

### ***1. Знания, как способ представления информации. Иерархия способов представления информации.***

Знания входят в состав иерархии способов представления информации. На нижнем уровне этой иерархии находится шум, состоящий из информационных элементов, которые не представляют интереса и могут лишь затруднить восприятие и представление данных. На более высоком уровне находятся бесформатные данные, содержащие элементы данных, которые в принципе могут представлять определенный интерес.

На следующем уровне находится информация, т.е. обработанные данные, явно представляющие интерес для пользователей. За этим уровнем следует уровень знаний, на котором представлена настолько важная информация, что ее следует надежно хранить и обеспечить выполнение над ней необходимых операций.

Почти на самом верхнем уровне иерархии над уровнем знаний находится уровень метазнаний. Префикс мета означает “свыше” или “дальше”. Метазнания представляют собой знания об обычных и экспертных знаниях с целью реализации процесса их интерпретации и планирования вывода.

Наконец, вершиной всех знаний является мудрость, рассматриваемая в ее философском толковании. Мудрость — это метазнания, позволяющие определять наилучшие цели в жизни и находить пути их достижения.

### ***2. Знания, как способ представления информации. Классификация знаний.***

Знания делятся:

По способу получения:

- априорные знания — это знания предшествуют знаниям, полученным с помощью органов чувств, и не зависят от них, рассматриваются как универсально истинные, и эти знания невозможно опровергнуть, не впадая в противоречия;

- апостериорные - это знания, полученные с помощью органов чувств, но эти знания не являются абсолютно достоверными и могут быть опровергнуты в процессе получения новых знаний;

По содержанию:

- процедурные знания-это знаниями о том, как сделать то или другое, это знания, которые задают последовательность действий, которые должны быть выполнены и последовательность целей которые должны быть достигнуты;

- декларативные знания - называют знания о том, является ли некоторое утверждение истинным или ложным;

- эпизодические знания — это знания которые используют исходя из схожих ситуаций, аналогичности тематики предыдущих эпизодов, которые сохраняются, обобщаются и снова применяются в новой ситуации;

- эвристические знания — относятся к практическим правилам, выведены на основе собственного опыта и позволяют выявить разницу между специалистом и новичком;

- метазнания знания - представляют собой знания об обычных и экспертных знаниях с целью реализации процесса их интерпретации и планирования вывода;

По форме существования:

- формализованные — это знания которые можно описать, документировать и рассказать другим людям;

- неформализованные — это продукт личного опыта человека, которые утверждают эти убеждения, моральные ценности и взгляды.

### ***3. Знания, как способ представления информации. Особенности знаний.***

Представление знаний — вопрос, возникающий в когнитологии (науке о мышлении), в информатике и в исследованиях искусственного интеллекта. Под термином «представление знаний» чаще всего подразумеваются способы представления знаний,

ориентированные на автоматическую обработку современными компьютерами, и, в частности, представления, состоящие из явных объектов.

Как правило, выделяют следующие свойства знаний, используемых экспертными системами (ЭС):

1. Внутренняя интерпретируемость. Каждая информационная единица имеет уникальное имя, по которому система находит и вызывает ее. В памяти хранится статическая структура информационных единиц.

2. Структурированность. Информационные единицы должны обладать гибкой структурой. Каждая единица может быть включена в состав другой, и каждую единицу можно разбить на отдельные составляющие.

3. Связность. В базе знаний между информационными единицами должна быть реализована семантическая связь. Семантика может носить декларативный или процедурный характер. В зависимости от типа связи различают отношения структуризации, функциональные, каузальные и семантические отношения.

4. Семантическая метрика. В базе знаний часто полезно задавать отношение, характеризующее ситуационную близость информационных единиц, то есть силу ассоциативной связи между информационными единицами. Такое отношение дает возможность выделять в информационной базе типовые ситуации.

5. Активность. Выполнение программ в интеллектуальных системах должно инициироваться текущим состоянием базы знаний. Особенности знаний: внутренняя интерпретируемость; знания структурированные; наличие ситуационных связей.

