

FERNANDA HEMBECKER



**PARSER BASEADO EM CASOS
NA COMPREENSÃO DE DIÁLOGO:
UMA PROPOSTA DE APLICAÇÃO DO
DIRECT MEMORY ACCESS PARSING**

Dissertação apresentada ao Programa de Pós-Graduação em Informática Aplicada da Pontifícia Universidade Católica do Paraná como requisito parcial para obtenção do título de Mestre em Informática Aplicada.

Área de Concentração: Metodologia e Técnicas de Computação

Orientador: Prof. Dr. Bráulio Coelho Ávila

Co-orientador: Prof. Dr. Edson Emílio Scalabrin

CURITIBA

2002



Hembecker, Fernanda

Parser Baseado em Casos na Compreensão de Diálogo: uma Proposta de Aplicação do Direct Memory Access Parsing. Curitiba, 2002. 97p.

Dissertação – Pontifícia Universidade Católica do Paraná. Programa de Pós-Graduação em Informática Aplicada.

1. Compreensão de Linguagem Natural 2. Raciocínio Baseado em Casos
3. Direct Memory Access Parsing. 4. Diálogo I. Pontifícia Universidade Católica do Paraná. Centro de Ciências Exatas e de Tecnologia. Programa de Pós-Graduação em Informática Aplicada II-t



ATA DA SESSÃO PÚBLICA DE DEFESA DE DISSERTAÇÃO DE MESTRADO
DO PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA APLICADA
DA PONTIFÍCIA UNIVERSIDADE CATÓLICA DO PARANÁ

DEFESA DE DISSERTAÇÃO Nº 064

Aos 13 dias do mês de julho de 2002 realizou-se a sessão pública de defesa da dissertação **“Parser Baseado em Casos na Compreensão de Diálogo: uma Proposta de Aplicação do Direct Memory Access Parsing”**, apresentada por **Fernanda Hembercker** como requisito parcial para a obtenção do título de **Mestre em Informática Aplicada**, perante uma Banca Examinadora composta pelos seguintes membros:

Prof. Dr. Bráulio Coelho Ávila
PUCPR (Presidente)

assinatura

parecer (aprov/ reprov.)

aprovado

Prof. Dr. Edson Emílio Scalabrin
PUCPR

assinatura

aprovado

Prof. Dr. Hilton J. S. de Azevedo
CEFET - PR

assinatura

REPROVADO

Prof. Dr. Ricardo Luís de Freitas
PUC - Campinas

assinatura

aprovado

Conforme as normas regimentais do PPGIA e da PUCPR, o trabalho apresentado foi considerado *aprovado* (*aprovado/reprovado*), segundo avaliação da maioria dos membros desta Banca Examinadora. Este resultado está condicionado ao cumprimento integral das solicitações da Banca Examinadora, conforme registrado no Livro de Defesas do programa.

Prof. Dr. Carlos Maziero
Diretor do PPGIA PUCPR



Data e assinatura, após homologação da defesa pelo colegiado.

Aos meus pais,
Rogemil e Liliam
pelo amor e apoio constantes

Agradecimentos

Ao amigo e orientador Prof. Dr. Bráulio Coelho Ávila pela sugestão do tema e constante apoio no desenvolvimento deste trabalho.

Ao Prof. Dr. Edson Emílio Scalabrin pelos comentários e sugestões valiosas referentes ao tema deste trabalho.

À minha família, que deu todo o suporte no decorrer dos estudos e acreditou na conclusão deste trabalho.

Aos amigos Eduardo C. Hamerski Jr. e Cláudio R. Carvilhe pela amizade e constante apoio.

Ao mestrando Emerson L. dos Santos pela companhia nas pesquisas e pelos comentários referentes ao desenvolvimento deste trabalho.

Ao pessoal dos laboratórios AgP, LASIN e LUCI pela amizade e pelos momentos de descontração.

À equipe do Parque Tecnológico II pelo suporte técnico.

À PUCPR pela oportunidade concedida.

À CAPES pelo auxílio financeiro.

Àqueles que, de um modo ou de outro, contribuíram para a conclusão deste trabalho.

Sumário

Lista de Figuras	iv
Lista de Tabelas	vi
Resumo	vii
Abstract	viii
Capítulo 1	
Introdução	1
Capítulo 2	
Cognição e Memória	5
2.1. Considerações Iniciais	5
2.2. Evolução das Ciências Cognitivas.....	6
2.3. Processos de Informação	8
2.3.1. Significado e Propósito.....	8
2.3.2. Representação	8
2.3.3. Descrição Formal.....	9
2.4. Sistemas da Mente	10
2.5. Princípios Básicos da Ciência Cognitiva.....	14
2.5.1. Psicologia Cognitiva e Inteligência Artificial.....	15
2.5.2. Representação de Conhecimento.....	16
2.6. Memória Dinâmica.....	18
2.6.1. Conceitualização de Memória Dinâmica.....	19
2.6.2. Memória Dinâmica vs. Scripts	20
2.7. Memory Organization Packages.....	22
2.7.1. Frames e MOPs	22

2.7.2. Recuperação de Conhecimento em MOPs	25
2.8. Considerações Finais	28

Capítulo 3

Raciocínio Baseado em Casos	29
3.1. Considerações Iniciais	29
3.2. Inteligência Artificial e Cognição	29
3.2.1. Raciocínio Humano	30
3.2.2. Lembrança	31
3.2.3. A Ilusão do Controle	33
3.3. Fundamentos de Raciocínio Baseado em Casos	35
3.3.1. Casos	37
3.3.2. Ciclo de Funcionamento	39
3.3.3. Indexação	46
3.3.4. Base metodológica	47
3.4. Considerações Finais	49

Capítulo 4

Parser Baseado em Casos	50
4.1. Considerações Iniciais	50
4.2. Compreensão da Linguagem Natural	50
4.3. Fundamentos de Parsers Baseados em Casos	55
4.4. Direct Memory Access Parsing — DMAP	57
4.4.1. Índices	58
4.4.2. Busca em Memória	61
4.4.3. Falhas e Aprendizado	64
4.5. Considerações Finais	67

Capítulo 5

Sistema ComPor	68
5.1. Considerações Iniciais	68
5.2. Diálogo	68

5.3. Diálogo no Sistema ComPor	70
5.3.1. Modelagem dos Conceitos.....	71
5.3.2. Estruturação das Respostas do Sistema	74
5.3.3. Aprendizado na Execução de um Diálogo.....	76
5.3.4. Validação do Sistema	77
5.4. Modelagem/Execução de um Diálogo.....	78
5.4.1. Modelagem da Memória.....	79
5.4.2. Modelagem das Respostas do Sistema	83
5.4.3. Execução de um Diálogo.....	85
5.5. Considerações Finais	87

Capítulo 6

Conclusões	88
6.1. Principais Resultados Obtidos.....	89
6.2. Trabalhos Futuros	91

Referências Bibliográficas	92
---	-----------

Lista de Figuras

Figura 2.1	MOPs sobre eventos de comunicação	25
Figura 2.2	Hierarquia Abstrata sobre M-MTRANS	26
Figura 3.1	Resolução de problemas baseado em casos anteriores.....	35
Figura 3.2	Ciclo de funcionamento.....	40
Figura 3.3	Recuperação de casos	41
Figura 3.4	Adaptação de casos.....	43
Figura 3.5	Revisão de casos	45
Figura 3.6	Armazenamento de casos	45
Figura 4.1	Taxa de acerto.....	54
Figura 4.2	Representação de memória e índices.....	56
Figura 4.3	Análise conceitual e compreensão baseada em memória.....	58
Figura 4.4	Interpretação de índices	59
Figura 4.5	Memória e índices: caso de estudo	62
Figura 4.6	Esquema de busca do DMAP	64
Figura 4.7	Falha de especialização	65
Figura 4.8	Tipos de interseção de marcadores.....	66
Figura 5.1	Casamento de padrões no ELIZA.....	69
Figura 5.2	Arquitetura genérica do sistema ComPor.....	70
Figura 5.3	Estruturação dos conceitos na memória	72
Figura 5.4	Estrutura de memória: diálogo em uma livraria	74
Figura 5.5	Módulos principais do sistema ComPor.....	79
Figura 5.6	EMOPs: Editor de MOPs	80
Figura 5.7	Definição de hierarquia e de empacotamentos	81
Figura 5.8	Definição de slots visando as respostas do sistema.....	82

Figura 5.9	Definição de índices	82
Figura 5.10	Arquivo de estrutura de memória	83
Figura 5.11	Arquivo de índices	83
Figura 5.12	Gerenciador de Respostas.....	84
Figura 5.13	Arquivo de respostas	85
Figura 5.14	Execução de Diálogo	86

Lista de Tabelas

Tabela 2.1	Aplicações teóricas de abordagens computacionais.....	11
Tabela 2.2	Representação abstrata de um frame	23
Tabela 2.3	Tabela de busca em MOPs	27
Tabela 4.1	Lexicon: Sistemas tradicionais	52
Tabela 4.2	Combinação de significados	53
Tabela 5.1	Esquema de organização de planos	76

Resumo

Este trabalho tem como objetivo a concepção e a implementação de um sistema que viabilize a execução de um diálogo entre usuários e computadores através da compreensão automática da linguagem natural. Voltado ao idioma português, este sistema deve receber as sentenças de entrada do usuário, efetuar um processo de compreensão para então emitir respostas coerentes com esta entrada. Em geral, os modelos propostos para o tratamento automático de textos se baseiam na criação de primitivas neutras capazes de mapear grande parte do significado das palavras, ou em procedimentos sequenciais que analisam aspectos gramaticais do texto. Utilizando uma abordagem diferenciada, o sistema desenvolvido neste trabalho efetua a compreensão a partir de uma memória pré-modelada que representa os conceitos sobre o domínio do conhecimento, as informações sobre o tratamento da linguagem natural e ainda os procedimentos que conduzem o diálogo de forma inteligente. Como um todo, a intenção é aproximar-se dos processos cognitivos humanos e construir um sistema artificial controlado que reproduza a compreensão de linguagem natural. Para atingir estes objetivos, o foco do trabalho baseia-se em estudos referentes às características cognitivas humanas e aos processos computacionais de Memória Dinâmica, Raciocínio Baseado em Casos e *Parsers* Baseados em Casos.

Palavras-Chave: Compreensão de Linguagem Natural. Raciocínio Baseados em Casos. *Direct Memory Access Parsing*. Diálogo.

Resumo

Este trabalho tem como objetivo a concepção e a implementação de um sistema que viabilize a execução de um diálogo entre usuários e computadores através da compreensão automática da linguagem natural. Voltado ao idioma português, este sistema deve receber as sentenças de entrada do usuário, efetuar um processo de compreensão para então emitir respostas coerentes com esta entrada. Em geral, os modelos propostos para o tratamento automático de textos se baseiam na criação de primitivas neutras capazes de mapear grande parte do significado das palavras, ou em procedimentos sequenciais que analisam aspectos gramaticais do texto. Utilizando uma abordagem diferenciada, o sistema desenvolvido neste trabalho efetua a compreensão a partir de uma memória pré-modelada que representa os conceitos sobre o domínio do conhecimento, as informações sobre o tratamento da linguagem natural e ainda os procedimentos que conduzem o diálogo de forma inteligente. Como um todo, a intenção é aproximar-se dos processos cognitivos humanos e construir um sistema artificial controlado que reproduza a compreensão de linguagem natural. Para atingir estes objetivos, o foco do trabalho baseia-se em estudos referentes às características cognitivas humanas e aos processos computacionais de Memória Dinâmica, Raciocínio Baseado em Casos e *Parsers* Baseados em Casos.

Palavras-Chave: Compreensão de Linguagem Natural. Raciocínio Baseados em Casos. *Direct Memory Access Parsing*. Diálogo.

Abstract

This work proposes the conception and implementation of a system that executes a dialog between users and computers through automatic comprehension of natural language. Focused inclusive on Portuguese, this system should receive the users' sentences, execute the comprehension process and finally show coherent answers with this sentences. Generally, the models proposed to natural language processing are based on neutral primitives that represent the meaning of the words, or on sequential procedures that analyze the text considering the grammatical aspect. Using a distinctive method, the system developed in this work comprehend sentences considering a memory structure that represents concepts about an specific domain, information about the natural language usage and methods that guide the dialogue in an intelligent manner. Generally, the main intention is to approach the cognitive human processes and create a controlled artificial system that recreate the natural language comprehension. In this context, the bulk of this dissertation is based on researches about human cognitive aspects as well in computational processes as Dynamic Memory, Case-Based Reasoning and Case-Based Parsing.

Keywords: Natural Language Comprehension. Case-Based Reasoning. Direct Memory Access Parsing. Dialogue.

Abstract

This work proposes the conception and implementation of a system that executes a dialog between users and computers through automatic comprehension of natural language. Focused inclusive on Portuguese, this system should receive the users' sentences, execute the comprehension process and finally show coherent answers with this sentences. Generally, the models proposed to natural language processing are based on neutral primitives that represent the meaning of the words, or on sequential procedures that analyze the text considering the grammatical aspect. Using a distinctive method, the system developed in this work comprehend sentences considering a memory structure that represents concepts about an specific domain, information about the natural language usage and methods that guide the dialogue in an intelligent manner. Generally, the main intention is to approach the cognitive human processes and create a controlled artificial system that recreate the natural language comprehension. In this context, the bulk of this dissertation is based on researches about human cognitive aspects as well in computational processes as Dynamic Memory, Case-Based Reasoning and Case-Based Parsing.

Keywords: Natural Language Comprehension. Case-Based Reasoning. Direct Memory Access Parsing. Dialogue.

Capítulo 1

Introdução

A crença de que um dia as pessoas poderão interagir com computadores através da fala tem sido um dos assuntos preferidos da ficção científica. Seguindo esta mesma linha de pensamento, muitos acreditam que o diálogo representará a forma mais natural e simples de se interagir com computadores.

Entretanto, a fala não é uma atividade isolada, baseada apenas na emissão de sons. Em um diálogo, por exemplo, é necessário compreender o que uma pessoa diz para então chegar às próprias conclusões e finalmente emitir uma resposta. Em todo este ciclo de raciocínio, estão envolvidas diversas capacidades cognitivas humanas que têm sido intensamente estudadas por vários grupos de pesquisa. Em geral, com estes estudos, busca-se compreender e reproduzir comportamentos relativos à inteligência humana de forma que esta reprodução possa trazer algum ganho de informações, de processamento ou de qualidade às atividades humanas. E, para que isto ocorra, pesquisas estão sendo realizadas em conjunto nas áreas de neurociência, psicologia, lingüística, filosofia e computação.

Para muitos, os computadores são vistos como verdadeiros cérebros eletrônicos capazes de resolver os mais difíceis problemas. Se relacionados à Inteligência Artificial, esta visão vai muito mais além. A resolução de cálculos matemáticos, combinação de bases de dados e a manipulação de imagens pode superar e muito o raciocínio humano, quando o fator de comparação é a velocidade de raciocínio e a precisão dos dados. Entretanto, uma atividade simples efetuada pelos humanos como a compreensão de um texto, não é tarefa trivial para um computador.

Uma linha de pesquisa em crescente ascensão em Inteligência Artificial vem sendo o tratamento da linguagem natural. Desde a década de 60, pesquisadores visam construir

sistemas computacionais capazes de receber como entrada um texto qualquer e apresentar como saída a compreensão do mesmo. Atualmente, as técnicas voltam-se principalmente às análises sintática, semântica e morfológica das sentenças, porém, alguns pesquisadores questionam estas técnicas quando indagados sobre o grau de compreensão que as mesmas atingem. Nelas, utiliza-se unicamente o texto de entrada e um conjunto de regras de tratamento bastante específicas para efetuar a compreensão das suas sentenças, sendo estes alguns dos principais pontos de dúvida. Uma vez que aparentemente as pessoas identificam o significado de uma palavra de forma bastante fácil, talvez possa haver algo falho no tratamento lingüístico efetuado por computadores.

Marvin Minsky [MIN85] afirma que a consciência humana está relacionada a fatos do passado e não a fatos do presente, como costumeiramente pensado. Ou seja, é necessário se basear em informações previamente conhecidas para que se possa compreender uma nova situação. Seguindo esta mesma linha de pensamento, embasado em aspectos da Ciência Cognitiva e nos processos de compreensão da mente humana, Roger Schank [SCH82] afirma que a inteligência é um processo baseado em memória. Isto significa que ao recuperar crenças ou informações previamente conhecidas, pode-se utilizá-las para efetivamente compreender uma nova situação. Além disso, a memória não é considerada uma estrutura estática onde são apenas armazenadas as informações conhecidas. Ao longo do tempo, a compreensão de uma nova situação pode adicionar novos conhecimentos à memória; e esta pode, ainda, conduzir processos de raciocínio se possuir informações para tal. A memória não é apenas um depósito; ela é uma estrutura dinâmica de armazenamento e processamento de conhecimento.

Fundamentado nas idéias de Schank, Christopher Riesbeck [RIE89] optou por investir suas teorias no tratamento automático da linguagem natural. Sua proposta é viabilizar a compreensão de textos através de recursos da memória, desviando-se de regras bem definidas aprendidas na escola. Assim, ele sugere uma inversão no modelo tradicional de compreensão: ao invés de se construir um novo conceito a partir das informações extraídas do texto, deve-se verificar se um conceito já conhecido valida a interpretação do mesmo. Como conseqüência, se uma porção do texto não se relacionar a conceitos da memória, estes são devidamente adaptados à estrutura existente e passam a incorporar a memória como um novo conhecimento aprendido. Desta forma, compreender significa identificar a conexão entre o texto de entrada e as estruturas de memória representadas no sistema.

De acordo com o cenário apresentado nos parágrafos anteriores, a proposta deste trabalho visa a concepção e a implementação de um ambiente que permita a compreensão automática da linguagem natural e a utilize em um sistema de diálogo. Assim, de forma a realizar a interação com o usuário, inicialmente este sistema deve permitir a criação de uma estrutura de conhecimento própria sobre um determinado assunto. Esta estrutura deve ser capaz de fornecer informações que possam ser requisitadas pelo usuário durante o diálogo com o sistema. Possuindo estas informações, o sistema deve ainda viabilizar a construção de diferentes esquemas de diálogo, onde seja possível prever as entradas do usuário e construir automaticamente respostas coerentes com esta entrada. Ou seja, como um todo, o sistema deve viabilizar a construção de um contexto voltado à execução de um diálogo.

Com o desenvolvimento dos estudos e do sistema proposto, tem-se como objetivo desenvolver um sistema computacional na área de Inteligência Artificial bastante próximo a diversos preceitos da Ciência Cognitiva. A intenção é reproduzir partes do comportamento cognitivo humano buscando auxiliar a interação entre usuários e computadores através de uma interface simples de diálogo em linguagem natural. Além de enriquecer estudos referentes à compreensão da linguagem, os benefícios da aplicação de um sistema com estas características poderiam ser aproveitados em sistemas computacionais da área de educação, bem como em sistemas comerciais onde há interação direta entre homens e máquinas. Em pesquisas científicas, por sua vez, diferentes características da memória podem ser exploradas assim como outros processos cognitivos além da compreensão, como o aprendizado, por exemplo.

O presente documento é composto por 6 Capítulos principais que fundamentam desde as idéias iniciais de como reproduzir a compreensão humana em computadores até detalhes do sistema desenvolvido.

No Segundo Capítulo é apresentada uma visão genérica da Ciência Cognitiva, uma linha de pesquisa interdisciplinar focada no estudo do comportamento humano e que exerce grande influência na área de Inteligência Artificial. Em especial, são apresentados tópicos sobre a memória humana, representação e reconhecimento de informações.

O Raciocínio Baseado em Casos, fundamentado no Capítulo 3, surge na área de computação como um meio de reproduzir o pensamento, o aprendizado, a memória e o planejamento humano.

Seguindo preceitos de Raciocínio Baseado em Casos, no Capítulo 4 são apresentados detalhes de *Parsers* Baseados em Casos, uma abordagem inovadora utilizada na compreensão de linguagem natural. Um enfoque especial é dado à proposta de Riesbeck [RIE89], o DMAP — *Direct Memory Access Parsing* — por representar o ponto principal deste trabalho.

O sistema ComPor, resultado dos estudos apresentados nesta dissertação, é detalhado no Capítulo 5.

E, finalmente, no Sexto Capítulo são apresentadas as conclusões deste trabalho, benefícios da utilização do sistema ComPor e sugestões para trabalhos futuros.

Capítulo 2

Cognição e Memória

2.1. Considerações Iniciais

Pesquisadores em psicologia, lingüística, ciência da computação, filosofia e neurociências perceberam que tentavam responder as mesmas perguntas sobre a natureza da mente humana. Todos estavam desenvolvendo métodos de investigação similares e complementares, embora não conhecessem o trabalho das outras linhas de pesquisa e voltassem seu foco de estudo às características principais de cada linha em particular. A junção de todas estas pesquisas resultou na criação do campo de pesquisa interdisciplinar intitulado *Ciência Cognitiva*.

A palavra “cognitivo” refere-se à percepção e conhecimento. Desta forma, Ciência Cognitiva pode ser vista como a ciência da mente. Pesquisadores desta nova linha buscam compreender a percepção, o pensamento, a lembrança, a compreensão da linguagem, a aprendizagem e outros fenômenos mentais. Suas pesquisas são bastante diversas, incluindo, por exemplo, a observação do comportamento de crianças, a programação de computadores para a resolução de problemas complexos e o estudo dos princípios neurais do cérebro humano [STI95].

Enfocando a participação da informatização, este capítulo apresenta uma caracterização genérica das ciências cognitivas. Em especial, apresenta ainda sua aplicação na compreensão da linguagem natural, ponto central deste trabalho. A intenção não é fazer um estudo aprofundado sobre as ciências cognitivas, mas sim mostrar a participação efetiva da Inteligência Artificial no estudo da natureza da mente humana.

2.2. Evolução das Ciências Cognitivas

Todos os dias, as pessoas realizam uma ampla gama de tarefas mentais; seja na resolução de um trabalho escolar, na tomada de decisões frente a um problema de trabalho, na elaboração de uma resposta qualquer ou nas atitudes e no comportamento diante da sociedade. O fato de estudar o funcionamento da mente é importante para muitas atividades práticas, como por exemplo: os educadores precisam saber a natureza do pensamento de estudantes para planejar melhor seu método de ensino, e programas computacionais podem simular o comportamento humano inteligente e auxiliar a resolução de problemas complexos. Entretanto, o estudo da mente não é tarefa fácil.

Pesquisadores em Ciência Cognitiva vêem a mente humana como um sistema complexo que recebe, armazena, recupera, transforma e transmite informações. Estas operações são chamadas *computações* ou *processos de informação* [STI95]. O início das pesquisas cognitivas foi marcado pelo trabalho de filósofos gregos, como Platão e Aristóteles, que buscavam compreender a mente humana e suas operações. Para isto, Platão afirmava que o conhecimento humano era inato, independente da experiência dos sentidos. Descartes e Leibniz, outros filósofos renomados, acreditavam que o conhecimento poderia ser obtido apenas pelo pensamento e pelo raciocínio, defendendo assim suas teorias sobre o Racionalismo. Aristóteles, por sua vez, defendia que o conhecimento era representado em termos de regras, a partir das quais poderiam ser efetuadas deduções, considerando ainda que as regras eram aprendidas com a experiência.

Estudos sobre a mente foram dominados pela área da filosofia até o século XIX, época em que surgiu a psicologia experimental com Wilhelm Wundt. Buscava-se o estudo de operações mentais de forma mais sistemática através da aplicação de métodos laboratoriais. Entretanto, estes procedimentos foram substituídos pelo Behaviorismo, visão que negava a existência da mente e restringia-se ao exame da relação entre estímulos e respostas comportamentais obtidas. Um representante desta linha foi J.B. Watson.

Em 1956, houve uma revolução nos estudos da psicologia. George Milles sintetizou em seus estudos que a capacidade do pensamento humano é limitada. Em seguida, despontaram os pesquisadores da área computacional, como John McCarthy, Marvin Minsky, Allen Newell e Hebert Simon. McCarthy defendia a Inteligência Artificial embasada em lógica formal, Newell e Simon basearam seus estudos na criação e aplicação de regras e Minsky criou esquemas de conceitos para a representação de conhecimento. Os estudos já se

encontravam então na década de 70, quando psicólogos começaram a demonstrar interesse na imaginação mental.

Muitas pesquisas experimentais e sobre computação começaram a se desenvolver na década de 80, quando Roger Schank sugeriu a teoria de Raciocínio Baseado em Casos para simular o comportamento da mente humana. Esta linha de pesquisa mostrou-se bastante interessante em diversas áreas do conhecimento. Em especial, a junção de conceitos e sistemas de Raciocínio Baseado em Casos aplicados à compreensão da linguagem natural, compõem o foco deste trabalho, como será mostrado nos capítulos subseqüentes.

Outras linhas de pesquisa na área de computação, associando conhecimentos sobre psicologia e fisiologia, culminaram na criação de sistemas de Redes Neurais, que buscam simular a organização do cérebro humano através dos neurônios a fim de obter um sistema inteligente.

A partir deste breve histórico, baseado em [THA98], pode-se perceber o interesse e os progressos obtidos na área das ciências cognitivas. O estudo da mente humana ainda é um mistério, mas vem obtendo avanços significativos com a união das diversas áreas de conhecimento que o homem domina. Inicialmente, testes eram realizados com seres humanos a fim de analisar o comportamento mental. Atualmente, para complementar os experimentos psicológicos sobre raciocínio dedutivo, formação de conceitos, imaginação e resolução de problemas, por exemplo, desenvolvem-se sistemas computacionais que simulam aspectos da mente humana.

No momento em que neurocientistas estão preocupados com a organização do sistema nervoso, lingüistas estudam a estrutura da linguagem humana e a natureza da aquisição da linguagem. Filósofos pesquisam sobre lógica e significado, esclarecendo conceitos fundamentais da ciência cognitiva, como informação e conhecimento; ao passo que em computação, os pesquisadores viabilizam meios de reproduzir o comportamento humano inteligente [STI95].

Em resumo, a Ciência Cognitiva pode ser vista como soma das áreas de conhecimento previamente mencionadas. “Cientistas cognitivistas”, mesmo que especializados em uma ou outra linha de pesquisa, beneficiam de forma marcante o cruzamento de todas as pesquisas. O trabalho interdisciplinar conduz a resultados mais ricos e interessantes, uma vez que há uma convergência teórica e experimental das conclusões sobre o comportamento da mente humana.

2.3. Processos de Informação

O principal objetivo das ciências cognitivas é explicar como as pessoas realizam seus pensamentos, abordando tipos diversificados de resolução de problemas, aprendizado e a maneira com que a mente humana efetua estas operações. Para tanto, deve-se analisar com maior profundidade o que significam os *processos de informação* que, como dito anteriormente, refletem o comportamento da mente ao receber, armazenar, recuperar, transformar e transmitir informações.

2.3.1. Significado e Propósito

Para compreender os processos de informação, deve-se observar o seu significado, ou seja, o propósito da sua utilização. A idéia é que um processo de informação tipicamente permite que um sistema ou entidade obtenha respostas sistemáticas dado um determinado conjunto de objetos em um ambiente bem definido. Ou seja, esta tarefa exige a análise do conteúdo da informação que se manipula, ou então seria impossível obter uma resposta satisfatória.

Analisando uma simples conta de adição, pode-se conceituar a aplicação de símbolos e semântica aos processos de informação. “Quando um dispositivo é alimentado com dois símbolos que representam dois números, ele deve produzir um terceiro símbolo que representa a soma dos dois primeiros”. Aqui, especifica-se o que o dispositivo deve fazer e não como deve fazer, assumindo que este dispositivo possui algum tipo de representação de números.

Com o propósito de construir um novo sistema, deve-se possuir um entendimento global do que se busca alcançar. Deve-se especificar, por exemplo, o tipo de informação a ser utilizada, quais os relacionamentos existentes, quais manipulações são permitidas, que resultado é esperado, entre outros. Em vista disto, um grande esforço na área de Inteligência Artificial envolve a descoberta dos tipos de conhecimento que caracterizam o comportamento humano inteligente.

