

ОСНОВЫ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

Лекция Вердыш А.В. для ОО ВНО. Минск. 2023 г.

Часть 1

Представления людей об интеллекте с развитием цивилизации всё более изменяется. Когда-то даже простой переписчик книг или счетовод считался тружеником интеллектуального труда. В нашу эпоху уже не являются интеллектуальной задачей многие разделы высшей математики, если для их решения существует определённый алгоритм.

Сейчас принято считать интеллектуальными **задачи, которые не поддаются алгоритмизации** в традиционном смысле этого слова. Это задачи, для решения которых требуются необходимо работать с нечёткими, неконкретными, ненадежными, расплывчатыми или нетрадиционными знаниями.

Человек — это самый сложный из доступных для нашего восприятия объект, а способность мышления — его главное свойство (**атрибут**).

Искусственный интеллект — наука, поставившая своей целью изучение и моделирование атрибута человека. Какова природа мышления? Какие процессы происходят в нашем организме, когда мы думаем, чувствуем, видим, понимаем?

Возможно ли в принципе понять, как работает наш мозг, и заставить мыслить неживую природу? На протяжении тысячелетий мы задавались этими вопросами, но до сих пор точного ответа нет.

Толчком для развития послужило изобретение вычислительных, цифровых машин. С одной стороны, она стимулировала интерес общественности к новому научному направлению, выразившийся в ускоренном финансировании проектов правительствами. **С другой стороны, кибернетика стала объектом весьма резкой критики более «трезво мыслящих» ученых.**

Вскоре после признания искусственного интеллекта самостоятельной отраслью науки произошло его разделение на два основных направления: **нейрокибернетику и кибернетику «черного ящика».** Первое из этих направлений иногда называют **низкоуровневым, или восходящим, а второе — высокоуровневым, или нисходящим.**

Основную идею нейрокибернетики можно сформулировать следующим образом. Единственный объект, способный мыслить, — это человеческий мозг. Поэтому любое мыслящее устройство должно быть обязательно выполнено по образу и подобию человеческого мозга, воспроизводить его структуру, его принцип действия.

Таким образом, нейрокибернетика занимается аппаратным моделированием структуры мозга и его деятельности.

Как известно, мозг человека состоит из большого количества взаимосвязанных нервных клеток — нейронов. Поэтому усилия нейрокибернетиков сосредоточены на разработке элементов, подобных нейронам, и объединении этих элементов в системы — нейросети и нейрокомпьютеры. Первые нейросети и нейрокомпьютеры были предложены и созданы американскими учеными В. Мак-Каллоком, В.Питтсом и Ф. Розенблаттом в конце 1950-х годов. Это были устройства, моделирующие человеческий глаз и его взаимодействие с мозгом. Устройства умели распознавать буквы алфавита, однако были чувствительны к их написанию.

Сегодня нейрокомпьютерные и нейросетевые технологии являются одним из наиболее перспективных и быстро развивающихся разделов искусственного интеллекта. Крупных успехов в этой области добились японские исследователи.

Ими создан компьютер нового поколения — нейрокомпьютер, моделирующий структуру мозга и имеющий обширную базу знаний. Значительных успехов в этой области добились российские ученые. Отечественные нейрокомпьютеры уже давно применяются для управления сложными техническими объектами военного назначения.

В отличие от нейрокибернетики **кибернетика «черного ящика»** не придает значения принципу действия мыслящего устройства. Главное, чтобы оно адекватно моделировало его функциональную деятельность. Это направление искусственного интеллекта ориентировано на **поиски алгоритмов решения интеллектуальных задач с использованием существующих компьютеров независимо от их аппаратной базы.**

Поставив перед собой задачу моделирования функций мозга, ученые столкнулись с серьезной проблемой. Оказалось, что несмотря на многовековую историю исследований ни одна из существующих наук (философия, психология, лингвистика и др.) не смогла предложить сколько-нибудь конкретный алгоритм человеческого мышления. Поэтому кибернетикам пришлось создавать собственные модели мышления.

