Система - посредник, заключение договора на поставку.

Инженерия знаний - область информатики, в рамках которой проводятся исследования по представлению знаний в ЭВМ, поддержание их в актуальном состоянии и манипулировании ими.

Knowledge system - система основанная на знаниях.

СОЗ СБЗ СУБД ЭС ИС СИИ - система искусственного интеллекта.

Структура системы, основанной на знаниях.

БЗ механизм получения решения

ИНТЕРФЕЙС

БЗ - это модель, представляющая в ЭВМ знания, накопленные в определенной предметной области. Эти знания должны быть формализованы. Знания формируются с помощью модели, а затем представляются с помощью определенного языка.

В БЗ обычно выделяются знания о конкретных объектах и правила. Эти правила исполняются как механизм получения решений, для того, чтобы из исходных фактов вывести новые.

Интерфейс обеспечивает ведение диалога на языке, близком пользователю.

Методы, основанные на использовании логических выводов, часто используется в инженерии знаний.

Понятие предметной области.

Объект – то что существует или воспринимается в качестве отдельной сущности.

Основные свойства: дискретность; различие.

При представлении знаний используется прагматический подход, т.е. выделяются те свойства объекта, которые важны для решения задач, которые будет решать создаваемая система. Поэтому система, основанная на знаниях, имеет дело с предметами, которые являются абстрактными объектами. Предмет выступает в роли носителя некоторых свойств объекта. Состояние предметной области может меняться со временем. В каждый момент времени состояние предметной области характеризуется множеством объектов и связями. Состояние предметной области характеризуется ситуацией.

Временные

Устойчивые

События

Процессы

Состояния

Постоянные свойства и соотношения

Динамическая

Статическая

Ситуация

Концептуальные средства описания предметной области.

Концептуальная модель отражает наиболее общие свойства. Для того, чтобы детализировать описание нужны языки. Характерными чертами концептуальных средств описания предметной области являются абстрактность и универсальность. Их можно использовать для описания любой предметной области.

Понятие класса объекта.

Понятие объекта – понятие множеств. Сходные между собой объекты объединяются в классы. В разные моменты времени одному и тому же классу могут соответствовать разные множества объектов.

К – класс объекта.

Кt – множество объектов класса К в момент времени t.

К = ∪ t Кt

Пример:

Группа (1999) = { ИА-1-99, ИА-1-98, … , ИА-1-94, ИБ-1-99,…}

Группа (1998) = { ИА-1-98, ИА-1-97, … , ИА-1-93, ИБ-1-98,…}

∀ t Кt = { … }

Преподавательская должность = { профессор, доцент, старший преподаватель, преподаватель, ассистент}

син

зел

1 4 Геометрическая фигура, форма квадрат, цвет синий.

желт

син

2 5

зел

3

(К : А1 К1, А2К2, … , АnКn)

имя атрибут название

классов классов

объектов атрибутная пара

Идентификация объектов может быть прямая и косвенная. В случае прямой используются имена объектов, порядковые номера объектов; косвенная основана на использовании свойств объектов.

Атрибут может быть компонентом. Под атрибутом понимается свойство, характеристика, название компонентов.

Пример:

(Геометрическая фигура:

форма Геометрическая форма

цвет Цвет)

Пары имя атрибута и значение атрибута часто совпадают.

Пример ситуации:

(Лекция:

лектор Фамилия\_лектора,

место №\_аудитории,

тема Название\_темы,

слушатель Код\_группы,

день День\_недели,

время Время\_начала)

Ситуация – показана связь между «преподаватель» и «слушатель», остальные характеристики данной ситуации.

Роли участников ситуации:

* лектор
* место
* слушатель

Характеристики ситуации:

* день
* время

(К: А1К1,А2К2, … , АnКn) – представление знаний в виде некоторой структуры.

(К: АiКi)

Пример:

(дата, число, день\_месяца)

(дата, месяц, название\_месяца)

(дата, год, год)

(геометрическая\_фигура, форма, геометрическая\_форма)

(геометрическая\_фигура, цвет, цвет)

Такому представлению знаний соответствует представление знаний в виде отдельных фактов.

(К: АiКi)

# (К: А1К1,А2К2, … , АnКn)

# Представления знаний об объектах делятся на:

# классы объекта (структура данных)

# знания о конкретных объектах (о данных)

Классы объекта.