2.3.2. Representação

Processos de informação envolvem claramente a manipulação de informação, sendo necessário estipular um modo de representação para a mesma. Pode-se inicialmente, estipular um conjunto de símbolos básicos que, através de um conjunto de regras — uma sintaxe —

permita a criação de símbolos mais complexos. No caso dos números, por exemplo, o conjunto de símbolos de 0 a 9 representa os símbolos básicos, e a concatenação deles permite a criação de números mais complexos. Entretanto, é necessário também compreender o que estes símbolos representam. O significado de um símbolo complexo, por exemplo, é composto pela junção do significado dos símbolos básicos. Além disso, a organização destes símbolos influencia diretamente o seu significado.

Um exemplo claro que ilustra a importância e a aplicação da sintaxe e da semântica pode ser representado com a utilização de numerais. Para o símbolo “10” representar uma dezena ou o valor “dez”, define-se a sintaxe como conjunto de numerais $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ e a semântica como o sistema decimal. Se for mantida uma porção da sintaxe, como $\{0, 1\}$ e adotada a semântica do sistema binário, o símbolo “10” não representa mais uma dezena, mas sim o valor “dois”.

Procedimentos de manipulação que processam uma entrada simbólica e fornecem uma saída também simbólica, ou seja, que analisam e constroem estruturas sintáticas, são denominados algoritmos [STI95]. Uma vez que o significado das estruturas simbólicas está implícito em sua sintaxe, estes procedimentos resultam em respostas significativas dentro do domínio de conhecimento que foi estipulado.

2.3.3. Descrição Formal

Um algoritmo é um procedimento formal ou um sistema porque é definido de acordo com a forma de representação da informação e não com o seu significado. Em uma visão mais simplista, não vai além da manipulação de padrões de acordo com a representação estipulada. Os procedimentos possuem significado, mas este é adquirido indiretamente através dos símbolos ou padrões que manipula.

Desta forma, se algoritmos podem ser executados sem considerar o significado das informações, pode-se seguir este mesmo conceito para a compreensão da mente humana. Por exemplo, ao dirigir um automóvel, sabe-se que virando o volante para a direita, o carro mudará sua trajetória para esta direção. Não é necessário conhecer as engrenagens que compõem o sistema de direção do automóvel tampouco as forças da física que atuam sobre o motor para que o mesmo funcione adequadamente. São conhecidos apenas os procedimentos “virar o volante” e “pressionar o acelerador”, sem haver a necessidade de um raciocínio profundo sobre estas ações.

Pesquisadores em ciências cognitivas buscam desenvolver processos de informação distinguindo:

- operações formais em símbolos;
- o relacionamento entre símbolos; e
- o significado que os símbolos e seus relacionamentos podem representar.

Um dos objetivos é a criação de um organismo ou máquina que obtenha um comportamento significativo através do processamento formal, sendo que a este processamento, são associadas estruturas simbólicas que fornecem um relacionamento representativo do mundo real [STI95]. Simplificadamente, busca-se descobrir princípios fundamentais altamente genéricos e explicativos sobre processamento de informação.

2.4. Sistemas da Mente

“Algo que a Ciência Cognitiva ainda não conseguiu realizar foi criar uma teoria unificada que explica a ampla gama de fenômenos psicológicos, no sentido de que a teoria evolucionária e a genética unificam os fenômenos biológicos e a teoria da relatividade e do quantum unificam a teoria da física” [THA98]. Diferentes cientistas cognitivistas argumentam que a mente é um sistema baseado em lógica, regras, conceitos, analogias, imagens ou conexões. Entretanto, se for feita a pergunta: *“Que tipo de sistema é a mente?”*, não se pode afirmar que ela é composta por apenas uma destas abordagens, mas sim, que é uma composição de todas elas, criando um sistema extraordinariamente complexo e possibilitando uma diversidade de tipos de pensamento.

Paul Thagard apresenta em [THA98] um estudo detalhado sobre as abordagens citadas no parágrafo anterior. Após suas avaliações, Thagard concluiu que a compreensão computacional-representacional da mente contribui muito para a compreensão teórica e para a aplicação prática, mas salienta que nenhuma abordagem pode ser caracterizada como a explicação mais clara ou mais completa das capacidades cognitivas humanas.

Tabela 2.1: Aplicações teóricas de abordagens computacionais

	Representação	Solução do Problema	Aprendizado	Linguagem
Lógica	proposição operadores predicados quantificadores	dedução plausibilidade	generalização abdução	análise gramatical
Regras	se - então	procura encadeamento para frente encadeamento para trás	<i>chunking</i> generalização abdução	gramática pronúncia soletração
Conceitos	figuras com pistas esquemas escritas	busca herança ativação expandida	abstração de exemplos combinação conceitual	léxico semântica
Analogias	alvo e fonte relações causais adaptação	recuperação busca	armazenamento formação de esquemas	metáfora
Imagens	visual, motora, etc.	busca, manipulação	prática imaginária	esquemas de imagens
Conexões	unidades e ligações	satisfação de restrições paralelas	ajustes do peso de retropropagação	esquemas de linguagem

Considerando o poder representacional e computacional, a plausibilidade psicológica e neurológica, bem como a aplicabilidade prática das abordagens apresentadas na Tabela 2.1, Thagard salienta os aspectos positivos de cada uma delas, qualificando-as adequadamente sob aspectos de: representação, solução do problema, aprendizado e linguagem.

Voltando a análise ao tratamento de linguagem natural, cada abordagem apresenta características particulares que justificam ou não sua utilização para este tipo de atividade:

- a *lógica formal* possui um considerável poder representacional, mas ao mesmo tempo apresenta algumas limitações por não captar todas as sutilezas da linguagem natural. Para sentenças extensas ou que exijam uma representação mais elaborada, a criação de proposições é bastante difícil, não sendo esta uma abordagem comumente utilizada em tratamento de linguagem natural;
- *sistemas baseados em regras* possuem uma representação concisa para o formato das regras: “SE x ENTÃO y”. Entretanto, as regras representam uma

estrutura muito fechada, não sendo favoráveis a adaptações. Desta forma, para a compreensão de sentenças em linguagem natural, seria necessário construir uma base de regras muito extensa a fim de prever todos os tipos de sentença de entrada;

- na linguagem falada ou escrita, os *conceitos* são costumeiramente representados por palavras, e, ao estabelecer inter-relacionamentos entre conceitos, pode-se obter uma poderosa forma de se representar conhecimento. Entretanto, uma palavra ou um conceito por si só pode apresentar ambigüidade na sua interpretação, sendo este um dos maiores problemas no tratamento da linguagem natural. Se não houver uma informação adicional para sua interpretação, não há como solucionar automaticamente este problema;
- muitos experimentos psicológicos apóiam a idéia de que a *imagem* visual faz parte do pensamento. Como aprender noções como “embaixo” e “em cima”, por exemplo, sem a ajuda de um instrumento visual? Entretanto, no tratamento automático de linguagem natural, é muito mais fácil trabalhar com palavras escritas do que com análise de figuras. Atualmente, não há registro de trabalhos de manipulação de imagens como processamento central para a compreensão de linguagem natural;
- os *modelos conexionistas* são aplicados ao processamento de linguagem natural, mas são, por sua estrutura, bastante complexos e exigem um grande esforço computacional. Muitos pesquisadores que defendem idéias conexionistas acreditam que esta venha a ser a melhor forma de tratar a linguagem natural, mas atualmente esbarra-se em problemas quanto à quantidade de neurônios que podem ser representados computacionalmente e ao poder de processamento das máquinas atuais; e
- em especial, *analogia* está salientada na Tabela 2.1 por representar o escopo deste trabalho e merecer maior atenção. A fim de efetivar a compreensão da linguagem natural, a utilização de analogias tem se mostrado uma abordagem inovadora e bastante interessante. Partindo do princípio que as pessoas entendem umas às outras ao comparar uma situação com outras previamente vivenciadas ou conhecidas, Christopher Riesbeck [RIE86] propôs uma nova abordagem para a compreensão da linguagem natural a partir da utilização de

analogias; intitulando seu trabalho de DMAP — *Direct Memory Access Parsing*.

Não apenas Christopher Riesbeck como também outros pesquisadores da área de computação apresentaram um grande interesse nos processos de compreensão do cérebro humano. Para Roger Schank e Chip Cleary [SCH94], por exemplo, freqüentemente as pessoas são submetidas a lembranças e comparações em virtude das experiências do dia-a-dia. A atividade de “lembrar” é um reflexo da busca constante da mente humana por informações antigas que possam facilitar a compreensão ou o aprendizado de uma nova situação. Constantemente, as pessoas acumulam casos e os comparam com outros já conhecidos a fim de entender e inclusive prever o que se pode esperar em seguida.

As analogias desempenham um papel muito importante no pensamento humano, diversificando-se em áreas como resolução de problemas, tomada de decisão, explanações e comunicação. Tipicamente, o raciocínio analógico compreende quatro estágios [THA98]:

- enfrentar o problema-alvo a ser resolvido;
- relembrar um problema-fonte semelhante ao qual já se conhece a solução;
- comparar os problemas fonte e alvo, identificando os componentes correspondentes; e
- adaptar o problema-fonte a fim de produzir uma solução para o problema-alvo.

Em geral, mecanismos automáticos de compreensão de linguagem natural precisam inicialmente pré-processar a sentença de entrada a fim de mapeá-la a um formato neutro que represente o seu significado. Na proposta de Riesbeck, entretanto, com o uso de analogias, este procedimento inicial é descartado, sendo a sentença trabalhada diretamente para obtenção do seu significado. Comparando seus procedimentos com os quatro estágios de compreensão apresentados por Thagard, pode-se visualizar o trabalho de Riesbeck como:

- ler uma palavra da sentença de entrada;
- encontrar na memória do sistema possíveis conceitos que se relacionem à palavra;

- à medida que outras palavras são lidas e outros conceitos são identificados, resolver problemas de ambigüidade e encontrar os conceitos mais específicos;
- e
- a fim de interpretar diretamente em uma nova leitura a sentença trabalhada, atualizar os conceitos da memória.

Desta forma, percebe-se claramente os estágios a serem cumpridos no raciocínio analógico em um ambiente computacional:

- *lembrança*: entre todos os conceitos representados na memória, recuperar aqueles que se relacionam ao problema fonte;
- *comparação*: dentre os conceitos — ou casos — recuperados, identificar aqueles que melhor representam o problema fonte; e
- *adaptação*: adaptar o(s) conceito(s) selecionado(s) na comparação para que o tratamento do problema fonte seja incorporado na memória, permitindo que em uma próxima leitura, o mesmo problema seja tratado diretamente.

2.5. Princípios Básicos da Ciência Cognitiva

Paulo Abrantes, em [ABR93], afirma que a diferença expressiva entre as abordagens utilizadas pelos antigos gregos e os cientistas cognitivistas de hoje encontra-se na utilização da tecnologia recente para estudar o comportamento da mente humana. Com participação marcante neste processo atual, encontra-se a aplicação de computadores na tentativa de simular a inteligência e testar as teorias propostas sobre a estrutura dos processos mentais. Abrantes salienta alguns dos pressupostos básicos sobre ciência cognitiva:

- o estudo do conhecimento humano requer uma investigação das capacidades representacionais e computacionais da mente;
- o estudo científico da mente deve ser desenvolvido a partir de uma perspectiva interdisciplinar; e
- os computadores, apesar de diferirem materialmente dos organismos, fornecem bons modelos para o estudo do sistema cognitivo humano.

O primeiro ponto indica que os estudos sobre os modelos representacionais da mente humana não são propriedade exclusiva de pesquisadores modernos da ciência cognitiva. Muito tempo antes do surgimento do computador e da sua aplicação em cognição, o homem buscava entender seus processos mentais, em especial envolvendo conhecimento, lembrança, comunicação e aprendizado.

Qualquer estudo relativo ao sistema cognitivo humano é bastante complexo, havendo neste caso um grande ganho de conhecimento a partir da interdisciplinaridade de pesquisas. De acordo com o segundo pressuposto, pesquisadores cognitivistas objetivam a criação de um campo unificado de pesquisa, englobando filosofia, matemática, lingüística, psicologia, neurociências e computação.

Em Ciência Cognitiva, uma vertente intitulada “funcionalismo”, reflexo do 3º pressuposto, sustenta que, apesar da diferença material existente entre computadores e seres humanos, ambos possuem a capacidade de processar informações. O interesse direciona-se então ao estudo da estrutura dos sistemas de processamento, a fim de explicar como estes sistemas se organizam na produção de estados cognitivos. Visando entender os modelos de organização, o funcionalismo divide-se em duas linhas de pesquisa: funcionalismo lógico-computacional (FLC) e funcionalismo neurocomputacional (FNC). Enquanto o primeiro enfatiza o aspecto lógico ou abstrato dos sistemas organizacionais, o segundo aborda os aspectos físicos responsáveis por sua organização [ABR98].

De acordo com as duas subdivisões supracitadas, é marcante a participação da Inteligência Artificial em FLC ao estudar os aspectos racionais da cognição, enfatizando a resolução de problemas, representação de conhecimento, a demonstração de teoremas e regras para manipulação de símbolos, a compreensão automatizada e a aprendizagem, entre outros. FNC, por sua vez, é conhecida também como Conexionismo, Redes Neurais ou Sistemas de Processamento de Informações em Paralelo. A ênfase neste caso não está centrada nos aspectos racionais dos sistemas cognitivos, mas nos seus aspectos adaptativos e seus elementos físicos (ou sub-simbólicos), na tentativa de representar uma rede de neurônios.

2.5.1. Psicologia Cognitiva e Inteligência Artificial

As questões pesquisadas em Inteligência Artificial são bastante próximas daquelas estudadas em psicologia cognitiva, diferindo, entretanto, nos métodos adotados para efetuar as pesquisas [STI95]. É possível desenvolver e testar programas computacionais que expressem

características da inteligência humana, podendo-se variar suas entradas e analisar os diferentes resultados obtidos. Para tanto, é necessário detalhar e formalizar a representação e os processos de uma teoria, havendo assim um controle genérico de todo o procedimento envolvido. Em psicologia, contudo, o controle sobre os processos cognitivos é bastante limitado, não podendo haver mudanças de teoria ou reprogramações para a realização de testes. Os avanços requerem um grande esforço, mas ao contrário dos sistemas artificiais, lida-se diretamente com a mente humana. Deve-se salientar que os métodos de ambas as linhas de pesquisa são complementares, podendo ainda fazer uso de técnicas ou teorias de outras áreas que englobam as ciências cognitivas.

Juntamente com a psicologia, a Inteligência Artificial tem mostrado um grande interesse no tratamento de conhecimento, enfocando aquisição, representação, recuperação e atualização do mesmo. Para solucionar problemas complexos e obter um comportamento automático inteligente, é necessária uma grande quantidade de conhecimento, representada de forma tal que possa ser devidamente manipulada para a obtenção de uma resposta satisfatória.

2.5.2. Representação de Conhecimento

Na área de Representação de Conhecimento, três especialistas — Randal Davis, Howard Schrobe e Peter Szolovits — elaboraram uma revisão crítica acerca do seu estado da arte. A partir desta revisão, apresentaram como resultado uma síntese com 5 princípios básicos sobre representação de conhecimento e seu papel na área de Inteligência Artificial, como descrito em [SOW00]:

- *Representação de conhecimento implica “substituição”*: objetos físicos, eventos e relacionamentos que não podem ser manipulados diretamente por computadores são devidamente substituídos por símbolos, formalizando assim a representação interna destas entidades. Através da manipulação destes símbolos, tem-se a simulação do mundo externo e viabiliza-se o raciocínio sobre o mesmo.
- *Representação de conhecimento é um conjunto de comprometimentos ontológicos*: a ontologia determina a categoria de entidades que podem existir no domínio de uma determinada aplicação. Sua representação através de

variáveis e seus tipos correspondentes determinam o comprometimento ontológico.

- *Representação de conhecimento é uma teoria parcial para o raciocínio inteligente:* a estrutura do programa reflete a estrutura do problema. Diferentes situações requerem abordagens diferenciadas, sejam elas lógicas, procedurais ou estruturadas, por exemplo. Ao descrever o comportamento do sistema e suas interações, a representação de conhecimento fornece suporte para o raciocínio sobre as entidades do domínio em questão.
- *Representação de conhecimento é um meio para a computação eficiente:* além de representar, um sistema de inteligência artificial deve codificar o conhecimento de forma a ser efetivamente processado por um computador. Conforme a necessidade, os axiomas do sistema podem ser codificados em diferentes linguagens, obtendo um melhor resultado de acordo com sua aplicação.
- *Representação de conhecimento é um meio de expressão de idéias:* a comunicação entre o Engenheiro de Conhecimento e o Especialista do Domínio deve ser clara para ambas as partes, garantindo assim a correta modelagem do sistema e facilitando a troca de idéias entre o engenheiro e o especialista.

As linguagens naturais podem ser vistas como as linguagens de representação de conhecimento de mais alto nível a que temos contato. Seu grande poder de expressão e sua flexibilidade, algumas das suas maiores características, são também alguns dos seus grandes obstáculos sob o ponto de vista computacional, o que pode comprometer a eficiência de um sistema.

Segundo Roger Schank [SCH71], as pessoas não criam ou interpretam uma sentença isoladamente. Elas falam com a intenção de comunicar alguma “coisa”, e é a descoberta desta “coisa” que caracteriza o domínio de estudo dos pesquisadores interessados em linguagem natural. O que se procura identificar é uma teoria que corresponda à capacidade humana de compreender um outro ser humano. Uma vez que a reação do ouvinte é uma das únicas formas de avaliar o que ele compreendeu — visto que não há uma medida específica para tal — parte desta teoria deveria incluir as regras de decisão e heurísticas que são utilizadas pelo

ouvinte no momento em que ele compreende uma sentença e dá início à criação da sua resposta.

Normalmente assume-se que consciência é saber o que está acontecendo no momento atual, ter uma sensação ou um conhecimento sobre o que está ocorrendo e por quê. Marvin Minsky, entretanto, afirma que consciência não reflete o presente, mas o passado, uma vez que se relaciona à lembrança de pensamentos recentes. Em [MIN85], apresenta ainda a seguinte questão: “*Como nós, seres humanos, efetivamente compreendemos algo?*”. Para ele, na maioria das vezes, a compreensão é realizada através de analogias, ou seja, para cada novo pensamento há a lembrança de algo que já se conhece. Assim, as novidades parecem similares àquilo que temos como habitual e, através do uso de sinais, símbolos ou nomes, a mente transforma o desconhecido em comum.

Desta forma, a memória participa efetivamente dos processos de compreensão, havendo não apenas a captura de conhecimento como também a adaptação frente a novos conceitos. Afirmações como estas culminaram no estudo da memória humana e na representação da sua estrutura e do seu comportamento em um ambiente computacional, favorecendo inclusive o desenvolvimento de sistemas baseados em uma memória dinâmica, como apresentado a seguir.

2.6. Memória Dinâmica

Marvin Minsky, em [MIN85], apresenta o seguinte comentário: “*geralmente falamos em memória como sendo coisas que guardamos em caixas na nossa mente, como objetos que mantemos nos armários de nossas casas*”. Roger Schank, por sua vez, afirma em [SCH94] que: “*temos a visão comum, porém incorreta, de que memória é simplesmente um depósito onde armazenamos nosso conhecimento quando não o estamos utilizando. (...) A visão correspondente de lembrança é que quando precisamos de uma porção de conhecimento, vamos até nossa memória e então a recuperamos*”.

Com estas afirmações, ambos se sentem insatisfeitos com definições que apresentam memória como um meio estático de armazenamento de informações. Para eles, o potencial da memória humana, que envolve a maneira como o conhecimento é representado, acessado, recuperado e manipulado é muito mais rico e, por conseguinte, mais complexo.

Na década de 80, na tentativa de simular computacionalmente o comportamento da memória humana, Roger Schank criou um esquema de representação e manipulação de

conhecimento intitulado Memória Dinâmica. Para ele, alteração é a regra e os fatos estáticos não são os mais significativos. Qualquer coisa aprendida varia com o tempo, e a manipulação desta alteração é uma importante parte da inteligência, tanto para máquinas quanto para os homens [SCH99].

2.6.1. Conceitualização de Memória Dinâmica

As idéias de Schank referentes à Memória Dinâmica têm sua base fundamentada em estudos referentes à memória humana. O ponto central é que humanos possuem uma memória complexa que sofre mudanças constantes. Memória certamente é um local onde se armazena conhecimento, mas seu conceito vai muito mais além. É um local onde se processam conhecimentos, alterando dinamicamente o que já se conhece.

Na tentativa de criar um sistema computacional inteligente, simulando desta forma o comportamento humano, Schank acredita que devem ser recriadas situações onde seja necessário alterar a memória do sistema, afetando, em alguns casos, a própria estrutura da memória. Desta forma, ele afirma em [SCH94] que para construir uma memória dinâmica, sua estrutura não deve ser afetada somente por processamentos externos, deve haver um comportamento interno à própria memória. Seu dinamismo está representado nas alterações que uma nova experiência pode requerer. É, por natureza, um sistema de aprendizagem.

Por exemplo, o sistema de uma biblioteca precisa ser previamente modelado para prever todas as categorias de livros a serem armazenados. Classificados por categorias, os livros apresentam informações sobre autor, editora, ano e assuntos relacionados. Entretanto, se for necessário criar ou eliminar algumas categorias, boa parte do sistema será afetada e precisará ser atualizada. Desta forma, este sistema não é dinâmico, pois é alterado com muita dificuldade e requer intervenção externa. Um especialista, em contrapartida, pode alterar facilmente o sistema de classificação do seu conhecimento de acordo com seus interesses ou quando um conhecimento específico implicar alterações. Ele pode relacionar uma informação à outra com bastante facilidade e ainda modificar sua estrutura de classificação após uma análise do que conhece. Tudo isto pode ser feito sem que ele perceba o que está acontecendo, pois o especialista possui uma memória dinâmica.

A memória humana é alterada dinamicamente no momento em que armazena uma nova informação e realiza alguma abstração ou especialização sobre o conhecimento já armazenado. À medida que surgem novas experiências, elas perturbam de alguma forma o

estado da memória, devendo ser armazenadas corretamente nesta estrutura. Estas perturbações podem ser causadas por experiências favoráveis à memória atual ou mesmo contrárias a elas, causando assim alterações de maior ou menor intensidade.

Estas características, bem observadas por Roger Schank, contribuíram para o desenvolvimento do conceito de Memória Dinâmica. Suas pesquisas culminaram na criação de uma estrutura de memória que permitiu a recriação deste comportamento de alteração, permitindo não só o aprendizado de novas experiências como também a compreensão automática de sentenças escritas em linguagem natural.

2.6.2. Memória Dinâmica vs. Scripts

Para compreender a variedade de tipos de estruturas que compõem a memória humana, Schank e Cleary em [SCH94] apresentam o seguinte exemplo: “*Quais os tipos de resposta podem ser obtidos para a pergunta ‘quando Bill Clinton nasceu?’ ?*”. Diferentes pessoas fornecem diferentes respostas que, em geral, variam entre três estratégias:

- sabe-se a resposta diretamente;
- sabe-se a idade de Clinton em alguma ocasião importante, como no momento em que assumiu a presidência e, desta forma, calcula-se sua idade atual; e
- não se sabe sua idade, mas tenta-se compará-lo com outra pessoa ao qual se conhece a idade.

Neste exemplo, para uma mesma pergunta, apresentam-se os seguintes tipos de resposta:

- *pontual*: localizando o fato na memória e fornecendo a informação requisitada;
- *baseada em inferências*: a partir de uma informação presente na memória, faz-se uma inferência sobre a possível resposta correta; e
- *baseada em casos*: a partir de casos representados na memória, encontra-se o que mais se relaciona à questão apresentada e adapta-se sua resposta.

Desta forma, percebe-se o relacionamento que existe entre a estrutura da memória e o tipo de processamento efetuado pelo sistema. As estratégias de memória dependem

diretamente do seu modo de organização, identificando de diferentes formas o conhecimento que se busca.

Na década de 70, o processamento de linguagem natural tinha como base o esquema de representação criado por Schank e Riesbeck [SCH99] denominado "*script*". Em sua definição inicial, *scripts* eram vistos como uma estrutura que descrevia uma seqüência apropriada de eventos em um contexto particular, ou ainda, como uma seqüência pré-determinada de ações que definia uma situação bem conhecida. A idéia era que o processo de inferência poderia ser reduzido uma vez que algumas inferências eram representadas diretamente no *script* relevante às questões apresentadas.

Scripts são facilmente utilizados para compreender ou prever as ações executadas por uma pessoa, uma vez que ela faz aquilo que se espera, de acordo com o *script* que determina suas ações. Ao entrar em um restaurante japonês, por exemplo, pode-se assumir que o cliente está em um "sushi bar", que está usando palitos e não garfos para comer e que está bebendo saquê. Estas expectativas podem ser criadas uma vez que o *script* "restaurante japonês" foi modelado de acordo com as experiências vividas por clientes neste tipo de restaurante. Ou seja, *scripts* cobrem situações bastante específicas e são baseados na experiência e repetição de fatos.

Entretanto, as pessoas também possuem em sua memória informações genéricas associadas a diferentes situações. Assim, pode-se dizer que existem na memória dois tipos de informação:

- estruturas específicas baseadas nas experiências;
- estruturas genéricas contendo informações mais abstratas.

A diferença então entre memória baseada em *scripts* e memória baseada em estruturas genéricas é que, no primeiro caso, o que se conhece sobre uma dada situação vem de experiências idênticas ou bastante similares. Se não forem encontrados na memória *scripts* iguais ou bastante similares à entrada, não há como garantir uma resposta satisfatória do sistema. No segundo caso, entretanto, a partir de estruturas mais abstratas, pode-se aplicar em uma dada situação informações acumuladas sobre situações diferentes. Aquilo que é generalizado sobre uma dada informação é tido como verdade, dando ao sistema mais informações no processo de manipulação da entrada. Este processo de generalização de

informações fornece mais agilidade ao sistema, além de representar ainda um certo ganho de informações; o que contribui diretamente para a construção de uma memória realmente dinâmica.

Aliando a idéia de expectativas que podem ser previstas em sistemas de *scripts* e o conceito de generalização de informações, Roger Schank viabilizou a criação de estruturas dinâmicas de memória através do conceito de MOPs — *Memory Organization Packages*, apresentado na seção a seguir.

2.7. Memory Organization Packages

Em [SCH94], Schank afirma que as estratégias da memória humana dependem, em parte, das estruturas mentais que a organizam a fim de que se possa localizar um conhecimento quando necessário. É implausível acreditar que todas as estruturas representadas na memória são ativadas cada vez que se deseja responder um questionamento, então, ao invés disso, são utilizadas estruturas organizacionais que identificam o conhecimento relevante para uma dada situação. Estas estruturas organizacionais auxiliam a classificação e a localização de informações de mais baixo nível como fatos ou casos, por exemplo. Auxiliam ainda a captura de informações através de generalizações e permitem a criação de previsões que podem guiar um processo de raciocínio¹.

A estrutura de conhecimento representada por MOPs envolve preceitos básicos de Inteligência Artificial, como *frames*, abstrações e redes de herança [MIN85]. O diferencial está no comportamento dinâmico apresentado pela estrutura de memória que, nestes conceitos tradicionais, é estática. Basicamente, os MOPs contêm as mesmas informações apresentadas em *scripts*, organizadas, porém, através de uma rede de conceitos inter-relacionados.

2.7.1. Frames e MOPs

Em geral, pode-se afirmar que o objetivo de um sistema de memória é possibilitar o armazenamento e a recuperação de conhecimento. No padrão de modelagem sugerido por Roger Schank, este conhecimento é representado através de MOPS, um tipo de *frames* criado inicialmente para representar memória episódica, ou seja, eventos particulares e suas generalizações.

