

XIV Conferencia Latinoamericana de Informática
17avas. Jornadas Argentinas de Informática
e Investigación Operativa.
Buenos Aires, Setiembre 1988.

VERIFICAÇÃO DE BASE DE CONHECIMENTO

JOSE CARLOS BARBOSA DAMSKI

SMART SYSTEMS CONSULT. E INF. S/C LTDA

Av. Francisco José Longo 149/117

cep 12245 São José dos Campos - S.P.

Tel. (0123) 22-3244

RESUMO : No presente trabalho é mostrado como verificar Bases de Conhecimento para Sistemas Especialistas, juntamente com a Aquisição de Conhecimento. É visto também como implementar um sistema que faça parte da verificação e análise da B.C.

INTRODUÇÃO :

Dentro da área de Inteligência Artificial, os Sistemas Especialistas destacam-se pelas várias aplicações em que podem atuar de forma simples e flexível.

Existem muitos ambientes para desenvolvimento de Sistemas Especialistas disponíveis hoje no mercado internacional, para diversos equipamentos (de grande a pequeno porte) e variadas finalidades (desde de ambientes demonstrativos até ambientes profissionais).

Com o uso destes ambientes são desenvolvidas muitas aplicações, desde a escolha de um vinho até a configuração automática de ambientes computacionais.

Mas, curiosamente, muitos destes ambientes não auxiliam em duas fases importantes na construção de Sistemas Especialistas, que são: a Aquisição de Conhecimento e a Verificação da Base de Conhecimento.

A Aquisição de Conhecimento é um problema muito complexo que aborda desde a relação homem x homem até a filosofia (epistemologia). Neste trabalho não abordaremos objetivamente este assunto.

Depois do "conhecimento" adquirido, este é expresso em uma (várias) representação(ções) formando a Base de Conhecimento.

A partir daí pode-se desenvolver testes para a

verificação (quanto a correção) desta Base.

Nosso trabalho visa apresentar soluções para esta etapa, tão importante para a confecção de Sistemas Especialistas.

DESENVOLVIMENTO DE SISTEMAS ESPECIALISTAS

Existem diversas formas de representar "conhecimento" em Sistema Especialista: regras de produção, quadros ("frames"), redes semânticas, etc. A mais difundida atualmente é a de regras de produção, inclusive por sua semelhança ao cálculo proposicional e de predicados da lógica matemática.

Dentro do paradigma de representação por regras, existem diversas formas sintáticas de expressá-las. Existem também diversas formas de fazer inferências e calcular o valor das condições das regras.

Nosso escopo de trabalho é em regras sob uma forma sintática geral o suficiente para servir de exemplo de solução para outras formas sintáticas.

Acreditamos que o sistema de verificação de Base de Conhecimento deve ser feito "ad hoc", i.e., para cada ambiente de desenvolvimento de Sistemas Especialistas deve ser feito um particular sistema de verificação de Base de Conhecimento.

Para confeccionar um Sistema Especialista é necessário passar pelas fases :

1. Aquisição do Conhecimento
2. Formalização do Conhecimento
3. Verificação da Base de Conhecimento
4. Teste de campo do Sistema Especialista
5. Uso

que podem fazer ciclos entre diversas dessas fases. A fase 3 auxilia a fase 1 e garante melhor resultado em 4 e 5.

A fase de verificação é importante, pois a Aquisição de Conhecimento não garante que todo o conhecimento (ou que se deseja) está expresso e se esta correto semântica e pragmaticamente em relação ao especialista.

Tipos de Verificação

Para se concretizar a fase 3 do tópico anterior, dividimos a validação em 3 etapas:

- a) Validação em tempo de concepção das regras
 - a.1) Quando a regra é concebida, por quem quer que seja, o Engenheiro de Conhecimento deve pedir a aprovação por parte do especialista quanto a semântica das condições que compõem a regra. Deve ser feita também a avaliação pragmática da regra.
 - a.2) O especialista deve ser solicitado a avaliar também os cachos decisoriais (conjunto pequeno de regras que levam

a um objetivo).

b) Validação em tempo de prototipação

b.1) Quando o Sistema Especialista está na fase de protótipo, pode-se fazer teste de validação colocando o Especialista frente a frente com o Sistema Especialista. O Especialista deve fazer todos os testes que julgar conveniente e lançar mão dos recursos que o ambiente computacional pode lhe oferecer : explicação do raciocínio usado; mostrar todos os caminhos que o Sistema Especialista tenta para obter a solução, etc.

b.2) Uma variante do caso anterior é o Engenheiro de Conhecimento criar situações de teste e oferecer, separadamente, ao Sistema Especialista e ao Especialista e posteriormente confrontar as respostas.

c) Validação Automatizada

Nos 2 itens anteriores os métodos de validação estão centrados no homem, que por um lado é bom devido a sua grande capacidade de análise e por outro é ruim pois pode ficar incompleto e não abarcar toda a Base de conhecimento. Visto que é praticamente inviável um Engenheiro de Conhecimento conceber toda a Base de Conhecimento em sua memória (para B.C. de tamanho médio, mais de 500 regras).

Para contornar este problema, deve-se desenvolver uma ferramenta computacional que faça a análise da Base de Conhecimento, não excludente com as feitas nos itens a e b acima, mas que tenham um enfoque mais sintático e exaustivo em

relação a base total.