1. (К: А1К1,А2К2, … , АnКn)

Аi – имя атрибута

Кi – классы объекта, являются значением атрибута

К – имя класса

Пример:

(преподаватели:

ФИО фамилия\_с\_инициалами,

Должность преподпвательская\_должность)

1. (К: АiКi)

Пример:

(преподаватель, ФИО фамилия\_с\_инициалами,

преподаватель, должность преподпвательская\_должность)

1. К (К1,К2, … , Кn)
2. К (А1,А2, … , Аn)

Пример:

(преподаватель (фамилия\_с\_инициалами, преподпвательская\_должность),

преподаватель (ФИО, должность))

Представление знаний для первой формы:

(К: А1К1,А2К2, … , АnКn) кi ∈ Кi

Атрибутивное представление знаний:

Пример:

(преподаватель: - представляет собой

ФИО Семенов - некоторую структуру

Должность доцент) - данных

Представление знаний для второй формы:

(К: АiКi) к ∈ К, кi ∈ Кi

Атрибутивное представление знаний в виде отдельных фактов:

Пример:

(преподаватель1, ФИО, Семенов) - 1, 2 являются связками между

(преподаватель1, должность, доцент) - фактами

(преподаватель2, ФИО, Петров)

(преподаватель2, должность, ассистент)

Представление знаний для третьей формы:

К (К1,К2, … , Кn)

(преподаватель (Семенов, доцент) - позиционное представление знаний

Если имена атрибутов отсутствуют, а сами атрибуты записываются на определённых позициях, то – позиционноё представление знаний.

Представление знаний в виде «троек» - (объект, атрибут, значение).

Для представления неточных значений используются коэффициенты уверенности – (объект, атрибут, значение, коэффициент уверенности).

Пример:

(пациент1, диагноз, колит, К760)

0 ÷ 100

0 ÷ 1

-1 ÷ 1

0 – соответствует неопределенности.

отрицательное значение – степень уверенности в невозможности значения атрибута.

(пациент1, диагноз, гастрит, К740)

\*

(пациент1, диагноз, колит, К760)

(пациент1, диагноз, гастрит, К740)

\* (пациент, ФИО, Антонов, диагноз колит К760, гастрит К740)

Представление знаний о классе объекта называется минимальным, если при удалении одного из атрибутов приводит к тому, что оставшееся множество атрибутов перестает быть представлением данного класса объекта.

Пример:

Аренда (объект\_аренды, арендатор, арендодатель, срок\_аренды, плата).

Если удалить «срок\_аренды», получится купля-продажа, а если удалить «срок\_аренды» и «плата», то получиться подарок.

Представление знаний в реляционной базе данных.

Реляционная база данных – данные хранятся в позиционном формате.

Данные хранятся в виде таблицы, где название таблицы – имя класса. Каждому классу соответствует таблица или файл БД. Имя класса - название соответствующей таблицы. Имена атрибутов – соответствующие поля таблицы (столбец). Строки таблицы – записи БД. Записи соответствует запись в позиционном формате.

|  |  |  |  |
| --- | --- | --- | --- |
| А1 | А2 | . . . . . | Аn |
| К1 | К2 | . . . . . | Кn |

Преподаватели

|  |  |
| --- | --- |
| ФИО | должность |
| Семенов  Петров | Доцент  ассистент |

Понятие атрибута в позиционной БД сохраняется.

Запись К (А1,А2, … , Аn) называется отношение между атрибутами. Такая терминология используется в реляционной БД. Идея данных в реляционной БД основана на понятие «ключ».

Ключ – набор атрибутов отношения, значение которых однозначно определяет запись в файле.

## Квартира

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| город | улица | дом | корпус | квартира | площадь | количество комнат |
| Москва  Москва | Тверская  Тверская | 2  2 | 1  1 | 47  54 | 60  50 | 2  1 |

В данном случае ключ будет состоять из нескольких полей.

Кi sup Кj является подклассом класс sup подкласс; подкласс sup класс.

Кi  является подклассом Кj, если ∀t Ki t ∈ Kj t

(Если в любой момент времени t класс Кi является подклассом Кj)

Npr – классификация сети.

Классификация сети представляется в виде иерархической структуры.

Студент sup учащийся.

sup

sup

sup

sup

sup

учащийся

Ki part of Kj - является частью Ki part Kj

sup

sup

sup

sup

sup

sup

sup

sup

лаборант

техник

доцент

ассистент

профессор

методист

заведующий кафедрой

декан

студент

аспирант

обслуживающий персонал

преподаватели

административный персонал

учащиеся

оборудование

помещение

человек

sup

Материальный объект

sup

sup

sup

sup

sup

имя

понятие

ситуация

время

пространство

Нематериальный объект

Определенный объект

Неопределенный объект

объект

студент

Ki является частью Kj, если конкретный объект класса Ki является частью однозначно определенного объекта Kj.

part

part

sup

sup



время

part

part

part

part

секунда

минута

час

сутки

месяц

год

неделя

part

part

part

пространство

sup

sup

sup

sup

город

губерния

область

деревня

штат

населенный пункт

страна

Отношение принадлежности.

k isa K - является элементом

isa

Москва

город

Ki ius K - является составляющей

К1 ius K

K2 ius K

. . . . . . .

Kn ius K

Означает, что объект класса К состоит из объектов класса К1, К2, … , Кn, причем объект класса К может включать несколько объектов класса Кi.

**Лекция №4.**

Свойства отношений.

Отношения частичного порядка обладают свойством транзитивности.

# Ki sup Kj Kj sup Km

Ki sup Km

# Ki part Kj Kj part Km

Ki part Km

Если элемент является составляющей блока, а блок составлен…

Нет циклов в графе принадлежности.

