

Conhecimento e Raciocínio

Aula 2

Sistemas Periciais

2.1 Introdução

Viriato A.P. Marinho Marques

DEIS - ISEC

2010 / 2011

1. Introdução

1.1 Conceitos Base

Sistema Pericial (SP), Especialista ou *Expert System* (ES)

Um programa que exhibe, num dado domínio, um certo grau de perícia na resolução de problemas, comparável ao exibido por um perito humano.

Perito

Um indivíduo capaz de resolver problemas complexos num período de tempo reduzido;

Uma pessoa que, devido ao seu treino e experiência, é capaz de fazer coisas que as restantes pessoas não conseguem fazer;

Um perito recorre a truques e métodos próprios que pode mesmo ter dificuldade em explicar.

1.1 Generalidades

KBES - Knowledge Based Expert Systems

Para tornar alguns programas inteligentes é necessário dotá-los de conhecimento específico e de alta qualidade, prático, heurístico, acerca da área onde se pretende aplicá-los. O termo *KBES - Knowledge Based Expert Systems* traduz este facto.

O conhecimento é extraído de peritos do domínio através de entrevistas, questionários, observação, etc. e também de documentação.

Engenharia do Conhecimento (*Knowledge Engineering*)

Consiste na extracção de conhecimento a partir dos peritos e sua representação numa Base de Conhecimento (*Knowledge Base*).

Um *Knowledge Engineer* desenvolve modelos de Bases de Conhecimento.

A sua atenção foca-se sobre o conhecimento em si e no modo de o representar. O suporte (programa) utilizado não é assunto importante nesta etapa de desenvolvimento de um SP.

1.1 Generalidades

Base de Conhecimento

Uma Base de Conhecimento representa conhecimento.

- O que é exactamente o conhecimento?
- Como é representado?

O conhecimento pode assumir as mais diversas formas, mas geralmente inclui algum tipo de relação. Por exemplo:

1. Relação entre sintomas e causas
2. Relação entre objectos e seus atributos
3. Relação entre atributos e domínios
4. Relação de Classificação (*Is A*), de Composição (*Is Part Of*)...
5. Experiências passadas contextualizadas (*Case Based Reasoning - CBR*)
6. etc.

Para conferir permanência ao conhecimento podem utilizar-se bases de dados. Mas uma Base de Conhecimento **não é** uma Base de Dados.



1.1 Generalidades

Características de um SP

Um ES deve possuir as seguintes características:

1. Obter os resultados de uma forma prática e eficiente
2. Interagir com os utilizadores eventualmente em linguagem natural
3. Raciocinar usando descrições simbólicas
4. Poder trabalhar com dados incertos
5. Explicar as suas dúvidas e justificar as suas conclusões

Razões para o seu Desenvolvimento e Aplicação

As principais razões para desenvolvimento e aplicação de SP são:

1. Serem sistemas não perecíveis
2. Proporcionarem fácil e imediata transferência de conhecimento
3. Permitirem uma fácil documentação
4. Apresentarem um comportamento consistente
5. Serem baratos a médio prazo



1. Introdução

1.2 Tipos

Os SP são um dos produtos da IA de maior divulgação e sucesso. Consideraremos 3 tipos:

1. Protótipos
2. Protótipos com Aplicação de Campo
3. Baseados em Ferramentas de Desenvolvimento

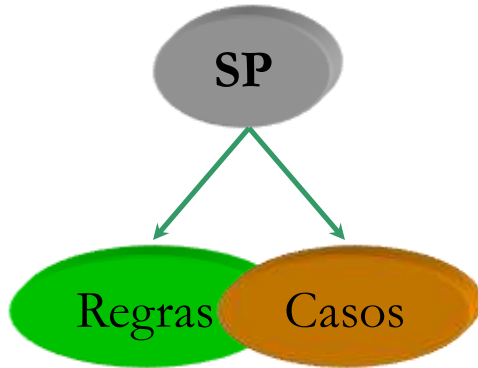
Consideraremos também duas grandes famílias base, sobre as quais focaremos a atenção:

1. Baseados em Regras
2. Baseados em Casos

Estas filosofias base podem sobrepor-se e ser complementadas com muitas outras técnicas da IA, nomeadamente Indução (*ID3*), Lógica Difusa (*Fuzzy Logic*), Algoritmos Genéticos (*AG*), etc.



1.2 Tipos



Os SP "mais famosos" são todos designados por nomes próprios interessantes, que normalmente aparecem em maiúsculas; alguns são elucidativos.