¹ Usa-se o termo genérico “processo de raciocínio” para qualquer processamento que possa ser efetuado sobre uma estrutura de memória modelada através de MOPs.

Estrutura de um Frame

Um *frame* constitui a representação de um objeto, evento ou situação do mundo real, por exemplo. Identificado por um nome único em uma estrutura hierárquica, consiste de um conjunto de *slots* (ou atributos) que determinam a relação atributo-valor de uma determinada característica deste objeto.

Desta forma, os *slots* são compostos por duas partes:

- papel (*role*): é o nome do *slot*, identificando-o no conceito em que é definido. Em um mesmo *frame*, não há *slots* com nomes iguais; e
- conteúdo (*filler*): é o valor assumido pelo slot. Este valor pode ser um dado propriamente dito, como o nome de uma pessoa, ou pode referenciar um outro conceito existente na estrutura de memória.

A Tabela 2.2 ilustra a representação abstrata de um *frame*.

Tabela 2.2: Representação abstrata de um frame

Representação	Significado
joão-1: homem nome = "João da Silva" idade = 22	Pessoa do sexo masculino cujo nome é João da Silva e cuja idade é de 22 anos. Este conhecimento é internamente referenciado pelo identificador "joão-1"
Evento-3: evento ator = joão-1 ação = ingerir objeto = maçã-92	Evento onde o ator João da Silva ingere uma maçã. Este conhecimento é internamente referenciado pelo identificador "evento-3".

Ligações entre os Conceitos

Um mecanismo natural de dedução nos *frames* é realizado através da herança². Os conceitos são representados em uma estrutura hierárquica de modo que os mais especializados podem herdar as propriedades dos conceitos mais genéricos.

Da mesma forma que em *frames*, para viabilizar este procedimento na memória dinâmica, são estabelecidos diferentes relacionamentos entre os MOPs através de ligações. Assim, um processo de raciocínio pode caminhar entre os MOPs e, obedecendo a estrutura

² Em geral, na tentativa de definir o valor exato de um atributo, sistemas computacionais apresentam problemas quando a árvore de conhecimento apresenta múltipla herança (ou seja, quando um conceito específico possui mais de uma ligação para diferentes conceitos mais abstratos). Pesquisas voltadas a este problema podem ser encontradas em [AVI01a] [AVI01b] [AVI01c] e [AVI01d].

modelada, efetuar suas inferências. Os tipos de ligações existentes apresentados por Riesbeck em [RIE89] e ilustrados na Figura 2.1 são:

- *abstrações*: representam os conceitos mais genéricos na estrutura organizacional. Se for observado um conceito num certo nível da hierarquia e outro a pelo menos um nível acima, diz-se que este segundo conceito abstrai o primeiro. A abstração permite que o conceito filho (conceito de nível mais baixo) herde características do conceito pai (conceito de nível mais alto), admitindo-se como verdadeiras todas as informações herdadas.
- *especializações*: definem conceitos mais especializados em relação às abstrações. Um conceito que esteja pelo menos um nível abaixo de outro, é caracterizado como uma especialização deste conceito (ou seja, é o conceito filho). Em geral, as especializações contêm informações mais específicas sobre os conceitos representados na memória pois representam um maior grau de especificidade da informação. Parte-se de informações genéricas ou mais abstratas e, com a definição de características mais detalhadas, definem-se as especializações.
- *instanciações*: referem-se a ocorrências específicas de um conceito. Caracterizam-se como um tipo de especialização particular pois são os nós folhas na rede que estrutura a memória. Representam, desta forma, o último nível de especialização.
- *empacotamentos*: representam as cenas de um *script*. São caracterizados por eventos específicos que ocorrem em um MOP, ou seja, informações que complementam um conceito mas que, efetivamente, não participam da sua definição. São caracterizados pelo par “atributo-valor”, onde atributo é um *slot* e o seu valor aponta para outro conceito. Na Figura 2.1, são representados pelos arcos nomeados.

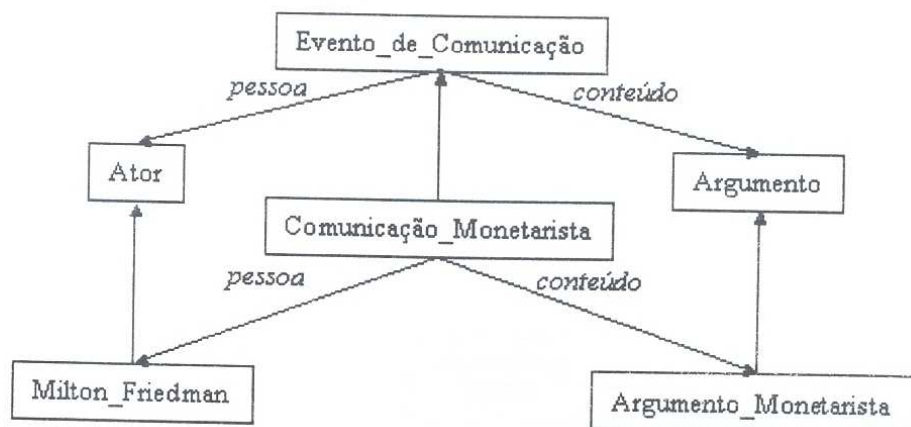


Figura 2.1: MOPs sobre eventos de comunicação

No exemplo da Figura 2.1, “Evento_de_Comunicação” é o MOP mais abstrato na hierarquia, tendo como especialização “Comunicação_Monetarista”. “pessoa” e “conteúdo” são empacotamentos que apontam para os conceitos “Ator” e “Argumento”, respectivamente. Os conceitos “Milton_Friedman” e “Argumento_Monetarista”, por sua vez, representam os conceitos mais específicos da hierarquia, caracterizando-se desta forma como instâncias de “Ator” e “Argumento”.

Todas estes relacionamentos ou ligações são de fundamental importância, uma vez que determinam quais são as informações disponíveis em determinados estágios do processo de raciocínio.

2.7.2. Recuperação de Conhecimento em MOPs

O processo central de recuperação de conhecimento em uma estrutura de memória baseada em MOPs pode ser definido genericamente como um processo de busca [RIE89]. A idéia básica é a seguinte: dado um MOP de entrada e o seu conjunto de *slots*, deve-se encontrar na memória o MOP (ou conceito) mais específico que seja compatível com a definição dos *slots* do MOP de entrada. Busca-se então por um conceito previamente modelado na memória com exatamente o mesmo conjunto de slots que se procura. Caso este conceito não seja encontrado, cria-se um novo conceito na memória de acordo com algumas restrições, ou então, executam-se procedimentos de tratamento de falha.

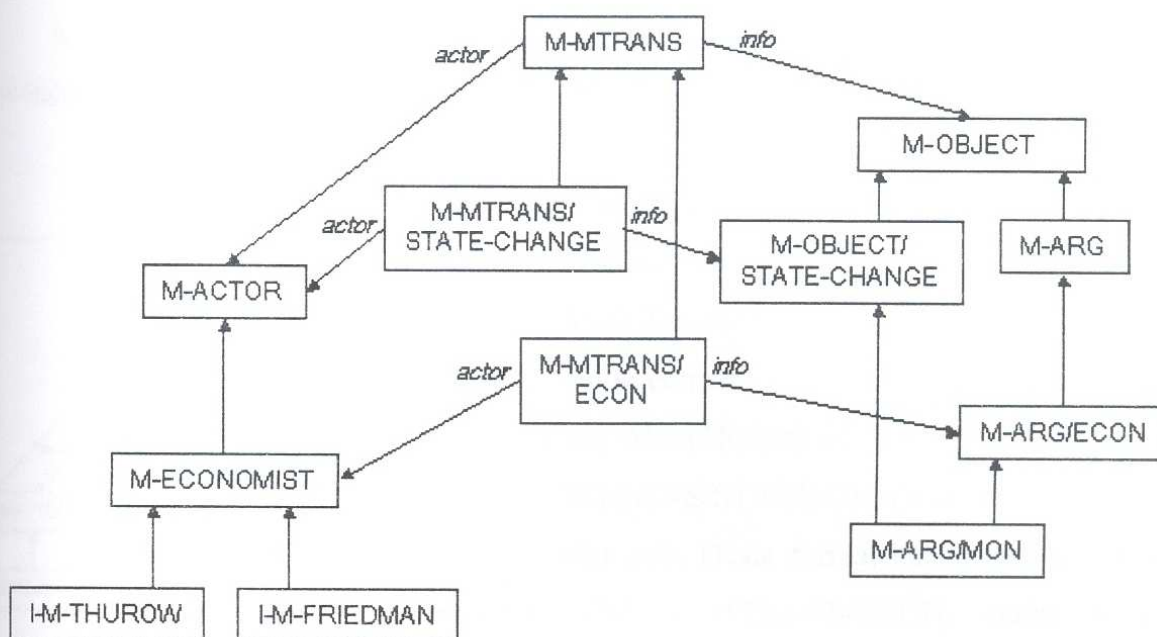


Figura 2.2: Hierarquia Abstrata sobre M-MTRANS

A ilustração da Figura 2.2 refere-se ao seguinte exemplo³:

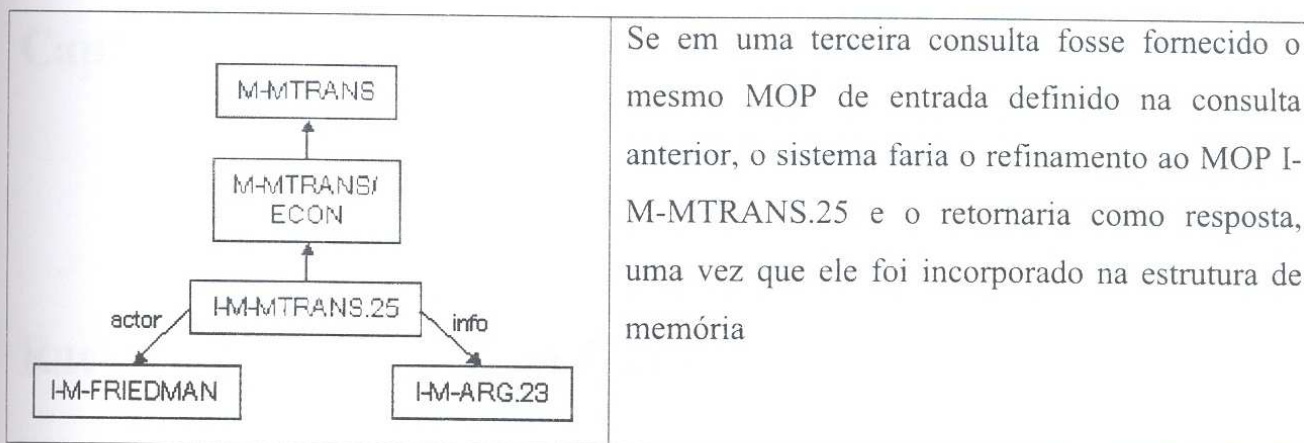
- Conceitos iniciados por “M-” representam abstrações e aqueles iniciados por “I-M-” representam instâncias;
- M-MTRANS é um conceito na memória que abstrai eventos de comunicação. Suas duas especializações são M-MTRANS/ECON, que representa um argumento causal fornecido por um economista, e M-MTRANS/STATE-CHANGE, que representa a previsão de uma mudança de estados;
- Associados a M-MTRANS estão os empacotamentos *actor*, que aponta para M-ACTOR e indica o ator do evento; e *info*, que aponta para M-OBJECT que representa o “objeto mental” de um M-MTRANS. M-OBJECT contém o slot *content* que, se preenchido com uma informação causal sobre economia, tem-se a especialização M-ARG/ECON. Caso contrário, se preenchido com uma informação sobre mudança de estado, tem-se a especialização M-MOBJECT/STATE-CHANGE; e
- I-M-THUROW e I-M-FRIEDMAN representam dois economistas que podem ser os atores de um evento M-MTRANS.

³ Este exemplo foi recuperado de [RIE89] e apresenta uma das 11 primitivas que denotam Dependências Conceituais. Informações mais específicas podem ser encontradas em [SCH77].

A partir desta estrutura previamente modelada na memória, pode-se iniciar a busca por um conceito. Os procedimentos de busca estão apresentados na Tabela 2.3.

Tabela 2.3: Tabela de Busca em MOPs

<pre> graph TD M-MTRANS -- actor --> M-MTRANS_STATE_CHANGE[M-MTRANS/STATE-CHANGE] M-MTRANS -- info --> M-OBJECT_STATE_CHANGE[M-OBJECT/STATE-CHANGE] M-MTRANS_STATE_CHANGE -- actor --> M-ACTOR M-MTRANS_STATE_CHANGE -- actor --> M-MTRANS_ECON[M-MTRANS/ECON] M-ACTOR -- actor --> M-ECONOMIST M-MTRANS_ECON -- actor --> M-MTRANS_STATE_CHANGE M-MTRANS_ECON -- info --> M-OBJECT_STATE_CHANGE M-OBJECT_STATE_CHANGE -- actor --> M-OBJECT M-OBJECT_STATE_CHANGE -- actor --> M-ARG_ECON[M-ARG/ECON] </pre>	<p>Dado o MOP M-MTRANS com os <i>slots</i> (<i>actor</i> = M-ECONOMIST) e (<i>info</i> = M-ARG/ECON) como entrada. Estes <i>slots</i> são iguais aos definidos em M-MTRANS/ECON mas não aos de M-MTRANS/STATE-CHANGE devido ao valor do <i>slot info</i>. Desta forma, há um refinamento até o MOP M-MTRANS/ECON, sendo o mesmo retornado como resultado.</p>
<pre> graph TD M-MTRANS -- actor --> M-ACTOR M-MTRANS -- actor --> M-MTRANS_ECON[M-MTRANS/ECON] M-MTRANS -- info --> M-OBJECT_STATE_CHANGE[M-OBJECT/STATE-CHANGE] M-ACTOR -- actor --> M-ECONOMIST M-MTRANS_ECON -- actor --> M-MTRANS_STATE_CHANGE[M-MTRANS/STATE-CHANGE] M-MTRANS_ECON -- actor --> M-MTRANS_ECON M-MTRANS_ECON -- info --> M-OBJECT_STATE_CHANGE M-OBJECT_STATE_CHANGE -- actor --> M-OBJECT M-OBJECT_STATE_CHANGE -- actor --> M-ARG_ECON[M-ARG/ECON] M-ARG_ECON -- actor --> M-ARGMON M-ARGMON -- actor --> M-ARG M-MTRANS_ECON -- actor --> I-M-MTRANS_25[I-M-MTRANS.25] I-M-MTRANS_25 -- actor --> I-M-FRIEDMAN I-M-MTRANS_25 -- actor --> I-M-ARG_23[I-M-ARG.23] I-M-FRIEDMAN -- actor --> M-ECONOMIST I-M-ARG_23 -- actor --> M-ARGMON </pre>	<p>Em uma segunda consulta à memória, dá-se como entrada o MOP M-MTRANS com os <i>slots</i> (<i>actor</i> = I-M-FRIEDMAN) e (<i>info</i> = I-M-ARG.23), onde I-M-ARG.23 é uma instância de M-ARG/MON. Da mesma forma que na primeira consulta, há um refinamento ao conceito M-MTRANS/ECON, entretanto, a entrada possui informações mais específicas do que as encontradas neste conceito. Desta forma, o sistema cria um novo conceito na memória, I-M-MTRANS.25, especializando-o a partir de M-MTRANS/ECON e estabelecendo os empacotamentos <i>actor</i> e <i>info</i> para I-M-FRIEDMAN e I-M-ARG.23, respectivamente.</p>



Se em uma terceira consulta fosse fornecido o mesmo MOP de entrada definido na consulta anterior, o sistema faria o refinamento ao MOP I-M-MTRANS.25 e o retornaria como resposta, uma vez que ele foi incorporado na estrutura de memória

Este processo de refinamento baseia-se na comparação dos *slots* de entrada com os *slots* do conceito mais específico encontrado na memória. Se houver uma igualdade, o sistema já possui todas as informações sobre a resposta desejada, não sendo necessária nenhuma intervenção na memória. Caso contrário, com a nova especialização criada, o sistema consegue elaborar um novo relacionamento, “aprendendo” mais informações sobre os conceitos modelados na memória.

2.8. Considerações Finais

Neste capítulo foram apresentados conceitos genéricos sobre ciências cognitivas e suas aplicações na área de computação. São visíveis não só as contribuições obtidas com a junção destas áreas, como também os avanços da informática ao longo dos anos na tentativa de reproduzir os processos da mente humana.

Em especial, pesquisas referentes ao comportamento da memória humana têm-se mostrado de grande importância para a criação de sistemas de compreensão de linguagem natural. Nesta linha de pesquisa, despontaram importantes trabalhos como a formalização de conceitos, de Marvin Minsky; os Scripts e a Memória Dinâmica de Roger Schank.

Enfocando ainda a utilização de uma memória computacional, Roger Schank propôs a abordagem de Raciocínio Baseado em Casos a fim de viabilizar a resolução de novos problemas a partir de outros problemas previamente resolvidos. O capítulo seguinte apresenta os conceitos principais desta abordagem como um dos fundamentos do trabalho proposto nesta dissertação.

Capítulo 3

Raciocínio Baseado em Casos

3.1. Considerações Iniciais

Estudos têm mostrado que seres humanos, quando necessitam desenvolver alguma forma de raciocínio para solucionar ou interpretar problemas, acabam muitas vezes se apoiando em experiências anteriores [KOL93]. Desta forma, procurou-se reproduzir este comportamento em sistemas computacionais a fim de reaproveitar conhecimento e utilizar esta técnica na resolução automática de problemas.

Diferente dos paradigmas tradicionais da Inteligência Artificial, o Raciocínio Baseado em Casos (*Case-Based Reasoning* — CBR) não se fundamenta na generalização de conhecimento ou de procedimentos para resolver problemas. Seu foco principal são conhecimentos específicos, devidamente armazenados, de situações ou problemas previamente experimentados e tratados. A intenção é fazer uso destas informações para resolver novos problemas que possam surgir, aproveitando desta forma o conhecimento que foi adquirido na resolução de problemas anteriores.

Este capítulo apresenta uma visão geral sobre Raciocínio Baseado em Casos, detalhando seu paradigma e os métodos que o compõem.

3.2. Inteligência Artificial e Cognição

Do ponto de vista da pesquisa científica, a Inteligência Artificial busca compreender processos da mente humana como pensamento, aprendizado, memória e planejamento a fim de identificar o comportamento humano inteligente. Sob o aspecto tecnológico, os estudos buscam criar artefatos que reproduzam este comportamento a fim de que possam ser aplicados em tarefas automatizadas. CBR enquadra-se nestas duas áreas, sendo considerado uma teoria

psicológica do conhecimento humano que fornece fundamentação para uma nova tecnologia de sistemas computacionais inteligentes que podem resolver problemas e adaptá-los para uma nova situação [FRE96].

CBR se fundamenta principalmente na idéia de que o conhecimento de experiências passadas (denominadas *casos*) pode guiar o comportamento humano e, sendo assim, um sistema automático pode utilizar experiências prévias (problemas e soluções conhecidos) para interpretar e resolver problemas novos. Sua metodologia se baseia em dois princípios da natureza do mundo [LEA96]:

- problemas similares possuem soluções similares; e
- os tipos de problemas tendem a se repetir.

Conseqüentemente, problemas futuros têm grandes chances de ser semelhantes aos problemas atuais e a utilização de técnicas de lembrança e reutilização de conhecimento compõem uma estratégia bastante efetiva de raciocínio.

3.2.1. Raciocínio Humano

Em geral, as pessoas associam a inteligência humana ao conhecimento de várias regras. Se isto fosse verdade, peritos seriam formados apenas com a leitura de diferentes livros a respeito de um assunto particular e, seguindo a mesma linha de pensamento, poderia ser dito que alunos dos primeiros anos de graduação seriam os mais inteligentes, uma vez que neste período exige-se muito a leitura de livros, artigos e manuais. Faz sentido dizer que a inteligência vem do conhecimento, mas muito deste conhecimento está expresso de forma diferente daquela encontrada nos livros [SCH95].

Ao resolver um problema, geralmente o raciocínio humano trabalha com lembranças de casos similares já vivenciados pela pessoa. Ao esperar na fila do banco, são lembradas outras situações em que isto ocorreu. Ao entrar em um restaurante, sabe-se o comportamento esperado uma vez que outras ocasiões como esta já foram presenciadas mais de uma vez. Constantemente, as pessoas estabelecem algum tipo de comparação ou analogia entre situações similares e, de alguma forma, ainda aprendem com isto. Ao entrar em um “sushi bar” pela primeira vez, por exemplo, sabe-se que o garçom irá trazer o menu e que servirá o prato desejado, entretanto, pode-se não saber como usar os palitos para comer. Neste caso, uma nova informação é aprendida e este novo episódio é armazenado na memória. Em uma

visita futura, o comportamento adotado não será mais surpreendente uma vez que se possui o conhecimento prévio sobre a situação.

A todo o momento, as pessoas acumulam casos e os comparam com outros já conhecidos na tentativa de compreender a situação, de aprender algo novo e inclusive de prever o próximo caso que possa surgir. Estas previsões podem conduzir a novos casos e também a casos onde aconteceu algo de errado, onde houve uma falha. De acordo com esta última situação, a previsão pode ser útil para impedir que um erro se repita ou para alertar a pessoa que algo de errado pode acontecer, afetando assim as suas ações.

3.2.2. Lembrança

Especialistas humanos não são sistemas de regras, eles são bibliotecas de experiências [RIE89]. E estas bibliotecas são adaptáveis, à medida que recebem novos casos que podem modificar, substituir, ampliar ou contrariar experiências prévias. De alguma forma, uma informação nova perturba o conhecimento anterior, devendo este se adaptar à nova informação, encaixando-a na estrutura existente da melhor maneira. Uma estrutura de memória que não se adapta a novas experiências, ou seja, que não aprende, não apresenta grande utilidade.

Além desta capacidade de adaptação, deve ser fácil encontrar informações nesta memória. A primeira tarefa é identificar as experiências que se relacionam àquilo que se busca, mas como estabelecer estes relacionamentos? Os episódios da memória podem se relacionar por possuírem características semelhantes ou por contradizerem uns aos outros. Podem ainda se relacionar de diferentes formas de acordo com o contexto em questão, afetando assim a recuperação dos casos na memória.

Um fenômeno que esclarece alguns problemas de aprendizado e recuperação de informações é a lembrança [RIE89]. Roger Schank afirma que a lembrança é o ponto central para a compreensão, pois, para ele, compreender algo novo significa encontrar informações na memória que se relacionem a esta novidade. A lembrança de um objeto físico pode conduzir à lembrança de outro objeto, assim como a lembrança de uma situação ou um evento pode conduzir à lembrança de outro.

Schank afirma ainda que a natureza da memória e o processo de compreensão revelam características significantes:

- se é encontrado um episódio na memória porque se estava procurando por ele, pergunta-se: “*Como era conhecida a existência deste episódio de forma que ele pudesse ser encontrado?*”; e
- se a explicação para a lembrança é que acidentalmente encontra-se um caso na memória, pergunta-se: “*Por que este acidente ocorre e que relevância este acidente tem para o processamento?*”.

Marvin Minsky [MIN85] expõe ainda uma terceira pergunta: “*Quanto pode ser lembrado?*”. Algumas vezes, as pessoas lembram-se de fatos que explicitamente não sabem conhecer e, como resposta, teorias antigas da psicologia afirmam que os seres humanos podem se lembrar de tudo. Entretanto, Minsky suspeita que na verdade não se pode lembrar tudo nem de uma experiência particular. Inconscientemente, apenas alguns estados podem ser lembrados por terem sido classificados como úteis, perigosos, interessantes ou significativos por uma ou outra razão, por exemplo. Não seria razoável armazenar grandes quantidades de memórias não classificadas se, a cada momento, fosse desejado procurar por apenas uma delas.

As características listadas por Schank e as idéias de Minsky coincidem em um ponto: classificação. Durante o raciocínio, não se busca conscientemente por um episódio único na memória uma vez que pode-se não saber da existência deste episódio. O que se conhece é a classificação de casos similares ao que está sendo trabalhando, podendo-se, desta forma, encontrar estruturas na memória que contêm experiências relacionadas. Assim sendo, a lembrança ocorre quando é encontrada a estrutura de memória mais apropriada que auxiliará o processamento da nova experiência.

Certamente, quanto mais se sabe sobre um assunto, um maior número de informações (presumivelmente, informações de melhor qualidade) podem ser lembradas ou recuperadas em uma determinada situação. Desta forma, espera-se que especialistas em áreas do conhecimento particulares lembrem-se de experiências relacionadas a estas áreas e, quando submetidos a determinados problemas, saibam resolvê-los mais facilmente (uma vez que já conhecem o problema) ou adaptem seu conhecimento de forma a resolver um novo problema.

Independente de resolverem problemas novos ou similares, ou de obterem respostas de sucesso ou erro, o raciocínio baseado em casos é aprendido através da experiência. Um complemento à idéia de “raciocínio por lembrança” está o princípio que “raciocínio é lembrado”, ou seja, raciocínio e aprendizagem estão intimamente ligados [LEA96]. Todo o esforço despendido em uma situação, se não for armazenado, será totalmente desperdiçado. Por isso, uma estrutura baseada em casos é atualizada à medida que novos casos são aprendidos, de forma que este novo conhecimento possa ser utilizado no futuro.

3.2.3. A Ilusão do Controle

Ao acordar no meio da noite, uma pessoa tenta voltar a dormir, mas não consegue de imediato. Procura relaxar, “desligar a mente” e descansar, mas fica pensando em como será o dia seguinte, nos problemas que enfrentou dias atrás, nas reuniões, conversas e trabalhos que virão pela frente e parece não conseguir controlar a sua mente. Por que isto ocorre?

Com o exemplo exposto acima, Roger Schank [SCH99] apresenta a importância em se diferenciar pensamento racional e pensamento não-consciente. A maioria do conhecimento que as pessoas detém, que as habilita a fazer diferentes coisas, em geral não é conscientemente conhecido. Ao executar tarefas do dia-a-dia, como escovar os dentes, dirigir um carro, conversar ou compreender um noticiário, não é necessário pensar sobre cada passo que compõem estas tarefas. Elas são executadas de forma “automática”, como se fosse seguido um roteiro que foi aprendido com o passar do tempo através das experiências vivenciadas. Certamente, pode-se tomar decisões sobre a escolha de uma profissão, para onde viajar nas férias ou qual caminho seguir, mas para Roger Schank, mesmo este tipo de pensamento não é feito de forma consciente e racional. Para ele, o processo do pensamento é muito mais complexo que efetuar algumas simples decisões.

Existe uma grande diferença entre conhecimento e consciência. Muitas vezes, as pessoas não sabem necessariamente o que estão fazendo ao executarem uma tarefa. Ao serem questionadas sobre alguma lei da física, por exemplo, podem citar precisamente algumas fórmulas, quem as desenvolveu e para que servem. São conhecidos os estudos e testes que foram realizados e pode-se descrevê-los em detalhe tendo domínio do assunto. Contudo, na leitura e compreensão de uma sentença, a situação é diferente. Como exemplo, pede-se que seja definido o significado da palavra “casa”. Facilmente, pode-se defini-la como um tipo de moradia; entretanto, dependendo do contexto da sentença, a mesma palavra pode representar

diferentes significados: “Paulo casa com Joana na catedral da cidade”, “Mova sua rainha para a casa da direita”, “O botão desta blusa está sem casa”. Assim, na tentativa de compreender qualquer sentença com a palavra “casa”, precisa-se rejeitar, de alguma forma, significados que não se relacionam ao contexto da sentença. Ou seja, pode-se analisar aspectos sobre nomes, tempos verbais e outras coisas que se aprendem na escola, mas, na verdade, não há explicações claras sobre como agem os processos humanos de compreensão; apenas compreende-se [SCH99].