K1 ins K2, K2 ins K3,…,Kn-1 ins Kj

Неверно, что Kn ins K1

K isa Ki sup Kj

K isa Kj

Москва isa город

Город sup Населенный пункт

Москва isa Населенный пункт

K1

K3

K2

R7

R6

K5

K4

R5

R4

R1

R2

R3

Операции над классами объектов.

С помощью операций над классами объектов можно определить новый класс объектов

Ki ins K

℧ Ki множество блоков, к примеру, телевизоров

Ki sup K

℧ Ki = K

Материальные объекты делятся на три класса

Условие ∪ Помещение ∪ Оборудование = Материальный объект

Человек ∪ Помещение = Человек ∪ Оборудование = Помещение ∪ Оборудование =∅

Размещение классов объекта

Человек (Фамилия, Имя, Отчество, Год\_Рождения, пол)

Пол={мужской, женский}

Мужчина, женщина = Человек\пол

K (K1, K2, K3, K4, K5)

K\K5 – Разбиение класса по классу К5.

Объединение всех этих классов есть человек.

Мужчина⋃Женщина=Человек

Мужчина⋃Женщина=∅

(Знание\_иностранного языка

Знающий человек,

Предмет иностранный\_язык)

(K A1K1, A2K2)

K1/(K, K2)

В результате разбиения мы получаем классы людей, знающих иностранный язык.

Концептуальной схемой предметной области называется множество классов объектов, заданных на нем отношений и операциями.

Шаблонные описания состояния предметной области:

Занятия K

<Преподаватель> проводит занятия по дисциплине <название дисциплины> в группе <код группы> в <день недели> на <номер пары> в <аудитории>.

Иванов И.И. проводит занятия по дисциплине ТОЭ в группе ИТ-1-98 в понедельник на 4 паре в Г-301.

(занятия:

преподаватель Преподаватель

дисциплина Название\_дисциплины

группа Код\_группы

день День\_недели

время Номер\_пары

место Аудитория)

Концептуальные модели предметной области – концептуальная схема вместе с множеством высказываний построенных по конечному набору шаблонов.

Диаграмма сущности и связи (ER – диаграмма)

Entety Relation Diagramm

Сущность

связь

Атрибуты сущности и связи

работает

кафедра

преподаватель

N 1

\*

На 1 кафедре работает N преподавателей. «\*» – знак преподавателя – можно найти кафедру.

Связь глагол или дополнение

замещение

Группа

дисциплина

Атрибуты – прилагательное, числители, размеры, место действия

факультет

обучение

Код специальности

Назв. Спец.

специальность

Расписание нагрузки

преподаватель

решает

Зав. кафедрой

занятие

группа

дисциплина

# Логические системы (модели), на основе единственного примера поставки товара в магазин.

Логические модели представления знаний.

Описание предметной области на одном из логических языков программирования, основано на исчислении предикат.

Язык многократного исчисления предикатов 1-го порядка. Многократная логика 1-го порядка.

Для составления этого языка :

Понятие сорта соответствует понятию классов объектов.

Множество сортов S

 - на множестве задаются функциями.

f-имя функции;

сорта аргументов;

В – сорт значения функции.

Z – сигнатура – это верхний уровень представления знаний в логических моделях.

Предикат - 

Т={0;1}

ложь истина

-константа сорта В

Рассмотрим в качестве примеров обработку деталей на производстве

2-токарных;

1-фрезерный;

S={Деталь, Станок, Операция, Тип\_детали, Тип\_станка, Время }

1) дет: Операция Деталь;

f A1 B

2) ст: Операция → Станок;

3) нач: Операция → Время

4) кон: Операция → Время

5) тип\_дет: Деталь → Тип\_детали

6) тип\_ст: Станок → Тип\_станка

7) 0 : → Время

C B

1: → Время

. В

.

.

t: → Время

8) ст\_вал:→Тип\_детали

вал\_мест: → Тип\_детали

9) фрез: →Тип\_станка

ток: →Тип\_станка

10) фрез\_торц:операция Т

ток\_обр: операция Т

11) +: Время\*Время Время

12): Время\*Время Т

Знания о конкретных объектах

(нижн. Уровень представления знаний) на языке многократного исчисления предикатов наз-ся структурой интегрированной сигнатурой

1. сигнатура
2. Структура интегр. Сигнатуры.
3. Для каждого имя сорта создаётся мн-во объектов этого сорта.

Деталь = {дет.1, дет.2, дет.3, дет.4}

Станок = {ст.1, ст.2, ст3}

Операция ={опер1,опер2, опер3, опер4, опер5, опер6, опер7, опер8}

Тип\_детали = {ст\_вал, вал\_мест}

Тип\_станка = {ток, фрез}

Время = {1,2,…,t}

Объединение всех множеств - универсум.

Каждой функции и предикатов из структуры в системе соответствует множество факторов.

1. дет.(опер.1)=дет1

дет.(опер.2)=дет1

дет.(опер.3)=дет2

…………………..

2) ст.(опер.1)= ст.3

ст.(опер.2)= ст.1

ст.(опер.3)= ст.3

…………………

3) нач.(опер.1)=0

нач.(опер.2)=5

нач.(опер.3)=5

…………………..

