



REPRESENTAÇÃO DO
CONHECIMENTO

INTELIGÊNCIA ARTIFICIAL COMO
REPRESENTAÇÃO E BUSCA

Inteligência artificial pode ser resumida como o estudo da representação e da busca por meio do qual a atividade inteligente pode ser executada em um dispositivo mecânico.



INTELIGÊNCIA ARTIFICIAL COMO
REPRESENTAÇÃO E BUSCA

- Tópicos da primeira conferência sobre IA – Dartmouth College, verão de 1956:
- 1. Computadores automáticos
 - 2. Como um computador pode ser programado para usar uma linguagem?
 - 3. Redes de neurônios
 - 4. Teoria do tamanho de um cálculo
 - 5. Autoaperfeiçoamento
 - 6. Abstrações
 - 7. Aleatoriedade e criatividade



INTELIGÊNCIA ARTIFICIAL COMO REPRESENTAÇÃO E BUSCA

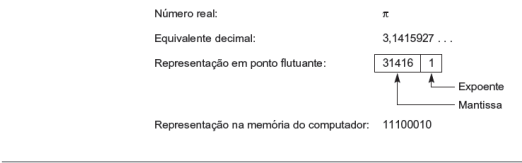
- Sistemas de representação
 - *Abstração, expressividade e eficiência.*
 - Uma representação deve fornecer um arcabouço *natural*.
 - *Expressar conhecimento para solução de problema*
 - *Tornar conhecimento disponível ao computador e ajudar o programador na organização*
 - Linguagens utilizadas são ferramentas para resolver problemas



INTELIGÊNCIA ARTIFICIAL COMO REPRESENTAÇÃO E BUSCA

- Representação de números em ponto flutuante pelo computador.

Figura II.1 Diferentes representações do número real π .



CAPTURANDO INTELIGÊNCIA:O DESAFIO DA IA REPRESENTAÇÃO E INTELIGÊNCIA

- A questão da *representação*, ou de como captar, da melhor forma possível, os aspectos críticos da atividade inteligente para uso em um computador, ou mesmo para a comunicação com os seres humanos, tem sido um tema constante ao longo dos sessenta anos da história da IA.



CAPTURANDO INTELIGÊNCIA: O DESAFIO DA IA
NECESSIDADE DE UMA BOA REPRESENTAÇÃO

- Representação útil, eficiente e significativa
 - Computador não gaste muito tempo em computações desnecessárias



CAPTURANDO INTELIGÊNCIA:
O DESAFIO DA IA

- Três abordagens para a representação, predominantes na comunidade de pesquisa em IA durante esse período de tempo:
 - Solução de problemas por métodos fracos.
 - Solução de problemas por métodos fortes.
 - Abordagens de inteligência distribuídas e incorporadas ou baseadas em agente.



INTELIGÊNCIA ARTIFICIAL COMO
REPRESENTAÇÃO E BUSCA

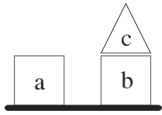
- Em geral, os problemas que a IA tenta resolver não são adequados às representações oferecidas pelos formalismos mais tradicionais, como os arranjos.
- A inteligência artificial está mais preocupada:
 - com a solução qualitativa de problemas que com a sua solução quantitativa;
 - com o raciocínio que com o cálculo;
 - com a organização de grandes e variadas quantidades de conhecimento, que com a implementação de um único algoritmo bem definido.



INTELIGÊNCIA ARTIFICIAL COMO REPRESENTAÇÃO E BUSCA

O cálculo de predicado: predicados e argumentos do mundo de blocos.

Figura II.3 Um mundo de blocos.



$\forall X \neg \exists Y \text{ sobre}(Y,X) \Rightarrow \text{livre}(X)$



INTELIGÊNCIA ARTIFICIAL COMO REPRESENTAÇÃO E BUSCA

Figura II.4 Descrição por rede semântica de um azulão.