#### **4. Модели представления знаний**

Модели представления знаний обычно делят на логические (формальные) и эвристические (формализованные) [6]. В основе логических моделей представления знаний лежит понятие формальной системы (теории). Примерами формальных теорий могут служить исчисление предикатов и любая конкретная система продукций. В логических моделях, как правило, используется исчисление предикатов первого порядка, дополненное рядом эвристических стратегий. Эти методы являются системами дедуктивного типа, т.е. в них используется модель получения вывода из заданной системы посылок с помощью фиксированной системы правил вывода. Дальнейшим развитием предикатных систем являются системы индуктивного типа, в которых правила вывода порождаются системой на основе обработки конечного числа обучающих примеров.

В логических моделях представления знаний отношения, существующие между отдельными единицами знаний, выражаются только с помощью тех небогатых средств, которые предоставляются синтаксическими правилами используемой формальной системы.

В отличие от формальных моделей эвристические модели имеют разнообразный набор средств, передающих специфические особенности той или иной проблемной области. Именно поэтому эвристические модели превосходят логические как по возможности адекватно представить проблемную среду, так и по эффективности используемых правил вывода. К эвристическим моделям, используемым в экспертных системах, можно отнести сетевые, фреймовые, продукционные и объектно-ориентированные модели. Следует отметить, что продукционные модели, используемые для представления знаний в экспертных системах, отличаются от формальных продукционных систем тем, что они используют более сложные конструкции правил, а также содержат эвристическую информацию о специфике проблемной среды, выражаемую часто в виде семантических структур.

К основным моделям представления знаний для экспертных систем относятся:

- продукционные;
- семантические сети;
- фреймы;

логические;  
функциональные;

**4.1 Продукционная модель,** или модель, основанная на правилах, позволяет представить знания в виде предложений типа:

ЕСЛИ А ТО В

Импликация может истолковываться в обычном логическом смысле как знак логического следования В из истинного А. Возможны и другие интерпретации продукции, например, А описывает некоторое условие, необходимое, чтобы можно было совершить действие В.

Если в памяти системы хранится некоторый набор продукций, то они образуют систему продукций. В системе продукций должны быть заданы специальные процедуры управления продукциями, с помощью которых происходит актуализация продукций и выполнение той или иной продукции из числа актуализированных.

В состав системы продукций входит база правил (продукций), глобальная база данных и система управления. База правил – это область памяти, которая содержит совокупность знаний в форме правил вида ЕСЛИ – ТО. Глобальная база данных – область памяти, содержащая фактические данные (факты). Система управления формирует заключения, используя базу правил и базу данных. Существуют два способа формирования заключений – прямые выводы и обратные выводы.

В прямых выводах выбирается один из элементов данных, содержащихся в базе данных, и если при сопоставлении этот элемент согласуется с левой частью правила (посылкой), то из правила выводится соответствующее заключение и помещается в базу данных или выполняется действие, определяемое правилом, и соответствующим образом изменяется содержимое базы данных.

В обратных выводах процесс начинается от поставленной цели. Если эта цель согласуется с правой частью правила (заключением), то посылка правила принимается за подцель или гипотезу. Этот процесс повторяется до тех пор, пока не будет получено совпадение подцели с данными.

При большом числе продукций в продукционной модели усложняется проверка непротиворечивости системы продукций, т.е. множества правил. Поэтому число продукций, с которыми работают современные системы ИИ, как правило, не превышают тысячи.

**4.2 Семантическая сеть-** это ориентированный граф, вершины которого - понятия, а дуги - отношения между ними.

Термин семантическая означает смысловая, а сама семантика - это наука, устанавливающая отношения между символами и объектами, которые они обозначают, т.е. наука, определяющая смысл знаков.

Понятиями обычно выступают абстрактные или конкретные объекты, а отношения - это связи типа: "это" ("is"), "имеет частью" ("has part"), "принадлежит", "любит". Характерной особенностью семантических сетей является обязательное наличие трех типов отношений:

класс - элемент класса;  
свойство - значение;  
пример элемента класса.

