

Лекция 2.

Раздел 2. Основы технологии СИИ, модели представления знаний и стратегии вывода решений

Тема 4. Методологические основы ИИ.

Тема 5. Классические техники обработки данных, связанные с ИИ.

Тема 6. Компьютерная поддержка извлечения знаний. Основные методы извлечения знаний.

План лекции.

1. Основы ИИ.
2. Методы извлечения знаний.

Основная часть.

Искусственный интеллект - раздел информатики, изучающий возможность обеспечения разумных рассуждений и действий с помощью вычислительных систем и иных искусственных устройств. При этом в большинстве случаев заранее неизвестен алгоритм решения задачи.

Точного определения этой науки не существует, так как в философии не решён вопрос о природе и статусе человеческого интеллекта. Нет точного критерия достижения компьютерами «разумности», хотя на заре искусственного интеллекта был предложен ряд гипотез, например, тест Тьюринга или гипотеза Ньюэлла-Саймона¹. На данный момент есть

¹ **Гипотеза Ньюэлла-Саймона** - «Физическая символьная система имеет необходимые и достаточные средства для произведения основных интеллектуальных операций». Именуется также *гипотеза о физической символьной системе*. Под интеллектуальными операциями подразумеваются действия сильного искусственного интеллекта.

Гипотеза была сформулирована Алленом Ньюэллом и Гербертом Саймоном в 1976 году. Основанием для гипотезы стало успешное применение созданной ими программы - универсального решателя задач для моделирования рассуждений человека. Другими словами, без символьных вычислений невозможно выполнять осмысленные действия, а способность выполнять символьные вычисления вполне достаточна для того, чтобы стать способным выполнять осмысленные действия. Таким образом, если мы полагаем, что животное, или человек, или машина действуют осмысленно, то значит, они каким-то образом выполняют символьные вычисления (ваш кот в какой-то мере - вычислительная машина). И наоборот, так как компьютер способен к подобным вычислениям, то на его основе *может* быть создан искусственный интеллект.

множество подходов как к пониманию задачи ИИ, так и созданию интеллектуальных систем.

Так, одна из классификаций выделяет два подхода к разработке ИИ:

- *нисходящий, семиотический* - создание символьных систем, моделирующих высокоуровневые психические процессы: мышление, рассуждение, речь, эмоции, творчество и т. д.;
- *восходящий, биологический* - изучение нейронных сетей и эволюционные вычисления, моделирующих интеллектуальное поведение на основе более мелких «неинтеллектуальных» элементов.

Эта наука связана с психологией, нейрофизиологией², трансгуманизмом³ и другими. Как и все компьютерные науки, она использует математический аппарат. Особое значение для неё имеют философия и робототехника. Искусственный интеллект - очень молодая

Гипотеза уязвима для критики, но так получилось, что большая часть исследований искусственного интеллекта пошла именно по пути создания символьных систем. Независимо от того, справедлива ли эта гипотеза, символьные вычисления – реальность программирования, и полезность подобной парадигмы в этой области трудно отрицать.

² **Нейрофизиология** – раздел физиологии, изучающий функции нервной системы. Задача нейронауки состоит в том, чтобы объяснить поведение человека, понять как мозг управляет миллионами нервных клеток, чтобы сформировать поведение и как на эти клетки влияет внешняя среда. *Целью нейрофизиологии* является формирование представления о функциональной организации нервной системы, нейронных механизмах организации рефлекторного поведения и принципах системной организации функций мозга. *Задачи нейрофизиологии*: дать понятие основным нервным процессам и взаимодействию отделов нервной системы друг с другом; дать представление об основах физиологии нервной ткани и центральной нервной системы человека; дать представление о принципах системной организации функций мозга.

³ **Трансгуманизм** (от лат. *trans* - сквозь, через, за и *homo* - человек) - философская концепция, а также международное движение, поддерживающие использование достижений науки и технологии для улучшения умственных и физических возможностей человека с целью устранения тех аспектов человеческого существования, которые трансгуманисты считают нежелательными - страданий, болезней, старения и смерти. Трансгуманисты изучают возможности и последствия применения таких технологий, опасности и преимущества их использования, рассматривая в том числе идею конвергенции биологических, информационных, познавательных и нанотехнологий.