В конце 50-х гг. XX в. появилась модель лабиринтного поиска. Согласно этому подходу решение интеллектуальной задачи выполнялось путем перебора огромного количества вариантов, который представлялся в виде движения по лабиринту. Создание таких алгоритмов, по словам их критиков, было не более разумно, чем попытки заново написать все книги, хранящиеся в Британском музее, посадив за пишущие машинки обезьян и надеясь, что обезьяны рано или поздно чисто случайно сумеют напечатать осмысленное слово, фразу или страницу. В настоящее время модель лабиринтного поиска признается тупиковой и имеет ограниченное использование в игровых компьютерных программах.

В начале 1960-х гг. началась эпоха эвристического программирования. Как писал автор этого термина американский математик Пойа, цель эвристики — исследовать методы и правила, как делать открытия и изобретения. Это очень сложная проблема.

Дело в том, что Архимед, выпрыгнувший из ванны с криком «Эврика», не объяснил, каким образом он догадался, что тело, погружённое в жидкость, теряет в своем весе ровно столько, сколько весит вытесненный им объем воды. Ньютон открыл закон всемирного тяготения, наблюдая за падением яблока. Менделеев пришел к принципу построения периодической таблицы во сне. Поэтов и музыкантов вдохновляют к творческим поискам возвышенные чувства, разобраться в которых **в принципе невозможно**.

Чтобы понять механизмы творческого мышления, авторы эвристического подхода провели эксперимент. Была отобрана группа студентов, не знакомых с математической логикой. Каждый студент должен был доказать самостоятельно одну или несколько теорем из учебника, не заглядывая в него. При этом ему вменялось в обязанность рассуждать вслух, делать любые записи, прекращать работу, если становилось ясно, что выбран неверный путь, и начинать все сначала.

Обработав магнитофонные записи, выкладки, черновики студентов, программисты нашли эвристики — способы, которыми пользовались студенты, доказывая теоремы. А затем с помощью этих эвристик была составлена программа, известная под названием «Логик-теоретик», которую принято считать родоначальницей эвристического программирования. И эта программа доказала все теоремы, какие были в учебнике, и сформулировала дополнительно те, которых не хватало до полной логической завершенности курса.

Наряду с указанными выше двумя подходами к проблеме моделирования мышления и создания искусственного интеллекта существует третий, названный эволюционным программированием (моделированием). Смысл этого подхода состоит в том, что процесс моделирования человека заменяется моделированием процесса его эволюции.

Серьезный прорыв в практических приложениях искусственного интеллекта произошел в середине 1970-х гг., когда, отказавшись от поисков универсального алгоритма мышления, программисты начали моделировать конкретные знания специалистов экспертов. Открылось новое направление искусственного интеллекта — экспертные системы. С появлением экспертных систем бизнес в сфере интеллектуальных информационных технологий впервые становится рентабельным.

С середины 1980-х гг. искусственный интеллект — это одно из наиболее привлекательных в коммерческом отношении направлений компьютерной индустрии. Растут ежегодные капиталовложения, создаются промышленные и военные экспертные системы.

В качестве альтернативы экспертным системам появляются и успешно завоевывают рынок нейросетевые и нейрокомпьютерные технологии, в которых, подобно процессам, происходящим в мозгу, **знания растворяются в межнейронных связях, а процесс программирования системы заменяется ее обучением**.

РАЗВИТИЕ ТЕХНОЛОГИЙ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

Искусственный интеллект — это обширная область исследований и разработок интеллектуальных систем, предназначенных для работы в трудно формализуемых областях деятельности человека. Для задач, решаемых методами искусственного интеллекта, характерно наличие большого числа степеней свободы с числом вариантов поиска решений, приближающимся к бесконечности. В отличие от жестко детерминированных компьютерных программ системы искусственного интеллекта сами ищут пути решения поставленной задачи.