Можно ввести несколько классификаций семантических сетей.

Например, по количеству типов отношений:

однородные (с единственным типом отношений);

неоднородные (с различными типами отношений).

По типам отношений:

бинарные (в которых отношения связывают два объекта);

парные (в которых есть специальные отношения, связывающие более двух понятий).

Наиболее часто в семантических сетях используются следующие отношения:

связи типа "часть-целое" ("класс-подкласс", "элемент-множество" и т.п.);

функциональные связи (определяемые обычно глаголами "производит", "влияет"...);

количественные (больше, меньше, равно...);

пространственные (далеко от, близко от, за, под, над...);

временные (раньше, позже, в течение...);

атрибутивные связи (иметь свойство, иметь значение...);

логические связи (и, или, не) и др.

Семантические сети, используемые для описания естественных языков, используют дуги типа "агент", "объект", "источник", "приемник", "время", "место", "цель". Агент - предмет, являющийся инициатором действия. Объект - предмет, подвергающийся действию. Источник - размещение предмета перед действием. Приемник - размещение предмета после действия. Время - момент выполнения действия. Место - место проведения действия. Цель - действие другого события.

Проблема поиска решения в базе знаний типа семантической сети сводится к задаче поиска фрагмента сети, соответствующего некоторой подсети, соответствующей поставленному вопросу.

Для реализации семантических сетей существуют специальные сетевые языки, например, NET и др. Широко известны экспертные системы, использующие семантические сети в качестве языка представления знаний - PROSPECTOR, CASNBT, TORUS.

Необходимо отметить ряд преимуществ семантической сети:

описание понятий и событий производится на уровне, очень близком к естественному языку;

обеспечивается возможность сцепления различных фрагментов сети;

отношение между понятиями и событиями образуют достаточно небольшое и хорошо формализованное множество;

для каждой операции над данными и знаниями можно выделить из полной сети, представляющей всю семантику (или все знания), некоторый ее участок, который охватывает необходимые в данном запросе смысловые характеристики.

Недостаток модели - сложность поиска вывода на семантической сети.

В сложных семантических сетях, включающих множество понятий, процесс обновления узлов и контроль связей между ними становится затруднительным. При этом количество опосредованных родовидовых связей между понятиями резко возрастает.

**4.3** Основная идея **фреймового подхода** к представлению знаний заключается в том, что все, что касается понятия или ситуации, не «размывается по сети», а представляется во фрейме.

Фреймом называется структура для описания понятия или ситуации, состоящая из характеристик этой ситуации и их значений.

Фрейм можно рассматривать как фрагмент семантической сети, предназначенный для описания понятий со всей совокупностью присущих им свойств. Графически фреймовая модель изображается аналогично семантической сети но принципиальное отличие состоит в том, что узел во фреймовой системе имеет вид фрейма, структуру которого можно представить так:

ИМЯ ФРЕЙМА :

(имя 1-го слота: значение 1-го слота),

(имя 2-го слота: значение 2-го слота),

- - - -

(имя N-го слота: значение N-го слота).

Ту же запись можно представить в виде таблицы, дополнив ее еще двумя столбцами. Дополнительные столбцы предназначены для описания типа слота и возможного присоединения к тому или иному слоту специальных процедур, что допускается в теории фреймов. В качестве значения слота может выступать имя другого фрейма; так образуют сети фреймов.

С каждым слотом может быть связана одна или несколько процедур, которые выполняются, когда изменяются значения слотов.

Чаще всего со слотами связываются процедуры:

Если – добавлено (выполняется, когда новая информация помещается в слот);

Если – удалено (выполняется при удалении информации из слота);

Если – нужно (выполняется, когда запрашивается информация из слота, а он пуст).

Эти процедуры могут следить за приписыванием информации к данному узлу и проверять, что при изменении значения производятся соответствующие действия.

Различают фреймы-образцы, или прототипы, хранящиеся в базе знаний, и фреймы-экземпляры, которые создаются для отображения реальных ситуаций на основе поступающих данных.