область исследований, старт которой был дан в 1956 году.

Подходы к изучению. Существуют различные подходы к построению систем ИИ. На данный момент можно выделить 4 различных подхода:

1. *Логический подход.* Основой для логического подхода служит Булева алгебра. Каждый программист знаком с нею и с логическими операторами с тех пор, когда он осваивал оператор IF. Свое дальнейшее развитие Булева алгебра получила в виде исчисления предикатов - в котором она расширена за счет введения предметных символов, отношений между ними, кванторов существования и всеобщности. Практически каждая система ИИ, построенная на логическом принципе, представляет собой машину доказательства теорем. При этом исходные данные хранятся в базе данных в виде аксиом, правила логического вывода как отношения между ними. Кроме того, каждая такая машина имеет блок генерации цели, и система вывода пытается доказать данную цель как теорему. Если цель доказана, то трассировка примененных правил позволяет получить цепочку действий, необходимых для реализации поставленной цели (такая система известна как экспертные системы). Мощность такой системы определяется возможностями генератора целей и машиной доказательства теорем. Добиться большей выразительности логическому подходу позволяет такое сравнительно новое направление, как нечеткая логика. Основным ее отличием является то, что правдивость высказывания может принимать в ней кроме да/нет (1/0) еще и промежуточные значения - не знаю (0.5), пациент скорее жив, чем мертв (0.75), пациент скорее мертв, чем жив (0.25). Данный подход больше похож на мышление человека, поскольку он на вопросы редко отвечает только да или нет.

2. *Под структурным подходом* мы подразумеваем здесь попытки построения ИИ путем моделирования структуры человеческого мозга. Одной из первых таких попыток был персептрон Френка Розенблатта. Основной моделируемой структурной единицей в персептронах (как и в большинстве других вариантов моделирования мозга) является нейрон. Позднее возникли и другие модели, которые большинству известны под термином нейронные сети (НС). Эти модели различаются по строению отдельных нейронов, по топологии связей между ними и по алгоритмам обучения. Среди наиболее известных сейчас

вариантов НС можно назвать НС с обратным распространением ошибки, сети Хопфилда⁴, стохастические нейронные сети. В более широком смысле такой подход известен как Коннективизм⁵.

⁴ **Нейронная сеть Хопфилда** (англ. *Hopfield network*) – полносвязная нейронная сеть с симметричной матрицей связей. В процессе работы динамика таких сетей сходится (конвергирует) к одному из положений равновесия. Эти положения равновесия определяются заранее в процессе обучения, они являются локальными минимумами функционала, называемого *энергией сети* (в простейшем случае - локальными минимумами отрицательно определённой квадратичной формы на n-мерном кубе). Такая сеть может быть использована как автоассоциативная память, как фильтр, а также для решения некоторых задач оптимизации. В отличие от многих нейронных сетей, работающих до получения ответа через определённое количество тактов, сети Хопфилда работают до достижения равновесия, когда следующее состояние сети в точности равно предыдущему: начальное состояние является входным образом, а при равновесии получают выходной образ.

⁵ **Коннекционизм** (англ. *connectionism*) - один из подходов в области искусственного интеллекта, когнитивной науки(когнитивистики), нейробиологии, психологии и философии сознания. *Коннекционизм* моделирует мыслительные или поведенческие явления процессами становления в сетях из связанных между собой простых элементов. Существует много различных форм *коннекционизма*, но наиболее общие используют нейросетевые модели. В рамках этого течения предпринимаются попытки объяснить умственные способности человека, используя искусственные нейронные сети. Философы начали проявлять интерес к коннекционизму, так как коннекционистский подход обещал обеспечить альтернативу классической теории разума и широко распространённой в рамках этой теории идеи, согласно которой механизмы работы разума имеют сходство с обработкой знакового языка цифровой вычислительной машиной. То, как именно и в какой степени парадигма коннекционизма составляет альтернативу классическим представлениям о природе разума, является предметом жарких споров, ведущихся в последние годы. Главный принцип коннекционизма состоит в предположении, что мыслительные явления могут быть описаны сетями из взаимосвязанных простых элементов. Форма связей и элементов может меняться от модели к модели. Например, элементы в сети могут представлять нейроны, а связи - синапсы. Другая модель может считать каждый элемент в сети словом, а каждую связь признаком семантического подобия и т.п.

