будет показано, что проблемная среда может иметь целый ряд разновидностей.

Свойства окружающей среды непосредственно влияют на выбор проектного решения в отношении программы агента.

Определение проблемной среды

В простом примере агента - пылесоса, мы определили показатели производительности, среду, а также исполнительные механизмы и датчики этого агента.

Все это можно рассматривать как проблемную среду. В литературе используется аббревиатура **PEAS** (*Performance, Environment, Actuators, Sensors* - производительность, среда, исполнительные механизмы, датчики).

При разработке любого агента первый этап всегда должен заключаться в определении проблемной среды с наибольшей возможной полнотой.

Мир пылесоса - это очень простой пример, поэтому рассмотрим более сложную задачу: автоматизированный водитель такси.

Общее описание PEAS для автоматизированного такси

Тип агента	Показатели производительности	Среда	Исполнительные механизмы	Датчики
Водитель такси	Безопасная, быстрая и комфортная езда с соблюдением правил дорожного движения, максимизация прибыли, минимизация влияния на других участников дорожного движения	Дороги, различные транспортные средства, полиция, пешеходы, пассажиры, погода	Рулевое управление, тормоза, световые указатели, сигнал, акселератор (газ), компьютер, речь	Спидометр, датчики двигателя, видеорекамеры, микрофоны, сенсорный экран

Среда, в которой будет функционировать водитель такси? Любому водителю такси придется иметь дело с различными дорогами, начиная от сельских улочек и городских переулков до много-полосных автострад. На любых дорогах может присутствовать другой движущийся транспорт, пешеходы и бездомные животные, могут проводиться дорожные работы, встречаться полицейские машины, попадаться лужи и выбоины. Таксист также должен взаимодействовать с потенциальными и действительными пассажирами. Существуют и некоторые дополнительные аспекты. Движение на дорогах может быть правосторонним или левосторонним, поэтому придется учитывать этот факт, если требуется, чтобы агент был более гибким и мог успешно функционировать в различных странах. Очевидно, что чем более ограничена среда, тем проще задача проектирования.

В число **ИСПОЛНИТЕЛЬНЫХ** **МЕХАНИЗМОВ** автоматизированного такси обязательно войдут все механизмы, доступные водителю-человеку: управление двигателем посредством педали акселератора, ручное и ножное управление тормозами и рулевое управление. Кроме того, понадобятся средства вывода на экран дисплея или синтезатор голоса, чтобы общаться с пассажирами, и, возможно, какие-то механизмы и с другими транспортными средствами, вежливо или наоборот.

Главные **датчики** агента - водителя такси должны включать одну или более видеокамер, чтобы можно было наблюдать за происходящим, а также датчики для определения расстояния до других машин и препятствий.

Чтобы избежать штрафов за превышение скорости, в такси должен быть спидометр, а для правильного управления машиной, особенно на поворотах, в ней должен быть установлен акселерометр.

Для определения текущего состояния оборудования автомобиля достаточно будет обычного набора датчиков в двигателе, топливной и электрической системе. Как и многие люди-водители, агент-таксист может нуждаться в доступе к системе ГЛОНАСС, чтобы не потеряться в пути. И наконец, для получения от пассажира информации о требуемом пункте назначения потребуются сенсорный экран или система голосового

Рассмотрим основные элементы PEAS для целого ряда прочих типов агентов

Тип агента	Показатели-производительности	Среда	Исполнительные механизмы	Датчики
Медицинская диагностическая система	Успешное излечение пациента, минимизация затрат, отсутствие поводов для судебных тяжб	Пациент, больница, персонал	Вывод вопросов, тестов, диагнозов, рекомендаций, направлений	Ввод с клавиатуры симптомов, результатов лабораторных исследований, ответов пациента
Система анализа изображений, полученных со спутника	Правильная классификация изображения	Канал передачи данных от орбитального спутника	Вывод на дисплей результатов классификации определенного фрагмента изображения	Массивы пикселей с данными о цвете
Робот-сортировщик деталей	Процентные показатели безошибочной сортировки по лоткам	Ленточный конвейер с движущимися на нем деталями; лотки	Шарнирный манипулятор и захват	Видеокамера, датчики углов поворота шарниров
Контроллер очистительной установки	Максимизация степени очистки, продуктивности, безопасности	Очистительная установка, операторы	Клапаны, насосы, нагреватели, дисплеи	Температура, давление, датчики химического состава
Интерактивная программа обучения английскому языку	Максимизация оценок студентов на экзаменах	Множество студентов, экзаменационное агентство	Вывод на дисплей упражнений, рекомендаций, исправлений	Ввод с клавиатуры

В отличие от приведенных примеров некоторые **программные агенты** (называемые также **программными роботами** или **софтботами**) существуют в сложных, неограниченных проблемных областях.

Представьте себе программный робот, предназначенный для управления тренажером, имитирующим крупный пассажирский самолет. Этот тренажер представляет собой очень детально промоделированную, сложную среду, в которой имитируются движения других самолетов и работа наземных служб, а программный агент должен выбирать в реальном времени наиболее целесообразные действия из широкого диапазона действий.