Модель фрейма является достаточно универсальной, поскольку позволяет отобразить все многообразие знаний о мире через:

фреймы-структуры, для обозначения объектов и понятий (заем, залог, вексель);

фреймы-роли (менеджер, кассир, клиент);

фреймы-сценарии (банкротство, собрание акционеров, празднование именин);

фреймы-ситуации (тревога, авария, рабочий режим устройства) и др.

Важнейшим свойством теории фреймов является заимствованное из теории семантических сетей наследование свойств. И во фреймах, и в семантических сетях наследование происходит по АКО-связям (A-Kind-Of = это). Слот АКО указывает на фрейм более высокого уровня иерархии, откуда неявно наследуются, т.е. переносятся, значения аналогичных слотов.

Основным преимуществом фреймов как модели представления знаний является способность отражать концептуальную основу организации памяти человека, а также гибкость и наглядность.

Специальные языки представления знаний в сетях фреймов FRL (Frame Representation Language) и другие позволяют эффективно строить промышленные ЭС. Широко известны такие фреймо-ориентированные экспертные системы, как ANALYST, МОДИС.

**4.5 Модели представления знаний**, в которых вычисления представляются с помощью логики предикатов, называются **логическими**, а язык программирования, используемый для описания вычислений, называется логическим языком. В основе логических моделей лежит понятие формальной системы. Наиболее распространенной формальной системой, используемой для представления знаний, является исчисление предикатов первого порядка.

Исчисление предикатов - это формальный язык, в котором знания представляются в виде логических высказываний, которые могут иметь значения ИСТИНА, либо ЛОЖЬ. В дедуктивных системах доказательства теорем, построенных на основе исчисления предикатов, знания представляются в виде множества логических высказываний, дополненного фиксированным набором универсальных правил вывода - утверждений самого общего характера о взаимосвязях между посылками и заключениями, которые с позиций исчисления предикатов всегда справедливы. В этих системах утверждения, требующие доказательства, рассматриваются как теоремы, которые нужно подтвердить или опровергнуть, исходя из данного набора истинных утверждений. Важным компонентом логических систем являются стратегии применения правил вывода: в основе

функционирования современных дедуктивных систем лежит как правило принцип резолюции Дж.Робинсона, либо одна из его модификаций

Главным достоинством исчисления предикатов является его универсальность и декларативность. Утверждения, выраженные на этом языке, могут быть без изменения смысла преобразованы в форму, удобную для обработки на ЭВМ. В то же время этому формализму присущи и существенные недостатки: в системы, построенные на основе исчисления предикатов трудно встраивать процедурные и проблемно-ориентированные знания, они плохо справляются с противоречивостью знаний и др.. В связи с указанными недостатками исчисления предикатов получили развитие другие логические подходы, например, индуктивные системы, в которых сами правила вывода порождаются системой на основании обработки конечного числа обучающих примеров, строятся с применением многозначных или модальных логик. Разработаны также псевдофизические логики, особенностью которых является использование в правилах вывода конкретных знаний о свойствах отношений, реализуемых в проблемной области, и применение аппарата лингвистических переменных.

"Классическая" модель логического программирования описывается понятиями терм, атом, программа и запрос. Терм - это либо переменная, либо константа, либо составной терм вида  $f(t_1, \dots, t_n)$ , где  $f$  -  $n$ -арный функциональный символ, а  $t_1, \dots, t_n$  - термы. Константы удобно понимать как нульарные функциональные символы. Атомарная формула (атом) имеет вид  $w(t_1, \dots, t_n)$ , где  $w$  -  $n$ -арный предикатный символ (имя отношения), а  $t_1, \dots, t_n$  - термы

Программа - это множество правил. Правило имеет вид:

$A_0 \vdash A_1, \dots, A_m \ (m > 0),$

где  $A_0, A_1, \dots, A_m$  - атомы. Атом  $A_0$  называют заголовком, а  $A_1, \dots, A_m$  - телом правила. Тело может быть пустым (при  $m=0$ ). Такие правила называются фактами. Если атом  $A_0$  в заголовке правила имеет вид  $w(t_1, \dots, t_n)$ , то говорят, что это правило для предикатного символа  $w$ . Запрос (цель) имеет вид:

$\vdash C_1, \dots, C_r,$

где  $r > 0$  и  $C_1, \dots, C_r$  - атомы.