Eis alguns exemplos:

Regras: MYCIN
CATS-1
XCON
PROSPECTOR

Diagnóstico de infecções hospitalares
Diagnóstico de falhas em locomotivas
Configuração de mini-computadores DEC
Prospecção mineira

Casos: CYRUS
PATDEX
CHEF
CLAVIER
MEDIATOR

Informação acerca das viagens de Cyrus Vince
Diagnóstico de falhas em sistemas complexos
Criação de receitas culinárias
Programação de cargas de um autoclave
Conselheiro de questões jurídicas

1.2 Tipos

Regras de Produção

Uma Regra de Produção é literalmente uma proposição IF...THEN que permite obter uma dada **conclusão** a partir de **factos**.

Exemplo (MYCIN)

```
IF    the stain of the organism is gram positive
AND   the morphology of the organism is cocus
THEN  there is a suggestive evidence (0.7) that the organism
       is streptococcus
```

Contudo, é preciso compreender que estas regras não são equivalentes aos IF...THEN das linguagens procedimentais !

- As regras são executadas (disparadas, *fired*) em função dos factos (antecedentes) conhecidos num dado instante do processamento

1.2 Tipos



- O consequente de uma regra pode constituir um novo facto que assim determinar o disparo de outra(s) regra(s)
- Este mecanismo de disparo é controlado por um componente do SP designado por Motor de Inferência
- A mesma regra não é disparada mais que uma vez

De modo muito *naïf* um Motor de Inferência pode ser visto como um ciclo *While* que "activa" consequentes em função do valor verdadeiro ou falso dos antecedentes, enquanto não for encontrada uma resposta para o problema posto ao SP.

O Disparo das Regras

Enquanto !resposta

Para todas as regras não disparadas

Se antecedente verdadeiro

Activar consequente

Fim

Fim

1.2 Tipos

Casos

Um Caso é essencialmente a representação de uma ocorrência passada, à qual está associada a solução então utilizada.

Um caso é uma unidade de conhecimento contextualizado.

Raciocínio Baseado em Casos (*Case Based Reasoning - CBR*)

A solução para o Caso A será utilizada directamente para o caso B, ou adaptada ao caso B, se B for *suficientemente* semelhante a A.

A chave do paradigma CBR é o conceito de semelhança.

A semelhança é avaliada com base em valores de atributos de objectos do caso actual e dos casos passados.

Os casos são armazenados numa biblioteca de casos (*Case Library*).



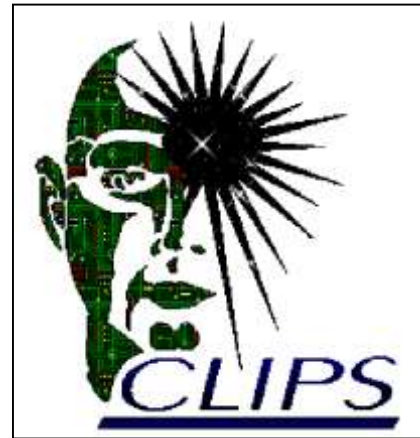
1. Introdução

1.3 Ferramentas de Desenvolvimento

Os SP aplicados em casos comuns e reais são desenvolvidos através de ferramentas próprias designadas por *shell* e *CBR shell*.

Existem muitas destas ferramentas:

- LPA *Flex/Flint*
- NASA *Clips*
- NRC *Fuzzy Clips*
- ExSys *Corvid*



1.3 Ferramentas

- | | |
|-----------------------|--|
| • Attar Software | <i>∞PertRule Knowledge Builder, etc.</i> |
| • GoldHill | <i>GoldWorks</i> |
| • MindBox | <i>Art*Enterprise</i> |
| • Oracle (ex-Halley) | <i>CLA Server, Easy Reasonner, etc.</i> |
| • Inductive Solutions | <i>Induce-IT</i> |
| • SHAI | <i>Esteem</i> (já não comercializado) |
| • I-Soft | <i>Recall</i> |
| • empolis (tecInno) | <i>kms-knowledge management suite</i> |
| • Kaidara (AcknoSoft) | <i>Kaidara Advisor (KATE)</i> |
| • Simon Fraser Univ. | <i>Case Advisor</i> |
| • Edinburgh Univ. | <i>CBR Shell</i> (para ensaios) |
| • DFKI | <i>myCBR</i> (open source) |



1. Introdução

1.4 Exemplos

Como se pode concluir, os SP são aplicáveis aos mais diversos domínios:

- Diagnóstico médico
- Diagnóstico de Falhas em equipamento
- Aconselhamento de investimento
- Selecção de produtos (*e-commerce*)
- Planeamento e Controlo de Produção
- Atribuição de portas aos aviões em aeroportos
- Gestão de cargas e descargas em portos (p.e. Singapura)
- Configuração de computadores
- Localização e projecto de instalações
- Criação de menus e receitas de cozinha
- Aconselhamento jurídico
- Projectos de arquitectura
- Auditorias
- etc.