3. *Эволюционный подход.* При построении систем ИИ по данному подходу основное внимание уделяется построению начальной модели, и правилам, по которым она может изменяться (эволюционировать). Причем модель может быть составлена по самым различным методам, это может быть и НС и набор логических правил и любая другая модель. После этого мы включаем компьютер и он, на основании проверки моделей отбирает самые лучшие из них, на основании которых по самым различным правилам генерируются новые модели. Среди эволюционных алгоритмов классическим считается генетический алгоритм

4. *Имитационный подход.* Данный подход является классическим для кибернетики с одним из ее базовых понятий черный ящик. Объект, поведение которого имитируется, как раз и представляет собой «черный ящик». Нам не важно, что у него и у модели внутри и как он функционирует, главное, чтобы наша модель в аналогичных ситуациях вела себя точно так же. Таким образом здесь моделируется другое свойство человека - способность копировать то, что делают другие, не вдаваясь в подробности, зачем это нужно. Зачастую эта способность экономит ему массу времени, особенно в начале его жизни.

В рамках гибридных интеллектуальных систем пытаются объединить эти направления. Экспертные правила умозаключений могут генерироваться нейронными сетями, а порождающие правила получают с помощью статистического обучения.

Многообещающий новый подход, называемый усиление интеллекта, рассматривает достижение ИИ в процессе эволюционной разработки как побочный эффект усиления человеческого интеллекта технологиями.

Основные методы извлечения знаний. Теория формальных систем и, в частности, математическая логика являются формализацией человеческого мышления и представления наших знаний. Если предположить, что можно аксиоматизировать наши знания и можно построить алгоритм, позволяющий реализовать процесс вывода ответов на запрос из знаний, то в результате можно получить формальный метод для получения неформальных результатов.

Логическое программирование возникло в эру ЭВМ как естественное желание автоматизировать процесс логического вывода, поэтому оно является ветвью теории формальных систем.

Логическое программирование (в широком смысле) представляет собой семейство таких методов решения задач, в которых используются приемы логического вывода для манипулирования знаниями, представленными в декларативной форме. Как писал Джордж Робинсон в 1984 году, в основе идеи логического программирования лежит описание задачи совокупностью утверждений на некотором формальном логическом языке и

получение решения с помощью вывода в некоторой формальной (аксиоматической) системе. Такой аксиоматической системой являются исчисление предикатов первого порядка, поэтому в узком смысле логическое программирование понимается как использование исчисления предикатов первого порядка в качестве основы для описания предметной области и осуществления резолюционного логического вывода.

Аксиоматической системой называется способ задания множества путем указания исходных элементов (аксиом исчисления) и правил вывода, каждое из которых описывает, как строить новые элементы из исходных элементов.

Под *аксиоматическим методом* понимают способ построения научной теории, при которой за ее основу берется ряд основополагающих, не требующих доказательств положений этой теории, называемыми аксиомами или постулатами.

Аксиоматический метод зародился в работах древнегреческих геометров. Вплоть до начала XIX века единственным образцом применения этого метода была геометрия Евклида.

В начале XIX века Н.И. Лобачевский и Я. Больяй, независимо друг от друга, открыли новую неевклидову геометрию, заменив пятый постулат о параллельных прямых на его отрицание. Их открытие стало отправной точкой для развития аксиоматического метода, который лег в основу теории формальных систем.

С накоплением опыта построения формальных теорий и попытками аксиоматизации арифметики, предпринятыми Дж. Пеано, возникла *теория доказательств*. Теория доказательств – это раздел современной математической логики и предшественница логического программирования.

Исчислениями называют наиболее важные из аксиоматических логических систем – исчисление высказываний и исчисление предикатов.

Формальная теория строится как четко определенный класс выражений, формул, в котором некоторым точным способом выделяется подкласс теорем данной формальной системы. При этом формулы формальной системы непосредственно не несут в себе никакого содержательного смысла, они строятся из произвольных знаков или символов, исходя лишь из соображений удобства.