O conhecimento racional é aquele que pode ser explicitamente declarado sob a forma de conceitos, fatos, experiências ou regras, por exemplo. Pode-se definir uma ordem lógica ou uma seqüência de passos que guiam o raciocínio e ele pode ainda ser adquirido através de livros ou outras fontes. Por sua vez, o conhecimento não-consciente está presente em atividades básicas mentais, mas não pode ser formalizado. Roger Schank afirma que este conhecimento é o coração de tudo que se sabe. Não se pode definir como as pessoas constroem sentenças enquanto falam ou escrevem e, de forma análoga, não se sabe como a leitura de uma sentença conduz à compreensão.

A resolução de problemas depende, presumivelmente, de conhecimento racional. Geralmente, as pessoas seguem corretamente seqüências de instruções para realizar alguma tarefa, mas, sem estas instruções, muitas vezes são incapazes de repetir a mesma tarefa. Isto ocorre porque não houve aprendizado ou qualquer tipo de ganho de conhecimento. Este cenário caracteriza o fato de que simplesmente seguir uma seqüência de regras não confere necessariamente conhecimento a quem as executa. O mesmo ocorre com computadores: não se pode dizer que, por seguirem seqüências lógicas de regras, um computador saiba o que está fazendo.

No final da década de 70, pesquisadores concluíram que o poder de um programa em resolver problemas dependia mais do conhecimento que possuía do que do formalismo ou esquema de inferência empregados. Surgiram então os Sistemas Especialistas [RUS95]. Estes sistemas são voltados a domínios específicos do conhecimento e compostos por conjuntos de regras que reproduzem a tomada de decisão de um especialista. Entretanto, como apresentado anteriormente, seguir uma seqüência de regras não significa que o sistema tem conhecimento do problema que está resolvendo. Se o mesmo problema ocorrer mais de uma vez, o sistema não possui nenhuma informação que possa ser aproveitada da resolução anterior e, desta forma, as regras devem ser executadas novamente, na mesma ordem, sem interrupções.

Em contrapartida, CBR é o oposto da hipótese de raciocínio inteligente através de regras. Em sistemas baseados em casos, o raciocínio é efetuado de acordo com experiências conhecidas que guiam a resolução de novos problemas. CBR envolve a utilização de uma memória dinâmica que reutiliza conhecimento e se adapta a novos casos. Não se pode afirmar que o raciocínio baseado em casos é necessariamente um processo não-consciente uma vez que se baseia na lembrança e, desta forma, apresenta um comportamento semelhante. Algumas vezes, uma lembrança ou um caso é recuperado com muita facilidade; em outras, não se sabe o motivo por que uma lembrança foi recuperada.

3.3. Fundamentos de Raciocínio Baseado em Casos

O princípio de um raciocinador baseado em casos é simples: a resolução de problemas é apoiada na solução de casos semelhantes resolvidos previamente. Segundo Christopher Riesbeck e Roger Schank [RIE89], raciocínio baseado em casos significa raciocínio sobre exemplos prévios. Ou seja, inicialmente deve-se determinar as similaridades entre o problema a ser resolvido e os casos armazenados na memória e, uma vez determinado o caso mais semelhante, adaptar a solução deste caso para que se possa resolver o problema em questão, como ilustrado na Figura 3.1.

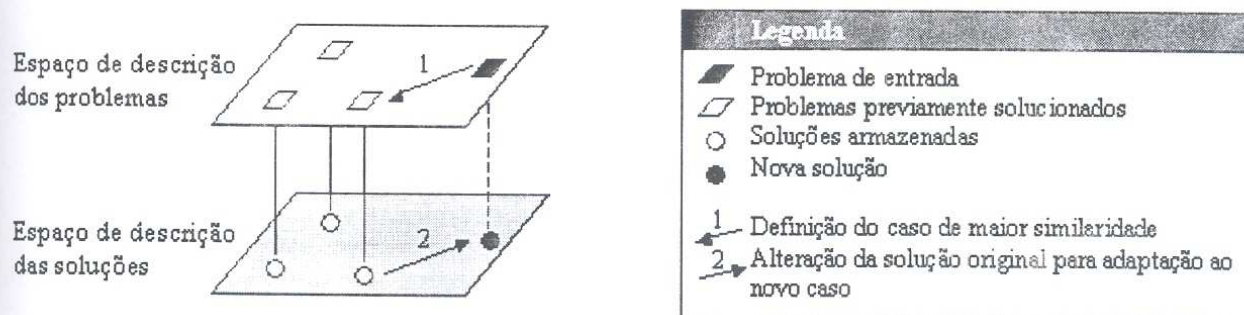


Figura 3.1: Resolução de problemas baseado em casos anteriores

As técnicas de CBR sugerem um modelo de raciocínio que incorpora resolução de problemas, compreensão e aprendizado; integrando tudo isso com processos na memória. Janet Kolodner [KOL93] apresenta as seguintes premissas que fundamentam este modelo:

- referenciar casos passados é vantajoso quando situações podem se repetir. Referenciar casos passados semelhantes é freqüentemente necessário para lidar com as complexidades de uma situação não conhecida. Desta forma, lembrar um

caso e utilizá-lo na resolução de casos futuros, integrando esta nova resolução à base de conhecimento, representa um processo de aprendizagem;

- como as descrições de problemas são geralmente incompletas, a compreensão ou a interpretação de um problema é um pré-requisito para o raciocínio. Um raciocinador baseado em casos não pode recuperar casos semelhantes se não compreender a situação que está tratando. Toda forma de raciocínio requer que o problema seja bem elaborado e que seja detalhado com clareza suficiente de modo que o raciocinador possa reconhecer a situação. Em CBR, entretanto, durante a resolução do problema, pode-se obter um maior entendimento do mesmo e, desta forma, recuperar um maior número de casos correspondentes que auxiliem a sua resolução final;
- como casos antigos nem sempre são idênticos ao novo caso, muitas vezes é necessário adaptar a solução antiga para adequá-la ao problema em questão;
- o aprendizado ocorre naturalmente como consequência do raciocínio. Se um novo procedimento é derivado da solução de um problema complexo e este procedimento não apresenta irregularidades, o mesmo é incorporado como uma solução para este tipo de problema. Uma vez aprendido, o procedimento é devidamente indexado na memória de casos do sistema e pode ser recuperado em consultas futuras. Se ocorrerem problemas na utilização deste caso, o raciocinador é avisado que o procedimento do caso tem apresentado falhas ou que seu índice pode estar incorreto. Pode-se tentar analisar os resultados do novo caso e tentar corrigir seus problemas. Quando um novo caso é inserido na biblioteca de casos, ele representa um refinamento do caso de que foi originado. Ambos devem ter seus índices definidos de forma que a recuperação dos mesmos seja feita nas situações apropriadas. Este processo incremental de reconhecimento e criação representa o processo de aprendizado de novos casos;
- as respostas obtidas pelo sistema bem como a análise das mesmas faz parte do ciclo de raciocínio e aprendizagem. Se não houver uma avaliação baseada nestas respostas, não se pode garantir o correto armazenamento de novos casos e pode haver um comprometimento na aprendizagem, onde casos previamente armazenados também podem ser afetados por uma alteração de indexação mal feita.

Desta forma, estas premissas sugerem que a qualidade de um raciocinador baseado em casos depende basicamente:

- da experiência prévia que possui;
- da habilidade de reconhecer similaridades entre casos novos e casos antigos;
- da habilidade de efetuar adaptações;
- da habilidade de realizar avaliação e reparos; e
- do correto armazenamento de novas experiências na biblioteca de casos.

Um raciocinador com uma menor quantidade de casos trabalhará com um grupo menor de problemas. Todavia, esta característica não significa que suas respostas serão piores que um sistema de maior porte. A qualidade de suas respostas dependerá da representação dos casos e da sua capacidade de interpretar, adaptar, avaliar e criar novas soluções [FRE96].

Cabe ressaltar que os estudos em CBR não abrangem todas as condições que conferem ao homem seu comportamento cognitivo. Desta forma, devem ser consideradas as suposições psicológicas que este paradigma apresenta:

- a memória é predominantemente episódica, sendo seu conteúdo básico representado pela experiência;
- a memória é indexada, podendo uma mesma experiência ser recuperada por índices ou caminhos diferenciados;
- a memória é dinâmica, sendo possível ocorrer alterações na sua estrutura ao longo do tempo; e
- as experiências guiam o raciocínio, uma vez que a interpretação e a compreensão de novas situações se dá sobre experiências já conhecidas.

3.3.1. Casos

Um raciocinador baseado em casos apresenta grande dependência na estrutura e no conteúdo da sua biblioteca de casos (também denominada estrutura de memória ou base de casos). Uma vez que um problema é resolvido através de experiências prévias, os processos de busca por casos conhecidos e de comparação entre casos são fundamentais para o bom

funcionamento do sistema. Outro ponto importante é o aproveitamento do conhecimento de novos casos resolvidos, que devem ser corretamente integrados à estrutura de memória.

Inicialmente, deve-se definir as porções de informação que irão caracterizar um caso, determinando assim uma maneira apropriada para a representação do seu conteúdo bem como da estrutura que definirá o relacionamento entre casos.

Em sistemas tradicionais de Inteligência Artificial, é muito comum encontrar a representação de conhecimento determinada de forma bastante genérica. Isto pode ser justificado porque, além de haver uma economia no armazenamento de informações, pode-se cobrir uma série de problemas com uma mesma informação. Outra característica é a possibilidade de lidar com situações onde não se tem domínio de todos os seus detalhes e, mesmo assim, dar continuidade ao raciocínio. CBR, entretanto, procura dar mais ênfase a conhecimentos específicos associados a situações bem particulares. Desta forma, Janet Kolodner [KOL93] afirma que o conhecimento é representado em um nível operacional, ou seja, torna explícito como uma tarefa foi realizada, como uma porção de conhecimento foi utilizada ou como os métodos para atingir um objetivo foram executados. Além disso, pode-se utilizar informações que não seriam representadas em uma modelagem mais genérica, dando um maior grau de especificidade à resposta.

Os casos podem ser modelados de acordo com as necessidades do sistema que se desenvolve. Quanto ao aspecto temporal, pode-se criar estruturas que representem a evolução do quadro clínico de um paciente ao longo do tempo, pode-se representar informações pontuais como as sentenças de um juiz ou a alteração de comportamento de sensores em um determinado intervalo de tempo. Os casos podem ainda representar episódios de solução de problemas, como a definição de um diagnóstico; podem associar a descrição de um problema a um resultado, como em um sistema jurídico; ou ainda fazer a combinação de todas estas representações. Além desta representação interna, os casos podem ainda estar associados em uma hierarquia ou relacionados através de índices.

A principal característica dos casos não reside exclusivamente nos seus modos de representação, mas também no seu comportamento. Como a idéia principal é o reaproveitamento de conhecimento, os casos devem relatar experiências. Desta forma, quando recuperados para a resolução de novos problemas, forma-se um contexto onde o conteúdo do caso recuperado está presumidamente de acordo com o problema [KOL93]. Quando situações similares são identificadas, não é necessário recomeçar todo o raciocínio; pode-se fazer uso de

informações já experimentadas e validadas que auxiliam ou apontam diretamente para a resolução do problema. A adaptação de um caso antigo para a resolução de um problema ainda não conhecido representa a criação de um novo caso. Desta forma, o conhecimento deste novo caso é analisado e então integrado à estrutura de memória para que possa ser recuperado futuramente, uma vez que indica uma nova experiência para o raciocinador.

Pode-se então resumidamente afirmar que:

- os casos representam conhecimentos específicos relacionados a determinadas situações, sendo desta forma, representados e utilizados de acordo com um contexto;
- não há como definir o melhor modelo de um caso, uma vez que sua representação é diretamente dependente do sistema em que é aplicado;
- casos recuperados da memória ensinam o raciocinador a alcançar seus objetivos através das experiências que relatam sobre os problemas resolvidos e as soluções destes problemas; e
- casos podem ser inseridos na base de casos de acordo com a resolução de novos problemas, refletindo assim novas experiências aprendidas pelo raciocinador.

3.3.2. Ciclo de Funcionamento

De uma forma genérica, como apresentado na Figura 3.2, pode-se resumir o comportamento de um raciocinador baseado em casos em quatro processos principais [BO99]:

- *recuperação*: definir o(s) caso(s) de maior similaridade ao problema proposto;
- *reutilização*: aproveitar informações conhecidas na resolução do problema novo;
- *revisão*: avaliar a nova solução criada a partir de conhecimento prévio; e
- *armazenamento*: reter o conhecimento do novo caso solucionado.

Recuperação

A recuperação de casos na memória está intimamente ligada à lembrança e, em um sistema baseado em casos, é essencial que esta tarefa seja executada de forma eficiente pois dela dependem todos os demais processos de raciocínio. Um caso é recuperado de acordo com as similaridades que apresenta com um problema em questão e, a partir deste caso, serão

criadas adaptações e a solução do problema. Por este motivo, deve-se dar atenção especial a esta tarefa.

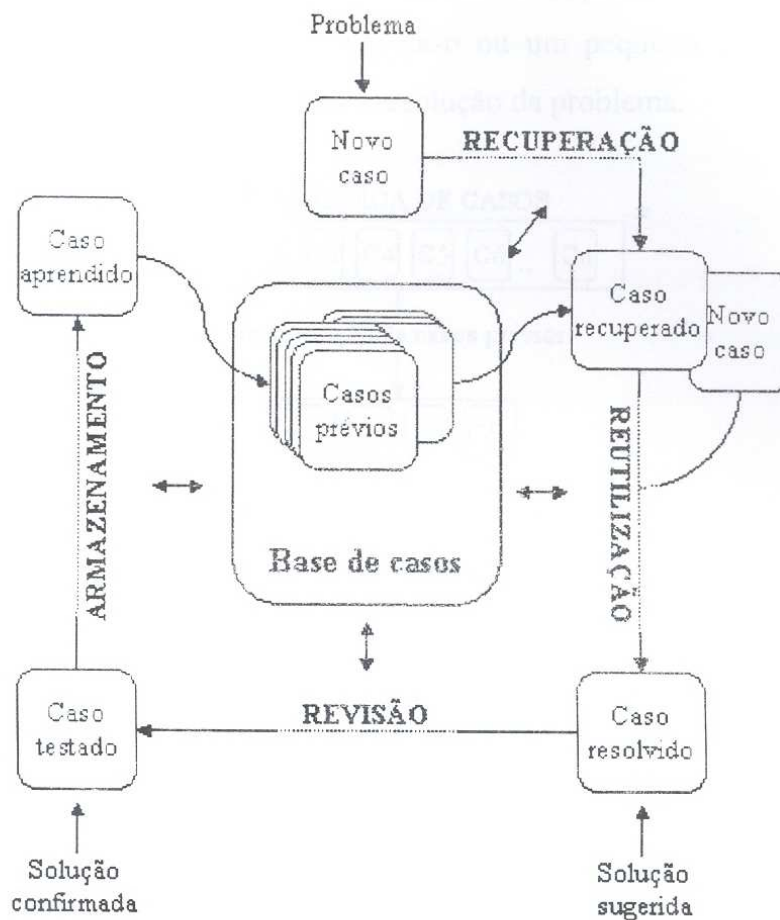


Figura 3.2: Ciclo de funcionamento

Aspectos que influenciam uma boa recuperação de casos referem-se ao tipo de representação de conhecimento dos casos, à estruturação destes na memória, à indexação dos mesmos, ao algoritmo de seleção/comparação e às intenções do sistema. Uma proposta seria executar tarefas como identificação de características, análise de correspondência inicial, busca e seleção [AAM94], ou, como ilustrado na Figura 3.3, poderiam ser executados dois sub-processos mais genéricos [FRE96]:

- *recuperar casos prévios*: a intenção é selecionar um conjunto de casos que apresentem potencialidade de serem aproveitados nas próximas etapas a serem executadas. A similaridade entre estes e o problema a ser resolvido são estipuladas em função de aspectos que caracterizam determinadas situações. Em geral, estes

aspectos são os elementos que compõem as chaves de indexação dos casos armazenados; e

- *selecionar o melhor subconjunto de casos*: a partir da seleção feita no passo anterior, é selecionado um único caso ou um pequeno grupo de casos que se mostram mais promissores para a resolução da problema.

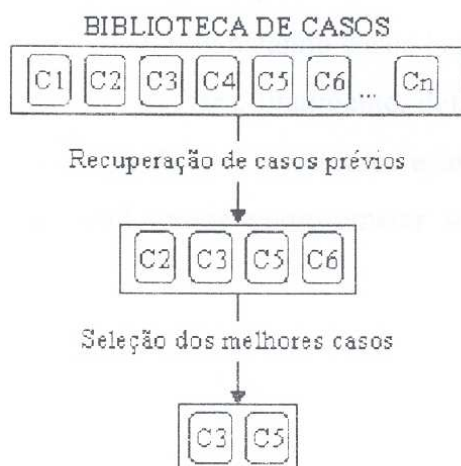


Figura 3.3: Recuperação de casos

Em geral, a recuperação é iniciada com a descrição de um problema e é finalizada quando se define o(s) melhor(es) caso(s) na memória relacionado(s) a este problema. Entretanto, há várias maneiras de se estabelecer equivalências. O primeiro passo seria determinar o número de aspectos que caracterizam os casos e um algoritmo eficiente que analise estes aspectos. Deve-se levar em conta que, para uma base de casos muito extensa, não é vantajoso estipular um grande número de aspectos, optando-se por selecionar aqueles que são realmente relevantes. Se isto não ocorrer, fatalmente surgirão problemas referentes ao tempo gasto na recuperação dos casos. O algoritmo de recuperação, por sua vez, pode representar ainda diferentes estratégias de recuperação. Ele pode ser modelado para recuperar casos em busca de um objetivo ou de um plano, pode pesquisar em múltiplos contextos e pode, ainda, buscar por histórias ou experiências prévias específicas [RIE89]. Em virtude destas características, deve-se conciliar a representação dos casos, a definição dos aspectos que os caracterizam e o método de recuperação.

Para garantir uma recuperação eficiente, em geral os algoritmos utilizam sistemas de indexação¹ que garantem a busca por casos corretos nos momentos apropriados. Os índices ou

¹ Detalhes específicos sobre indexação são apresentados na seção “3.3.3. Indexação” deste Capítulo.

rótulos (*labels*) que são associados aos casos, indicam as condições em que cada caso pode ser utilizado a fim de produzir inferências úteis à resolução de um problema. Programas encaram uma nova situação como uma chave de busca e, comparando-a com os índices dos casos armazenados, define aquele que pode trazer maior ganho de informações. Atualmente, os pesquisadores da área de CBR procuram identificar os tipos de índices que se mostram mais úteis, definindo conjuntos de vocabulários para índices, criando algoritmos e heurísticas para automatizar a escolha de um índice, organizando casos baseados nos índices, realizando a busca em memória utilizando os índices para, finalmente, definir o melhor caso recuperado [KOL93]. Esta tem sido uma das áreas de maior interesse e importância em CBR, uma vez que a recuperação de um caso errado pode comprometer seriamente a credibilidade do sistema.

Reutilização

Após a recuperação de um caso, deve-se analisar suas informações a fim de adaptá-las para a solução do problema atual, uma vez que o caso recuperado e o caso atual dificilmente são idênticos. Em geral, a construção de uma “solução aproximada” fundamenta-se na obtenção da solução do caso prévio (ou em partes da sua solução) e na comparação desta solução com o caso atual, para então serem definidas as partes do caso prévio que podem ser reutilizadas na solução do problema.

Segundo Schank e Riesbeck [RIE89], existem basicamente duas classes principais de adaptação:

- *adaptação por transformação ou adaptação estrutural*: regras de adaptação são aplicadas diretamente sobre a solução do caso recuperado. A solução do caso atual não é idêntica à solução do caso recuperado, entretanto, existe um conhecimento embutido nos operadores ou nas regras de adaptação que transformam a solução do caso prévio em uma solução para o caso atual. Este procedimento não aborda a maneira com que o problema prévio foi resolvido, enfocando desta forma apenas a equivalência de soluções. Percebe-se então que esta resolução é bastante dependente do contexto em que os casos estão inseridos e dos operadores de transformação; e
- *adaptação por derivação*: regras que reproduziram a solução do caso prévio são re-executadas para gerar a solução do caso atual. Para viabilizar este tipo de

adaptação, os casos não armazenam apenas as suas soluções como também os métodos que conduziram a elaboração destas. As diferenças entre os casos são analisadas e a seqüência de planos do caso prévio é atualizada com estas diferenças, garantindo desta forma a solução no novo problema. Em outras palavras, a solução original é adaptada sem sofrer alterações diretas, mas através da re-execução de partes do seu processo de resolução.

Um ponto importante a ser considerado no processo de adaptação são as técnicas utilizadas para avaliar o caso recuperado e o atual e desta forma determinar as características que precisam ser adaptadas. Deve haver heurísticas que definam o caminho a ser seguido para realizar esta tarefa, sempre de acordo com as intenções do sistema.

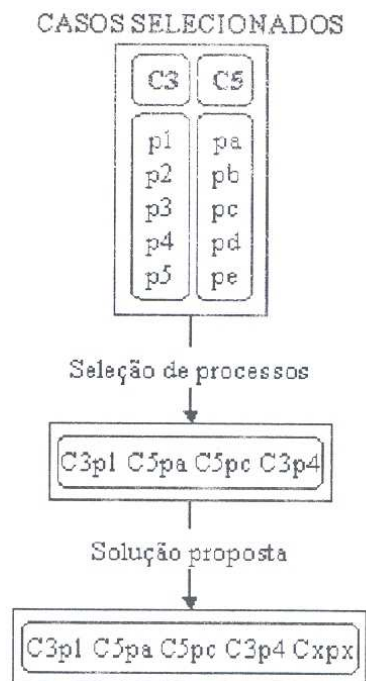


Figura 3.4: Adaptação de casos

Um exemplo de adaptação de casos pode ser visto na Figura 3.4. Supõe-se que os casos recuperados da memória foram C3 e C5, sendo o primeiro composto pelos processos p1, p2, p3, p4 e p5, e o segundo pelos processos pa, pb, pc, pd e pe. Na adaptação, o algoritmo identifica que estes casos representam apenas parcelas da solução e, desta forma, seleciona os processos relevantes de ambos. Para compor a nova solução, o algoritmo efetua a junção dos processos selecionados e gera um novo processo, Cxpx, que não era atendido por C3 e C5 [FRE96].

Revisão

A revisão de casos é um processo que visa validar a solução proposta pelo raciocinador. Inicialmente, o novo caso deve ser avaliado. Se não houver problemas, o aprendizado de uma nova experiência foi feito com sucesso; caso contrário, parte-se para a reparação do caso. Pode-se decompor a revisão em dois sub-processos [AAM94]:

- *avaliação da solução*: quando se possui todo o conhecimento necessário para avaliar a solução proposta, pode-se entender este procedimento como uma ação de validação. Contudo, se esta informação não está disponível, pode-se transpor o caso a uma situação real e analisar se os resultados da solução são satisfatórios. Se isto ocorrer, o caso é tido como correto, caso contrário, deve-se passar para a fase de reparo. Entretanto, se não for possível submeter o caso a uma situação real, pode-se optar pela utilização de um programa de simulação que gere a resposta correta para que esta seja comparada à solução proposta pelo raciocinador; e
- *reparo*: a reparação de casos envolve a detecção de erros e a recuperação ou a criação de explicações que justifiquem a causa dos erros. Uma alternativa interessante é manter na memória as situações que apresentaram falhas, de modo que o sistema possa utilizar estas informações na fase de reutilização para prever, manipular ou evitar possíveis deficiências na adaptação de casos. Em uma outra situação, pode-se construir mecanismos que utilizem a explicação da falha para readaptar a solução de modo que esta falha seja resolvida. Isto pode ser feito com a adição de novos processos à resolução proposta, por exemplo.

De forma resumida, o módulo de revisão possui conhecimentos gerais sobre como neutralizar ou compensar as causas de um erro em um determinado caso. Após a revisão, o caso pode então ser integrado à memória de casos do sistema ou pode ser avaliado e restaurado novamente. Uma ilustração sobre o processo de revisão pode ser visto na Figura 3.5.

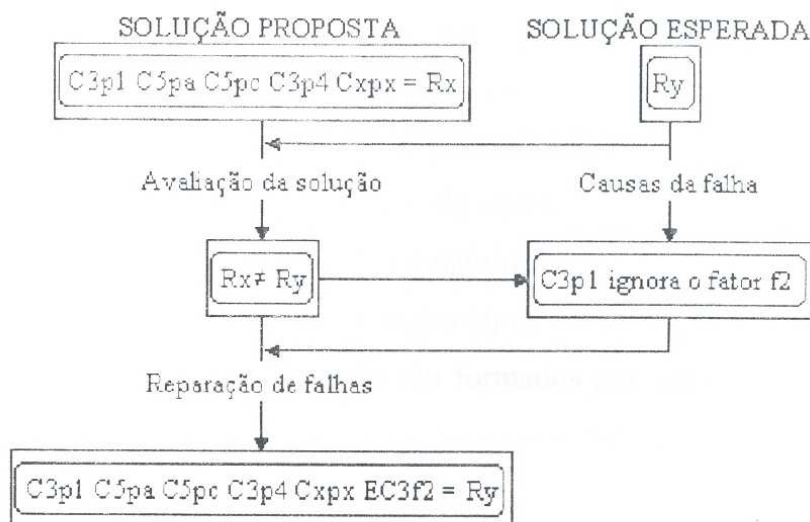


Figura 3.5: Revisão de casos

Armazenamento

Em CBR, a base de casos é atualizada não importando a forma com que um problema foi resolvido [AAM96]. Se a resolução foi obtida com a utilização de um caso prévio, um novo caso pode ser elaborado ou o caso antigo pode ser generalizado de forma a abranger o caso atual. Em outras circunstâncias, um novo caso é construído e posteriormente integrado à memória do sistema. Esta integração, contudo, deve considerar os aspectos relevantes do novo caso que podem ser utilizados para o aprendizado, como características principais do problema e da solução obtida. Além disso, pode-se considerar informações adicionais como justificativas da solução e explicações da sua obtenção, bem como informações sobre possíveis erros que possam ter ocorrido. Um exemplo ilustrativo pode ser visto na Figura 3.6.

No armazenamento, a importância reside principalmente na correta indexação da nova experiência aprendida. Índices muito genéricos podem recuperar o caso indiscriminadamente e uma indexação deficiente pode dificultar o reaproveitamento do caso.

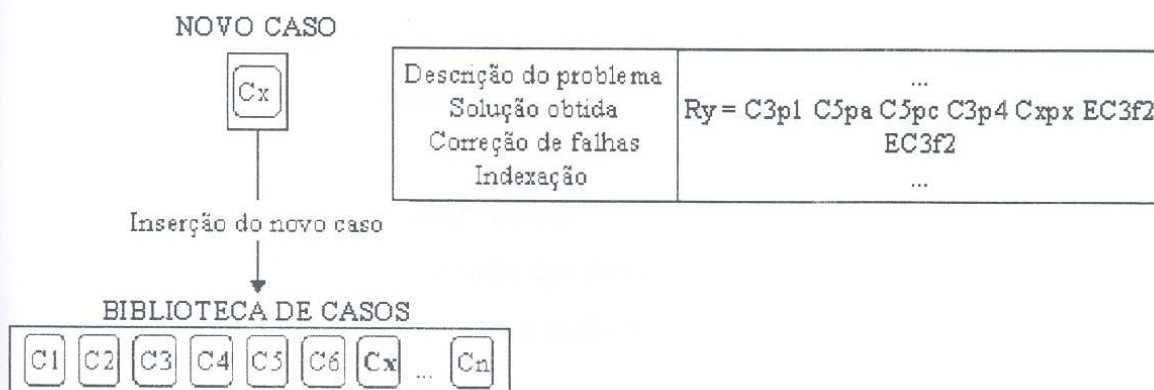


Figura 3.6: Armazenamento de casos

É possível afirmar que CBR é uma metodologia tanto para raciocínio quanto para aprendizado [KOL93], uma vez que utiliza os casos conhecidos na resolução de novos problemas e armazena as soluções obtidas para futuras interpretações. O aprendizado é efetuado gradativamente através da utilização do sistema ao longo do tempo, podendo desta forma aprender novas soluções para posterior reutilização como também aprender falhas que podem ser evitadas em situações futuras. Em princípio, estes sistemas podem aprender porque podem sofrer alterações, uma vez que não são formados por um conjunto restrito de regras [SCH82].