4) конц(опер.1)=5

конц(опер.2)=12

конц(опер.3)=0

…………………

5) тип\_дет(дет.1)=ст\_вал

тип\_дет(дет.2)=вал\_мест

тип\_дет(дет.3)=ст\_вал

тип\_дет(дет.4)=вал\_мест

………………….

6) тип\_ст. (ст.1)=ток

тип\_ст. (ст.2)=ток

тип\_ст. (ст.3)=фрез

………………….

10) фрез\_торц(опер1)

ток\_обр (опер2)

фрез\_торц(опер3)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| операция | деталь | станок | начало | конец | фрез\_торц | ток\_обр |
| Опер1 | Дет.1 | Ст.3 | 0 | 5 | 1 | 0 |
| Опер2 | Дет.1 | Ст.1 | 5 | 12 | 0 | 1 |
| Опер3 | Дет.2 | Ст.3 | 5 | 10 | 1 | 0 |
| Опер4 | Дет.2 | Ст.2 | 10 | 17 | 0 | 1 |
| Опер5 | Дет.3 | Ст.3 | 10 | 16 | 1 | 0 |
| Опер6 | Дет.3 | Ст.1 | 16 | 26 | 0 | 1 |
| Опер7 | Дет.4 | Ст.3 | 16 | 22 | 1 | 0 |
| Опер8 | Дет.4 | Ст.2 | 22 | 32 | 0 | 1 |

|  |  |
| --- | --- |
| Деталь | Тип\_дет |
| Дет.1 | Ст\_вал |
| Дет.2 | Ст\_вал |
| Дет.3 | Вал\_мест |
| Дет.4 | Вал\_мест |

|  |  |
| --- | --- |
| Станок | Тип\_ст |
| Ст.1 | Ток. |
| Ст.2 | Ток. |
| Ст.3 | Фрез. |

3) Составляющая : Логические формулы

Правила построения формул:

а)константа сорта А, есть терм сорта А

б)переменная принимающая значение из сорта А, есть терм сорта А

в)если сигнатура содержит функцию-

построенные термы сортов соответственно, то

 -есть терм сорта В

г)если сигнатура содержит предикат-

 ,термы построенных сортов

, то - есть атом.

д)если - термы одинакового сорта, то выражение , то есть атом

е)Атом есть формула правильно построенная (ППФ)Переменная, входящая в атом, является свободной в этом атоме.

ж)если  построенная формула в которую свободно входит переменные х сорта А , то выражения:

 также является ППФ, переменная “x” является

связанной (в новых файлах)

з)если  уже построенные формулы, то  , также

является ППФ

Примеры:

1. Представление Знания b=> опер2 выполнены на токарном станке

тип\_ст(ст(опер2))=nток

1. Опер2 выполн на ост.1 на ст.1 нач 5 конец 12

)

3)

Лекция 8 12.11.99.

# Метод резолюций



Метод резолюций доказывает невыполнимость.

Для использования этого метода необходимо исходную формулу привести к ДНФ.

ДНФ: 

- дизъюнкция литер

рii – атом или отрицание атома.

Потом ДНФ представляют в виде множества дизъюнктов



В методе резолюций – имеется одно правило вывода



В результате из 2-х дизъюнктов получаем новую, называется руовентой



 - получаем пустой дизъюнкт , который всегда ложный.

Если множество содержит пустой дизъюнкт , то оно является не выполнимым.



Получается пустой дизъюнкт, который доказывает что данное множество является невыполнимым.

Метод резолюций применяется до тех пор пока не получится пустой– дизъюнкт



m,n – const

подстановка вместо переменной константы –унификация.

В данном случае выполняем подстановку {n/y}:

Из (1)и (2) => a(x)c(x,n) (5)

Из (3) и (5) , выполняя ь подстановку {m/n}=> c(m,n) (6)

Из (4) и (6) без подстановок => 0

Принцип резолюций в Прологе

В Прологе используются хордовские дизъюнкты, т.е. дизъюнкты, содержащие одну литеру без отрицания.

На пример 

 => 

конъюнкция

без

отрицания

Могут использоваться дизъюнкты , которые вообще не содержат литер. –

это целевое утверждение на прологе: ? – a

a: - b,c,d.

b: - e,f.

c.

e.

f.

?-a

a(1)

a(2)

a(3)

|  |  |  |  |
| --- | --- | --- | --- |
| № шага | Целевой  дизъюнкт | Исходный  дизъюнкт | резольвета |
| 1  2  3  4  5  6 | ?- a.  ?-b,c,d  ?-e,f,c,d  ?-f,c,d  ?-c,d  ?-d | a:-b,c,d.  b:-e,f  e  f  c  d | -b,c,d.  -e,f,c,d  -f,c,d  -c,d  -d  0 |

Представление программы в виде графа

a: - b;c

b: - d,e

c: - g,f.

e: - i,h

g: - h,j

d.

f.

h.

?-a

«,» - и

«;» - или

Построение графа начинается с целевого дизъюнкта.

На графе видно какие и сколько решений имеет рассматриваемая задача.