INTELIGÊNCIA ARTIFICIAL COMO REPRESENTAÇÃO E BUSCA

Busca: dada uma representação, o segundo componente para solucionar problemas de forma inteligente é a busca.

- Busca em espaço de estados.
- Nó ou estado do grafo/ elos do grafo.
- Grafo de espaço de estados.



BUSCA

- Um agente com várias opções imediatas pode decidir o que fazer comparando diferentes sequências de ações possíveis.
- Esse processo de procurar pela melhor sequência é chamado de busca.
- Formular objetivo → buscar → executar



ESPAÇO DE ESTADOS

- O conjunto de todos os estados acessíveis a partir de um estado inicial é chamado de espaço de estados.
 - Os estados acessíveis são aqueles dados pela função sucessora.
- *O espaço de estados pode ser interpretado como um grafo em que os nós são estados e os arcos são ações.*



ESPAÇO DE BUSCA

- Muitos problemas de interesse prático possuem ESPAÇO de BUSCA tão grande que não podem ser representados por um grafo explícito.
- Elaboraões nos procedimentos de busca básicos vistos anteriormente, onde procura-se representar todo o espaço de busca num grafo, tornam-se necessários.



ELABORAÇÕES

- Três pontos básicos são importantes:
 - necessitamos ser especialmente cuidadosos em COMO formulamos esses problemas para a busca
 - devemos ter métodos para representar buscas em grandes grafos de maneira IMPLÍCITA.
 - Devemos utilizar métodos eficientes para a busca nesses grafos enormes.



MODELAMENTO = ARTE

- No problema do empilhamento de blocos não é difícil conceber uma estrutura de dados para representar os diferentes estados do problema e as ações capazes de alterá-los.
- Em problemas REAIS, isso é difícil. Requer uma profunda análise do problema
 - Simetrias
 - Ignorar detalhes irrelevantes
 - Encontrar abstrações apropriadas

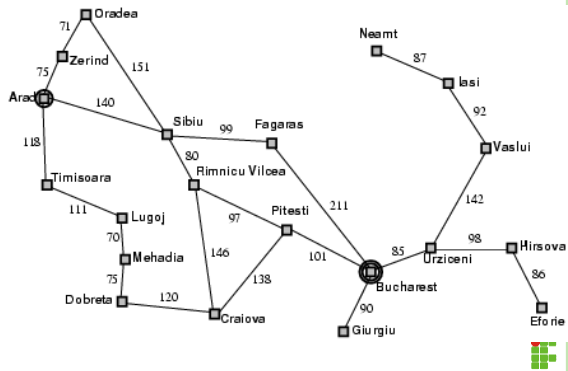


EXEMPLO: ROMÊNIA

- De férias na Romênia; atualmente em Arad.
- Vôo sai amanhã de Bucareste.
- Formular objetivo:
 - Estar em Bucareste
- Formular problema:
 - estados: cidades
 - ações: dirigir entre as cidades
- Encontrar solução:
 - sequência de cidades, ex., Arad, Sibiu, Fagaras, Bucareste.



EXEMPLO: ROMÊNIA



FORMULAÇÃO DE PROBLEMAS

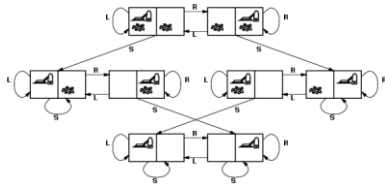
Um **problema** é definido por quatro itens:

- 1. **Estado inicial** ex., "em Arad"
 - 2. **Ações** ou **função sucessor** $S(x)$ = conjunto de pares estado-ação
 - ex., $S(\text{Arad}) = \{ \langle \text{Arad} \rightarrow \text{Zerind}, \text{Zerind} \rangle, \dots \}$
 - 3. **Teste de objetivo**, pode ser
 - **explícito**, ex., $x = \text{"em Bucharest"}$
 - **implícito**, ex., *Cheque-mate*(x)
 - 4. **Custo de caminho** (aditivo)
 - ex., soma das distâncias, número de ações executadas, etc.
 - $c(x,a,y)$ é o **custo do passo**, que deve ser sempre ≥ 0
- Uma **solução** é uma sequência de ações que levam do estado inicial para o estado objetivo.
- Uma **solução ótima** é uma solução com o menor custo de caminho.