3.3.3. Indexação

A principal característica de sistemas baseados em casos é a recuperação correta de casos da memória. Em CBR, o problema de indexação refere-se à identificação das condições em que casos conhecidos podem ser considerados durante a resolução de um problema [DOM93]. Recuperar o caso certo no momento certo reflete a boa reutilização de casos, como se estivesse sendo “ensinada uma lição” ao raciocinador. Para tanto, deve-se definir índices apropriados para os casos que são inseridos na base, considerando a organização destes — de modo que possam ser recuperados de forma eficiente — além dos critérios adotados no próprio algoritmo de recuperação.

Independente do modelo ou dos termos de indexação estabelecidos, os índices representam a interpretação de uma situação [KOL93]. Entre os vários pontos a se considerar na determinação de índices, salientam-se os seguintes:

- índices devem antecipar os termos a serem utilizados na recuperação de casos;
- a elaboração dos índices deve ser estipulada a partir de conceitos que caracterizam os casos, compondo-se assim um vocabulário de indexação;
- deve haver uma seleção destes conceitos, pois genéricos em demasia causam a recuperação indiscriminada de casos e muito específicos limitam a reutilização dos casos;
- índices devem considerar características que distinguem um caso do outro, possibilitando assim a previsão dos pontos mais importantes de cada caso; e
- com os índices, deve-se prever a razão por que se realiza uma busca.

Podem ser considerados úteis os índices² que selecionam um caso e, de alguma forma, este caso contribui com o processo de raciocínio do sistema. Ou seja, o caso recuperado está de acordo com o andamento do raciocínio e traz novas informações que também podem ser aproveitadas. Estas novas informações podem estar relacionadas à geração da solução do problema, à antecipação, explanação ou restauração de falhas e à avaliação da solução proposta. Por exemplo: um químico trabalha com um conjunto de hipóteses sobre possíveis falhas que possam ocorrer em seus testes de laboratório. Desta forma, os casos devem estar de acordo com este contexto e os índices devem apontar para possíveis falhas associadas às hipóteses.

Para finalizar, na tentativa de viabilizar o maior aproveitamento das experiências prévias, não se restringe a associação de apenas um índice a um determinado caso. Um caso pode referenciar a descrição de diferentes situações de acordo com o contexto em que é analisado e, desta forma, podem ser associados a ele diferentes índices ao mesmo tempo.

3.3.4. Base metodológica

As principais tarefas que todos os sistemas de CBR devem cumprir referem-se basicamente à identificação do problema ou da situação atual, seleção do caso prévio de maior similaridade, utilização do caso na resolução do novo problema, avaliação da proposta elaborada e integração da nova solução (agora novo caso) à memória do sistema. Todavia, os objetivos do sistema invariavelmente afetam a maneira com que estas tarefas são realizadas, de acordo com a metodologia adotada para a resolução dos problemas.

Segundo Agnar Aamodt [AAM94], o paradigma de CBR cobre uma série de diferentes métodos sobre organização, recuperação, utilização e indexação do conhecimento mantido nos casos. Quanto aos casos, pode-se por exemplo:

- representá-los como experiências específicas ou agrupá-los de acordo com suas generalidades;
- podem incluir em sua representação: o problema e sua solução, a descrição de uma situação e um resultado, ocorrência de falhas, explicação causal das falhas ou explicação dos métodos aplicados para derivar a solução;

² Maiores detalhes sobre a definição e aplicação de índices podem ser encontrado nos trabalhos de Janet Kolodner, como [KOL93].

- podem ser armazenados como estruturas de conhecimento independentes ou organizadas em unidades e subunidades distribuídas em uma estrutura de conhecimento;
- podem ser indexados termos de um vocabulário de indexação; e
- podem estar integrados em uma estrutura hierárquica ou plana.

E, quanto à execução do sistema:

- a solução pode estar diretamente associada às informações do caso prévio recuperado ou pode ser composta a partir de alterações deste caso de forma a cobrir aspectos do novo problema;
- a comparação de casos, a adaptação de soluções e o aprendizado podem ser guiados por métodos complexos em um determinado domínio do conhecimento, ou pode-se utilizar métodos mais simples e superficiais como comparação sintática, por exemplo;
- os métodos das diversas fases de raciocínio podem ser totalmente automáticos ou contar com a interação do usuário para suporte e guia de decisões; e
- a biblioteca de casos pode ser única ou distribuída, podendo a recuperação ser avaliada seqüencialmente ou em paralelo.

De acordo com estas características, pode-se modelar diferentes sistemas de CBR, desde que mantenham a idéia principal desta abordagem. Raciocínio Baseado em Casos pode ser visto então como uma terminologia genérica que abrange uma coleção de sistemas que utilizam métodos de solução de problemas baseados em conhecimento prévio.

A literatura conta com uma infinidade de sistemas de CBR desenvolvidos ao longo dos anos. Entre eles, pode-se citar alguns dos mais tradicionais, como: CHEF [HAM89], que adapta receitas de culinária; CASEY [KOT88], que diagnostica problemas de coração; JUDGE [BAI86], que atua no domínio de sentenças criminais; CYRUS [KOL84] e IPP [LEB80], como programas de compreensão de textos; HYPO [ASH87], como um raciocinador na área de patentes; COACH [COLL87], sobre futebol, entre outros. Cada programa salienta uma ou outra fase do raciocínio (recuperação, reutilização, revisão e

armazenamento) ou combina estas técnicas com diferentes modelos de representação de conhecimentos, como MOPs, por exemplo.

Uma linha de pesquisa que tem se mostrado inovadora com a utilização de CBR é a de compreensão de linguagem natural. Atualmente, os processadores de linguagem natural existentes se restringem a realizar tarefas de análises sintática ou semântica de sentenças, bem como o casamento de padrões entre palavras do texto. Sua funcionalidade não é questionada, porém, o que se pode dizer sobre o significado do texto trabalhado? Em geral estes sistemas não se baseiam em uma memória ou associam significado às palavras. Desta forma, Christopher Riesbeck e Roger Schank [RIE89] apresentaram sua proposta: aplicar um sistema de CBR para efetuar a compreensão de sentenças escritas em linguagem natural. Para tanto, uniram técnicas de CBR e Memória Dinâmica e fundamentaram os *Parsers Baseados em Casos*, base para a construção do DMAP — *Direct Memory Access Parsing*.

3.4. Considerações Finais

A aplicação de sistemas baseados em casos tem se mostrado bastante efetiva em diferentes áreas do conhecimento. A busca pela criação de metodologias que representem mais fielmente aspectos da cognição humana permanece uma área de pesquisa bastante atuante em Inteligência Artificial.

No tocante à compreensão de linguagem natural, a aplicação de técnicas de CBR ainda foi pouco explorada. O capítulo seguinte apresenta detalhes da teoria de *Parsers Baseados em Casos*, enfocando em especial os procedimentos específicos do DMAP.

Capítulo 4

Parsers Baseados em Casos

4.1 Considerações Iniciais

Como afirmou Christopher Riesbeck [RIE86], houve uma revolução na área de tratamento de linguagem natural. Esta revolução, porém, não se refere a aspectos sobre análise sintática ou análise semântica de sentenças. Ela enfoca o real conceito de compreensão automática de linguagem natural e o modo com que seu processamento é efetuado.

Proposto por Riesbeck, o DMAP — *Dinamic Memory Access Parsing* — é um *Parser* Baseado em Casos que representa em memória seus conhecimentos sobre linguagem natural. À medida que um texto é lido, os conceitos nele representados são ativados e é efetuada a compreensão do texto. Este mecanismo substitui a maioria dos procedimentos e instruções dos modelos tradicionais de *parser* — que são orientados à palavra — por um processo uniforme de busca em memória guiada por índices associados aos conceitos [MAR90].

Neste Capítulo serão apresentados os fundamentos de *Parsers* Baseados em Casos e, em especial, serão salientados os principais detalhes do DMAP, como a representação e a busca de informações efetuada na memória.

4.2 Compreensão da Linguagem Natural

O tratamento da linguagem natural vem sendo amplamente estudado a fim de melhorar a interação entre usuários e computadores. Neste aspecto, um dos focos principais da pesquisa refere-se ao processo de compreensão da linguagem natural.

A maioria das pessoas desconhece o quanto é elaborada e rica a linguagem humana, entretanto, estas características também a classificam como vaga e ambígua. Desta forma, não se pode ficar desapontado quando o computador falha ao tentar manipular informações

lingüísticas, como ocorre atualmente em sistemas de busca na internet ou na tradução automática de textos, por exemplo. George Miller [MIL01] afirma que ao mesmo tempo em que a comunicação torna-se mais importante, as limitações lingüísticas computacionais tornam-se mais frustrantes. À medida que mais e mais documentos são armazenados em sistemas computacionais, a incapacidade de compreender as informações neles contida restringe sua utilidade em diversas atividades. Entretanto, os computadores não são os culpados por esta situação; a linguagem é o centro do problema.

A primeira questão a ser tratada refere-se ao modo em que se associam significados às palavras. Estas podem ser consideradas as menores unidades de significação de uma linguagem, porém, mesmo independentes, muitas vezes não se restringem a um único sentido ou interpretação. A segunda questão, desta forma, torna-se mais complexa pois se refere à combinação de palavras em uma sentença. Ou seja, além de se considerar os diversos significados individuais, deve-se analisar as alterações de sentido que a combinação de palavras pode causar.

A linguagem pode ainda apresentar informações incompletas daquilo que se pretende transmitir, como no momento em que as pessoas deixam de mencionar dados considerados conhecidos; pode fornecer muita liberdade na utilização de palavras, expressões e significados, como em “xerocar”, “deletar” e na criação de gírias e dialetos; e, finalmente, pode haver inúmeras formas de se dizer a mesma coisa [RIC93].

Dentro deste contexto, inúmeros sistemas computacionais têm se voltado ao tratamento de diferentes características da linguagem natural, como análise sintática, semântica, morfológica, pragmática, entre outras, na tentativa de minimizar os problemas supracitados e viabilizar a processamento automático de sentenças. Basicamente, estes sistemas podem ser compostos por um dicionário léxico (ou *lexicon*) que apresenta o significado das palavras e um conjunto de regras que determinam o seu significado, ou podem possuir processamentos intermediários, que efetuam as análises apresentadas neste parágrafo, controlados por um processo global. Cabe ressaltar que a grande maioria dos sistemas atuais comumente voltam-se ao tratamento do idioma inglês.

Como no exemplo apresentado em [MAR90], para a entrada “*I went hunting this weekend and shot two bucks*”, parte do *lexicon* é composto como representado na Tabela 4.1. As instruções associadas a cada palavra devem conter todos os significados possíveis para uma palavra e o relacionamento destes com o significado de outras palavras. Além disso,

devem indicar como e quando cada interpretação individual deve ser considerada. Ou seja, percebe-se claramente que a definição de um dicionário em geral torna-se bastante extensa e complexa.

Tabela 4.1: Lexicon: Sistemas tradicionais

Lexicon	
Word	Instructions
"shot"	<p>Suggest HUNTING, SPENDING, WHISKEY, ...</p> <p>For HUNTING: If no HUMAN has already been seen then discard HUNTING else make that HUMAN the actor of HUNTING wait for an ANIMAL to be seen if an ANIMAL is seen then make that ANIMAL the object of HUNTING else discard HUNTING</p> <p>For SPENDING: ...</p>
"buck"	<p>Suggest DEER and MONEY; if one is used then discard the other.</p>

O processamento de uma sentença por etapas intermediárias, por sua vez, precisa efetuar uma série de atividades seqüenciais, como [RIC93]:

- *análise morfológica*: efetua a separação dos termos da sentença e análise individual dos mesmos (ex.: “do Mário” seria analisado como “de + o” e “Mário”);
- *análise sintática*: focaliza os relacionamentos válidos entre as palavras (ex.: “menino o vai à loja” seria rejeitado);
- *análise semântica*: as estruturas criadas na análise sintática recebem significado, de modo que palavras isoladas sejam mapeadas para os objetos apropriados na base de conhecimento e sejam criadas estruturas que correspondam ao modo como os significados das palavras isoladas combinam entre si;
- *integração do discurso*: faz-se uma análise da dependência existente entre frases (ex.: o termo “aquilo” pode se referir à palavra “bola” da frase anterior); e
- *análise pragmática*: a estrutura que representa o que foi dito é reinterpretada para determinar o que realmente se quis dizer (ex.: “você sabe que horas são” deve ser interpretada como uma solicitação para que as horas sejam informadas).

De acordo com suas características particulares, estes sistemas têm ainda que resolver um grande problema da linguagem natural que é a ocorrência da ambigüidade. Caracterizado como um problema fundamental a ser resolvido automaticamente, as pessoas estão acostumadas a resolver rotineiramente a ambigüidade mesmo sem perceber que a fazem. Entretanto, na tentativa de reproduzi-la computacionalmente, a resolução torna-se complicada quando se busca criar uma teoria sobre a maneira com que as pessoas fazem isto [MIL01].

Sistemas tradicionais de tratamento de linguagem natural baseiam-se em técnicas estatísticas de combinação de palavras. Por exemplo, baseado em um dicionário léxico, George Miller mostra em seu sistema que para a sentença "*But I have promises to keep and miles to go before I sleep*", a combinação dos possíveis significados dos termos chega ao impressionante número 3616013016000, como representado na Tabela 4.2.

Tabela 4.2: Combinação de significados

Palavra	Significados Individuais Possíveis	Combinação de Significados
but	11	11
I	3	33
have	16	528
promises	7	3696
to	21	77616
keep	17	1319472
and	5	6597360
miles	5	32986800
to	21	692722800
go	29	20088961200
before	10	200889612000
I	3	602668836000
sleep	6	3616013016000

Percebendo a inviabilidade do tratamento de um número tão grande de significados, buscou-se adicionar informações gramaticais que restringissem a combinação dos termos da sentença. Entretanto, mesmo especificando que "*but*" é uma conjunção, "*I*" é um pronome e assim por diante, e seguindo algumas regras gramaticais, a combinação de significados ainda ficou em 9660 possibilidades. Dando continuidade às pesquisas, juntamente com Claudia Leacock e Martin Chodorow [LEA98], Miller passou então a analisar a proximidade das

palavras no texto e obteve resultados mais satisfatórios. Estabelecendo um contexto geral (*topical context*) que avalia um intervalo de 50 palavras dentro de um texto, e um contexto local (*local context*), que avalia um intervalo de 4 palavras, determina-se a quantidade de significados que um termo pode assumir e realizam-se testes de contexto sobre o texto completo. Assim, para uma determinada palavra, fazem-se testes exaustivos sobre sua função sintática e alguns de seus significados mais freqüentes; considerando as demais palavras do contextos geral, local e a combinação destes. Com estes procedimentos, os pesquisadores chegaram a um acerto de 85% para determinadas palavras, como visto na Figura 4.1.

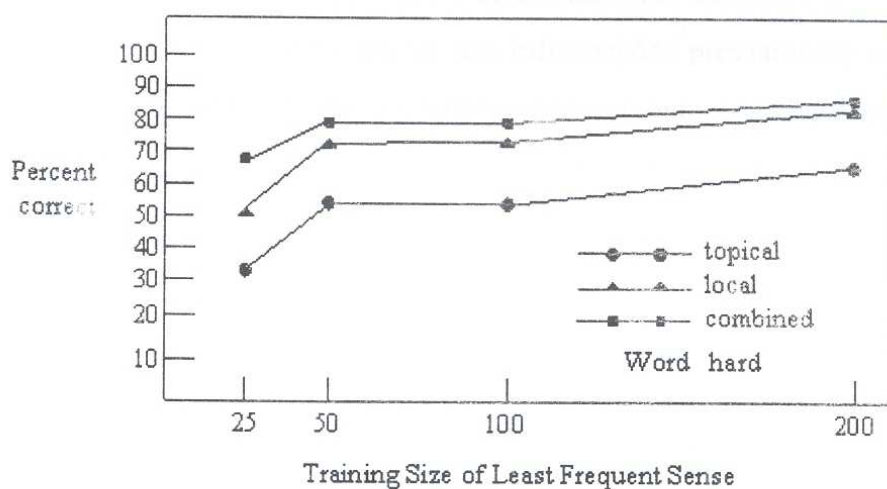


Figura 4.1: Taxa de acerto

Os procedimentos acima demonstrados, entretanto, não se enquadram nos preceitos da Ciência Cognitiva. Desta forma, não se pode afirmar que estão sendo reproduzidos os processos da mente humana como se desejaria. Psicólogos afirmam que quando uma pessoa se depara com uma palavra ambígua, em um primeiro momento mais de um significado pode ser ativado em sua mente. Na seqüência, aquele que for mais apropriado ao contexto é rapidamente escolhido e os demais são desprezados. Assim, pode-se dizer que não estariam sendo considerados apenas aspectos gramaticais; a compreensão em função de um contexto considera a situação representada no texto e um conhecimento geral que a pessoa possui.

As Seções seguintes apresentam conceitos referentes aos *Parsers* Baseados em Casos e à sua teoria que, claramente, reflete com mais fidelidade o comportamento de compreensão e tratamento de ambigüidades apresentado no parágrafo acima.

4.3 Fundamentos de Parsers Baseados em Casos

No final da década de 60, M. Quillian propôs a criação de uma teoria de compreensão de linguagem natural que integrasse um processo de *parser* a uma memória estruturada. Com o desenvolvimento do sistema TLC — *Teachable Language Comprehender*, sua intenção era que o computador lesse e compreendesse o conteúdo de jornais, livros, enciclopédias ou qualquer outro tipo de texto com uma habilidade comparável a dos seres-humanos [QUI69]. O TLC apresentou problemas no tocante à criação e à organização da estrutura memória, entretanto, a definição de compreensão foi uma das maiores contribuições deste trabalho.

A compreensão foi determinada como o relacionamento existente entre as afirmações explícitas ou implícitas de um texto e partes das informações previamente armazenadas no sistema. Ou seja, não há como efetuar a compreensão somente com base nas informações provenientes da entrada; é necessário possuir uma estrutura de armazenamento e recuperação de informações que possibilite a análise e a compreensão do texto. Desta forma, esta estrutura precisa ser previamente modelada de modo que permita a correta compreensão da entrada. Se os textos a serem interpretados relatarem fatos sobre a situação econômica de um país, por exemplo, a estrutura de conhecimento deve ser modelada de acordo com estas informações.

Parsers Baseados em Casos enquadram-se na situação supracitada. Seu objetivo é identificar quais dos conceitos armazenados em memória são mais relevantes a um determinado texto [RIE89]. Essencialmente, estes *parsers* se caracterizam como um processo de reconhecimento e, sendo assim, ressaltam-se dois pontos fundamentais:

- representação da memória; e
- processo de busca em memória.

Para viabilizar a compreensão, a memória do sistema armazena estruturas definidas no mesmo universo de conhecimento do texto de entrada. Ao ler uma palavra, o sistema ativa diversas destas estruturas — ou conceitos — que de alguma forma se relacionam a esta palavra. A partir de então, executando o processo de busca, o sistema identifica qual dos conceitos melhor se associa ao significado do texto.

O *Parser* Baseado em Casos realiza o processo de busca fazendo uso conjugado das palavras lidas na entrada com a representação dos conceitos em memória. Através de um sistema de indexação de conceitos, é criado um esquema de expectativas que conduz a busca

na definição do conceito mais apropriado. Os índices são representados por palavras em linguagem natural e definem a seqüência em que estas devem ser lidas na entrada para que um conceito na memória seja ativado. Se a seqüência de leitura for obedecida, as expectativas são cumpridas e o respectivo conceito é ativado. Desta forma, resolve-se diretamente o problema da ambigüidade, uma vez que a leitura de uma entrada, combinada à definição dos índices, conduz a um único conceito na memória.

É importante frisar que a representação da estrutura que organiza os conceitos e seus relacionamentos na memória, bem como a determinação dos índices, definem o desempenho do processo de reconhecimento. A tarefa de *parsing* é um problema de busca em memória. Para que seja executada corretamente, os conceitos devem ser previamente modelados na memória e estar de acordo com o domínio de conhecimento do sistema.

A Figura 4.2 representa a modelagem de uma porção de memória estruturada através de uma hierarquia de MOPs, seguindo o conceito de Memória Dinâmica. Nela são apresentados diferentes conceitos, o relacionamento entre eles (abstrações e empacotamentos), bem como índices associados a conceitos específicos. Se fosse recebida como entrada a sentença “Milton Friedman fala sobre aumento de juros”, o sistema ativaria os conceitos “Milton_Friedman” e “Evento_de_Comunicação”. Na seqüência, realizaria o refinamento de “Evento_de_Comunicação” para “Comunicação_Monetarista” e em seguida ativaria “Argumento_Monetarista”¹.

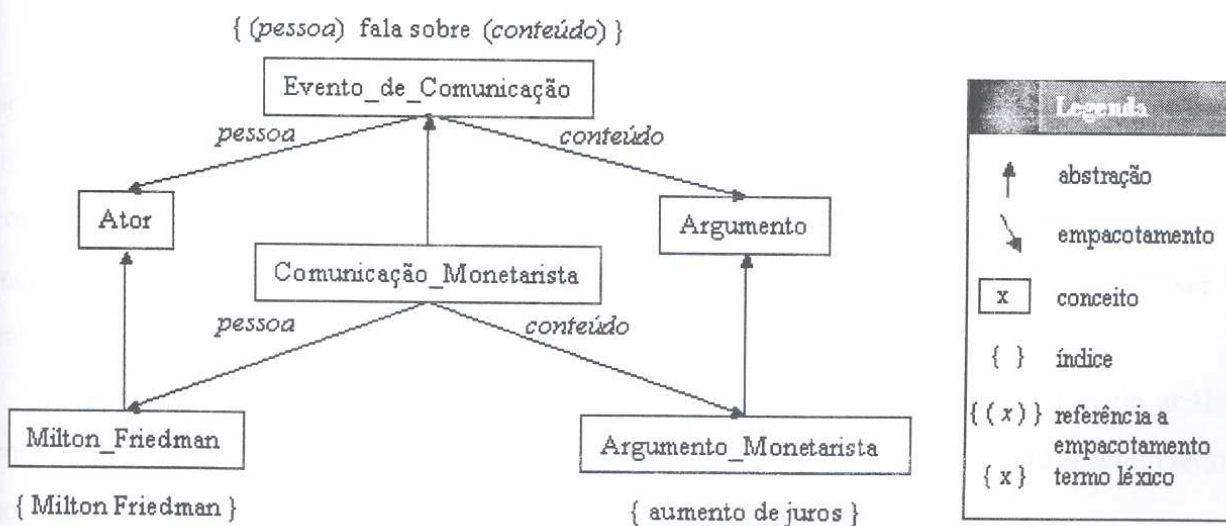


Figura 4.2: Representação de memória e índices

¹ Este processo de compreensão tem como base a busca por informações apresentada na seção “2.7.2. Recuperação de Conhecimento em MOPs”, Capítulo 2.

Uma das motivações para o trabalho com *parsers* baseados em casos deve-se ao fato de que em seu processamento, o tratamento de sentenças pode alterar os estados da memória e modificar o impacto da leitura de textos futuros. Isto ocorre uma vez que, automaticamente, o sistema pode criar novos conceitos na memória ou ativar diferentes conceitos após a leitura de uma sentença. Diferente de outros algoritmos de *parser* que definem claramente uma saída para o sistema como, por exemplo, através de uma sentença; o *parser* baseado em casos focaliza sua atenção à representação da memória. São mais significativas as conexões ativas e as expectativas criadas no decorrer do processo de busca do que a representação propriamente dita dos conceitos. Assim, a saída do sistema é caracterizada como um novo estado de memória.

Sistemas que se fundamentam na utilização de *Parsers Baseados em Casos* verificam o estado da memória após a manipulação de uma sentença. De acordo com as alterações encontradas ou dos conceitos ativados, conclusões sobre a compreensão são auferidas e o sistema dá andamento aos seus processos de raciocínio.

4.4 Direct Memory Access Parsing — DMAP

A criação do DMAP partiu da tentativa de unir as idéias de compreensão de M. Quillian ao conceito de Memória Dinâmica de Roger Schank. Por ser um *Parser Baseado em Casos*, questões sobre a organização, indexação e busca em memória são os pontos mais importantes da sua metodologia.

Muito se pesquisou sobre aspectos de análise ou dependência conceitual, definidas por Schank em [SCH77], mas estas técnicas conduzem à construção de uma descrição sintática intermediária relativa ao significado das sentenças. Através de *Parsers Baseados em Casos*, a compreensão pode ser efetuada sem a necessidade deste processo intermediário, sendo a memória utilizada não apenas para armazenar informações como também para processar as entradas do sistema.

A compreensão baseada em memória e a compreensão realizada através de análise conceitual apresentam uma estrutura semelhante, mas seus procedimentos são diferenciados, como exemplificado na Figura 4.3.

Um problema constante na compreensão de linguagem natural é a ocorrência de ambigüidades, onde uma única palavra pode representar uma grande variedade de significados, conforme apresentado na Seção 4.2. Uma possível solução para este problema é

inserir o texto ou as sentenças de entrada em um contexto bem definido, restringindo assim os significados e relacionamentos que as palavras podem assumir. Desta forma, é fortalecido o princípio de que para se efetivar a compreensão, é necessário identificar o relacionamento dos textos com um conhecimento previamente experimentado.

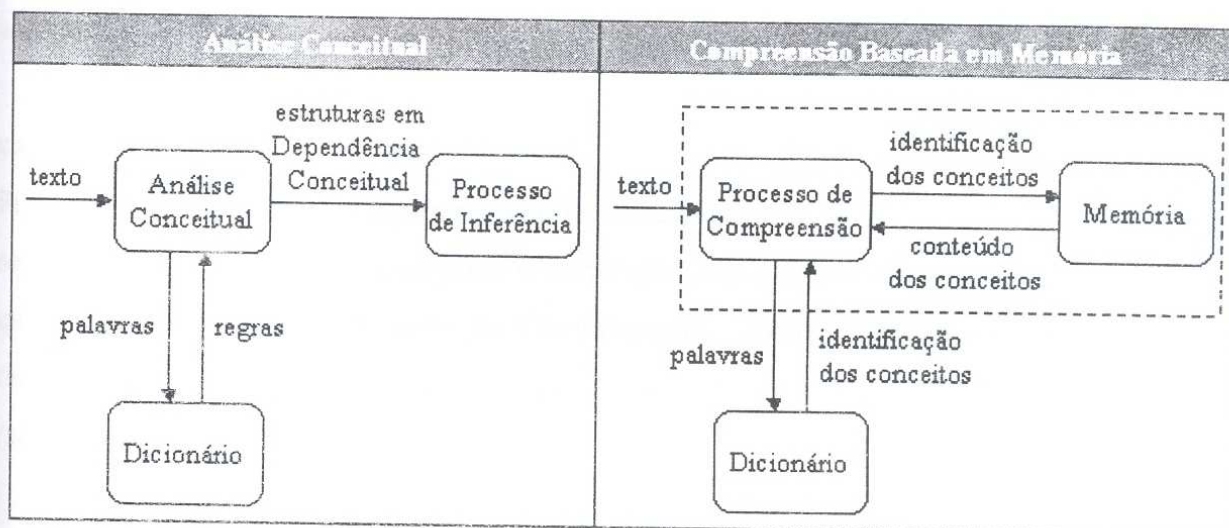


Figura 4.3: Análise conceitual e compreensão baseada em memória

O comportamento do DMAP se caracteriza como um processo de compreensão contínuo, guiado por um conjunto de índices que relaciona o texto de entrada e a memória do sistema. Os índices compõem um dicionário léxico dinâmico que é atualizado de acordo com as expectativas criadas pelo sistema e com o reconhecimento dos termos da entrada. A busca em memória, por sua vez, é realizada através de um mecanismo de busca bi-direcional que conta com a utilização de marcadores em uma estrutura de MOPs hierarquicamente estruturada. Este comportamento será detalhado nas seções a seguir.