Появление классической модели, обязано языку Пролог, который быстро завоевал популярность во всем мире и сейчас выступает как один из основных языков искусственного интеллекта

С точки зрения моделирования семантики предметной области логическая модель удобна для моделирования неклассических логик, отражающих закономерности предметной области, для моделирования продукционных правил и эвристик. Операционная семантика логических языков представляет собой логическую процедуру построения доказательств в формальном исчислении, что позволяет задачу для таких моделей мыслить как пару (условия, цель), где условия - это набор аксиом, а цель - теорема. Сдерживает развитие практических систем этого направления большая сложность логического вывода, который осуществляется в процессе решения каждой задачи пользователя, сложность совмещения процедурной и логической моделей, недостатки классической логики как формализма моделирования (принцип одной ошибки и т.п.).

**4.6 Модель представления знаний, в которой база знаний рассматривается как множество определений функций, называется функциональной моделью.** Эта модель основана на языке равенств и на операционном понимании равенств как правил редукций или правил переписывания термов. Равенствами называются формулы специального вида  $r=q$ , где  $r$  и  $q$  - термы. Под термами понимаются выражения, построенные синтаксически правильным способом из символов (имен) функций, переменных и констант (например,  $f(x)=x^2-x+5$ ). Программа предстает как пара  $(E, T)$ , где  $E$  - конечный набор равенств,  $T$  - исходный терм, а вычислительный процесс, порождаемый этой программой состоит из повторяющихся переписываний исходного терма до тех пор, пока это возможно. В последнем случае говорят о нормализуемости терма в его нормальную форму (т.е. итоговый

терм, к которому неприменима никакая аксиома из E) рассматривается как итоговый результат вычислений.

В равенства могут вкладываться самые различные математические смыслы. На равенства можно смотреть, например, как на утверждения, описывающие свойства объектов или как на уравнения. Заметим, что в последнем случае неизвестными могут считаться функции, и тогда равенства выступают в роли определений функций. Подобное многообразие смыслов порождает многообразие дескриптивных семантик языка равенств. Можно смотреть на равенства как на формулы некоторого логического (эквационального) исчисления, если нас интересует факт выводимости формулы из заданного набора аксиом, либо факт ее истинности в соответствующей математической структуре (теоретико-модельная семантика).

Среди функциональных моделей наиболее известной является бестиповое лямбда-исчисление, используемое для построения различных языков программирования (например, Лисп [46], SHEME [154], GEDANCEN [147], ML [130], [141], HOPE [119] и др. ([156]...).

С точки зрения моделирования семантики предметной области функциональная модель обладает как операционной, так и дескриптивной семантикой: функциональную программу можно рассматривать как вычислительное предписание и как конструктивное определение искомого объекта. В отличие от процедурного, функциональное программирование имеет теоретико-доказательное происхождение. Модель предметной области при функциональном подходе представляется набором определений функций, как правило, рекурсивных. Как и в процедурной модели, программа представляет собой запрос на решение одной задачи

Привлекательны следующие особенности функционального программирования:

- языки функционального программирования относятся к "что-языкам", в отличие от процедурных "как-языков";
- функциональное программирование обладает свойством прозрачности по ссылкам (функциональности), означающим, что каждое выражение языка определяет единственную величину, которую нельзя изменить ни путем вычисления, ни путем совместного использования различными функциями; в функциональных языках отсутствуют "побочные эффекты" - динамические изменения значений переменных в процессе вычислений;
- функциональные программы, как правило, оказываются короче своих процедурных аналогов, обладают более высокой степенью абстракции и доступности для понимания;
- функциональные программы пригодны для формальных преобразований с целью их оптимизации;
- системы функционального программирования поддаются естественному распараллеливанию и могут реализовываться на параллельных ЭВМ.