Введение в основы нечеткой логики

Нечеткие знания. При попытке формализовать человеческие знания исследователи вскоре столкнулись с проблемой, затруднявшей использование традиционного математического аппарата для их описания. Существует целый класс описаний, оперирующих качественными характеристиками объектов (*много, мало, сильный, очень сильный* и т. п.). Эти характеристики обычно размыты и не могут быть однозначно

интерпретированы, однако содержат важную информацию (например, «Одним из возможных признаков гриппа является *высокая* температура»).

Кроме того, в задачах, решаемых интеллектуальными системами, часто приходится пользоваться неточными знаниями, которые не могут быть интерпретированы как полностью истинные или ложные (логические true/false или 0/1). Существуют знания, достоверность которых выражается некоторой промежуточной цифрой, например 0,7.

Как, не разрушая свойства размытости и неточности, представлять подобные знания формально? Для разрешения таких проблем в начале 70-х американский математик *Лотфи Заде* предложил формальный аппарат нечеткой (fuzzy) алгебры нечеткой логики. Л. Заде ввел одно из главных понятий в нечеткой логике, - понятие лингвистической переменной.

Лингвистическая переменная (ЛП) – это переменная, значение которой определяется набором вербальных (словесных) характеристик некоторого свойства.

Например, ЛП «рост» определяется через набор (*карликовый, низкий, средний, высокий, очень высокий*).

Самым главным понятием систем, основанных на нечеткой логике, является понятие *нечеткого (под)множества*.

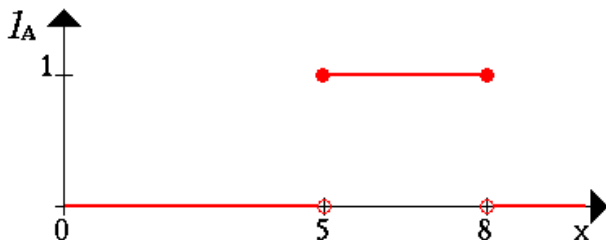
Из классической математики известно понятие *четких (определенных) множеств*.

Пример:

Рассмотрим множество X всех чисел от 0 до 10, которое назовем универсумом рассуждения. Определим подмножество A множества X всех действительных чисел от 5 до 8.

$$A = [5, 8]$$

Покажем характеристическую функцию множества A , эта функция ставит в соответствие число 1 или 0 каждому элементу в X , в зависимости от того принадлежит данный элемент подмножеству A или нет. Результат представлен на следующем рисунке:



Можно интерпретировать элементы, которым поставлена в соответствие 1, как элементы, находящиеся во множестве A , а элементы, которым поставлен в соответствие 0, как элементы, не находящиеся во множестве A .

Эта концепция используется во многих областях приложений. Но можно легко обнаружить ситуации, в которых данной концепции будет не хватать гибкости.

В данном примере опишем множество молодых людей. Более формально можно записать так

$$B = \{\text{множество молодых людей}\}$$

Так как, вообще, возраст начинается с 0, то нижний предел этого множества должен быть нулем. Верхний предел определить немного сложнее. На первый раз установим верхний предел, скажем, равным 20 годам. Таким образом, получаем B как четко ограниченный интервал, буквально:

$$B = [0, 20]$$

Возникает вопрос: почему кто-то в свой двадцатилетний юбилей - *молодой*, а сразу на следующий день уже *не молодой*? Очевидно, это структурная проблема, и если передвинуть верхнюю границу в произвольную точку, то можно задаться точно таким же вопросом.

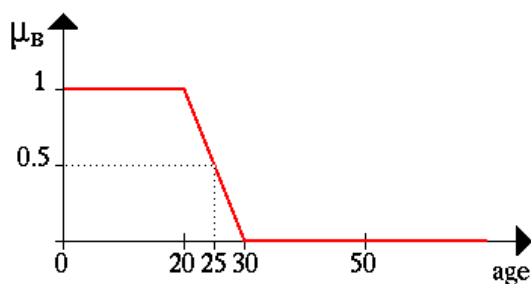
Более естественный путь получения множества B состоит в ослаблении строгого разделения на *молодых* и *не молодых*. Сделаем это, вынося не только (четкие) суждения *Да, он|она принадлежит множеству молодых людей* или *Нет, он|она не принадлежит множеству молодых людей*, но и более гибкие формулировки *Да, он|она принадлежит к достаточно молодым людям* или *Нет, он|она не очень молод|молода*.

Рассмотрим как с помощью нечеткого множества определить такое выражение, как *он|она еще молоды*.