4.4.1 Índices

Em sistemas computacionais tradicionais, é muito comum fazer o uso de índices, por exemplo, na busca de um conhecimento específico em uma base de informações muito extensa. Em geral, basta fornecer uma chave que identifique este conhecimento para que a recuperação seja feita de forma direta. Nestes casos, tem-se domínio da modelagem dos índices bem como das chaves que identificam a informação pesquisada.

O DMAP também utiliza índices para recuperar conceitos na memória, entretanto, a única informação disponível é o texto de entrada. Desta forma, questiona-se: "Como modelar um índice sendo que a informação a ser pesquisada não é conhecida de antemão?". Se a

entrada do sistema fosse caracterizada por chaves pré-definidas como no caso do parágrafo anterior, não seria realizada a compreensão da linguagem natural; seria resolvido um problema simples de busca em uma memória indexada. Para evitar esta limitação, a modelagem dos índices do DMAP se baseia em seqüências de termos da linguagem natural. Estas seqüências representam as expectativas do sistema, que podem ter sua interpretação aperfeiçoada ao longo do processo de reconhecimento.

Explicitamente, os índices são apenas padrões de termos, mas implicitamente, consideram também a estrutura de memória [MAR90]. Um exemplo é apresentado na Figura 4.4. De acordo com o processo de reconhecimento da sentença “Milton Friedman fala sobre aumento de juros”, inicialmente o termo (*pessoa*) refere-se ao conceito “Ator” uma vez que o conceito *root* é “Evento_de_Comunicação”. Com o andamento do processo de reconhecimento, o conceito *root* é refinado para “Comunicação_Monetarista”, sendo que (*pessoa*) passa a se referir ao conceito “Milton_Friedman”.

Assim, de acordo com o cenário exposto, ressalta-se que na elaboração de índices compostos por empacotamentos deve-se analisar cuidadosamente a estrutura da memória.

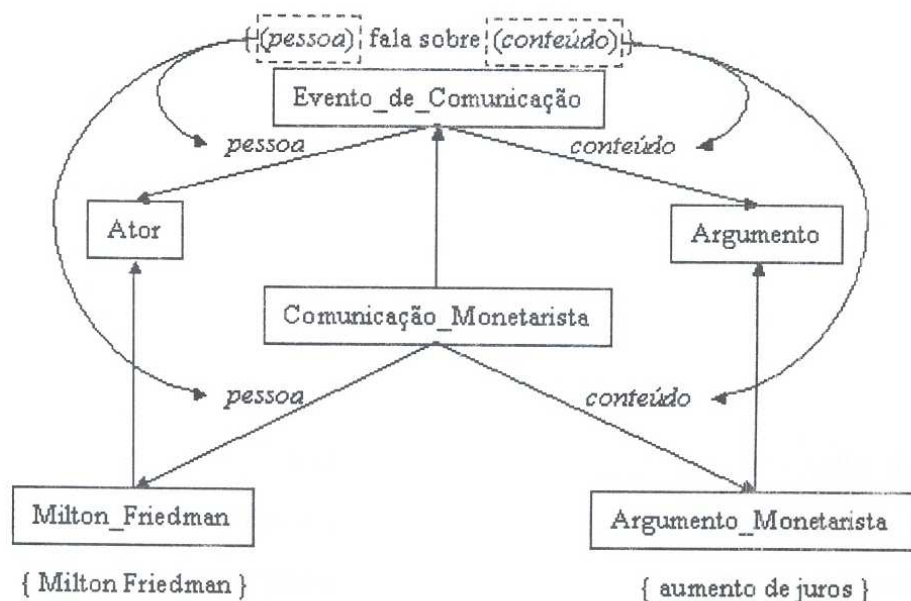


Figura 4.4: Interpretação de Índices

Um ponto importante sobre a análise dos índices combina as expectativas do sistema à atualização e manipulação de um dicionário léxico. Em um primeiro momento, para gerar as expectativas iniciais, todos os índices modelados no sistema têm o seu primeiro termo resolvido. Se este termo for um item léxico ou termo da linguagem natural, insere-se um

marcador de previsão deste termo no dicionário léxico, representando assim uma expectativa ao conceito a que se refere. No exemplo da Figura 4.4, isto ocorre com “Milton” e o conceito “Milton_Friedman”. Caso contrário, se o termo representar um empacotamento, o sistema posiciona um marcador de previsão no conceito a que o empacotamento se refere e utiliza este marcador no processo de busca em memória, detalhado a seguir. Isto ocorre no índice de “Evento_de_Comunicação” com o empacotamento (*pessoa*) e sua referência a “Ator”. Na seqüência, se a leitura de palavra da entrada coincidir com um termo do dicionário léxico, cumpre-se uma expectativa. Assim, o termo é retirado do dicionário e o próximo termo do índice em questão é resolvido. Quando o índice é resolvido por completo, tem-se o reconhecimento de um conceito da memória.

O exemplo anterior ilustra os aspectos característicos de índices:

- podem ser compostos por termos da linguagem natural ou por empacotamentos estabelecidos nos conceitos em que são definidos (os empacotamentos, em algum momento, referem-se a termos da linguagem natural);
- são responsáveis pela criação de expectativas do sistema;
- estabelecem a conexão entre a sentença de entrada e a memória do sistema;
- têm sua modelagem relacionada diretamente à estrutura de memória; e
- direcionam a interpretações mais aprimoradas de acordo com processos de refinamento.

Além disso, a criação de expectativas e previsões é responsável por:

- *criar um dicionário léxico dinâmico*: a atualização constante dos termos do dicionário conferem ao mesmo um comportamento dinâmico; e
- *resolver diretamente problemas de ambigüidade*: o termo “casa”, por exemplo, em um índice “{casa (*pessoa*)}” poderia se referir a um conceito sobre casamento, ao passo que em “{casa (*material*)}” poderia se referir a moradia.

4.4.2 Busca em Memória

Um dos desafios dos sistemas tradicionais de compreensão refere-se à criação de uma estrutura de controle centralizada encarregada de determinar a interpretação de um texto. O DMAP, por sua vez, tem a ausência desta estrutura como uma de suas maiores qualidades. No lugar da centralização, é realizada uma busca em memória com a utilização de um mecanismo de marcadores que se movimentam sobre a estrutura desta memória. Desta forma, são realizadas buscas parciais que podem ou não ser finalizadas de acordo com o conteúdo do texto e com o progresso de outras buscas [MAR90].

Semelhante ao processo de busca efetuado em memória dinâmica, apresentada no Capítulo 2, o DMAP aprimora este procedimento ao utilizar os índices e dois tipos de marcadores — de previsão e referencial — que realizam uma busca bidirecional na árvore de conhecimento. A Figura 4.5 ilustra um caso de estudo² apresentado por Charles Martin em [MAR90] e serve como base para o esquema genérico da compreensão detalhado a seguir:

- *sentença de entrada*: “Milton Friedman says ...”;
- *busca por previsão*: ao criar as expectativas/previsões iniciais do sistema, a partir do conceito *root OBJECT*, a resolução do termo (*actor*) do índice *Mtrans-1* — usado para reconhecer eventos de comunicação — cria um marcador de previsão para o conceito *HUMAN*;
- *busca por previsão*: de forma semelhante, a resolução do primeiro elemento de *Friedman-1* cria um marcador de previsão para o termo *milton*. Entretanto, por ser em um termo léxico, este marcador é inserido no dicionário léxico do sistema. Este índice é utilizado para reconhecer referências a Milton Friedman;
- *busca referencial*: a leitura da palavra “*Milton*” na entrada é uma referência a um item léxico, sendo criado um marcador de referência para este item. O dicionário é então consultado e retorna o marcador de previsão para o índice *Friedman-1*. A interseção dos marcadores de previsão e referencial satisfaz a previsão;
- *busca por previsão*: realiza-se a previsão do segundo termo do índice *Friedman-1*. Como este também resulta em um item léxico, um marcador de previsão para *friedman* é inserido no dicionário léxico;

² Para manter a fidelidade ao esquema apresentado por Charles Martin, detalhes deste exemplo foram mantidos em seu idioma original.

- *busca referencial*: ocorre uma nova interseção de marcadores após a leitura da palavra “Friedman” na entrada;

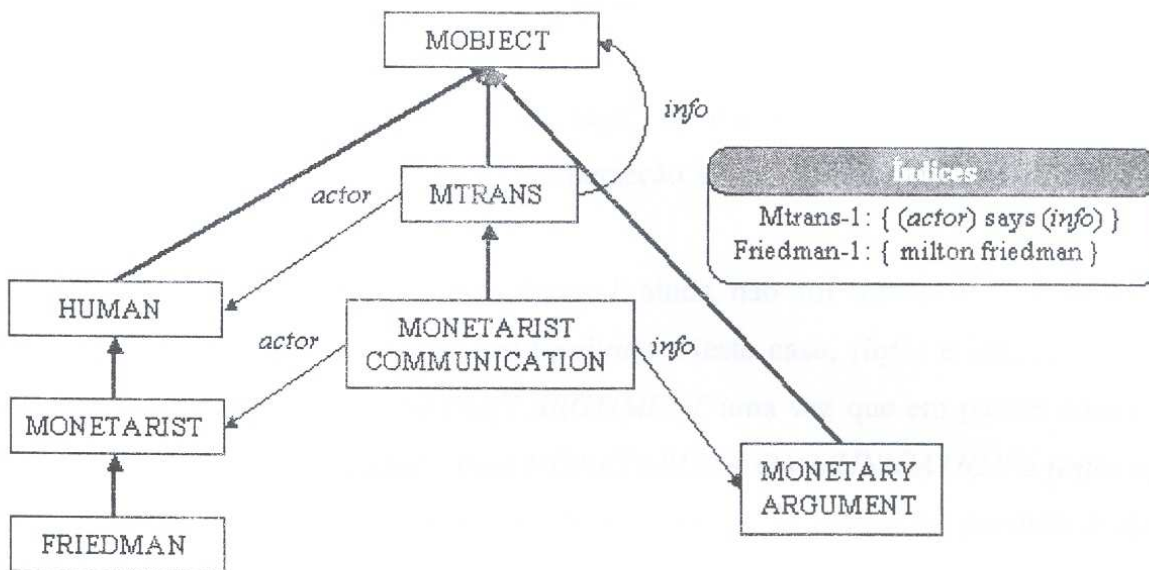


Figura 4.5: Memória e índices: caso de estudo

- *busca referencial*: como todos os termos do índice *Friedman-1* foram trabalhados, o índice é completamente reconhecido. Desta forma, um novo marcador de referência é criado para o conceito *Friedman*. Este marcador é repassado aos conceitos mais genéricos da hierarquia de abstrações, conduzindo o sistema a uma interseção com o marcador de previsão alocado em *HUMAN*;
- *refinamento de conceitos*: diferente das interseções anteriores, esta é mais interessante por ocorrer entre estruturas conceituais e não apenas entre termos léxicos. Neste caso, existe a possibilidade de se obter um maior ganho de informações a partir do conceito referenciado — *FRIEDMAN* — do que com o conceito previsto — *HUMAN*. Para que isto ocorra, é feito o refinamento do conceito raiz de forma que este seja o conceito mais especializado na hierarquia que abstraia o conceito referenciado. De acordo com o exemplo, *MONETARIST COMMUNICATION* é o conceito mais específico abaixo de *MTRANS* que ainda engloba *FRIEDMAN* ou suas abstrações. O conceito raiz para o marcador de previsão de *Mtrans-1* é alterado de *MTRANS* para *MONETARIST COMMUNICATION*³;

³ Alguns detalhes adicionais sobre conseqüências do refinamento de conceitos são apresentados na Seção 4.3.3. Falhas e Aprendizado.

- *busca por previsão*: a compreensão de Milton Friedman resolve o índice *Friedman-1* e o primeiro termo de *Mtrans-1*. Desta forma, a resolução do próximo termo deste índice leva à inclusão de um marcador de previsão referente a *says* no dicionário léxico;
- *busca referencial*: a leitura de “*says*” na entrada gera um novo marcador de referência, causando assim uma interseção léxica com o marcador de previsão de *Mtrans-1*;
- *busca por previsão*: como *Mtrans-1* ainda não foi tratado por completo, seu próximo elemento precisa ser resolvido. Neste caso, *(info)* é interpretado como uma referência a *MONETARY ARGUMENT* uma vez que em passos anteriores o conceito raiz foi refinado para *MONETARIST COMMUNICATION* e *(info)* aponta para um conceito na memória. Desta forma, o marcador de previsão é alocado corretamente e dá-se continuidade à compreensão.

De acordo com o exemplo exposto, reforça-se mais uma vez que a interpretação de uma sentença é realizada dentro de um contexto bem definido. Além disto, a utilização dos índices está presente em todo o processo de reconhecimento, uma vez que compõem os termos do dicionário léxico e, quando completos, efetivam o reconhecimento de um conceito na memória. Neste processo, é fundamental também a modelagem da hierarquia de abstrações dos conceitos e dos empacotamentos estabelecidos. Os dois tipos de marcadores são responsáveis pela identificação das expectativas — com o marcador de previsão — e pela confirmação destas — com o marcador referencial. Através do refinamento de conceitos, a interpretação da sentença é aprimorada e qualquer problema de ambigüidade é resolvido com esta atividade. Uma vez finalizada a completa interpretação de uma sentença, a memória do sistema reflete os conceitos ativados e os possíveis novos conceitos criados, sendo esta memória a representação da saída do sistema.

Um esquema genérico do funcionamento completo do algoritmo de busca do DMAP pode ser visto na Figura 4.6. Nela, “*índice referencial*” refere-se a índices compostos por termos léxicos e “*índice de previsão*” a índices que possuem empacotamentos, como {Milton friedman} e {(actor) says (info)}, respectivamente.

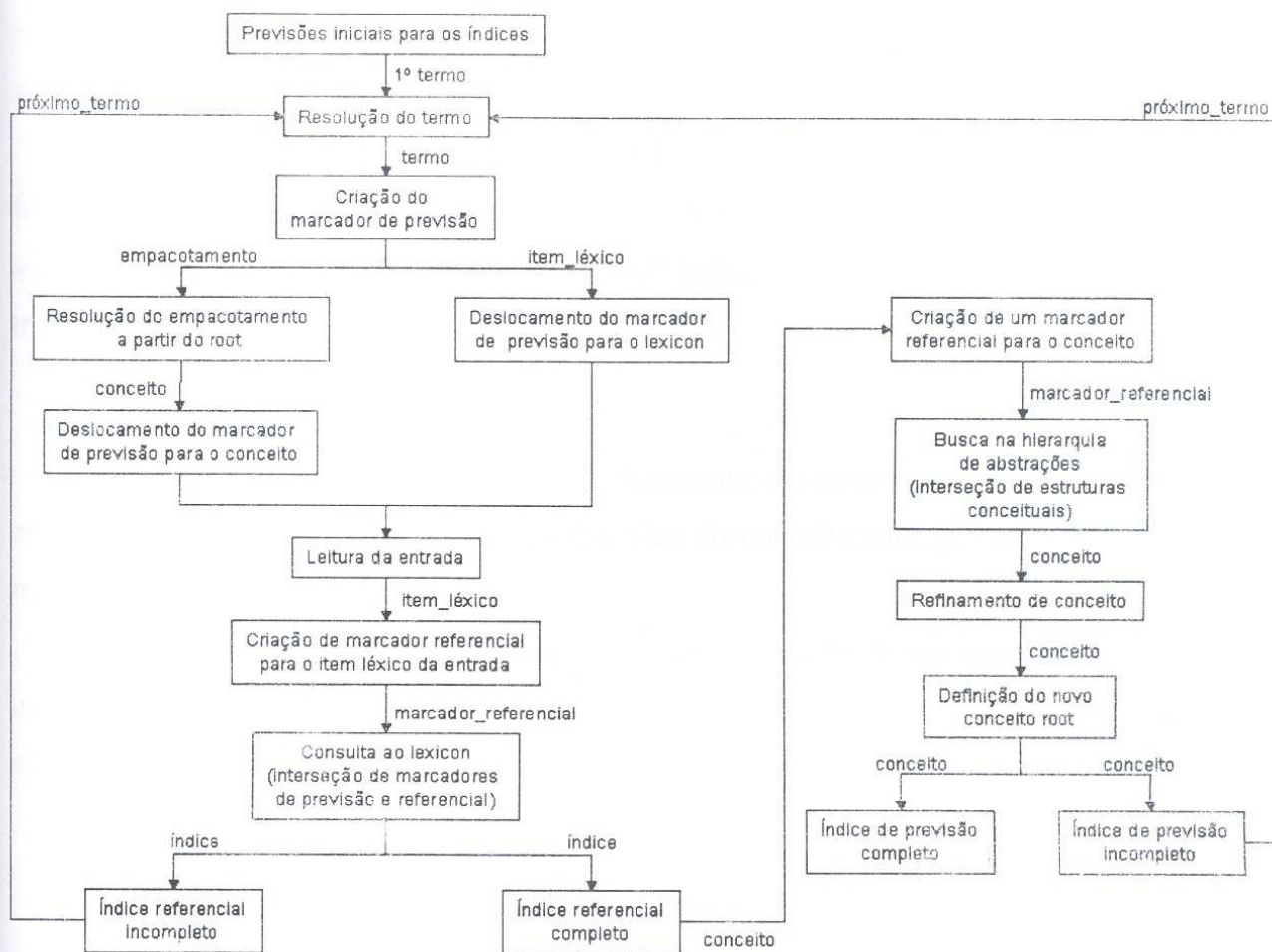


Figura 4.6: Esquema de busca do DMAP

4.4.3 Falhas e Aprendizado

Assume-se que na memória estão modelados os conceitos necessários para interpretar todas as sentenças que podem ser repassadas ao DMAP. Charles Martin [MAR90] relata que a busca por previsão depende da exatidão e especificidade do conceito raiz para resolver os índices referenciados, enquanto o refinamento de conceitos só pode localizar conceitos que já estejam representados na memória. Contudo, não é suficiente apenas reconhecer estruturas na memória. A fim de tratar textos diferenciados, como uma exigência dos problemas reais de compreensão da linguagem natural, o sistema deve possuir a capacidade de modificar a sua própria estrutura de memória e, desta forma, incorporar novos conhecimentos.

Quanto ao aprendizado, Martin afirma ainda que neste aspecto o sistema DMAP é limitado. Não há, por exemplo, teorias relacionadas à criação de novos índices, não sendo realizado desta forma o aprendizado de uma linguagem. Além disso, os novos conceitos criados podem ser definidos apenas como especializações de conceitos previamente

modelados na memória e não há o estabelecimento de novos empacotamentos entre os mesmos. Extensões como estas caracterizam futuras linhas de pesquisa sobre o DMAP.

Atualmente, todo o aprendizado no DMAP acontece em decorrência de eventuais falhas do processo de busca em memória. As duas formas em que este processo pode falhar são nomeadas falha de especialização (*specialization/specification failure*) e falha de interseção (*intersection failure*).

Falha de Especialização

Identificada quando não ocorre o refinamento de conceitos, ou seja, quando não é encontrada na memória uma instância específica do conceito raiz que abstraia um conceito referenciado.

A resolução desta falha pode ser feita com a inserção de um novo conceito que seja uma especificação da “melhor especificação” encontrada na memória. A partir deste novo conceito, cria-se então um empacotamento ao conceito referenciado.

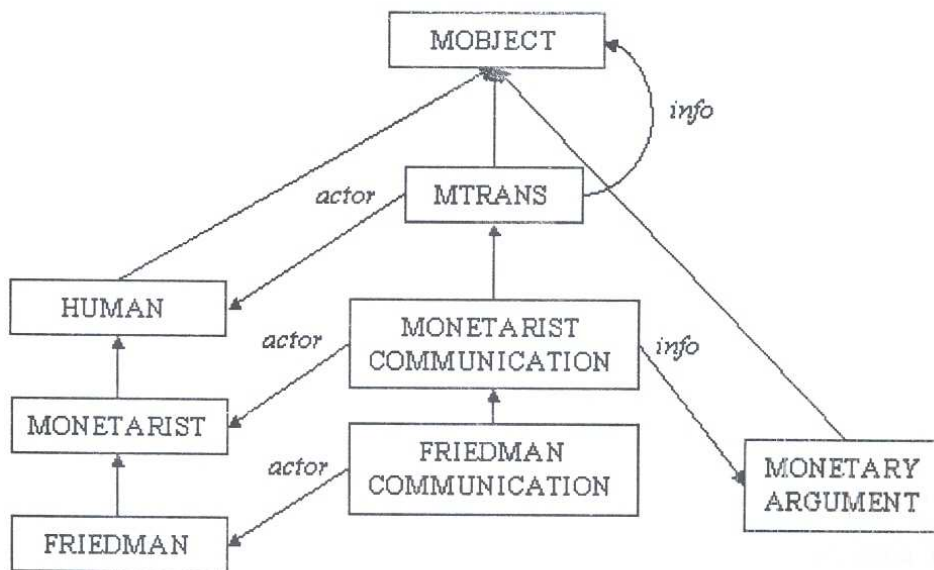


Figura 4.7: Falha de especialização

Na Figura 4.7, por exemplo, esta falha ocorre uma vez que o marcador de referência em *FRIEDMAN* possui mais informação que o conceito *MONETARIST*. Desta forma, o sistema cria um novo conceito, *FRIEDMAN COMMUNICATION*, como uma especialização de *MONETARIST COMMUNICATION* e estabelece um empacotamento *actor* apontando para *FRIEDMAN*. Se após a seqüência da leitura da entrada fosse reconhecida uma instância de *MONETARY ARGUMENT*, um novo empacotamento *info* seria criado a partir de *FRIEDMAN*

COMMUNICATION agrupando assim as informações referentes ao novo conceito da memória.

Uma vez inseridos na memória, os novos conceitos são interpretados sem qualquer distinção em relação aos que já estavam previamente modelados. Desta forma, pode-se afirmar que o sistema aprendeu e incorporou em sua base um novo conhecimento.

Falha de Interseção

Ocorre quando a interseção entre um marcador de referência (R) e um marcador de previsão (P) não caracteriza o relacionamento esperado entre ambos, ou seja, a referência não é uma especialização da previsão. Nestes casos, em geral, a interseção ocorre em um conceito que abstrai ambos os marcadores, como representado na Figura 4.8.

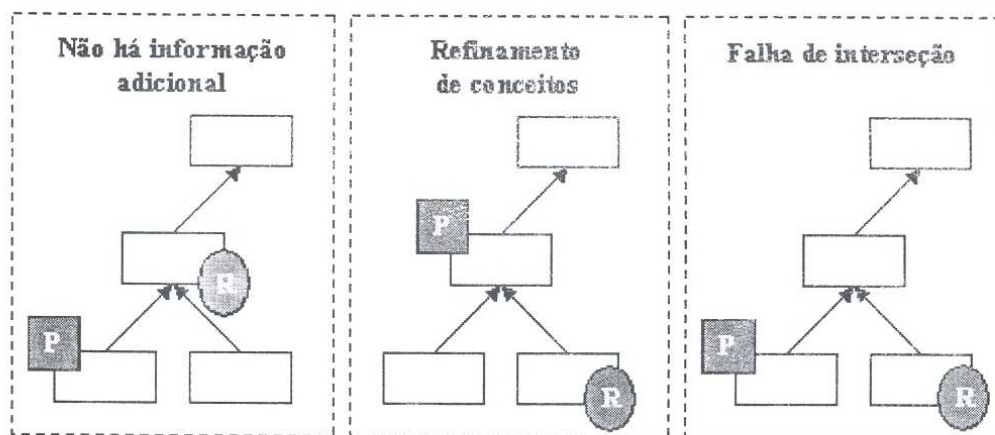


Figura 4.8: Tipos de interseção de marcadores

Para realizar a correção da falha, o sistema precisa identificar o que foi lido na entrada, o que era esperado pelo sistema e o contexto em que a falha se insere. Ou seja, as falhas precisam ser identificadas, representadas e explicadas para que o processamento continue. A explicação da falha é uma extensão natural do processo de interpretação, uma vez que esta informação é mantida na memória com a criação ou a ativação de um novo conceito. Este conceito, por sua vez, faz parte de uma sub-árvore do conhecimento que reflete o tratamento de falhas. Assim, estas estruturas são reconhecidas e manipuladas da mesma forma que os demais conceitos na memória, sendo o processo padrão de busca e construção capaz de procurar e implementar reparos relevantes [RIE89].

4.5 Considerações Finais

Englobando conceitos de Raciocínio Baseado em Casos e, desta forma, considerando capacidades cognitivas humanas, a aplicação de *Parsers* Baseados em Casos tem se mostrado bastante promissora na compreensão de linguagem natural. Com a utilização do DMAP, ao invés de sistemas recorrerem à representação de uma infinidade de regras na tentativa de analisar sintática e semanticamente uma sentença, a utilização de casos na memória viabiliza a compreensão direta destas sentenças. Além deste tratamento sem processos intermediários, existe ainda a possibilidade de haver aprendizado no sistema que pode, futuramente, ser utilizado na interpretação de novas sentenças.

A fim de facilitar a interação entre usuários e computadores, o Capítulo seguinte apresenta o sistema ComPor como o resultado da aplicação das teorias apresentadas até o momento neste trabalho.

Capítulo 5

Sistema ComPor

5.1. Considerações Iniciais

A proposta do sistema ComPor visa demonstrar a aplicação de *Parsers* Baseados em Casos na compreensão de sentenças em linguagem natural bem como na condução de um diálogo entre usuários e computadores. O fato de compreender uma sentença, e não de efetuar a mera comparação de palavras e a execução de regras pré-estabelecidas, permite ao mesmo tempo a compreensão e a condução de um diálogo controlado.

O sistema ComPor é a reunião de todos os conceitos previamente apresentados neste trabalho. Engloba a cognição no sentido de efetuar a compreensão da linguagem natural, reproduzindo processos da mente humana; baseia-se nos princípios de Raciocínio Baseado em Casos uma vez que faz uso de informações previamente conhecidas para resolver novos problemas; processa as sentenças de acordo com os fundamentos de *Parsers* Baseados em Casos; e faz uso conjugado destes *parsers* com uma estrutura de Memória Dinâmica, salientando as abordagens do DMAP e dos MOPs.

Este Capítulo apresenta os detalhes do sistema ComPor, partindo da estruturação de uma memória que viabilize um diálogo até a sua execução propriamente dita.

5.2. Diálogo

A iniciativa de criar sistemas computacionais capazes de dialogar com seus usuários não é novidade na área de Inteligência Artificial. Na década de 60, Joseph Weisenbaum criou o ELIZA [WEI66], um sistema capaz de simular uma conversa em inglês entre um paciente e seu psicoterapeuta. O sistema era capaz de responder várias frases razoavelmente coerentes com as frases de entrada do usuário, raciocinando sobre elas para posteriormente apresentar a

sua resposta. Na época, as características marcantes foram o aparente domínio da linguagem natural e a compreensão das sentenças.

Entretanto, o próprio Weisenbaum explica que tudo isto é mera ilusão. O programa burla a difícil tarefa de compreensão, reconhecimento e transformação de uma frase [WEI66]. Através do casamento de padrões, como representado na Figura 5.1, repetem-se palavras-chave da sentença e desta forma compõe-se uma resposta inteligível.