При этом они могут менять свои параметры и структуру, совершенствоваться и развиваться, жить самостоятельной, не зависящей от воли разработчика жизнью.

До недавнего времени считалось основным и наиболее полезным в развитии искусственного интеллекта- это разработка интеллектуальных систем, основанных на всех предыдущих знаниях. Создавались модели представлений знаний и базы знаний, образующих ядро экспертных систем.

В настоящее же время упор сделан на нейросетевые и нейрокомпьютерные технологии. Искусственные нейронные сети и нейрокомпьютеры в значительной мере заимствуют принципы работы головного мозга. Знания в них не отделены от процессора, а равномерно распределены и существуют неявно в виде сил синаптических связей. Такие знания не закладываются изначально, а приобретаются в процессе обучения.

Особо важным является **распознавание образов**. Это широкий круг проблем: распознавание **изображений, символов, текстов, запахов, звуков, шумов**. На рынке программных средств имеются системы, основанные на распознавании по признакам, оснащенные базами данных и знаний, имеющих возможность адаптации и обучения.

Однако в **последнее время становятся популярными гибридные системы, в которых наряду с технологиями экспертных систем используются и нейросетевые технологии.**

Игры и творчество. Традиционно искусственный интеллект включает в себя интеллектуальные задачи, решаемые при игре в шахматы, шашки, го, каллах. В основе этого направления лежит один из ранних подходов — лабиринтная модель плюс эвристики.

Кроме того, в современных программах-играх наиболее полно удалось реализовать центральную идею искусственного интеллекта — **обучение, самообучение и самоорганизацию.**

В широком смысле слова под **игрой** понимается **некая конфликтная ситуация**, участники которой своими действиями не только достигают своих личных целей, но и влияют на

достижимость целей другими участниками игры. Ясно, что **под такое толкование игры подпадают многие экономические, политические и военные конфликты.**

Компьютерное творчество представляет пока чисто теоретический интерес. Наибольший прогресс достигнут в сочинении компьютерной музыки. Разработаны различные модели художественного и поэтического творчества, имеющие больше познавательный, чем практический интерес.

Компьютерная лингвистика. Начиная с 50-х гг. XX в. и по настоящее время одной из популярных тем исследований искусственного интеллекта является область машинного перевода. Первая программа в этой области — переводчик с английского языка на русский. Первая идея — пословный перевод. В настоящее время используются более сложные структуры естественно-языковых интерфейсов, которые включают в себя:

морфологический анализ — анализ слов в тексте;

синтаксический анализ — анализ предложений, грамматики и связей между словами;

семантический анализ — анализ смысла каждого предложения на основе базы знаний, на которую ориентирована конкретная программа-переводчик;

прагматический анализ — анализ смысла предложений в окружающем контексте с помощью базы знаний.

Другой проблемой компьютерной лингвистики является разработка естественно-языкового интерфейса между человеком и машиной. Здесь немаловажную роль могут сыграть нейросетевые технологии, с помощью которых удастся научить компьютер правильному произношению слов. В проектах создания компьютеров новейших поколений решению этой проблемы уделено первостепенное внимание.

Интеллектуальные роботы. Роботы — это технические устройства, предназначенные для автоматизации человеческого труда. Само слово «робот» появилось в 20-х гг. XX в. Его автор — чешский писатель Карел Чапек.

В настоящее время в промышленности применяется огромное количество роботов-манипуляторов, работающих по жесткой схеме управления. В отличие от них интеллектуальные роботы обладают **способностью самообучаться и самоорганизовываться, адаптироваться к изменяющейся окружающей обстановке.**

Компьютерные вирусы. Сегодня трудно назвать компьютерного пользователя, избежавшего знакомства с этим видом программной продукции.

Последние поколения вирусов обладают всеми атрибутами систем искусственного интеллекта. Они свободно перемещаются по компьютерам, мутируют и размножаются, обучаются, меняют свои параметры и структуру.