SELECIONANDO UM ESPAÇO DE ESTADOS

- O mundo real é absurdamente complexo
 - o espaço de estados é uma **abstração**
- Estado (abstrato) = conjunto de estados reais
- Ação (abstrata) = combinação complexa de ações reais
 - ex., "Arad \rightarrow Zerind" representa um conjunto complexo de rotas, desvios, paradas, etc.
 - Qualquer estado real do conjunto "em Arad" deve levar a algum estado real "em Zerind".
- Solução (abstrata) = conjunto de caminhos reais que são soluções no mundo real
- A abstração é útil se cada ação abstrata é mais fácil de executar que o problema original.

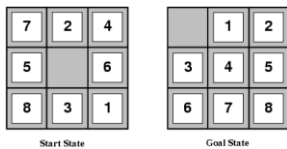
EXEMPLO 1: ESPAÇO DE ESTADOS DO MUNDO DO ASPIRADOR DE PÓ



- **Estados:** Definidos pela posição do robô e sujeira (8 estados)
- **Estado inicial:** Qualquer um
- **Função sucessor:** pode-se executar qualquer uma das ações em cada estado (esquerda, direita, aspirar)
- **Teste de objetivo:** Verifica se todos os quadrados estão limpos
- **Custo do caminho:** Cada passo custa 1, e assim o custo do caminho é o número de passos do caminho



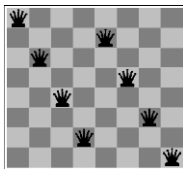
EXEMPLO 2: O QUEBRA-CABEÇA DE 8 PEÇAS



- **Estados:** Especifica a posição de cada uma das peças e do espaço vazio
- **Estado inicial:** Qualquer um
- **Função sucessor:** gera os estados válidos que resultam da tentativa de executar as quatro ações (mover espaço vazio para esquerda, direita, acima ou abaixo)
- **Teste de objetivo:** Verifica se o estado corresponde à configuração objetivo.
- **Custo do caminho:** Cada passo custa 1, e assim o custo do caminho é o número de passos do caminho



EXEMPLO 3: OITO RAINHAS
FORMULAÇÃO INCREMENTAL

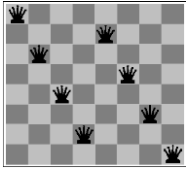


Quais solução

- **Estados:** qualquer disposição de 0 a 8 rainhas
- **Estado inicial:** nenhuma rainha
- **Função sucessor:** colocar 1 rainha em qualquer vazio
- **Teste:** 8 rainhas no tabuleiro, nenhuma atacada
- $64 \times 63 \times \dots \times 57 = 3 \times 10^{14}$ sequências para investigar



EXEMPLO 3: OITO RAINHAS
FORMULAÇÃO DE ESTADOS COMPLETOS



Quais solução

- **Estados:** disposições de n rainhas, uma por coluna, nas n colunas mais a esquerda sem que nenhuma rainha ataque outra
- **Função sucessor:** adicionar uma rainha a qualquer quadrado na coluna vazia mais à esquerda, de tal modo que ela não seja atacada
- Tamanho do espaço de estados: 2.057

ANALISANDO



PROBLEMAS DO MUNDO REAL

- Problema de roteamento
 - encontrar a melhor rota de um ponto a outro (aplicações: redes de computadores, planejamento militar, planejamento de viagens aéreas)
- Problemas de tour
 - visitar cada ponto pelo menos uma vez
- Caixeiro viajante
 - visitar cada cidade exatamente uma vez
 - encontrar o caminho mais curto
- Layout de VLSI
 - posicionamento de componentes e conexões em um chip
- Projeto de proteínas
 - encontrar uma sequência de aminoácidos que serão incorporados em uma proteína tridimensional para curar alguma doença.