<p>Padrão: (I need a X) Resposta: (What would it mean to you if you got a X?)</p> <p>Entrada: (I need a vacation) Transformação: (What would it mean to you if you got a vacation?)</p>	<p>Casamento de Padrões:</p> <ul style="list-style-type: none"> - I com I - need com need - a com a - X com vacation
---	---

Figura 5.1: Casamento de padrões no ELIZA

Atualmente, os sistemas ainda apresentam dificuldades técnicas referentes ao reconhecimento e à compreensão de termos; à criação de uma resposta razoável e natural; e ao gerenciamento de uma estratégia de diálogo flexível [GLA01]. A grande maioria dos sistemas é composta por uma gramática, que através de regras combina sintática e semanticamente a palavras de uma sentença, bem como por um gerenciador de diálogo, que analisa as entradas e saídas do sistema na tentativa de identificar as intenções do usuário. A modelagem do diálogo pode restringir a entrada do usuário, como em sistemas de pergunta e resposta (*Question-Answering Dialogs*) [CLA99] [SIM69], pode possuir interfaces mais genéricas voltadas à conversação (*Conversational Interfaces*) [KÖL99] e ainda podem ser aplicadas a sistemas de reconhecimento de voz (*Spoken Language Dialog Systems*) [GOR02]. Quanto maior a abrangência do sistema, ou seja, quanto maior a quantidade de sentenças a serem compreendidas e a variação da ocorrência destas sentenças, mais complicada é a modelagem da memória e do gerenciador de diálogo.

O sistema ideal deveria permitir que usuário perguntasse o que quer da forma que preferir [ALL01], contudo, a criação de um sistema com estas características ainda é um desafio a ser solucionado. As pesquisas voltam-se então a aplicações específicas, como reserva de vôos [SEN00], onde se tem um domínio maior sobre os diálogos que podem ser conduzidos.

5.3. Diálogo no Sistema ComPor

O sistema ComPor é caracterizado como um sistema computacional que viabiliza a criação, execução e o acompanhamento de diálogos efetuados entre homens e computadores em linguagem natural. Fundamentado em uma base de conhecimento modelada pelo próprio usuário, permite, através de uma interface amigável, o desenvolvimento e o estudo de diferentes esquemas de diálogos voltados ao idioma português.

Como características principais, o sistema não impõe qualquer limitação sobre o tema abordado na base de conhecimento ou sobre a criação de modelos específicos de diálogo, permitindo ainda diferentes combinações de ambos na execução propriamente dita de um diálogo. Ao contrário dos sistemas tradicionais, não faz uso de uma gramática para efetivar a compreensão das sentenças e também não possuiu um gerenciador central que analisa as entradas do usuário para então conduzir as respostas do sistema. Tendo o DMAP como núcleo, possui uma única fonte de conhecimento que é a sua memória, como ilustrado na Figura 5.2.

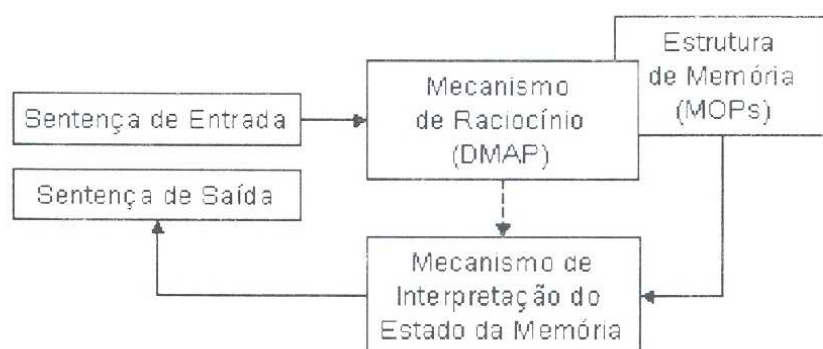


Figura 5.2: Arquitetura genérica do sistema ComPor

Desta forma, o sistema viabiliza a estruturação de uma memória própria para a execução de um diálogo, englobando não apenas as informações que podem ser requisitadas pelo usuário como também informações referentes à condução das sentenças trocadas entre ele e o sistema. Além disso, o ComPor efetiva ainda um aprendizado, utilizando mais uma vez a memória para manter o registro dos diálogos de maneira que estes possam ser reaproveitados em interações futuras ou servir como base para processamentos posteriores.

Como visto, a memória é considerada um dos pontos centrais deste trabalho, devendo-se desta forma salientar alguns aspectos relevantes à sua estruturação, principalmente no tocante à modelagem dos conceitos e das respostas do sistema.

5.3.1. Modelagem dos Conceitos

Como citado anteriormente, a memória do sistema armazena os conhecimentos sobre os diálogos a serem efetuados, relacionando assim informações gerais sobre o assunto do diálogo e informações que conduzam a interação com o usuário de forma coerente. Organizados em uma estrutura hierárquica de MOPs, estas informações são representadas sob o formato de conceitos que podem se relacionar através de relações hierárquicas ou empacotamentos. Na modelagem dos conceitos, é fundamental a análise destes relacionamentos bem como a criação dos índices de conceitos específicos.

Uma vez definida a área de aplicação de um diálogo, como um sistema de vendas em uma livraria, por exemplo; os conceitos da memória devem ser estruturados inicialmente de forma a representar informações sobre:

- *os possíveis objetos de diálogo*: livros comercializados e clientes da livraria;
- *ações/intenções do usuário ou fases do diálogo*: determinação de um livro a ser comprado, definição do tipo de pagamento ou fornecimento da identificação do cliente;
- *condução ou plano de diálogo*: a seqüência de ações que concretizam a compra de um livro.

Em geral, as ações dos usuários devem se relacionar aos objetos do diálogo, e a execução de uma seqüência de ações determina um plano de diálogo. Por exemplo: uma compra deve se relacionar a um livro específico e esta compra, seguida da sua confirmação e da identificação de um usuário, caracteriza um plano de compra definido como “Modelo1-L”.

Neste contexto, como representado na Figura 5.3, o relacionamento entre objetos, ações e planos é estabelecido da seguinte maneira:

- os conceitos referentes às ações possuem empacotamentos aos objetos; e
- os conceitos referentes aos planos possuem empacotamentos às ações.

- sua composição pode ser feita de:
 - *termos léxicos*: em geral representando as intenções do usuário, como em “Quero confirmar a compra”, “Cancelo o pedido” e “Desejo comprar um livro”; bem como na identificação individual dos objetos, como por exemplo o livro “Engines for Education” ou um cliente identificado pelo seu nome; e
 - *empacotamentos*: em geral representando os objetos a que as ações se referem, como em “{Quero comprar o livro (*objeto*)}”, ou um plano de diálogo, como na seqüência “{(L-total), (*Confirmação da Compra*), (*Efetuar Compra*)}” referente ao plano “Modelo1-L”.
- ao se referirem a conceitos que representam planos, devem ser compostos apenas por empacotamentos que apontam para conceitos que representam ações.

A Figura 5.4¹ ilustra parte de uma estrutura de memória modelada para compreender e conduzir diálogos referentes às vendas de uma livraria. A entrada da sentença “*Quero comprar o livro Engines for Education*” ativa os conceitos “M-Engines” e “M-L-Total”. Entretanto, “M-L-Total” refere-se à compra de um livro qualquer, sendo então criado o novo conceito “M-L-Total-0”, que relaciona a ação de compra ao livro específico “Engines for Education”. Na seqüência do diálogo, o usuário confirma a compra do livro, sendo identificado o conceito “M-Confirma Compra”. Tendo a confirmação, solicita-se que o usuário forneça o seu nome para que a compra seja efetuada. A partir de então, identifica-se na memória o conceito que se relaciona ao usuário e cria-se um novo conceito — “M-Efetua Compra-1” — ligando a realização da compra ao usuário em questão.

De acordo com o cenário exposto, é executado um plano de diálogo que modela a compra de um livro por um cliente da livraria. Este plano, representado no conceito “M-Modelo1-L” serve como base para a criação de “M-Modelo1-L-2” que relaciona os conceitos específicos sobre a compra efetuada, ou seja, “M-L-Total-0” e “M-Efetua Compra-1”. Assim, a partir de “M-Modelo1-L-2”, agrupam-se as informações fornecidas no diálogo e pode-se obter dados sobre o livro que foi comprado e sobre cliente que efetuou a compra.

¹ Nesta Figura estão representados apenas os empacotamentos utilizados no plano de diálogo sobre a compra de um livro específico.

conceito da seqüência. Desta forma, garante-se a conclusão de um plano de diálogo previamente estabelecido, satisfazendo as necessidades do usuário e registrando em memória a possível criação de novos conceitos.

Entretanto, apenas pelos empacotamentos definidos nos planos, não se pode garantir a correta seqüência de ativação dos conceitos. Para resolver este problema, na modelagem da memória do sistema, deve-se integrar a definição dos empacotamentos à modelagem dos índices dos conceitos. De acordo com o DMAP, o reconhecimento de um conceito relaciona-se à igualdade dos termos definidos nos seus índices e na sentença de entrada do sistema. Assim, os índices dos planos devem ser compostos por empacotamentos que se relacionam a conceitos que devem ser reconhecidos em uma ordem específica.

Além disso, os índices devem ainda ser considerados no momento da definição das respostas do sistema uma vez que eles representam a entrada do usuário. Ou seja, sabendo a entrada, pode-se facilmente modelar respostas coerentes, evitando desta forma a apresentação de informações desconexas que desviem do diálogo em andamento.

As respostas associadas aos conceitos são representadas por sentenças em linguagem natural ou pela composição destas com valores extraídos dos próprios conceitos. Uma resposta viável à pergunta "*Quero comprar um livro*" poderia ser "*Por favor, informe o nome do livro desejado*". Este seria o modelo mais simples de resposta, composto por uma sentença simples. Entretanto, para a pergunta "*Quero comprar o livro Engines for Education*", a resposta poderia ser "*Este livro custa R\$70,00 e o seu autor é Roger Schank. Confirma a compra do livro?*".

Neste último caso, esta seqüência de pergunta-resposta está associada ao conceito "L-Total" ilustrado na Figura 5.4. Após a compreensão da pergunta e da criação do conceito "L-Total-1", tem-se acesso às informações do livro Engines for Education e pode-se, desta forma, utilizar alguns de seus dados particulares para compor a resposta do sistema. Com a intenção de dar continuidade ao plano de diálogo, apresenta-se ainda um questionamento que conduz o usuário à confirmação ou ao cancelamento da compra do livro, na tentativa de conduzir o diálogo a dois planos previamente modelados.

Em resumo, estas são as características da estruturação das respostas do sistema:

- são representadas diretamente nos conceitos da memória;

- são modelados em função dos índices dos conceitos e dos planos de diálogo previamente modelados;
- são determinadas como sentenças simples, redigidas em linguagem natural, ou como sentenças combinadas a valores extraídos dos conceitos da memória; e
- devem induzir as entradas do usuário de forma a seguir os planos de diálogo.

Tabela 5.1: Esquema de organização de planos

Entrada do usuário ou Índice	Conceito ativado	Resposta do sistema	Seqüência do plano
Quero comprar um livro. Gostaria de comprar um livro.	M-Infôrmações dos Livros	Por favor, informe o nome do livro desejado	---
Quero comprar o livro {objeto}. Gostaria de comprar o livro {objeto}.	M-L-Total	O preço do livro é {objeto preço}, o autor é {objeto autor} e a editora é {objeto editora}. Confirma a compra do livro?	M-Confirma Compra, M-Cancela Compra
Quero saber o preço do livro {objeto}. Gostaria de saber o preço do livro {objeto}.	M-L-Preço	O preço do livro é {objeto preço}. Confirma a compra do livro?	M-Confirma Compra, M-Cancela Compra
Confirmo a compra.	M-Confirma Compra	Informe seu nome para efetuarmos a compra.	M-Efetua Compra
Meu nome é {cliente}. Me chamo {cliente}.	M-Efetua Compra	---	---
M-L-Total M-Confirma Compra M-Efetua Compra	M-Modelo1-L	Sua compra foi efetuada. Deseja outras informações?	---
M-L-Total M-Cancela Compra	M-Modelo2-L	Sua compra foi cancelada. Deseja outras informações?	---
M-L-Preço M-Confirma Compra M-Efetua Compra	M-Modelo3-L	Sua compra foi efetuada. Deseja outras informações?	---

A Tabela 5.1 apresenta uma esquematização da estrutura de conceitos, índices e respostas do sistema, visando a execução de planos de diálogo. Da mesma forma que a entrada pode ser caracterizada por frases diferenciadas, a ativação de um conceito pode dar seqüência a diferentes planos, dependendo da próxima entrada do usuário. Deve-se ressaltar que a coluna “Seqüência do Plano” é meramente ilustrativa, sendo a expectativa da entrada gerenciada internamente pelos índices dos conceitos que representam planos de diálogo.

5.3.3. Aprendizado na Execução de um Diálogo

Como apresentado anteriormente, o DMAP realiza um aprendizado no momento em que ocorre uma falha de especialização; e, por ter esta abordagem como núcleo, o sistema ComPor apresenta o mesmo comportamento.

Neste caso, entretanto, busca-se realizar este aprendizado em função das entradas do usuário e das ações modeladas na memória. Ou seja, uma ação pode estar modelada para efetuar a compra de um livro qualquer, mas como a entrada “*Quero comprar o livro Engines for Education*” causa a criação do conceito que relaciona a ação de compra ao livro Engines for Education, pode-se dizer que o sistema aprendeu a fazer a compra deste livro. Se um outro cliente acessar o sistema e efetuar a mesma compra, este novo conceito integrado à memória é ativado, havendo desta forma o reaproveitamento direto das informações da memória evitando a duplicidade de informações.

Da mesma forma, quando os planos de diálogo são efetuados com sucesso, uma nova instância do plano é criada e as informações específicas do diálogo executado são agrupadas nesta instância. Supondo que o sistema foi acessado no período de um mês e registrou diversos diálogos com sucesso, pode-se acessar as novas instâncias criadas e fazer uma avaliação das informações ali agrupadas. Ou seja, de acordo com os planos estabelecidos, pode-se extrair da memória diferentes tipos de informação e efetuar análises diferenciadas, como o volume de vendas no período, o número de acesso dos clientes, o balanço de vendas feitas pelo computador, entre outras. Atualmente o sistema não efetua este tipo de análise, entretanto, armazena estas informações na memória para possíveis processamentos futuros.

5.3.4. Validação do Sistema

De acordo com seus objetivos e com características das abordagens do DMAP e de Raciocínio Baseado em Casos, a validação do sistema ComPor pode ser analisada sob dois aspectos:

- compreensão da linguagem natural; e
- execução de um diálogo.

Em geral, sistemas tradicionais de processamento de linguagem natural possuem dados estatísticos que consideram a taxa de acerto do sistema e garantem assim sua validação através da análise destes dados. O ComPor, em contrapartida, por não seguir a mesma abordagem, pode ser validado em função dos conceitos que reconhece na memória. Ou seja, se as sentenças de entrada coincidirem com as expectativas do sistema e os conceitos corretos

forem ativados/criados na memória, pode-se dizer que o sistema compreendeu a sentença e efetuou corretamente seus processos de reconhecimento.

A avaliação de sistemas de diálogo pode ser um processo complexo de acordo com o aspecto que se analisa [CHU97]. Podem ser consideradas algumas medidas subjetivas, como utilidade, naturalidade, clareza, robusteza no tratamento de falhas e quantidade de sentenças trocadas [GIA96]. O sistema *Mercury Flight Reservation* [SEN00], por exemplo, considera a quantidade de reserva de passagens efetuadas com sucesso; as reservas que não apresentaram erro no itinerário e foram corretamente enviadas por *e-mail* ao usuário; e a quantidade de diálogos executados até o fim. No sistema ComPor, um diálogo poderia ser validado se o sistema reconhecesse corretamente os objetos, as ações e os planos de diálogo, identificando assim as intenções do usuário e informando as respostas corretas.

Entretanto, deve-se analisar que estas duas tarefas são efetuadas em função de uma memória pré-modelada da qual o sistema em si não tem controle. O ComPor habilita a modelagem desta memória, mas não garante que a mesma esteja de acordo com o modelo próprio para a execução de um diálogo. Como em sistemas baseados em casos, a solução de um problema está diretamente relacionada à modelagem dos casos. Ou seja, o processo de compreensão das sentenças, reconhecimento de conceitos e emissão das respostas é garantido, mas o diálogo pode apresentar falhas se a memória não for estruturada corretamente.

5.4. Modelagem/Execução de um Diálogo

O sistema ComPor foi inteiramente desenvolvido na linguagem LISP, no ambiente Allegro CL 6.1 para Windows. A definição dos arquivos que armazenam a estrutura de memória, os índices e as respostas do sistema também foi estabelecida nesta linguagem, entretanto, não é exigido do usuário nenhum conhecimento de programação para a manipulação do sistema. Através de uma interface amigável, permite-se a modelagem e a execução de esquemas de diálogo variados.

Para tanto, o sistema é composto por três módulos principais, acessados a partir da tela representada na Figura 5.5, que viabilizam:

- *a criação de uma estrutura de memória*: define-se a base de conhecimento do sistema através de uma estrutura de conceitos hierarquicamente organizados. Estes

conceitos devem ser indexados e ao mesmo tempo representar os objetos, as ações e os planos de diálogo;

- *a determinação das respostas do sistema:* de acordo com a memória modelada no passo anterior, são estipuladas as respostas que o sistema deve apresentar em virtude das entradas do usuário; e
- *a execução do diálogo:* habilita-se a execução e o acompanhamento do diálogo de acordo com a memória definida nos passos anteriores.

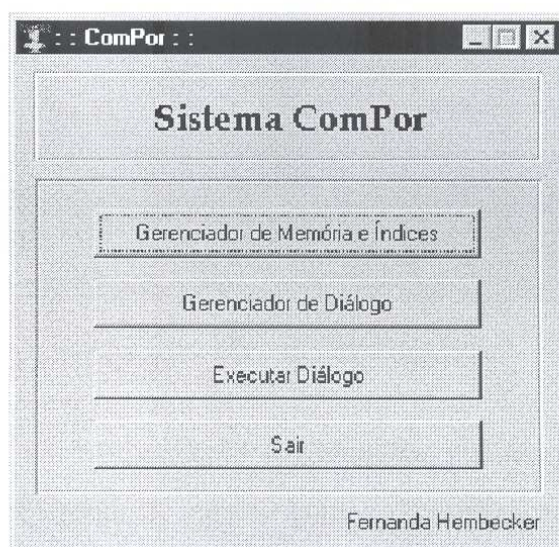


Figura 5.5: Módulos principais do sistema ComPor

5.4.1. Modelagem da Memória

A unidade básica de conhecimento neste sistema são os conceitos — ou MOPs. Desta forma, o primeiro passo a ser cumprido para a composição da memória refere-se à denominação de conceitos específicos e à determinação de suas características particulares, considerando ainda o seu posicionamento na estrutura hierárquica de conhecimento.

A Figura 5.6 apresenta o EMOP — Editor de MOPs do sistema ComPor, que permite a criação da estrutura de memória a partir da definição dos conceitos, seus slots, seus índices e seu relacionamento com os demais conceitos. O nome dos conceitos deve ser precedido de “:” e o dos slots de “:= =”.

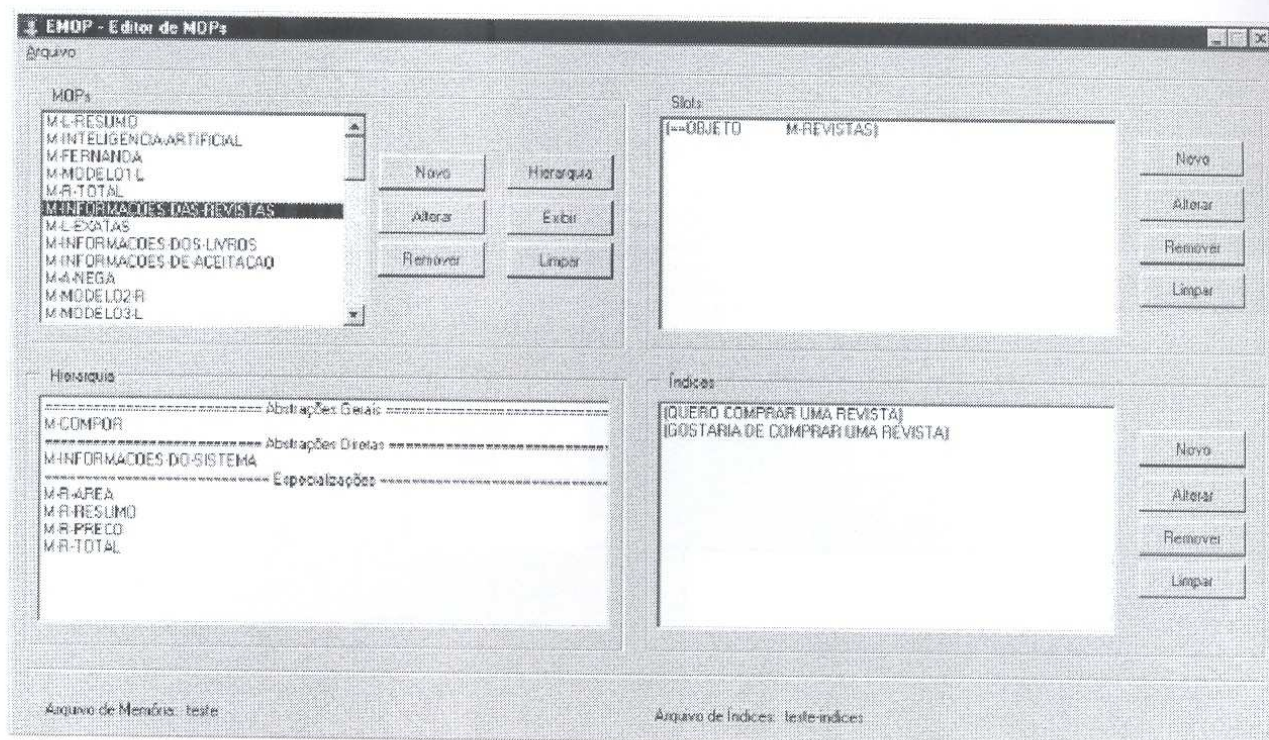


Figura 5.6: EMOPs: Editor de MOPs

Os conceitos podem ser diretamente manipulados, podendo ser inseridos, alterados ou removidos da memória; ocorrendo o mesmo com a determinação de seus slots, através da definição de atributos e valores que o especificam. Entretanto, a simples inserção de um conceito na memória não define o seu relacionamento com os conceitos já existentes. Por isso, é necessário determinar aqueles que o abstraem bem como aqueles que se especializam a partir dele.

Uma vez selecionado um conceito particular, pode-se determinar a sua posição na estrutura hierárquica da memória. Esta definição se dá em função dos demais conceitos previamente modelados, que são listados ao usuário e podem ser inseridos nas listas “*abstrações*” ou “*especializações*” do conceito referenciado, como apresentado na Figura 5.7. Com base ainda na estrutura hierárquica, pode-se criar e manipular os empacotamentos que um conceito possui. Internamente, os empacotamentos são representados como *slots*, mas há um diferencial entre eles e os *slots* tradicionais. A estes últimos, cabe a definição de valores específicos, como “10” e “Maria Luísa”, ao passo que os empacotamentos definem seus valores como conceitos da memória. Ou seja, um empacotamento “aponta” para um conceito já modelado.

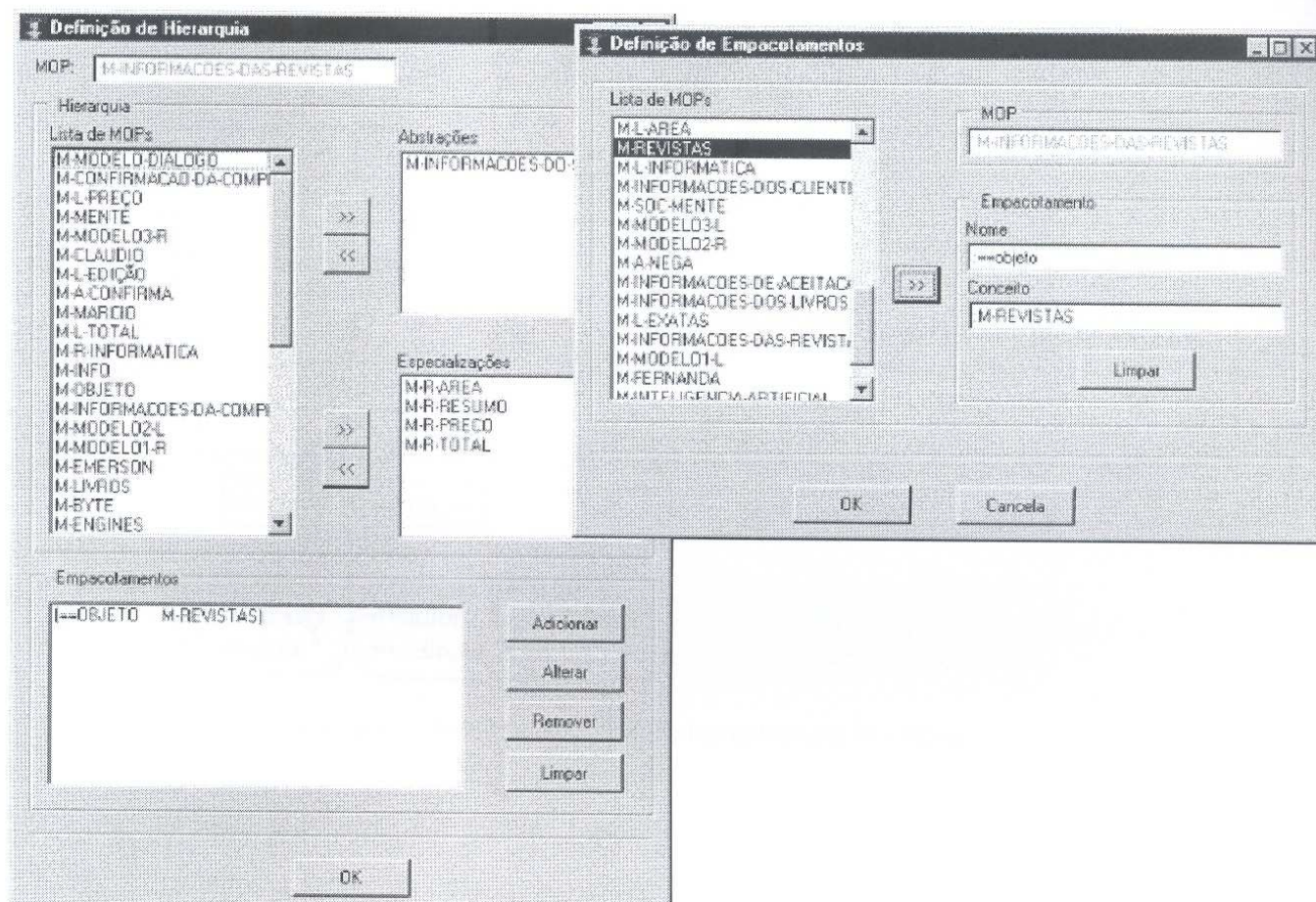


Figura 5.7: Definição de hierarquia e de empacotamentos

A definição dos conceitos em uma estrutura hierárquica implica na herança de propriedades. Este comportamento afeta:

- *a definição das respostas do sistema*: as respostas podem ser compostas por sentenças simples ou pela composição destas com valores extraídos dos conceitos. Estes valores referem-se ao conteúdo dos *slots*, com uma particularidade: o conjunto de *slots* de um conceito mais específico deve ser o mesmo definido em um conceito mais genérico que o abstrai e que, ao mesmo tempo, recebe um empacotamento. Por exemplo: na Figura 5.8, o conceito genérico “M-Livros”, que recebe um empacotamento de “M-Informações dos Livros”, deve conter os atributos comuns dos livros “Engines for Education” e “A Sociedade da Mente”. Para a montagem da resposta, são apresentados os *slots* definidos em “M-Livros”, garantindo assim que todos os valores requisitados sobre livros particulares estão representados na memória; e

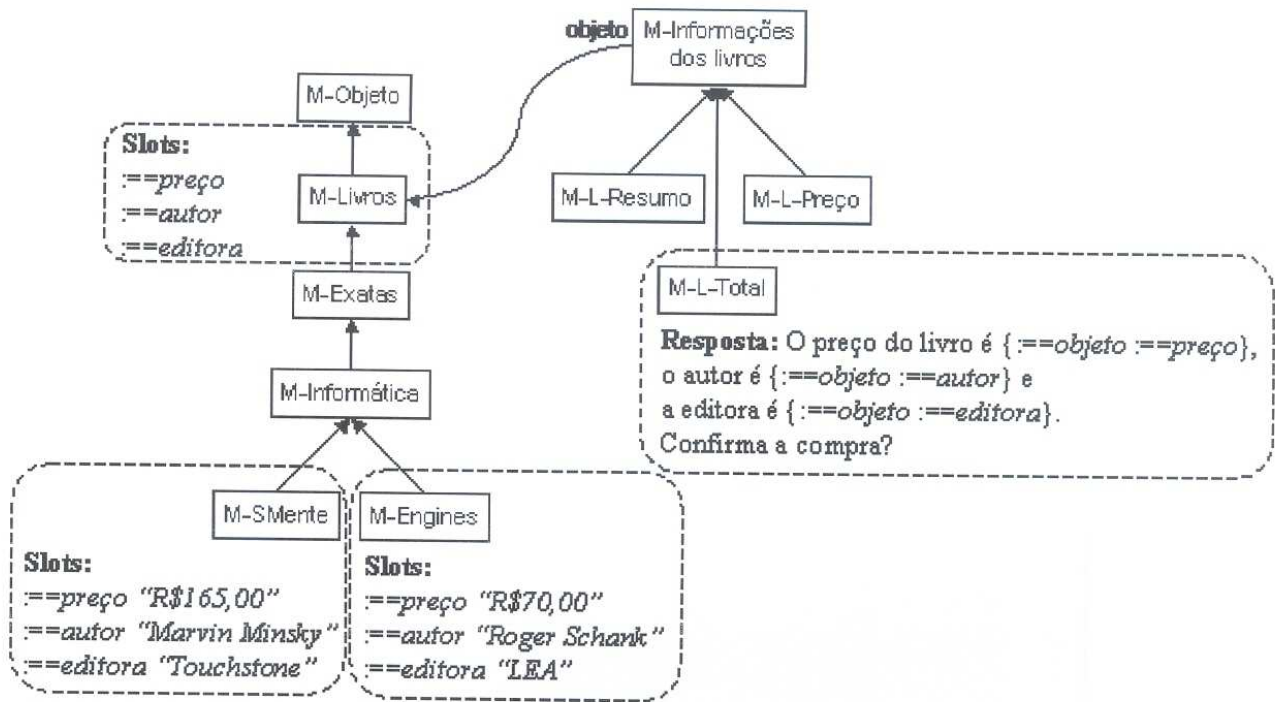


Figura 5.8: Definição de slots visando as respostas do sistema

- *a composição de um índice*: os empacotamentos que compõem um índice são restritos àqueles modelados no próprio conceito e àqueles herdados da árvore de abstrações. Assim, é habilitada a entrada livre de termos léxicos mas é apresentada a lista dos possíveis empacotamentos que podem ser referenciados na definição dos índices, como visto na Figura 5.9.