В первом примере мы кодировали все элементы универсума рассуждения с помощью 0 или 1. Простой способ обобщить данную концепцию - ввести значения между 0 и 1. Реально можно даже допустить бесконечное число значений между 0 и 1, называемое единичным интервалом $I = [0, 1]$.

Интерпретация чисел при соотнесении всех элементов универсума рассуждений становится теперь более сложной. Конечно, снова число 1 ставится в соответствие (соотносится) тому элементу, который принадлежит множеству B , а 0 означает, что элемент точно не принадлежит множеству B . Все другие значения определяют степень принадлежности ко множеству B .

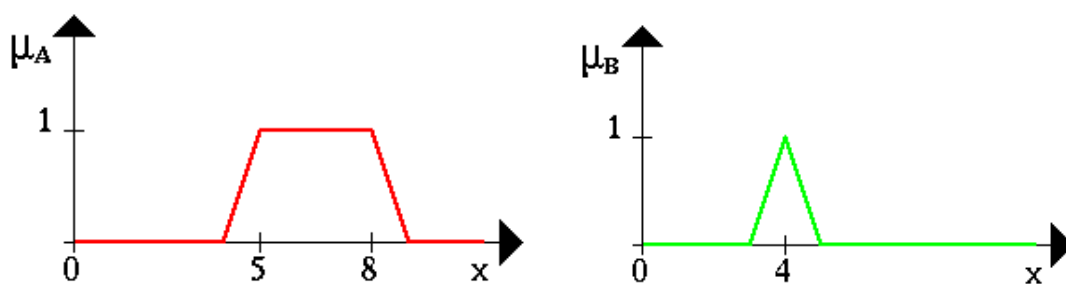
Для наглядности приведем характеристическую функцию множества молодых людей, как и в первом примере.



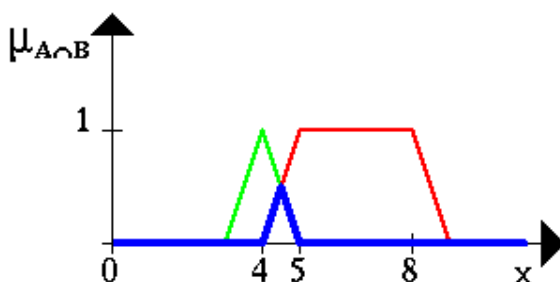
То есть 25-летние все еще *молоды* со **степенью 50 процентов**.

Аналогично действиям с обычными множествами нам потребуется определить *пересечение*, *объединение* и *отрицание* нечетких множеств. В своей самой первой работе по нечетким множествам Л.А. Задэ предложил **оператор минимума** для **пересечения** и **оператор максимума** для **объединения** двух нечетких множеств. Легко видеть, что эти операторы совпадают с обычными (*четкими*) объединением и пересечением, только рассматриваются степени принадлежности 0 и 1.

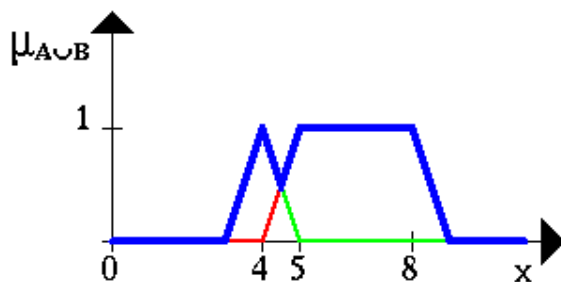
Чтобы пояснить это, приведем несколько примеров. Пусть A нечеткий интервал от 5 до 8 и B нечеткое число *около* 4, как показано на рисунке.



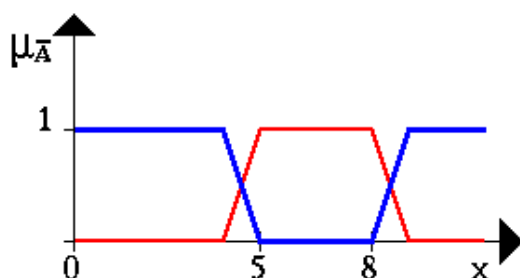
Следующий пример иллюстрирует нечеткое множество *между 5 и 8 И (AND) около 4* (синяя линия).