Воздействие компьютерных вирусов значительно возросло с появлением сетей. По прогнозам специалистов, неприятности, которые мы испытываем сегодня, представляются ничтожными по сравнению с теми перспективами, которые ожидают нас с проникновением компьютерных вирусов в сферу интеллектуальных роботов. Интеллектуальное математическое моделирование. Это компьютерное математическое моделирование с использованием методов искусственного интеллекта.

Интеллектуальные системы подобного рода имитируют творческую деятельность математика-профессионала, занимающегося решением краевых задач математической физики. Они обладают базами знаний, содержащими нужные теоремы, математические зависимости и эвристические правила, обобщающие опыт и интуицию математика-профессионала, способны к обучению с помощью учителя и к самообучению.

Основные технологические понятия

Данные — это отдельные факты, характеризующие объекты, процессы и явления предметной области, а также их свойства (атрибуты).

На ЭВМ данные изменяются, последовательно проходя следующие этапы:

данные, существующие как результат измерений и наблюдений;

данные на материальных носителях информации — в таблицах, протоколах, справочниках;

структуры данных в виде диаграмм, графиков, функций;

данные в компьютере на языке описания данных;

базы данных.

Знания связаны с данными, основываются на них, но представляют собой результат мыслительной деятельности человека, обобщают его опыт, полученный в ходе практической деятельности.

Знания — это выявленные закономерности предметной области.

При обработке на ЭВМ знания трансформируются аналогично данным:

знания, существующие в памяти человека как результат обучения, воспитания, мышления;

знания, помещенные на материальных носителях — учебниках, инструкциях, методических пособиях, книгах;

знания, описанные на языках представления знаний и помещенные в компьютер; .

Базы знаний

Для **хранения данных** используются базы данных. Для них характерны большой объем и относительно небольшая стоимость информации. Для **хранения знаний** используются базы знаний. Они, наоборот, отличаются сравнительно небольшими объемами, но исключительно дорогими информационными массивами.

Знания могут быть классифицированы на **поверхностные** — знания о видимых взаимосвязях между отдельными событиями и фактами в предметной области, и **глубинные** — абстракции, аналогии, схемы, отображающие структуру и процессы в предметной области.

Кроме того, знания можно разделить на процедурные и декларативные.

Исторически первичными были процедурные знания, т.е. знания, растворенные в алгоритмах. Они управляли данными. Для их изменения требовалось изменять программы.

Например, (фрагмент VBA – кода):

1. $a=5$
2. $S=a*a$
3. Площадь_квадрата= S (25)

Первые два оператора представляют собой данные, третий оператор — знание. Оно является результатом интеллектуальной деятельности древних геометров и представляет собой закон, выражающий площадь квадрата через его сторону.

С развитием искусственного интеллекта приоритет данных постепенно изменялся и все большая часть знаний сосредоточивалась в структурах данных, т.е. увеличивалась роль **декларативных знаний**.

Существуют десятки способов представления декларативных знаний для различных предметных областей. Большинство из них может быть сведено к следующим классам: продукционные; фреймы; семантические сети.

МЕТОДЫ ПРЕДСТАВЛЕНИЯ ЗНАНИЙ

Продукционная система состоит из трех основных компонентов. Первый из них — это база правил типа ЕСЛИ (условие), ТО (действие): ЕСЛИ холодно, ТО надеть шубу; ЕСЛИ идет дождь, ТО взять зонтик, и т.п.

Вторым компонентом является рабочая память, в которой хранятся исходные данные к задаче и выводы, полученные в ходе работы системы.

Третий компонент — механизм логического вывода, использующий правила в соответствии с содержимым рабочей памяти.

Рассмотрим конкретный пример. В базе правил экспертной системы имеются два правила:

Правило 1: ЕСЛИ «намерение — отдых» и «дорога ухабистая», ТО «использовать джип».