- Два решения

задачи

Продукционная модель представления знаний.

Основа для данной модели – это продукционные правила, которые имеют следующий вид

* продукционное правило >:=<имя правила>

Eсли <посылка> то < заключение> [КД=<коэффициент доверия>]

Примеры:

Правило 5

Если пол=женский

И сложение=мелкое

И вес=65 лет\_или\_больше

То относительный\_вес=изменчивый

Коэффициент доверия определяется числом 0-100

Правило 27

ЕСЛИ перспектива=отличная

И риск=высокий

ТО фактор=0 КД=10

В общем случае посылка может быть логическим выражением.

Если посылка истинна, то истинно и заключение, т.е. в заключение может быть указано какое-либо действие , которое выполняется, если посылка истинна

<посылка>::<условие>[И<условие>И…И<условие>]

<условие>::=<объект>=<значение>

объект, атрибут, значение, коэфициент доверия- представление знаний в виде четвёрки

<заключение>::=<объект>=<заключение>

<факт>:=<объект>=<значение>КД=<коэффициент доверия>

Один и тот же объект может иметь разные значения.

Многозначные объекты – объекты, которые могут иметь несколько достоверных значений.

Если объект не объявлен, как многозначный, то он может иметь несколько значений, то они не должны быть достоверными, т.е. КД= 100

## Список значений

## Список разрешённых

## значений

объект

вопрос

Для объектов, значение которое запрашивается у пользователя.

Какое сложение?

1. Мелкое
2. Среднее разрешённые значения
3. Крупное

Каков возраст

1. меньше 25
2. от 25 до 55
3. больше 55

Коэффициент доверия посылки=min(Кдусл)

 - факта, полученного в результате выполнения правила

перспектива=отличная КД=50

риск=высокий КД=70

фактор=0

Базовая структура продукционной модели представления знаний

Рабочая память(база данных)

Исходные данные

Результат

## База правил

Интерпритатор правил

Лекция 9 (Конец)

|  |  |  |  |
| --- | --- | --- | --- |
| №  шага | Конфликтное  Множество правил | Выполнение  правила | Выведенный  факт |
| 1  2  3  4  5 |  |  |  |

Выводы заканчиваются , когда достигается целевая вершина, либо не осталось применимых правил, а цель не достигнута.

Обратные выводы – выполняются сверху вниз (выводами ориентирующих на цель)

П6 П7

С2 С3

П 1 П2 П3 П4 П5

С 4 С5  С6 С7 С8

F1 F2 F3 F4 F5

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| №  шага | Цель | Конфликтное множество правил | Выполнение  правил | Подцели | Факт |
| 1  2  3  4  5  6  7  8 | С1  С2  С3  С4  С5  С6  С7  С8 | П6,П7  П1,П2  П3 | П6  П1  П3 | С2,С3  С1,С5,С3  С6,С7,С8 | F1  F2  F3  F4  F5 |

Цель – «продолжительность» –цель задаётся именем объекта.

Она сопоставляется с заключением правил и выбирается правило с заключением ,

в которых есть имя объекта. Выбираем правило, которое содержит целевой объект,

мы формируем гипотезу

Продолжительность=60

П7

В процессе гипотеза либо подтверждается либо опровергается. Выводы продолжаются до тех пор, пока какая либо не будет подтверждена, либо не будут исчерпаны все возможные гипотезы.

Используется меньшее количество проверок, т.к. в правиле бывает несколько условий и одно заключение.

# Двунаправленные выводы.

Сначала выполняются прямые выводы, на основе небольшого количества данных, в результате формируется гипотеза для подтверждения или опровержения выполняются другие выводы.

Для проверки условий правил используется аппарат активации правил, который выделяет на каждом шаге те правила, в которых проходит проверка условий.

Должны быть использованы также условия. В условиях правил выделяются индивидуальные, а затем общие.

Общие правила – правила условий применимости. Сфера применимости.

Обобщённая структура продукционного правила.

(i); Q; P; A; =B; N

(i)– имя правила:

Q –сфера применения правила;

P – условие применимости првила (логическое условие)

A=>B – ядро правило, где А- посылка, а В- заключение;

N – поставленное условие, определяетдействия, которые выполняются в случае выполнения ядра.

Р – при истинности активизируется ядро правила.

A111

P11

A112

Q1

…

……..

……..

P21

P22

P23

P12

Q2

Фрейм – структура данных для представления стереотипной ситуации

(к: А1К1, A2K2, ….,AnKn)

(к: A1k1, A2k2,….,An kn)

(имя файла:

имя слота1 (значение слота1)

имя слота2 (значение слота2)

………………………………..

имя слота n (значение слота n))

Протофрейм – знания о классе объектов.

Фрейм- экземпляр- получается из протофрейма при заполнении слотов конкретными значчениями.

В структуру фреймов обычно включают системные слоты. К системе слота относятся:

Слоты определим фреймродитель, слот, указываемый на прямые дочерние фрейма.

В качестве системы слотов: слоты, содержащие сведения о создателе программы, о её модификации.