1.4 Exemplos

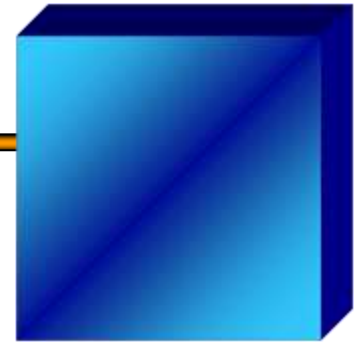


Exemplo 1

General Electrics CATS-1

David Smith, engenheiro chefe de manutenção de locomotivas diesel-eléctricas na General Electric's (GE), trabalhava nesta companhia há 40 anos e ia reformar-se. Viajava por todo o país para aconselhar e treinar jovens engenheiros. Contudo este método revelou-se pouco eficaz em termos de disseminação de *know-how*, porque Smith...era uma só pessoa. Em 1980 a GE decidiu implementar um SP. Os engenheiros passaram meses a entrevistar Smith e 3 anos a desenvolver um protótipo. Em 1983 nascia o CATS-1. Este SP passou a permitir que um técnico jovem diagnosticasse virtualmente qualquer tipo de avaria passando apenas alguns minutos ao computador. Simultaneamente, a GE instalou um SP em cada centro de manutenção para difusão do *know-how* corporativo, dado que também podiam explicar as conclusões a que chegavam.

1.4 Exemplos



Exemplo 2 - Interface e Justificação (*Why & How*)

Blue Box

BlueBox: Please characterize the degree of anxiety of Mr.Blue

User: MILD

BlueBox: Is the treatment of the patient planned in California ?

User: WHY ?

BlueBox: This will aid in determining the types of treatment recommended:
It has already been established that:

ECT era de
difícil
prescrição na
Califórnia
devido à
legislação

ECT is a treatment under consideration. Therefore

IF The treatment is NOT planned in California

AND Mr.Blue has given his consent for electro-convulsive
therapy

THEN ECT is the type of treatment recommended (RULE 87)

1.4 Exemplos



Exemplo 3

Demo ExSys - Selecção de uma Máquina de Vídeo

The use of an ES for product selection provides an emulation of the interaction a customer would have with an experienced human salesperson.

- The system interacts with the customer by asking questions to obtain data on customer needs. When a customer's answer indicates more details are needed in a specific area, the system asks follow-up questions.
- When the customer desires features that can not all be met with a single product, the system ranks products to find the "best fit" and explains the options. The customer is given the best alternatives, even though none may exactly match their request. Traditional database approaches to product selection eliminate a product if it does not meet all user requests.
- The ES presents overall comments on the user needs and information about how each recommended product will fit the user's needs. The same assistance that would be provided by a human expert can be delivered via the expert system.

1.4 Exemplos



Exemplo 4

Demo CLIPS - O Nome de um Animal (Classificação)

The origins of the C Language Integrated Production System (CLIPS) date back to 1984 at NASA's Johnson Space Center. At this time, the Artificial Intelligence Section had developed over a dozen prototype ES. However, despite extensive demonstrations of the potential of ES, few of these applications were put into regular use [...] due to the use of LISP. Three problems hindered the use of LISP based expert system tools within NASA: the low availability of LISP, the high cost of state-of-the-art LISP tools and hardware, and the poor integration of LISP with other languages.

The AI Section felt that the use of C would eliminate most of these problems [...]. To meet all of its needs in a timely and cost effective manner, it became evident that the AI Section would have to develop its own C based ES tool. The prototype version of CLIPS was developed in the spring of 1985 in a little over two months. Particular attention was given to making the tool compatible with expert systems under development. Thus, the syntax of CLIPS was made to very closely resemble the syntax of a subset of the ART ES tool developed by Inference Corporation. Although originally modelled from ART, CLIPS was developed entirely without assistance from Inference or access to the ART source code.

1.4 Exemplos

The logo for SADEX, featuring the word "SADEX" in a bold, yellow, sans-serif font. To the right of the text is a small, stylized graphic of a person in a purple suit standing on a globe.

CBR Fuzzy ES for Fault Diagnosis

Exemplo 5

Viriato M. Marques - SADEX - Diagnóstico de Falhas (Fuzzy CBR ES)

- Base de dados relacional e arquitectura *three-tiered* para acesso via Internet à biblioteca de casos supervisionada por um perito do domínio.
- Capacidade de aprendizagem baseada em relatos dos técnicos de manutenção, que podem incluir informação difusa.
- Critérios de semelhança actualizados com base em princípios heurísticos onde intervêm quantidade de informação, frequência de ocorrência e dispersão de observações tipo por classes de falhas.
- Permite a descrição de ocorrências por valores numéricos, lógicos e termos linguísticos, definidos num modelo que simula o interface homem/ mundo.
- Árvores de classificação de equipamentos e componentes permitem realizar adaptação transformacional baseada em princípios taxonómicos.
- O diagnóstico incorpora meta-conhecimento expresso por regras
- Proporciona acesso a procedimentos de manutenção, suportando campos multimédia.