Нечеткое множество *между 5 и 8 ИЛИ (OR) около 4* показано на следующем рисунке (снова синяя линия).



Следующий рисунок иллюстрирует операцию отрицания. Синяя линия - это **ОТРИЦАНИЕ** нечеткого множества A .



Экспертные системы, базовые понятия

Экспертная система - это программа (на современном уровне развития человечества), которая заменяет эксперта в той или иной области.

ЭС предназначены, главным образом, для решения практических задач, возникающих в слабо структурированной и трудно формализуемой предметной области. ЭС были первыми системами, которые привлекли внимание потенциальных потребителей продукции искусственного интеллекта.

Архитектура экспертной системы. Выше уже говорилось, что экспертные системы нашли широкое применение в самых разных областях человеческой деятельности. На первый взгляд это должно привести к тому, что между ними не будет ничего сходного. В действительности в самом общем виде экспертные системы построены на одних и тех же принципах и поэтому обладают одинаковой архитектурой.

Каждая полностью оформленная экспертная система имеет четыре существенные компоненты:

- 1) база знаний;
- 2) машина логического вывода;
- 3) модуль усвоения знаний;

4) система объяснений.

Система, основанная на знаниях, в принципе может обойтись без модуля усвоения знаний (обучающая компонента) или без системы объяснений, но считается, что истинно экспертная система должна иметь все четыре перечисленные компоненты.



Схема экспертной системы

База знаний. База знаний содержит факты (утверждения) и правила. Факты представляют собой информацию краткосрочную или постоянную о состоянии объекта. Постоянная информация есть предопределенные факты, которые заносятся в базу данных. Предопределенных фактов база знаний может и не содержать. Краткосрочная информация - это факты, определяемые и изменяемые в ходе консультации с пользователем. Эти факты представляют собой информацию о конкретном состоянии конкретного объекта, относящегося к предметной области. Например: "у пациента температура".

Правила представляют собой долговременную информацию о том, как порождать новые факты и гипотезы из того, что уже известно. Наиболее простым средством представления правил являются так называемые правила продукций. Такие правила имеют формат

ЕСЛИ <утверждение> ТО <действие>

Под утверждением здесь понимается как утверждение, констатирующее какой-либо факт, так и логическое утверждение, проверяющее, выполнены ли некоторые условия. Под действием понимается как собственно действие, так и логический вывод.

Примерами возможных правил могут служить следующие:

***ЕСЛИ** у пациента насморк **И** у пациента высокая температура **ТО** пациент болен гриппом.*

Следует помнить, что знания в виде правил не воплощены в какой-либо программе, а являются данными для программы, способной их интерпретировать.

Знания могут дополнительно иметь коэффициенты достоверности, т.е. некоторые числовые коэффициенты, показывающие степень уверенности в том, что знание истинно.

Например, приведенное правило не является абсолютно достоверным. Ясно, что большинство больных гриппом имеют высокую температуру и насморк, грипп является наиболее распространенным заболеванием с этими симптомами, но такие же симптомы могут возникнуть и при других болезнях. Поэтому следует отметить возможную ошибку при использовании знаний. Самый простой (но не единственный) подход заключается в следующем: выбирать коэффициенты достоверности из отрезка $[0,1]$, считая, что нуль соответствует ложному знанию, единица - истинному, а любое из промежуточных значений - знанию, не являющемуся достоверно ни истинным, ни ложным.

Следует помнить, что правила продукций не единственный способ представления правил. Для этой цели могут использоваться также деревья решений, семантические сети и исчисления предикатов. Представления фактов и правил взаимосвязи зависят от конкретной области, для которой разрабатывается экспертная система.

Машина вывода. Машина вывода (или машина логического вывода) предназначена для принятия решения о конкретном вопросе, интересующем пользователя и относящемся к выбранной предметной области. Работа машины вывода обычно заключается в получении некоторой гипотезы и проверке ее на основе имеющихся знаний. Такая работа может быть реализована тремя способами.

Главная особенность машины логического вывода заключается в том, что ей приходится работать с ненадежными данными. Действительно, во-первых, машина строит гипотезы на основе знаний, которые, как уже отмечалось, сами по себе недостоверны; во-вторых, некоторые знания вырабатываются на основе ответов пользователя, являвшихся также ненадежными, поскольку восприятие событий зависит от личности человека.