Este sistema deve atuar em:

1. Verificação da consistência
 - 1.1 Regras redundantes
 - 1.2 Regras conflitantes
 - 1.3 Regras mais gerais (que outras)
 - 1.4 Condições desnecessárias (nas premissas)
 - 1.5 Detecção de encadeamento cíclico nas regras
2. Verificação da Completeza
 - 2.1 Referência de atributos fora do escopo (contexto)
 - 2.2 A não referência de atributos do escopo
 - 2.3 O encadeamento completo das regras
 - 2.4 Mostrar as condições não dedutíveis pelas regras
 - 2.5 Verificar se todas as conclusões finais recaem sobre o universo final da aplicação
3. Análise dos Fatores de Certeza
 - 3.1 Conclusões com F.C. conflitantes
 - 3.2 Verificação dos problemas citados no item 1 acima, levando em consideração os F.C.
 - 3.3 Conclusões que nunca atingem o limiar de veracidade

No item "c" acima, os sub-itens 1 e 2 são discutidos no próximo tópico e o sub-item 3 em [5].

IMPLEMENTAÇÃO DA VERIFICAÇÃO MECANIZADA

Nossa implementação é baseada na representação de regras de produção na forma :

SE

$P_1 (T_1, T_2, \dots, T_n) \text{ E}$

$P_2 (T_1, T_2, \dots, T_m) \text{ E}$

ENTÃO

$P_n (T_1, T_2, \dots, T_j)$

onde P_i são símbolos predicativos e T_i são termos que pode ser constantes ou variáveis (estas são expressas por um "*" antes do nome).

Além do conjunto de regras, temos também a estrutura do dicionário.

O dicionário é constituído pela definição do escopo de cada termo. O escopo é o conjunto de valores (atômicos) que tornam o predicado verdadeiro. Na realidade este tipo de definição já existe em alguns ambientes de desenvolvimento de Sistemas Especialistas, como sendo a definição de um parametro. Por ex:

COR (*X)

é definido no dicionário todas as cores que são válidas no lugar da variável "x".

A partir deste conceito resolveremos muitos problemas [3], conforme um deles, abaixo.

Um dos problemas é a ocorrência de ciclos ("loops") durante o processo de inferência sobre uma Base de Conhecimento. Por exemplo, suponha a seguinte Base de Conhecimento com 3 regras:

```

R1  -  SE          A(*X)
        ENTÃO      B(*X)

R2  -  SE          C(*Y)
        ENTÃO      A(*Y)

R3  -  SE          B(*Z)
        ENTÃO      C(*Z)

```

em inferência regressiva (encadeamento para trás). Notamos claramente o ciclo a partir da regra 1, que para resolver B(*X) é necessário resolver A(*X). Para resolver A(*X), unifica-se com a conclusão da regra 2, tornando-se necessário resolver C(*Y). Resolvendo com a regra 3, precisa-se resolver B(*Z) voltando a regra 1 fechando o ciclo.

Para saber se este ciclo é possível de acontecer, e caso seja, em qual escopo, devemos calcular: (E abrevia escopo ; # simboliza intersecção)

1. Na regra 1 : $E[*X] = E[B(*X)] \# E[A(*X)]$
2. fazendo $*Y = *X$ temos $E[A(*Y)] = E[*X]$
3. Na regra 2 : $E[*Y] = E[A(*Y)] \# E[C(*Y)]$
4. fazendo $*Z = *Y$ temos $E[C(*Z)] = E[*Y]$
5. Na regra 3 : $E[*Z] = E[C(*Z)] \# E[B(*Z)]$

Se $E[*Z] \neq E[B(*X)] \neq$ vazio então

O ciclo pode acontecer no escopo resultante da intersecção

Se-não

O ciclo nunca irá acontecer.

Como foi visto nesta solução, existem outros problemas que podem ser resolvidos de maneira semelhante, conforme detalhado em [3].

CONCLUSAO

Poucos são os sistemas que possuem um módulo para verificação da Base de Conhecimento.

Devido a diversidade de formas de escrevermos regras de produção, acreditamos que cada usuário deva desenvolver seu módulo de verificação automatizada e executar os processos manuais com bastante frequência e critério.

Continuaremos nosso trabalho no sentido de dar mais poder de expressão ao escopo e tentar resolver o problema dos predicados pré-definidos (extra-lógicos) que não tem (pode não ter) uma definição precisa de seu escopo.

REFERENCIAS BIBLIOGRAFICAS :

1. CASANOVA, M.A.; GIORNO, F.A.C.; FURTADO, A.L. ,
Programação em Lógica, V Escola de Computação, 1986.
2. CHANG, CHIN-LIANG ; LEE, R.C. , Symbolic Logic and

Mechanical Theorem Proving, Academic Press, 1973.

3. DAMSKI, J.C.B. ; OMAR, N. - Um sistema para análise de uma Base de Conhecimento, Anais do IV SBIA, Uberlândia, 279-286, 1987.
4. GONÇALVES, C.A. , Aquisição e Representação do Conhecimento para Sistemas Especialistas, Tese de Doutorado, FEA/USP, 1986.
5. SENNE, E.L.F., Compatibilidade de regras de produção usadas como representação do conhecimento em sistemas de consulta, Tese de Mestrado, ITA, 1980.