В структуру входят:

* указатель наследования;
* указатель типа данных;
* демоны и т.п.

ЯЗЫК ФМС (FMS).

Указатели наследования могут быть:

U – unique – уникальный

S – same- какой-то

R – range – указатель границ;

0 –override – игнорировать

U – во фреймах разных уровней с одинаковыми именами будут различными.

S – слоты наследования значений из слотов высшего уровня с такими же именами

U

Человек

Вес=60

***Ребёнок***

Вес=30

***Миша***

Вес=32

Значение нижнего уравнения должно лежать внутри границ определённого в верхнем уравнении.

R

Человек

Вес=2-200

***Ребёнок***

Вес=2-50

***Миша***

Вес=32

Если значение не задано то оно наследуется из слота верхнего уравнения, а если оно задано , то наследование игнорируется.

**О**

Человек

Вес=60

***Ребёнок***

Вес=30

***Миша***

Вес=32

Лекция 11 3.12.99

Сочетание сетевой и фреймовой модели в системе представления знаний OPS-5

В этом языке есть продукционные правила и базы данных

<Аз-элемент>::=(<объект> {|<атрибут> <значение>}+)

{}+ - Может повторятся несколько раз

<Элемент –вектор>::=({ значение})

<ЭРП>::=< аз-элемент> | < элемент-вектор>

( Вещество  класс кислота

 имя 

 цвет бесцветная )

(Порядок – задач: Источник, утечки Ограждения)

Что собой представляют правила :

<Правило>::=(Р<имя правила> <посылка>  <заключение>)

<Посылка>::={<условие>}+

<Условие>::=<образец> | - <образец>

<Образец>::= <простой образец> | <образец с дезъюнкцией> | <образец с конъюнкцией>

<Простой образец>::=({значение>}+) |

# ( Порядок задач <первый> )

(<Объект> [{<атрибут> <значение>}+] )

# (Вещество )

В образце не обязательно указываются все атрибуты данного класса , т.е. мы можем записать

(Вешество  класс кислота

 имя <вещ> )

т.е. переменная кислота –вещ получит значение 

<образец с дизъюнкцией>::= (<объект> {<атрибут> <<{< значение>}+>>}+)

Значение с соответствующего атрибута элемента работой памяти должно совпадать с одним из элементов указанных в данном листе, хотя бы с одним. Эти значения задаются конкретными словами.

# (Вещество  класс кислота

 имя <вещ>

 цвет <<бесцветный желтый>> )

<образец с конъюнкцией>::= (< объект> {< атрибут>{{< значение>}+}}+)

Список значений может задаваться и в виде ограничений

# (Двигатель  мощность {<х> 100 <х> 200} )

(Двигатель  мощность 160)

<заключение>:={<действие>}+

<действие>::=(make < ЭРП> | remove <ссылка> | (modif <ссылка> {<атрибут>< значение>} +)

# (Р координировать \_а

(цель  состояние активный

 имя координировать )

Если цель находится в состоянии координировать и порядок задач не определён,

то создать

(Порядок задач ) –>

(make цель  состояние активный

 имя упорядочить задачи)

(modif1  состояние ожидания))

ссылка указывает , что модифицироваться будет элемент рабочей памяти

Стратегия решения задач основана на явном задании цели

Выполнение

1. сопоставление с элементами памяти в результате формируется конфликтное

множество правил

1. Выбор правил из конфликтного множества
2. Выполнение действий, указанных в заключении правил

Выполняется до тех пор, пока не будет достигнута цель.

# Приобретение знаний

* извлечение знаний из источника , преобразование их в нужную форму , а также

перенос в базу знаний интеллектуальной системы.

Знания делятся на :

* объективизированные ;
* субъективные

Объективизированные – знания , представленные во внешних источниках –

книгах, журналах, НИР.

* форматизированные, т.е. представлены в виде законов, формул, моделей, алгоритмов.

Субъективные – знания, которые являются экспертными и эмперическими не представлены

во внешней форме.

Знания экспертом является неформализованными, представляют собой множество эвристических приёмов и правил, позволяют находить подходы к решению задач и выдвигать гипотезы , которые могут быть подтверждены или опровержены.

Знания могут быть получены в процессе наблюдения за каким-либо объектогм.

Режимы работы инженера по знаниям, консультолога в процессе приобретения знаний.

1. протокольный анализ

* записываются рассуждения вслух в процессе решения задач.

О.с. составляются протоколы, которые анализируются

1. Интервью - ведется диалог с экспериментом, направленный на приобретение знаний.
2. Игровая имитация профессиональной деятельности.

Методы интервьюирования.

1. Рубление на ступени выделяются связи, позволяющие строить иерархические структуры
2. Репертуальная рещётка предлагаются 3 понятия и требуется назвать отличие 2-х понятие 3-его. Эксперту предлагается пара понятий и требуется назвать общие свойства =>

сформировать классы.