Модуль усвоения знаний. Знания представляют собой редкий и дорогой ресурс, однако ни одна экспертная система не может без них обходиться. Традиционно их получение для базы знаний достигалось через использование высокооплачиваемого специалиста в предметной области, с которым работал специалист по технологии знаний. В течение нескольких месяцев они обговаривали закодированный вариант знаний эксперта.

Такой подход имеет следующие недостатки:

1. Высокая стоимость (два высокооплачиваемых специалиста используются длительное время);
2. Увеличение временных затрат;
3. Сложности, связанные с поиском компетентных специалистов, не занятых в других проектах.

Поэтому остро ощущается потребность в автоматизации усвоения знаний. Уже существуют системы, в которых этот процесс в значительной степени автоматизирован.

Обычно он строится следующим образом. Первоначально база знаний может содержать некоторые элементарные знания. Их получение не требует привлечения специалистов в рассматриваемой области. Например, знания могут быть взяты из обычного учебника или справочника. Затем с помощью этих знаний анализируются возможные примеры решения проблем данной области. При таком рассмотрении могут быть получены новые знания, выделяемые модулем усвоения знаний и заносимые им в базу знаний.

В дальнейшем модуль усвоения знаний занимается анализом проблем, решенных полученной экспертной системой, производя новые знания.

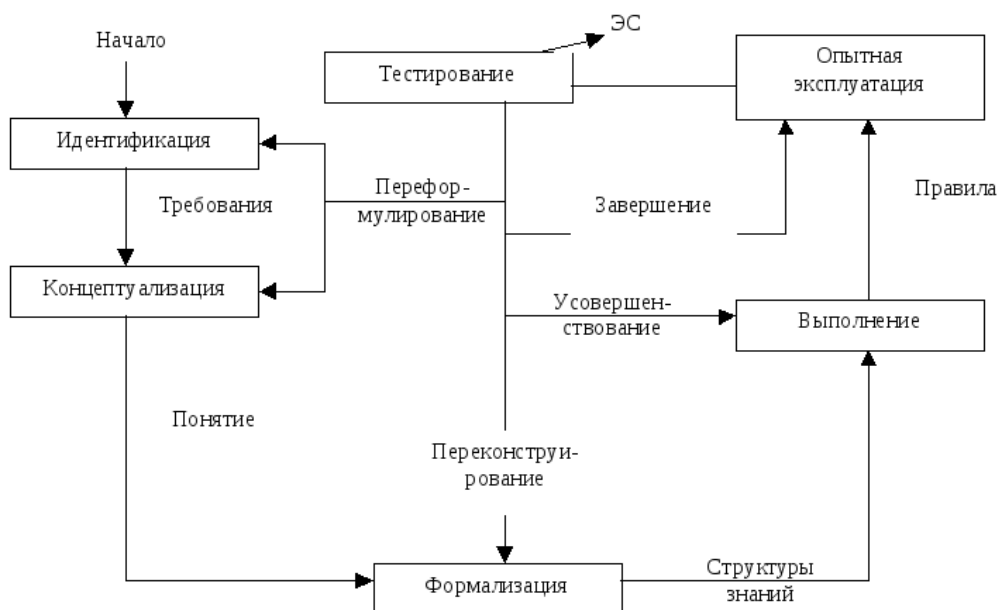
Система объяснений. Последней основной компонентой экспертной системы является система объяснений. Обычно это модуль (группа модулей), обеспечивающий возможность разъяснения пользователю, почему экспертная система приняла какое-либо заключение.

Может показаться, что эта компонента имеет слишком малое значение, чтобы являться необходимой частью экспертной системы. Однако, представим себе, что машина логического вывода пришла к заключению, значительно отличающемуся от предполагаемого вначале пользователем. Чем объяснить такое расхождение: недостатками базы знаний, ошибкой машины логического вывода или априорным заблуждением пользователя?

Для ответа на этот вопрос экспертная система должна уметь объяснить, почему и как она получила именно этот вывод, а не другой. Особенно важно, чтобы пользователь в любой момент общения с системой мог выяснить, как получено очередное заключение, если для него оно неочевидно.