Правило 2: ЕСЛИ «место отдыха — горы», ТО «дорога ухабистая».

Допустим, что в рабочую память поступили исходные данные:

«намерение — отдых»; «место отдыха — горы».

Механизм вывода начинает сопоставлять образцы из условных частей правил с образцами, хранимыми в рабочей памяти.

Если образцы из условной части имеются в рабочей памяти, то условная часть считается истинной, в противном случае — ложной.

В данном примере при рассмотрении правила 1 оказывается, что образец «намерение — отдых» имеется в рабочей памяти, а образец «дорога ухабистая» отсутствует, поэтому условная часть правила 1 считается ложной. При рассмотрении правила 2 выясняется, что его условная часть истинна. Механизм вывода выполняет заключительную часть этого правила, и образец «дорога ухабистая» заносится в рабочую память. Правило 2 при этом выбывает из числа кандидатов на рассмотрение.

Снова рассматривается правило 1, условная часть которого теперь становится истинной, и содержимое рабочей памяти пополняется образцом «использовать джип». В итоге правил, которые можно было бы применять, не остается и система останавливается.

В рассмотренном примере приведен прямой вывод — от данных к поиску цели. Однако применяют и обратный вывод — от цели для ее подтверждения к данным. Продемонстрируем этот способ на нашем примере. Допустим, что наряду с исходными данными «намерения — отдых»; «место отдыха — горы» имеется цель «использовать джип».

Согласно правилу 1 для достижения этой цели требуется выполнение условия «дорога ухабистая», поэтому условие становится новой целью. При рассмотрении правила 2 оказывается, что условная часть этого правила в данный момент истинна, поэтому рабочая память пополняется образцом «дорога ухабистая». При повторном рассмотрении правила 1 подтверждается цель «использовать джип».

При обратном выводе система останавливается в двух случаях:

либо достигается первоначальная цель, либо кончаются правила. При прямом выводе система останавливается только тогда, когда кончаются правила, либо при появлении в рабочей памяти специально предусмотренного образца, например, «использовать джип».

В приведенном примере на каждом этапе прямого вывода можно было использовать только одно правило. В общем же случае на каждом этапе вывода таких правил несколько, и тут возникает проблема выбора. Например, введем в рассмотрение еще одно правило.

Правило 3: ЕСЛИ «намерение — отдых», ТО «нужна скорость».

Кроме того, введем условие останова системы — появление в рабочей памяти образца «использовать джип».

Теперь на первом этапе прямого вывода появляется возможность применять либо правило 2, либо правило 3. Если сначала применить правило 2, то на следующем этапе можно будет применять правило 1 и правило 3. Если на этом этапе применить правило 1, то выполнится условие останова системы, но если прежде применить правило 3, то потребуются ещё один этап вывода.

Этот пример показывает, что выбор применяемого правила оказывает прямое влияние на эффективность вывода. В реальной системе, где имеется множество правил, **появляется проблема их оптимального выбора.**

Если на каждом этапе логического вывода существует множество применимых правил, то это множество носит название конфликтного набора, а выбор одного из них называется разрешением конфликта.

Аналогичная ситуация возникает и при обратном выводе. Например, дополним предыдущий пример еще одним правилом.

Правило 4: ЕСЛИ «место отдыха — пляж», ТО «дорога ухабистая».

Если на основании этого условия подтверждается Цель «использовать джип», то для достижения первоначальной цели достаточно применить только одно правило 1, однако, чтобы подтвердить новую цель «дорога ухабистая», открывается возможность применения правила 1, нужно использовать либо правило 2, либо правило 4. Если сначала применить правило 2, то это будет самый удачный выбор, поскольку сразу же

можно применить и правило 1. С другой стороны, если попытаться применить правило 2, то, поскольку образца «место отдыха — пляж», который является условием правила 4, в рабочей памяти не существует, кроме

того, не существует правила, - заменяющего его, данный выбор является неудачным: И лишь со второго захода, применяя правило 2, можно подтвердить цель “Дорога ухабистая».