Еще одним примером может служить программный робот, предназначенный для просмотра источников новостей в Internet и показа клиентам интересующих их сообщений.

Для успешной работы ему требуются определенные способности к обработке текста на естественном языке, он должен в процессе обучения определять, что интересует каждого заказчика, а также должен уметь изменять свои планы динамически, допустим, когда соединение с каким-либо из источников новостей закрывается или в оперативный режим переходит новый источник новостей.

Internet представляет собой среду, которая по своей сложности соперничает с физическим миром, а в число обитателей этой сети входит много искусственных агентов.

Структура агентов

Рассмотрим как организовано внутреннее функционирование агентов.

Задача искусственного интеллекта состоит в разработке **программы агента**, реализующей функцию агента, т.е. отображающей последовательность восприятия на действия.

Будем предполагать, что эта программа должна работать в своего рода вычислительном устройстве с физическими датчиками и исполнительными механизмами. В целом совокупность этих компонентов образует **архитектуру агента**.

Таким образом, структуру агента можно представить следующей формулой:

$$\textit{agent} = \textit{architecture} + \textit{program}.$$

Очевидно, что выбранная программа должна быть подходящей для существующей архитектуры. Так, если в программе предполагается выдача таких указаний, как **ИДТИ**, то в архитектуре следует предусмотреть наличие ног.

Архитектура может представлять собой обычный персональный компьютер или же может быть реализована в виде роботизированного автомобиля с несколькими бортовыми компьютерами, видеокамерами и другими датчиками.

В общем случае архитектура должна обеспечить передачу в программу результатов восприятия, поступающих от датчиков, выполнение самой программы агента и передачу исполнительным механизмам выбранных программой вариантов действий по мере их генерации.

Программы агентов

Будем рассматривать программы агентов, которые имеют одну и ту же структуру: они принимают от датчиков в качестве входных данных результаты текущего восприятия и возвращают исполнительным механизмам выбранный вариант действия.

Необходимо указать на различие между программой агента, которая принимает в качестве входных данных текущие восприятия, и функцией агента, которая принимает на входе всю историю восприятия.

Программа агента получает в качестве входных данных только результаты текущего восприятия, поскольку больше ничего не может узнать из своей среды; если действия агента зависят от всей последовательности актов восприятия, то агент должен сам запоминать результаты этих актов восприятия.

Для записи программ агентов будем использовать формальный язык, определяемый как множество строк, в котором каждая строка представляет собой последовательность символов. Наш язык состоит из бесконечного множества строк, поэтому нужен простой способ, позволяющий охарактеризовать это множество.

Для этого будем применять **контекстно-свободную грамматику**, поскольку каждое выражение в этом случае будет иметь одну и ту же форму в любом контексте.

В качестве способа оформления грамматики используем формальную систему под названием **форма Бэкуса-Наура** или **БНФ** (Backus-Naur form, BNF), входящую в школьную программу «Информатика». Для выполнения лабораторных работ я предлагаю методические указания *«Определение языков программирования с помощью формы Бэкуса-Наура»*, размещенные в библиотеке моего сайта.

Рассмотрим простую программу агента, которая регистрирует последовательность восприятия, а затем использует полученную последовательность для доступа по индексу к таблице действий с целью определения, что нужно сделать.

Таблица для мира пылесоса, рассмотренная ранее явно представляет функцию агента, воплощаемую данной программой агента.

Чтобы создать рационального агента таким способом, проектировщики должны сформировать таблицу, которая содержит подходящее действие для любой возможной последовательности актов восприятия.

function TABLE-DRIVEN-AGENT(*percept*) **returns** действие *action*
persistent: *percepts*, последовательность восприятия, исходно пустая
table, таблица действий, индексированная по последовательностям
восприятия, исходно полностью определенная
добавить результат восприятия *percept* в конец последовательности
percepts Action ← LOOKUP (*percepts*, *table*)
return *action*

Программа TABLE-DRIVEN-AGENT вызывается для каждого нового восприятия и каждый раз возвращает действие. Она сохраняет полную последовательность восприятия в памяти

Для понимания реальных параметров рассмотрим случай автоматизированного такси: визуальные входные данные с одной видеокамеры (обычно их восемь) поступают со скоростью примерно 70 Мбайт/с (30 кадров в секунду, в кадре 1080 x 720 пикселей с 24 битами цветовой информации).

На один час езды потребуется справочная таблица более чем с $10^{600\,000\,000\,000}$ записей! Даже для игры в шахматы поисковая таблица должна содержать не менее 10^{150} записей.

Ошеломляющий размер этих таблиц означает, что:

- ни один физический агент не будет иметь достаточно места для хранения подобной таблицы;
- любому проектировщику просто не хватит времени для ее создания;
- ни один агент не сможет обучиться всему, что содержится во всех правильных записях таблицы на собственном опыте.

Однако, несмотря на все сказанное выше, программа TABLE-DRIVEN-AGENT действительно делает то, что мы хотели (при условии, что таблица²⁵ заполнена правильно): реализует желаемую функцию агента.