Кроме возможности комментировать результаты, использование системы объяснений обладает также полезным побочным эффектом - она способна расположить пользователя к дальнейшей работе с экспертной системой, т.е. повышает свою конкурентоспособность. Таким образом, система объяснений является необходимой компонентой экспертной системы.

Методика построения



В настоящее время сложилась определенная технология разработки ЭС, которая включает следующие шесть этапов: идентификация, концептуализация, формализация, выполнение, тестирование и опытная эксплуатация.

Методика (этапы) разработки ЭС:

Этап идентификации. Этап идентификации связан, прежде всего, с осмыслением тех задач, которые предстоит решить будущей ЭС, и формированием требований к ней. Результатом данного этапа является ответ на вопрос, что надо сделать и какие ресурсы необходимо задействовать (идентификация задачи, определение участников процесса проектирования и их роли, выявление ресурсов и целей).

Идентификация задачи заключается в составлении неформального (вербального) описания, в котором указываются: общие характеристики задачи; подзадачи, выделяемые внутри данной задачи; ключевые понятия (объекты), их входные (выходные) данные; предположительный вид решения, а также знания, относящиеся к решаемой задаче.

Этап концептуализации. На данном этапе проводится содержательный анализ проблемной области, выявляются используемые понятия и их взаимосвязи, определяются методы решения задач. Этот этап завершается созданием модели предметной области (ПО), включающей основные концепты и отношения. На этапе концептуализации определяются следующие особенности задачи: типы доступных данных; исходные и выводимые данные, подзадачи общей задачи; используемые стратегии и гипотезы; виды взаимосвязей между объектами ПО, типы используемых отношений (иерархия, причина - следствие, часть - целое и т.п.); процессы, используемые в ходе решения; состав знаний, используемых при решении задачи; типы ограничений, накладываемых на процессы, используемые в ходе решения; состав знаний, используемых для обоснования решений.

Этап формализации. На данном этапе определяются состав средств и способы представления декларативных и процедурных знаний, осуществляется это представление и в итоге формируется описание решения задачи ЭС на предложенном (инженером по знаниям) формальном языке.

Выходом этапа формализации является описание того, как рассматриваемая задача может быть представлена в выбранном или разработанном формализме. Сюда относится указание способов представления знаний (фреймы, сценарии, семантические сети и т.д.) и определение способов манипулирования этими знаниями (логический вывод, аналитическая модель, статистическая модель и др.) и интерпретации знаний.

Этап выполнения. Цель этого этапа - создание одного или нескольких прототипов ЭС, решающих требуемые задачи. Затем на данном этапе по результатам тестирования и опытной эксплуатации создается конечный продукт, пригодный для промышленного использования. Разработка прототипа состоит в программировании его компонентов или выборе их из известных инструментальных средств и наполнении базы знаний.

Главное в создании прототипа заключается в том, чтобы этот прототип обеспечил проверку адекватности идей, методов и способов представления знаний решаемым задачам. Создание первого прототипа должно подтвердить, что выбранные методы решений и способы представления пригодны для успешного решения, по крайней мере, ряда задач из актуальной предметной области, а также продемонстрировать тенденцию к получению высококачественных и эффективных решений для всех задач предметной области по мере увеличения объема знаний.

Этап тестирования. В ходе данного этапа производится оценка выбранного способа представления знаний в ЭС в целом. Для этого инженер по знаниям подбирает примеры, обеспечивающие проверку всех возможностей разработанной ЭС.

Различают следующие источники неудач в работе системы: тестовые примеры, ввод-вывод, правила вывода, управляющие стратегии.

Этап опытной эксплуатации. На этом этапе проверяется пригодность ЭС для конечного пользователя. Пригодность ЭС для пользователя определяется в основном удобством работы с ней и ее полезностью. Под полезностью ЭС понимается ее способность в ходе диалога определять потребности пользователя, выявлять и устранять причины неудач в работе, а также удовлетворять указанные потребности пользователя (решать поставленные задачи). В свою очередь, удобство работы с ЭС подразумевает естественность взаимодействия с ней (общение в привычном, не утомляющем пользователя виде), гибкость ЭС (способность системы настраиваться на различных пользователей, а также учитывать изменения в квалификации одного и того же пользователя) и устойчивость

системы к ошибкам (способность не выходить из строя при ошибочных действиях неопытных пользователей).