Методика работы конитолога по формированию поля знания

Включает 2 этапа

1. подготовительный

1.1. Чёткая подготовка задачи , которая должна решать система

* 1. Знакомство конит с литовой
  2. Выбор экспертов
  3. Знакомство экспертов с копией
  4. Знакомство эксперта с популярной методикой по искусственному интеллекту
  5. Формирование с копии поля знания

1. Основной этап
   1. накачка поля знания в режиме
   2. командная работа косметолога – анализ протокола, определение связей между понятиями , готовит вопросы к эксперту
2. Подкачка поля знания – задача вопросов эксперту
3. Формализация концептуальной задачи.
4. Проверка полноты модели

Если модель неполная , то используется 2-ое приближение.

Лекция 12 10.12. 99.

Нечёткие множества

[10,40] – толщина изделий

малая [10;20]

средняя [20;30]

большая [30;40]

степень 

принадл

1

0.7

0.1

х

10 15 40 толщина изделий

 - нечёткое множество

х - универсальное множество

 х - образуют совокупность пар А



 - называется функцией принадлежности нечёткого множества .

Значения функции принадлежности для конкретного элемента Х называется

# Степенью принадлежности

** - носитель нечёткого множества**

** **

Нормальным нечётким множеством называется множество для которого





0.6

 - нечёткое множество

Х - универсальное множество

Х - образуют совокупность пар А



 :  - называется функцией принадлежности нечёткого множества .

Значение функции принадлежности для конкретного элемента Х называется **степенью**

**принадлежности**

- носитель нечёткого множества

&

Нормальным нечётким множеством называется множество для каждого





0,6 x

Если приводить к нормальной форме => нужно поделить все её значения на .

Пример:

Пусть функция принадлежности задаётся целым числом от 10 до 40

Определить понятие малая толщина изделия.

 

1 . .

. 

.

.

.

.

| | | | | | | | x x

10 11 12 13 14 15 16 17 18 18

## Операции над нечёткими множествами

1. Объединение нечётких множеств





1. Пересечение нечётких множеств





1. Дополнение нечёткого множества





 

  x x

μ‾A

x

Начало 12 и 13 лекции.

μA∩B

x

1. Декартовое произведение нечетких множеств.

‾A1,‾A2,….,‾An

x1,x2,…,xn

x1∈ X1  x2∈ X2  … xn∈Xn

‾A1 x‾A2 x … x‾An = {<μx (x1,x2,…,xn )/( x1,x2,…,xn )>}

μx (x1,x2,…,xn ) = min{μA1 (x1), μA2 (x2)…μAn (xn) }

‾A = {<1/10>, <0.8/15>, <0.2/20>}

‾B = {<0.7/2>, <0.5/4>}

‾A x‾B = {<0.7/(10,2)>, <0.7/(15,2)>, <0.2/(20,2)>, <0.5/(10,4)>, <0.5/(15,4)>, <0.2/(20,4)>}

1. Возведение нечеткого множества в степень.

‾An = {<μAn(x)/x>}

‾A2 = con(‾A) - концентрация

‾A0.5 = dil(‾A) – растяжение

μA

1 ‾A2

‾A0.5

0<x<1 x

x2 < x x0.5 > x

μA

1

x

Методы определения функции принадлежности.

Немного больше 2. От 0 до 5.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| x | 0 | 1 | 2 | 3 | 4 | 5 |
| n1 | - | - | - | 10 | 8 | 4 |
| n2 | 10 | 10 | 10 | - | 2 | 6 |

μA = n1 / (n1 + n2)

μA (0) = 0

μA (1) = 0

μA (2) = 0

μA (3) = 1

μA (4) = 0.8

μA (5) = 0.4

Метод рангирования.

ч

u

p

ч

u

p

u ч

р

0 Xчастота

редко иногда часто

Нечеткая переменная.

р u ч 1

x

μ

x

L

<α, x, ‾A>

α - имя нечеткой переменной

х – область ее определения

‾А – смысл, нечеткое множество определяет семантику нечеткой переменной

Лингвистическая переменная.

<β, T, Х, G, M>

β - имя лингвистической переменной

Т – базовая терм множество – образует имена нечетких переменных {редко, иногда, часто}, являющихся лингвистическими переменными

Х – носитель лингвистических значений [0; 1] – область определения

G – синтаксическая процедура

М – семантическая процедура

Синтаксическая процедура в виде грамматических терминов, символы которых составляют термы из терм множеств {и, или, не}, модификаторы типов {очень, слегка, не и т.д.}

β - частота

Т = {редко, иногда, часто}

X = [0;1]

Не редко

Очень часто

Такие термины вместе с исходными образуют производную терм множества.

Т\* = Т ∪ G (T)

Семантические процедуры позволяют переписать термо-нечеткую семантику.

М(α1 или α2) = ‾А1 ∪‾А2

(α1 , х1 , ‾А1)

(α2 , х2 , ‾А2)

М(α1 и α2) = ‾А1 ∩‾А2

М(^α) = ^‾А1

М(очень α) = con (‾A )

М(слегка α) = dil (‾A )

Сценарий.

Является классом фреймовых моделей представления знаний, где в обобщенной и структурной форме представлены знания о последовательности действий, событий типичных для предметной области. Рассмотрим стереотип каузальный сценарий – определяет последовательность действий необходимых для достижения целей, это фреймовая модель.