Следует обратить внимание на то, что при обратном выводе правило 3, которое не оказывает прямого влияния на достижение цели, не принималось в расчет с самого начала. Таким образом, для обратных выводов характерна тенденция исключения из рассмотрения правил, не имеющих прямого отношения к заданной цели, что позволяет повысить эффективность вывода.

Продукционная модель — это наиболее часто используемый способ представления знаний в современных экспертных системах. Основными преимуществами продукционной модели являются наглядность, высокая модульность, легкость внесения изменений и дополнений, простота механизма логического вывода.

Фреймы

Фрейм — это модель абстрактного образа, минимально возможное описание сущности какого-либо объекта, явления, события, ситуации, процесса. Фрейм состоит из имени и отдельных единиц, называемых слотами.

Например, слово «автомобиль» вызывает у слушающих образ устройства, способного перемещаться, имеющего четыре колеса, салон для шофера и пассажиров, двигатель, руль. Это описание абстрактного образа «автомобиль» является минимальным и из него ничего нельзя убрать без потери его сущности.

Он имеет однородную структуру:

ИМЯ ФРЕЙМА

Имя 1-го слота: значение 1-го слота

Имя 2-го слота: значение 2-го слота

.....

Имя NN-го слота: значение N-го слота.

В качестве значения слота может выступать имя другого фрейма. Таким образом фреймы объединяются в сеть. **Свойства фреймов наследуются сверху вниз.**

Каждый слот может содержать следующую информацию:

- значение параметра, представленного слотом;

- ссылка на другой фрейм;
- информация по умолчанию (значения слотов, которые должны быть использованы для типа объекта и сохраняются для него, если они не были изменены в ходе моделирования);
- дескрипторы для значений слотов (конкретные значения конкретных свойств стереотипных объектов, которые используются для определения соответствия новых объектов стереотипу, описанному в фрейме);
- присоединенная (связанная) процедура;
- неинициализированные значения. Слоты могут оставаться незаполненными, пока они не понадобятся для решения какой-либо задачи;
- метainформация — информация о допустимых значениях для заполнения слотов.

Незаполненный фрейм называется **протофреймом**, а заполненный — **экзофреймом**. Роль протофрейма как оболочки в экзофрейме весьма важна. Эта оболочка позволяет осуществлять процедуру внутренней интерпретации, благодаря которой данные в памяти системы не безлики, а имеют вполне определенный, известный системе смысл.

Например, фрейм «Солдат» имеет ссылки на вышестоящие фреймы: «Человек» и «Млекопитающее». Поэтому на вопрос: «Может ли студент мыслить?» — ответ будет положительным, так как этим свойством обладает вышестоящий фрейм «Человек».

Если одно и то же свойство указывается в нескольких связанных между собой фреймах, то приоритет отдается нижестоящему фрейму. Так, **возраст** фрейма «Солдат» не наследуется из вышестоящих фреймов.

Основным преимуществом фреймов как способа представления знаний является наглядность и гибкость в употреблении. Кроме того, фреймовая структура согласуется с современными представлениями о хранении информации в памяти человека.

Семантические сети

В основе этого способа представления знаний лежит идея о том, что любые знания можно представить в виде совокупности понятий (объектов) и отношений (связей). Семантическая сеть представляет собой ориентированный граф, вершинами которого являются понятия, а дугами — отношения между ними. Сам термин «**семантическая**» означает **смысловая**.

Основным **преимуществом этой модели** является наглядность представления знаний, а также соответствие современным представлениям об организации долговременной памяти человека.

Недостаток — сложность поиска вывода, а также сложность корректировки, т.е. удаления и дополнения сети новыми знаниями.