BUSCA DE SOLUÇÕES

- Idéia: Percorrer o espaço de estados a partir de uma **árvore de busca**.
- *Expandir* o estado atual aplicando a função sucessor, *gerando* novos estados.
- Busca: seguir um caminho, deixando os outros para depois.
- A *estratégia de busca* determina qual caminho seguir.

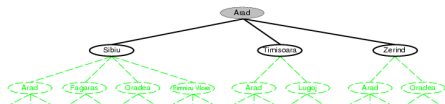


EXEMPLO DE ÁRVORE DE BUSCA



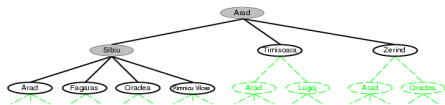
Estado inicial

EXEMPLO DE ÁRVORE DE BUSCA



Depois de expandir Arad

EXEMPLO DE ÁRVORE DE BUSCA



Depois de expandir Sibiu

DESCRIÇÃO INFORMAL DO ALGORITMO GERAL DE BUSCA

```
function TREE-SEARCH( problem, strategy) returns a solution, or failure
  initialize the search tree using the initial state of problem
  loop do
    if there are no candidates for expansion then return failure
    choose a leaf node for expansion according to strategy
    if the node contains a goal state then return the corresponding solution
    else expand the node and add the resulting nodes to the search tree
```



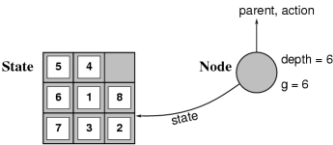
ÁRVORE DE BUSCA NÃO É EQUIVALENTE A ESPAÇO DE ESTADOS!

- Há 20 estados no mapa da Romênia (espaço de estados), mas infinitos caminhos a percorrer. Portanto a árvore de busca, neste caso, tem tamanho infinito.
- Caminho infinito: Arad-Sibiu-Arad-Sibiu-Arad-...



ESTADOS VS. NÓS

- Um estado é uma (representação de) uma configuração física
- Um nó é uma estrutura de dados que é parte da árvore de busca e inclui estado, nó pai, ação, custo do caminho $g(x)$, profundidade



- A função Expand cria novos nós, preenchendo os vários campos e usando a função sucessor do problema para gerar os estados correspondentes.
- A coleção de nós que foram gerados, mas ainda não foram expandidos é chamada de **borda** (ou fringe)
 - Geralmente implementados como uma fila.
 - A maneira como os nós entram na fila determina a estratégia de busca.



ALGORITMO GERAL DE BUSCA EM ÁRVORE

```
function TREE-SEARCH( problem, fringe) returns a solution, or failure
  fringe ← INSERT( MAKE-NODE( INITIAL-STATE[problem]), fringe)
  loop do
    if fringe is empty then return failure
    node ← REMOVE-FRONT(fringe)
    if GOAL-TEST[problem](STATE[node]) then return SOLUTION(node)
    fringe ← INSERT ALL( EXPAND(node, problem), fringe)

function EXPAND( node, problem) returns a set of nodes
  successors ← the empty set
  for each action, result in SUCCESSOR-FN[problem](STATE[node]) do
    s ← a new NODE
    PARENT-NODE[s] ← node, ACTION[s] ← action, STATE[s] ← result
    PATH-COST[s] ← PATH-COST[node] + STEP-COST(node, action, s)
    DEPTH[s] ← DEPTH[node] + 1
    add s to successors
  return successors
```