Ключевая задача ИИ заключается в выяснении того, как создавать программы, которые в рамках возможного обеспечивают рациональное поведение агента с использованием небольшого объема программного кода, а не обширных таблиц с множеством записей.

Существует много примеров, свидетельствующих о том, что подобную задачу можно успешно решить в других областях.

Например, огромные таблицы квадратных корней, которыми пользовались инженеры и школьники до 1970-х годов, теперь заменены программой из пяти строк, работающей в электронных калькуляторах, - для выполнения расчетов в ней применяется метод Ньютона.

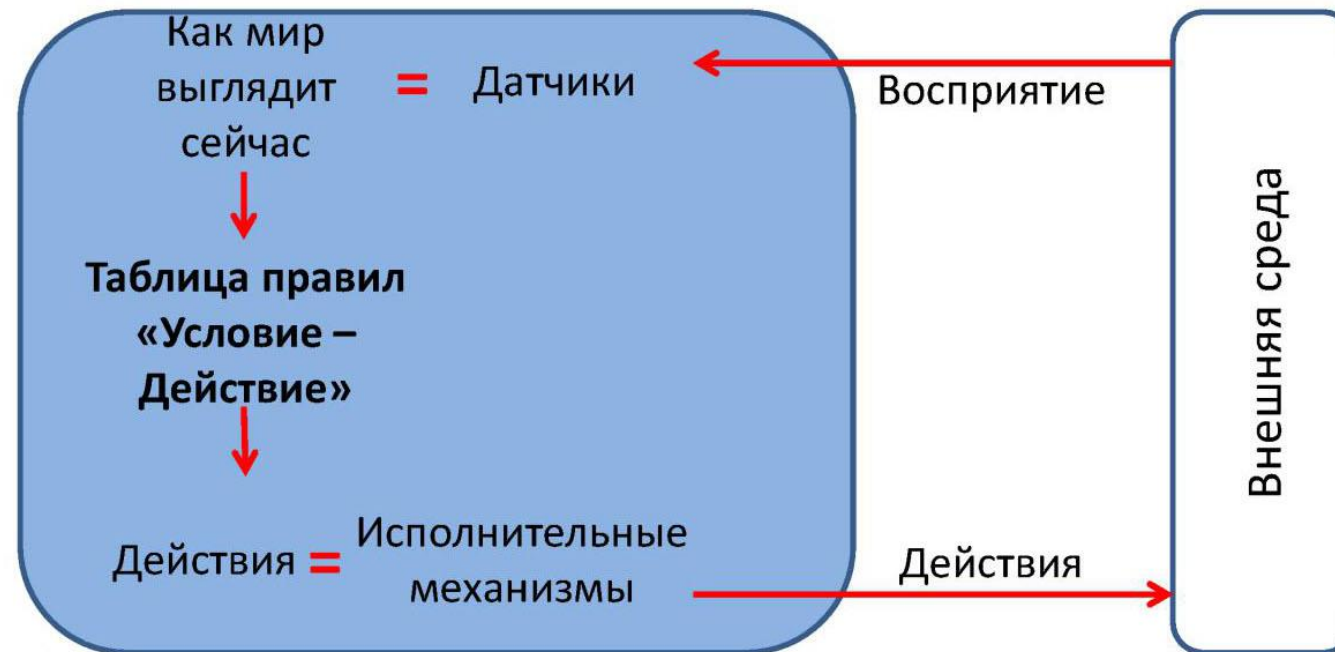
Рассмотрим четыре основных вида программ агентов, которые воплощают принципы, лежащие в основе почти всех интеллектуальных систем:

- простые рефлексные агенты;
- рефлексные агенты, основанные на модели;
- агенты, действующие на основе цели;
- агенты, действующие на основе полезности.

Затем приведено описание в общих терминах того, как преобразовать агентов всех этих типов в обучающихся агентов.

Простые рефлексные агенты

Простой рефлексный агент вырабатывает действие только на основе текущего акта восприятия, игнорируя всю остальную их историю.



Очень простые, очень надёжны, но ограниченный интеллект.

Например, агент-пылесос, таблица функций которого приведена ранее, является простым рефлекторным агентом, поскольку его решения основаны только на информации о текущем местоположении и о том, содержит ли оно мусор.

Программа для данного агента имеет следующий вид:

```
function REFLEX-VACUUM-AGENT(location, status) returns действие action  
if status = Dirty then return Suck  
else if location = A then return Right  
else if location = B then return Left
```

Простое рефлекторное поведение встречается и в более сложных средах. Рассмотрим водителя автоматизированного такси. Если автомобиль перед вами стал тормозить и его стоп-сигналы засветились, вы должны заметить это и также немедленно начать торможение. Другими словами, необходима определенная обработка визуальных сигналов для выявления ситуации, которую мы обозначим как "*car-in-front-is-braking*" - автомобиль впереди тормозит. Ее обнаружение инициирует в программе агента некоторую связь с действием *initiate-braking*, т.е. начать торможение. Подобную связь называют **правилом "условие-действие"**, которое можно сформулировать как

```
if car-in-front-is-braking then initiate-braking
```


Благодарю за внимание!