ЭКСПЕРТНЫЕ СИСТЕМЫ

Знания, которыми обладает специалист в какой-либо области, можно разделить на формализуемые и плохо формализуемые. Формализуемые знания излагаются в книгах и руководствах в виде законов, формул, моделей, алгоритмов. **Формализуемые знания характерны для точных наук**, таких как математика, физика, химия, астрономия.

Науки, которые принято называть описательными, обычно оперируют с плохо формализуемыми знаниями.

К таким наукам можно отнести, например, зоологию, ботанику, экологию, социологию, педагогику, медицину и др.

Существуют неформализуемые знания, которые вообще не попадают в книги и руководства в связи с их неконкретностью, субъективностью, приблизительностью. Знания этого рода являются результатом многолетних наблюдений, опыта работы, интуиции. Они обычно представляют собой множество эмпирических и эвристических приёмов и правил.

Такие знания передаются из поколения в поколение в виде определенных навыков, ноу-хау, секретов ремесла. Есть также знания, которые не могут быть выражены ни в математическом виде, ни в терминах обычного человеческого языка. Такими знаниями обладают религиозные деятели, экстрасенсы, контактеры, шаманы.

Класс задач, относящихся к **неформализуемым** и **плохо формализуемым** знаниям, значительно больше класса задач, для которых знания формализуемы. Этим объясняется особая популярность и широкое практическое применение экспертных систем, которые открыли возможность применения компьютерных технологий в предметных областях, в которых знания плохо формализуемы.

Экспертные системы — это сложные программные комплексы, аккумулирующие знания специалистов в конкретных предметных областях и тиражирующие эти знания для консультаций менее квалифицированных пользователей.

Обычно в состав экспертной системы входят следующие взаимосвязанные между собой модули:

база знаний — ядро экспертной системы, совокупность знаний предметной области, записанная на машинном носителе в форме, понятной эксперту и пользователю;

интеллектуальный редактор базы знаний — программа, представляющая инженеру-когнитологу и программисту возможность создавать базу знаний в диалоговом режиме. Она включает в себя системы вложенных меню, шаблонов языка представления знаний, подсказок (пер-режим) и других сервисных средств, облегчающих работу с базой знаний;

интерфейс пользователя — комплекс программ, реализующих диалог пользователя с экспертной системой на стадии как ввода информации, так и получения результатов;

решатель (синонимы: дедуктивная машина, блок логического вывода) — программа, моделирующая ход рассуждений эксперта на основании знаний, имеющихся в базе знаний;

подсистема объяснений — программа, позволяющая пользователю получать ответы на вопросы: «Как была получена та или иная рекомендация?» и «Почему система приняла такое решение?». Ответ на вопрос «Как?» — это трассировка всего процесса получения решения с указанием исполняющих фрагментов базы знаний, т.е. всех шагов цепи умозаключений. Ответ на вопрос «Почему?» — ссылка на умозаключение, непосредственно предшествовавшее полученному решению, т.е. отход на один шаг назад.

В коллектив разработчиков экспертной системы входят как минимум четыре специалиста (или четыре группы специалистов):

эксперт, инженер-когнитолог, программист, пользователь. Возглавляет коллектив **инженер-когнитолог** — ключевая фигура при разработке систем, основанных на знаниях. Обычно это руководитель проекта, в задачу которого входит организация всего процесса создания экспертной системы. С одной стороны, он должен быть специалистом в области искусственного интеллекта, а с другой — разбираться в предметной области, общаться с экспертом, извлекая и формализуя его знания, передавать их программисту, кодирующему и помещающему их в базу знаний экспертной системы.

Экспертная система работает в двух режимах — приобретения знаний и решения задач или консультаций.

В режиме приобретения знаний происходит формирование базы знаний. В режиме решения задач общение с экспертной системой осуществляет конечный пользователь.

Обычно знания, которыми располагает эксперт, различаются степенью надежности, важности, четкости. В этом случае они снабжаются некоторыми весовыми коэффициентами, которые называют коэффициентами доверия. Такие знания обрабатываются с помощью алгоритмов нечеткой математики.