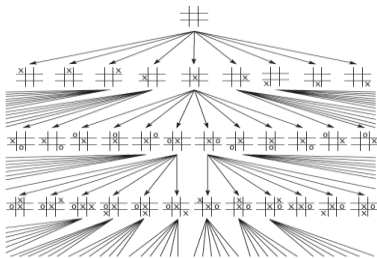
ESTRATÉGIAS DE BUSCA

- Uma estratégia de busca é definida pela escolha da **ordem da expansão de nós**
- Estratégias são avaliadas de acordo com os seguintes critérios:
 - completeza**: o algoritmo sempre encontra a solução se ela existe?
 - complexidade de tempo**: número de nós gerados
 - complexidade de espaço**: número máximo de nós na memória
 - otimização**: a estratégia encontra a solução ótima?
- Complexidade de tempo e espaço são medidas em termos de:
 - b**: máximo fator de ramificação da árvore (número máximo de sucessores de qualquer nó)
 - d**: profundidade do nó objetivo menos profundo
 - m**: o comprimento máximo de qualquer caminho no espaço de estados (pode ser ∞)



SISTEMAS DE REPRESENTAÇÃO

Figura 11.5 Parte do espaço de estados para o jogo da velha.



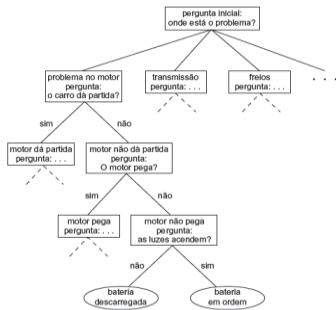
SISTEMAS DE REPRESENTAÇÃO

Figura II.6 Descrição do espaço de estados do primeiro passo no diagnóstico de um problema automotivo.



BUSCA

Figura II.7 Descrição do espaço de estados de diagnóstico do problema automotivo.



BUSCA

- Apesar dessa aparente universalidade, a busca em espaço de estados não é, por si só, suficiente para produzir um comportamento inteligente na resolução de problemas.
- Ela é muito mais uma ferramenta importante para o desenvolvimento de programas inteligentes.
- Busca exaustiva.
- Regras heurísticas.

UMA BREVE HISTÓRIA DOS ESQUEMAS REPRESENTACIONAIS DE IA

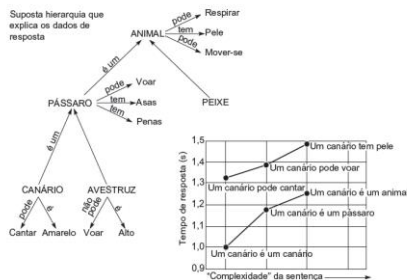
- Teorias associacionistas do significado:
 - As teorias *associacionistas*, seguindo a tradição empirista na filosofia, definem o significado de um objeto em termos de uma rede de associações com outros objetos.
 - Para os associacionistas, quando os seres humanos percebem um objeto, essa percepção é mapeada primeiro em um conceito.
 - Esse conceito é parte do nosso conhecimento completo do mundo e está conectado através de relacionamentos apropriados a outros conceitos.

REPRESENTAÇÃO DO CONHECIMENTO

- Questões da representação do conhecimento
 - As *redes semânticas*
 - As *dependências conceituais*
 - Os *roteiros (scripts)*
 - *Quadros (frames)*
 - Os *grafos conceituais*
 - *Agentes*
 - Distinção entre um *esquema* representacional e o *meio* de sua implementação.

REPRESENTAÇÃO DO CONHECIMENTO

Figura 7.1 Rede semântica desenvolvida por Collins e Quillian em sua pesquisa sobre o armazenamento de informação pelo homem e seu tempo de resposta (Harmon e King, 1985).

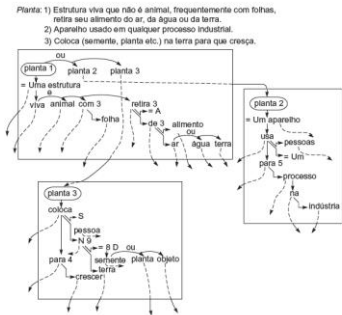


TRABALHOS INICIAIS EM REDES SEMÂNTICAS

- Sistema de grafos existenciais de Charles S. Pierce.
- MIND de Shapiro (1971)
- Quillian, 1967 - conceito de palavra, planos, nó de interseção.