(kcus имя:

имя слота 1(значение слота 1);

имя слота 2(значение слота 2);

…

имя слота n(значение слота n))

(kcus

деятель

цель деятеля

посылка

ключ

следствие

системное имя)

Посылка определяет действия, которые должны быть выполнены раньше ключевого действия, необходимые для его действия. Последствие – заключительное действие. Системное имя – сценарий.

(kcus «тушение пожара»:

деятель (S: )

цель деятеля (С: «прекращение пожара»)

П11, П12 посылки (cus: «поиск средств тушения»R1, «транспортные средства тушения»)

К1  ключ (f: «использование средств тушения для полного прекращения огня»)

следствие (Р: «прекращение огня»)

системное имя (sys: cus\*1))

R1 – быть раньше

(kcus «поиск средств тушения »:

деятель (S: )

цель деятеля (С: «нахождение средств тушения»)

П121, П22 посылки (cus: «определение координат местонахождения средств тушения»R1, «перемещение к месту нахождения средств тушения»)

К2  ключ (f: «схватывание средств тушения»)

следствие (Р: «нахождение у места расположения средств тушения»)

системное имя (sys: cus\*2))

(kcus «транспортировка средств тушения к месту пожара»:

деятель (S: )

цель деятеля (С: «доставка средств тушения к месту пожара»)

П31, П32 посылки (cus: «наличие средств тушения»R1, «определение координат места пожара»)

К3  ключ (f: «движение к месту пожара»)

следствие (Р: «нахождение на месте пожара средств тушения»)

системное имя (sys: cus\*3))

Пополнение знаний на основе сценария.

<посылки>R:<ключ>

Последовательность действий:

Д = cus: П11 R1 cus: П12 R1 K1 =

П21R1П22R1K2 П31R1П32R1K3

= П21R1П22R1K2 R1 П31R1П32R1K3 R1 K1

Посылки определяют действия, которые должны быть выполнены раньше ключевого действия, необходимы для его действия. Следствие заключительное действие. Системное имя сценарий.

Пополнение знаний на основе псевдофизических логик.

Р1 – посадка самолета

Р2 – подача трапа

Р3 – выход пассажиров из самолета

Р4 – подача автобуса

Р5 – прибытие на аэровокзал

Структура текста на лингвистическом уровне представляется следующей формулой:

TS = PR4dt&P1R3 10,αP2&P2R1P3&P4R3 2,αP5

t = 15 часов 20 минут

PR4dt , P1R3 10,αP2 → P2R4 dt + 10

P1R3 10,αP2 → P1R1P2

P4R3 2,αP5 → P4R1P5

TS\* = P1R1P2& P1R1P3& P2R1P3& P4R1P5

Модели и методы обобщений знаний.

Под обобщением понимается процесс получения знаний, объясняющих имеющиеся факты, а так же способных классифицировать, объяснять и предсказывать новые факты. Исходные данные представляются обучающей выборкой. Объекты могут быть разбиты на классы. В зависимости от того, заданы или нет априорные разбиения объектов на классы, модели обобщения делятся на модели обобщения по выборкам и по классам.

Кνi

ν+ = {01+, 02+…0nj+} – положительная выборка.

Может задаваться отрицательная выборка ν- = {01-, 02-…0ьj-}

Требуется найти такое правило, которое позволяет установит, относится или нет объект к классу Kj.

В моделях обобщения по данным выборка представляется множеством объектов класса. Методы обобщения делятся на методы обобщения по признакам и структурно-логические методы обобщения.

Z = {z1, z2, …, zr}

Zi = {zi1, zi2, …, zini}

Объект характеризуется множеством значений признаков Qi = {z1j1, z2j2, …, zrjr}.

Структурно-логические методы обобщения используются для представления знаний об объектах, имеющих внутреннюю структуру среди структурно-логических методов. Можно выдвинуть два направления: индуктивные методы нормального исчисления и методы обобщения на семантических сетях.

Алгоритм обобщения понятий по признакам.

hij =

1, если i-ый признак имеет j-ое значение

0, в противном случае

Правила определения принадлежности объектов к некоторому классу представляются в ряде логических формул элементами которых являются hij и функции ∩, ∪, ‾.

Пример:

Z = {z1, z2} {пол, возраст}

Z1 = {z11, z12} {м, ж}

Z2 = {z21, z22, z23} {молодой, средний, старый}

νj+ = {01+, 02+} νj- = {01-, 02-, 03-}

01+ = (z11, z21) 02+ = (z11, z22)

01- = (z11, z23) 02- = (z12, z21) 03- = (z12, z22)

&i hij - обобщенное конъюнктивное понятие

0 = max(xij – 1/αi), где 0 – критерий, xij – частота появления некоторого значения признака, αi – количество признаков.

Пример:

0 = 3/5 – 1/2 = 0.1

νj+ = {01+, 02+} νj- = {01-}

ν-1+ = 0 ν-1- = {02-, 03-}

ν-

ν+

ν-1-

ν-1+

νj-

νj+