В процессе опытной эксплуатации коэффициенты доверия могут подвергаться корректировке. В этом случае говорят, что происходит обучение экспертной системы. Процесс обучения экспертной системы может производиться автоматически с помощью обучающего алгоритма либо путем вмешательства инженера-когнитолога, выполняющего роль учителя.

Этапы и технология разработки экспертных систем

В процессе разработки экспертные системы проходят определенные стадии, в результате которых создаются различные версии, называемые прототипами:

демонстрационный прототип — экспертная система, которая решает часть требуемых задач, демонстрируя жизнеспособность метода инженерии знаний. Работает, имея в базе знаний всего 50...100 правил. Время разработки такой экспертной системы — 6... 12 мес.;

исследовательский прототип — экспертная система, которая решает все требуемые задачи, но неустойчива в работе и неполностью проверена. База знаний содержит 200... 500 правил. Разработка занимает 3...6 мес.;

действующий прототип — надежно решает все задачи, но для решения сложных задач может потребоваться много времени и памяти. База знаний содержит 500... 1000 правил. Время разработки — 6... 12 мес.;

промышленная экспертная система — обеспечивает высокое качество решения всех задач при минимуме времени и памяти, что достигается переписыванием программ с использованием более совершенных инструментальных средств и языков низкого уровня.

База знаний содержит 1000...1500 правил. Время разработки — 1...1,5 года;

коммерческая экспертная система — отличается от промышленной тем, что помимо собственного использования она может продаваться различным потребителям. База знаний содержит 1500...3000 правил. Время разработки — 1,5...3 года. Стоимость — 0,3...5 млн долларов и выше.

Технология разработки экспертных систем состоит из следующих этапов:

1. Идентификация (постановка задачи). На этапе устанавливаются задачи, которые подлежат решению, выявляются цели разработки, требования к экспертной системе, ресурсы, используемые понятия и их взаимосвязи, определяются методы решения задач. Цель этапа — сформулировать задачу, охарактеризовать поддерживающую ее базу знаний и таким образом обеспечить начальный импульс для развития базы знаний.

2. Концептуализация. Проводится содержательный анализ проблемной области, выявляются используемые понятия и их взаимосвязи, определяются методы решения задач.

3. Формализация. Определяются способы представления всех видов знаний, формализуются основные понятия, определяются способы интерпретации знаний, оценивается адекватность целям системы зафиксированных понятий, методов решения, средств представления и манипулирования знаниями.

4. Выполнение. Осуществляется наполнение экспертом базы знаний. Процесс приобретения знаний разделяют на извлечение знаний из эксперта, организацию знаний, обеспечивающую эффективную работу системы, и представление знаний в виде, понятном экспертной системе. Из-за эвристического характера знаний их приобретение является весьма трудоемким.

5. Тестирование. Эксперт и инженер по знаниям в интерактивном режиме, используя диалоговые и объяснительные средства, проверяют компетентность экспертной системы. Процесс тестирования продолжается до тех пор, пока эксперт не решит, что система достигла требуемого уровня компетентности.

6. Опытная эксплуатация. Проверяется пригодность экспертной системы для конечных пользователей. По результатам этого этапа может потребоваться модификация экспертной системы.

7. Модификация. В ходе создания экспертной системы почти постоянно производится ее модификация: переформулирование понятий и требований, переконструирование представления знаний и усовершенствование прототипа.

Усовершенствование прототипа осуществляется в процессе циклического прохождения через этапы выполнения и тестирования для отладки правил и процедур вывода.

Переконструирование выбранного ранее способа представления знаний предполагает возврат с этапа тестирования на этап формализации.

Если возникшие проблемы еще более серьезны, то после неудачи на этапе тестирования может потребоваться возврат на этап концептуализации и идентификации. В этом случае речь идет о переформулировании понятий, используемых в системе, т.е. перепроектировании системы заново.