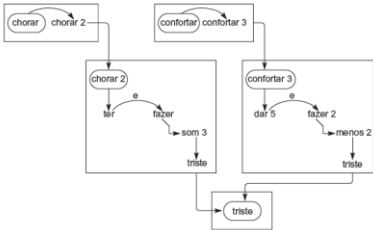
REPRESENTAÇÃO DO CONHECIMENTO

Figura 7.3 Três planos representando três definições da palavra "planta" (Quillian, 1967).



REPRESENTAÇÃO DO CONHECIMENTO

Figura 7.4 Interseção entre "chorar" e "confortar" (adaptado de Quillian, 1967).



SISTEMA DE COMPREENSÃO DA LING NATURAL

- Determina o significado de um corpo de texto por meio da construção de coleções desses nós de intersecção
- Escolher entre múltiplos significados de palavras procurando o significado com o menor caminho de intersecção para outras palavras da sentença
- Responder a uma gama flexível de consultas com base em associações entre conceitos das palavras das consultas e os conceitos do sistema



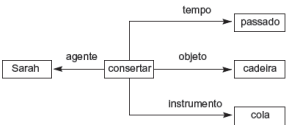
PADRONIZAÇÃO DOS RELACIONAMENTOS EM REDES

- Estrutura de casos dos verbos em inglês.
- Agente, objeto, instrumento, localização e tempo.
- Quadro de caso.



PADRONIZAÇÃO DOS RELACIONAMENTOS EM REDES

Figura 7.5 Representação por quadro de caso da sentença “Sarah consertou a cadeira com cola”.



REPRESENTAÇÃO DO
CONHECIMENTO

- Teoria da *dependência conceitual* de Roger Schank (Schank e Rieger, 1974):
 - ATOs** ações
 - PFs** objetos (produtores de figuras)
 - AAs** modificadores de ações (auxiliares de ações)
 - Afs** modificadores de objetos (auxiliares de figuras)
- Relacionamentos de *dependência conceitual*
- Regras de *sintaxe conceituais*

REPRESENTAÇÃO DO
CONHECIMENTO - ROTEIROS

- Um *roteiro* é uma representação estruturada que descreve uma sequência estereotipada de eventos em um contexto particular.
- O modelo de roteiro foi originalmente concebido por Schank e seu grupo de pesquisa (Schank e Abelson, 1977) como um meio de organizar estruturas de *dependência conceitual* formando descrições de situações típicas.

REPRESENTAÇÃO DO
CONHECIMENTO - ROTEIROS

- Os roteiros são usados em sistemas de compreensão de linguagem natural para organizar uma base de conhecimento em termos das situações que o sistema deve compreender.
- Os componentes de um roteiro são:
 - Condições de entrada ou descritores do mundo.
 - Resultados ou fatos que sejam verdadeiros quando o roteiro tiver terminado.
 - Acessórios ou “coisas” que suportam o conteúdo do roteiro.

REPRESENTAÇÃO DO CONHECIMENTO

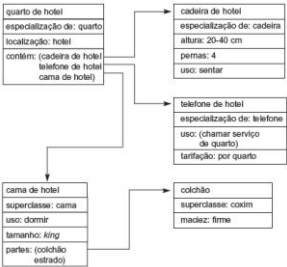
- *Papéis são as ações que os participantes individuais desempenham.*
- *Cenas.*
- Exemplo 2.1.1:
“John foi a um restaurante ontem à noite. Ele pediu um bife. Quando pagou a conta, percebeu que não tinha mais dinheiro. Ele correu para casa, pois tinha começado a chover.”

REPRESENTAÇÃO DO CONHECIMENTO

- 2.1.5 Quadros:
- “Eis aqui a essência da teoria dos quadros: quando alguém encontra uma nova situação (ou modifica substancialmente o seu entendimento sobre um problema), recupera da memória uma estrutura chamada “quadro”. Essa estrutura é um arcabouço memorizado que deve ser adaptado para se adequar à realidade, alterando detalhes, conforme a necessidade.” (Minsky, 1975).
- Exemplo dos quarto de hotel.

REPRESENTAÇÃO DO CONHECIMENTO

Figura 7.12 Parte de uma descrição por quadro de um quarto de hotel. “Especialização” indica um ponteiro para uma superclasse.



REPRESENTAÇÃO DO CONHECIMENTO

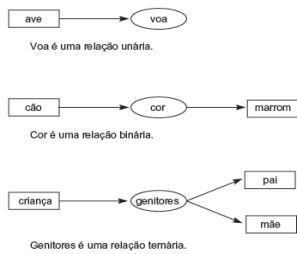
- As partições do quadro contêm informações como:
 1. Informação sobre a identificação do quadro.
 2. Relação desse quadro com outros quadros.
 3. Descritores de requisitos para um quadro.
 4. Informação procedimental sobre o uso da estrutura descrita.
 5. Informação padrão do quadro.

GRAFOS CONCEITUAIS: UMA LINGUAGEM DE REDE

- Introdução aos grafos conceituais:
 - Um *grafo conceitual* é um grafo finito, conectado e bipartido. Os nós do grafo são *conceitos* ou, então, *relações conceituais*. Os grafos conceituais não usam arcos rotulados; em vez disso, os nós de relações conceituais representam relações entre conceitos.

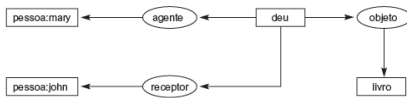
REPRESENTAÇÃO DO CONHECIMENTO

Figura 7.14 Relações conceituais de diferentes aridades.



REPRESENTAÇÃO DO CONHECIMENTO

Figura 7.15 Grafo de "Mary deu o livro a John".



REPRESENTAÇÃO DO CONHECIMENTO

Tipos, indivíduos e nomes

Figura 7.16 Grafo conceitual indicando que o cão de nome Emma é marrom.

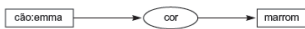
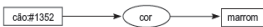
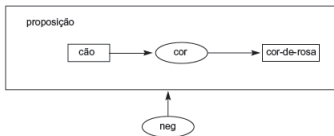


Figura 7.17 Grafo conceitual indicando que um determinado (ainda não identificado) cão é marrom.



GRAFOS CONCEITUAIS E LÓGICA

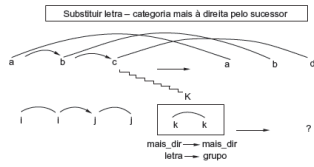
Figura 7.25 Grafo conceitual da proposição "não existem cães cor-de-rosa".



COPYCAT

- o espaço de trabalho, a rede deslizante e a estante de códigos.

Figura 7.27 Um estado possível do espaço de trabalho da copycat. São mostrados vários exemplos de ligações e elos entre as letras; adaptado de Mitchell (1993).



Múltiplas representações, ontologias e serviços de conhecimento

COMPONENTES COPYCAT

- Espaço de trabalho
 - Estrutura global. Lugar onde as estruturas perceptivas são construídas hierarquicamente sobre as entradas
- Rede deslizante
 - Reflete a rede de conceitos ou associações potenciais para os componentes da analogia. Rede semântica dinamicamente deformável
- Estante de códigos
 - Fila probabilística controlada por prioridade contendo codeletas. Que interagem com objetos do espaço.



SOLUÇÃO DE PROBLEMAS DISTRIBUÍDA E BASEADA EM AGENTES

- Solução de problemas orientada a agentes: definição
 - “Agente”
 - “Sistema baseado em agentes”
 - “Sistemas multiagentes”



REPRESENTAÇÃO DO
CONHECIMENTO

- Exemplos de um paradigma orientado a agentes e seus desafios
- Fabricação
- Controle automatizado
- Os quatro critérios para um sistema de agentes inteligentes incluem resolvidores de problemas que são situados, autônomos, flexíveis e sociais.



REPRESENTAÇÃO DO
CONHECIMENTO

- Telecomunicações
- Sistemas de transporte
- Gerenciamento de informação
- Comércio eletrônico (e-commerce)
- Jogos e teatro interativos