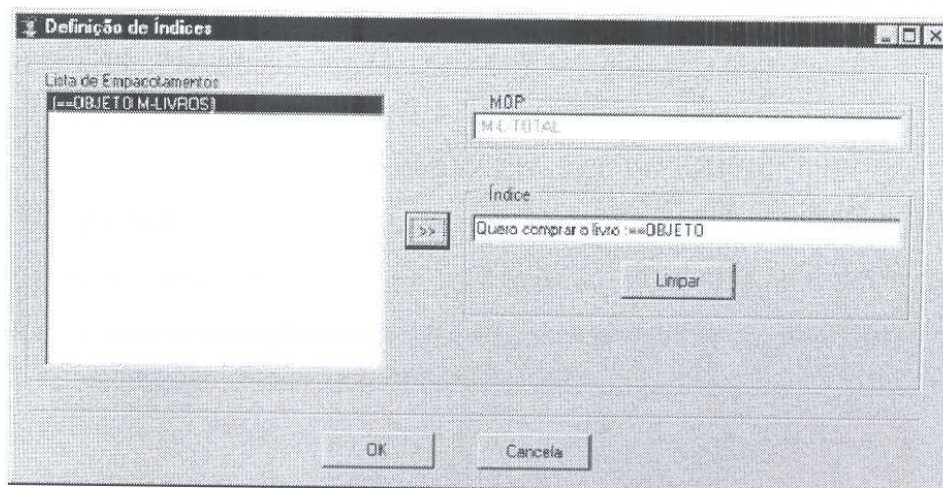


Figura 5.9: Definição de índices

Uma vez definida a estrutura de memória, englobando os relacionamentos, conteúdo e os índices dos conceitos, estas informações podem ser automaticamente armazenadas em dois

arquivos diferentes: um referente à estrutura de memória e outro referente aos índices. Desta forma, pode-se determinar diferentes combinações entre estruturas de memória e de índice, fazendo testes variados sobre a base de conhecimento.

A Figura 5.10 apresenta uma porção do conhecimento modelado em uma estrutura de memória e armazenado em arquivo em formato LISP.

```
(DEFFRAME :M-INFORMACOES-DOS-LIVROS (:M-INFORMACOES-DO-SISTEMA) (
(==OBJETO :M-LIVROS)))

(DEFFRAME :M-L-TOTAL (:M-INFORMACOES-DOS-LIVROS) ())

(DEFFRAME :M-LIVROS (:M-OBJETO) (
(==NOME NIL)
(==AREA NIL)
(==RESUMO NIL)
(==PRECO NIL)
(==AUTOR NIL)
(==EDITORA NIL)))

(DEFFRAME :M-ENGINES (:M-INTELEGENCIA-ARTIFICIAL) (
(==NOME "Engines for Education"
(==AREA "Inteligência Artificial")
(==RESUMO "...")
(==PRECO "R$70,00")
(==AUTOR "Roger Schank")
(==EDITORA "LEA")))
```

Figura 5.10: Arquivo de estrutura de memória

A Figura 5.11, por sua vez, representa o arquivo de índices em formato LISP referente à estrutura de memória apresentada na Figura anterior.

```
(DEFTOKSEQ :M-L-TOTAL O NOME DO LIVRO É :==OBJETO)
(DEFTOKSEQ :M-L-TOTAL QUERO COMPRAR O LIVRO :==OBJETO)
(DEFTOKSEQ :M-L-TOTAL GOSTARIA DE COMPRAR O LIVRO :==OBJETO)

(DEFTOKSEQ :M-ENGINES ENGINES FOR EDUCATION)

(DEFTOKSEQ :M-MODELO1-L :==L-TOTAL :==CONFIRMAÇÃO-DA-COMPRA :==EFETUAR-COMPRA)
```

Figura 5.11: Arquivo de índices

5.4.2. Modelagem das Respostas do Sistema

As respostas emitidas automaticamente pelo sistema devem ser coerentes com as entradas do usuário e ao mesmo tempo conduzir o diálogo à execução de planos previamente estabelecidos na memória. A partir do Gerenciador de Respostas, apresentado na Figura 5.12,

pode-se carregar os arquivos referentes à estrutura de memória e aos índices e utilizá-los como guia na elaboração das respostas do sistema.

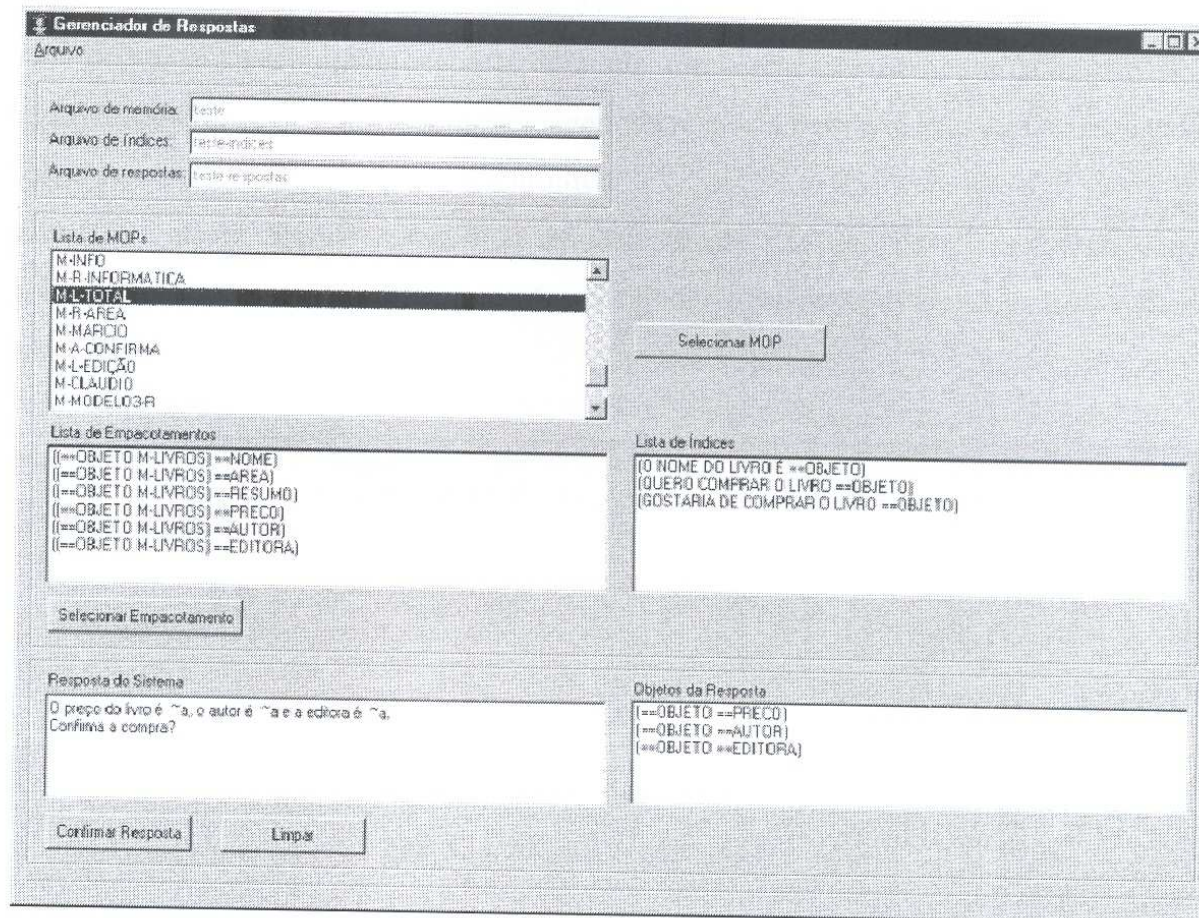


Figura 5.12: Gerenciador de Respostas

Após a seleção de um conceito específico, são apresentados os índices e os empacotamentos a ele relacionados. Desta forma, sabe-se qual a entrada esperada do usuário — representada nos índices — e os dados disponíveis para a composição das respostas — através da lista de empacotamentos.

A montagem da resposta pode ser feita através da entrada livre de texto na caixa “Resposta do Sistema” ou pode ainda recuperar informações de conceitos modelados na memória. Estas informações somente podem ser recuperadas dos empacotamentos que o conceito possui ou que herda de um conceito superior. Desta forma, estes empacotamentos são apresentados na lista “Lista de Empacotamentos” e, uma vez selecionados, são automaticamente adicionados à resposta do sistema. Nestes casos, na resposta é inserida a

marcação “~a”² e o atributo escolhido é adicionado à lista “Objetos da Resposta”. Esta lista obedece a ordem de inserção dos atributos, garantindo a correta composição da resposta por parte do sistema. A fim de dar continuidade à execução do plano de diálogo, pode-se ainda inserir uma pergunta à estrutura da resposta de modo a induzir a próxima entrada do usuário. Uma vez finalizada, deve-se confirmar a montagem da resposta para que esta seja armazenada em memória e posteriormente salva em um arquivo.

Gravada em um arquivo específico e independente dos demais (de memória e de índices), a estrutura de respostas definida no Gerenciador de Resposta assemelha-se ao código representado na Figura 5.13.

```
(defun F-M-L-TOTAL (frame)
  (format nil "O preço do livro é ~a, o autor é ~a e a editora é ~a.
  Confirma a compra?"
  (encontrar-slot *fm* frame '(:==OBJETO :==PRECO))
  (encontrar-slot *fm* frame '(:==OBJETO :==AUTOR))
  (encontrar-slot *fm* frame '(:==OBJETO :==EDITORA))))

(add-callback-to-frame *snp* :M-L-TOTAL 'F-M-L-TOTAL)
(add-callback-text-to-frame *snp* :M-L-TOTAL "O preço do livro é ~a, o autor
é ~a e a editora é ~a.Confirma a compra?" '(:==OBJETO :==PRECO) (:==OBJETO
:==AUTOR) (:==OBJETO :==EDITORA)))
```

Figura 5.13: Arquivo de respostas

Os arquivos de memória, índice e resposta não possuem dependência explícita na sua utilização, ou seja, pode-se criar diferentes ambientes a partir da combinação destes três arquivos. Entretanto, deve-se obedecer uma mesma estrutura de memória nos três arquivos, de modo que o sistema seja capaz de raciocinar sobre a memória e executar corretamente o diálogo. O que pode fazer, por exemplo, é definir uma estrutura uniforme e testar variações de planos de diálogo, modelagem de índices e modelagem de respostas.

5.4.3. Execução de um Diálogo

A execução de um diálogo representa o último estágio no sistema ComPor. Após definidas a estrutura de memória e índices, bem como a estrutura de respostas, pode-se finalmente executar um diálogo e analisar seu impacto com a interação com um usuário.

Ao carregar os arquivos referentes à memória, aos índices e às respostas, suas informações são devidamente combinadas e reestruturadas na memória do sistema. Desta

² Esta marcação é adotada de acordo com padrões da linguagem de programação LISP.

forma, considera-se que as entradas do usuário devem estar de acordo com esta memória; ou seja, não se pode esperar que toda e qualquer sentença de entrada seja compreendida uma vez que a compreensão é efetuada em um contexto bem definido de acordo com um conhecimento pré-estabelecido.

De acordo com a tela “Execução do Diálogo”, representada na Figura 5.14, o usuário deve redigir sua sentença de entrada na caixa “Entre com a sentença”, confirmando em seguida ao pressionar “OK”. Neste momento, seguindo os preceitos do DMAP, a compreensão da sentença é efetuada palavra por palavra, sendo que este procedimento é apresentado na caixa “Processo de reconhecimento – DMAP”. No momento em que estas palavras coincidem com a seqüência de termos do índice de um conceito, tem-se a ativação deste conceito e a apresentação da resposta do sistema. A resposta pode então ser vista na caixa “Andamento do Diálogo” e, após sua apresentação, o sistema aguarda uma nova entrada do usuário. Nesta caixa, as sentenças apresentadas em letras maiúsculas referem-se às entradas do usuário, sendo as demais as respostas do sistema.

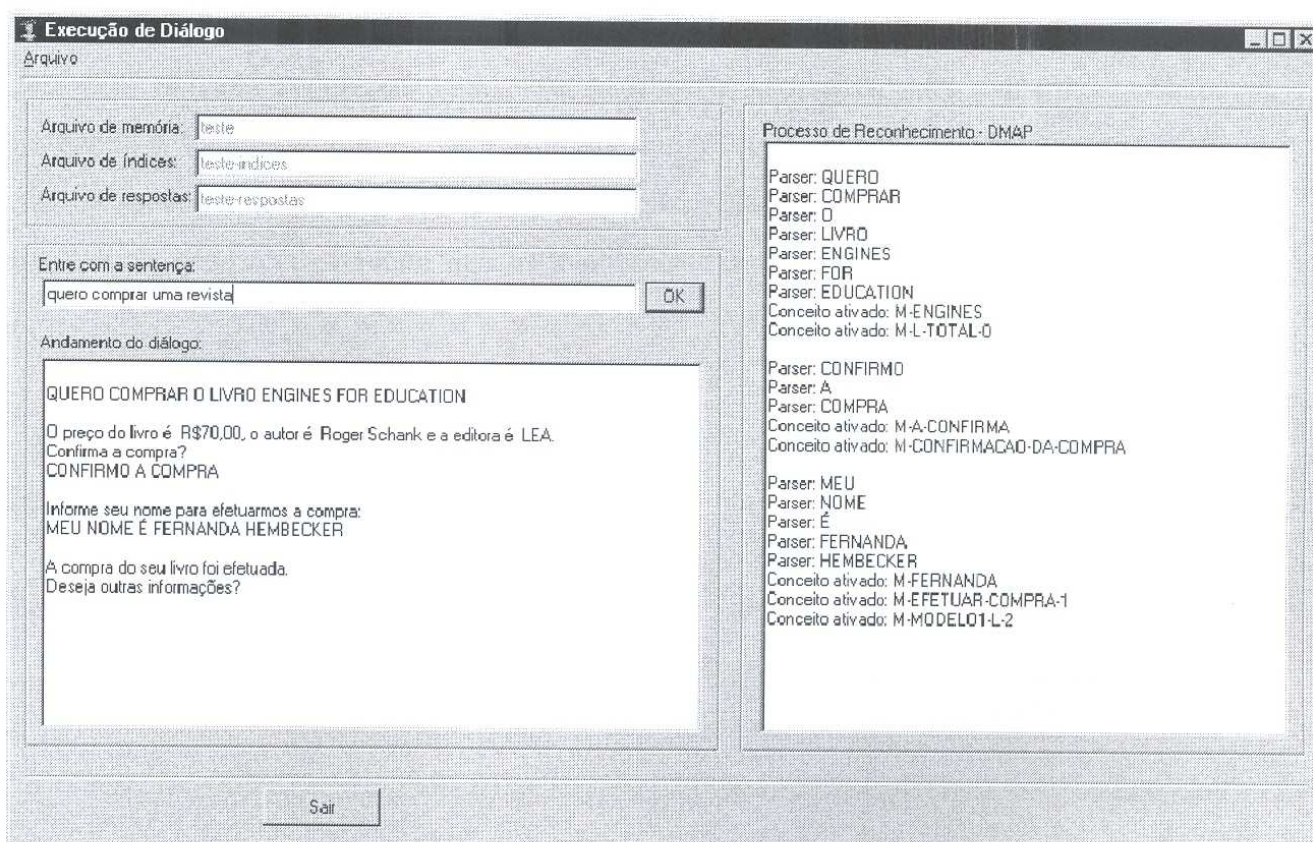


Figura 5.14: Execução de Diálogo

De acordo com a Figura 5.14, ao final da compreensão do diálogo, é reconhecido o conceito M-Modelo1-L, sendo então criada uma instância nomeada M-Modelo1-L-2. Esta, além de relacionar internamente as demais instâncias criadas no decorrer do diálogo (M-L-Total-0 e M-Efetuar-Compra-1), apresenta um novo questionamento ao usuário de forma que um novo diálogo seja iniciado. Neste caso, com a entrada de “*Quero comprar uma revista*”, seria ativado um novo conceito na memória e seria dada seqüência à execução de um novo plano.

5.5. Considerações Finais

O ambiente de modelagem e execução de diálogo provido pelo sistema ComPor mostrou que a utilização de *parser* baseados em casos — em específico do DMAP — representa uma abordagem efetiva na compreensão de sentenças escritas em português; desviando-se principalmente do foco de sistemas tradicionais de processamento de linguagem natural que exigem um conhecimento profundo sobre o idioma que se trabalha.

A ampla utilização da memória e sua capacidade de processar informações, não se limitando ao mero armazenamento, mostrou-se bastante útil e versátil na condução de diálogos entre homens e computadores; não havendo a necessidade de construção de um módulo gerenciador de diálogo ou de estruturas muito complexas de processamento. Além disso, as informações referentes aos diálogos não são perdidas e podem ainda ser utilizadas em sistemas de pós-processamento que as avaliem e extraiam outros tipos de informações úteis.

Capítulo 6

Conclusões

A Inteligência Artificial tem evoluído muito ao longo dos anos; e esta evolução não tem sido um processo isolado voltado apenas a ferramentas computacionais e ambientes automatizados. A contribuição direta de estudos referentes à Ciência Cognitiva tem provado cada vez mais que, de alguma forma, pode-se reproduzir o comportamento humano inteligente em ambientes artificiais altamente controlados.

O tratamento da linguagem natural é uma linha de pesquisa que interessa tanto a pesquisadores em Ciência da Computação quanto em Ciências Cognitivas. A compreensão de uma sentença, os mecanismos que a tornam possível e o reflexo da compreensão são alguns dos procedimentos estudados a fim de se obter o maior ganho qualitativo de processos automatizados. Entretanto, para se aproximar do comportamento cognitivo humano e assim realizar a correta compreensão de uma sentença, é necessário haver uma informação adicional que complemente os conhecimentos apresentados em um texto. Seguindo ainda os mecanismos da mente humana, pode-se concluir que esta informação adicional pode ser recuperada de uma memória, que armazena ao mesmo tempo informações sobre a linguagem, sobre a área do conhecimento e sobre a interpretação dos conceitos representados pelas palavras e seus relacionamentos. Assim, a compreensão pode ser efetuada de acordo com uma base de conhecimento da qual se tem domínio e que, eventualmente, pode ainda adquirir novos conhecimentos a partir da compreensão.

Registrado sob a forma um paradigma, o processamento baseado em memória foi intitulado Raciocínio Baseado em Casos por Roger Schank [RIE89] [SCH94] [SCH99]. Seus princípios se basearam no armazenamento ordenado de situações — ou casos — que, de acordo com semelhanças específicas, podem ser recuperados para auxiliar a resolução de um

problema. Na seqüência de suas pesquisas, enfocando desta vez o acesso, a recuperação, a manipulação e o aprendizado de informações, Roger Schank propôs a teoria da Memória Dinâmica. Nesta teoria, definiu-se memória como uma estrutura de conhecimento adaptável onde as informações são modeladas através de conceitos e dos relacionamentos entre eles. Com estas propostas, Schank acredita reproduzir partes do comportamento cognitivo humano, principalmente no tocante ao reconhecimento e ao aprendizado de informações.

Baseado nas técnicas tradicionais voltadas ao processamento da linguagem natural, Christopher Riesbeck [RIE86] [RIE89] questionou o grau de compreensão conferido por estas técnicas. Foi então que, unindo as propostas de Roger Schank e direcionando-as ao processo de compreensão de linguagem natural, Riesbeck propôs o DMAP — *Direct Memory Access Parsing* — com a finalidade de efetivar a real compreensão de uma sentença escrita, considerando os significados que as palavras representam em um texto e não apenas os aspectos gramaticais destas palavras.

De acordo a abordagem apresentada acima, este trabalho contou com a construção de um ambiente que viabilizou a criação e a execução de diferentes esquemas de diálogo entre homens e computadores, tendo o seu processamento principal caracterizado pela compreensão das sentenças escritas em português fornecidas pelo usuário do sistema. A partir da modelagem controlada de uma memória — que representa os conhecimentos gerais de um domínio específico bem como aspectos sobre a condução do diálogo —, possibilitou-se a identificação das intenções do usuário, a correta emissão de respostas por parte do sistema e o registro do diálogo realizado entre ambos. Seguindo os preceitos do DMAP, a memória do sistema foi tratada como uma unidade de armazenamento e processamento de informações, sendo ela o ponto central do sistema. Desta forma, excluiu-se a necessidade de implementação de sistemas de gerenciamento de diálogo ou de análise gramatical das sentenças de entrada. O sistema implementado, denominado ComPor, contou ainda com uma interface gráfica amigável, sem exigir do usuário a necessidade de aprender uma linguagem artificial para a definição da memória do sistema.

6.1. Principais Resultados Obtidos

A utilização do DMAP na compreensão de linguagem natural trouxe muitos benefícios na construção do sistema ComPor. Inicialmente, não se fez necessária a criação de um processo intermediário de tradução que transformasse a entrada do sistema em uma

linguagem artificial neutra para que, finalmente, fosse criada a base para o processo de compreensão. Além disso, o tratamento direto na memória excluiu a necessidade da utilização de um método gerenciador de diálogo, comum em sistemas tradicionais de diálogo.

A modelagem do conhecimento através de conceitos (MOPs) viabilizou o reaproveitamento de informações através da árvore de abstrações e permitiu o controle das informações a serem utilizadas na montagem das respostas do sistema, garantindo assim a consistência entre perguntas e respostas. Esta modelagem permitiu ainda a criação de planos de diálogo e o reaproveitamento de conceitos em diferentes planos, não havendo repetição ou duplicação de conhecimento. O acesso aos conceitos através de índices foi utilizado para garantir a compreensão da entrada e a coerência do diálogo, prevendo as possíveis sentenças redigidas pelo usuário e coordenando-as com um comportamento adequado por parte do sistema.

A abrangência de um esquema de diálogo não foi necessariamente ligada ao tamanho da estrutura de memória. Como os índices refletem as entradas do sistema, basta modelar conceitos referentes às principais ações do diálogo e aprimorar a definição dos índices, permitindo um maior grau de variabilidade de entradas para um mesmo conceito da memória. A criação de novos planos de diálogo também pode ser feita de forma simples, sem a necessidade de reorganizar as demais estruturas do sistema.

No tocante ao ambiente de modelagem viabilizado pelo ComPor, as estruturas de memória, índice e respostas foram armazenadas individualmente, não havendo qualquer relacionamento de dependência explícita entre as mesmas. Desta forma, viabilizou-se a criação e o teste de diferentes esquemas de diálogo, alterando-se a utilização de uma ou outra estrutura em um contexto bem definido. A própria estrutura do sistema foi criada de forma modular. Assim, a adaptação de novos módulos pode ser feita sem causar impacto nos demais módulos funcionais, permitindo a fácil expansão do sistema.

O sistema apresentou algumas limitações, entretanto, no tocante à composição dos índices e ao aprendizado automático. Seguindo os preceitos do DMAP, os termos da sentença de entrada e dos índices dos conceitos devem combinar entre si de forma exata. Ou seja, para aumentar a variabilidade das entradas aceitas, pode-se fazer necessário modelar um grande número de índices para um mesmo conceito. O aprendizado, por sua vez, é efetuado somente a partir de conceitos já modelados na memória. Desta forma, a inserção de novos conhecimentos limita-se aos níveis mais baixos da estrutura de conhecimento e não estabelece

novos relacionamentos entre conceitos pré-existentes. Além disso, com esta limitação do aprendizado, não se pode aprender novos índices, ou seja, novas entradas que usuário forneça e que não esteja pré-modeladas no sistema.

6.2. Trabalhos Futuros

Recentemente, muitas pesquisas têm sido feitas sobre o processamento automático da voz na tentativa de melhorar a interação dos usuários com seus computadores. Neste aspecto, poderia ser adaptado à entrada do sistema ComPor um mecanismo que realizasse a captura e a interpretação da voz humana, convertendo-a posteriormente a sentenças escritas para que estas pudessem ser interpretadas pelo sistema. Da mesma forma, ao invés de apresentar sentenças escritas na tela do computador, as respostas do sistema em função do diálogo poderiam ser novamente convertidas para voz.

A estrutura de diálogo como um todo, englobando a memória, índices, respostas e o processo de compreensão, poderia ainda ser associada a agentes inteligentes que atuassem em um sistema multi-agentes. Deste modo, ao efetuar uma negociação pela aquisição de um produto ou execução de uma tarefa, por exemplo, os agentes poderiam efetuá-la com a utilização da linguagem natural, criando um ambiente que facilitasse a interação com os usuários destes sistemas.

Analisando apenas a abordagem do DMAP, muito ainda pode ser investido nas ferramentas de aprendizado do sistema. Atualmente, apenas a criação de instâncias de conceitos específicos reflete um aprendizado mínimo. Aprendem-se apenas especializações de conceitos bem definidos ou novos relacionamentos entre estes mesmos conceitos. Seria interessante estudar formas de se aprender conceitos em níveis elevados da hierarquia de conhecimento, causando assim uma reestruturação da memória. Ainda sobre o aprendizado, os estudos poderiam se dirigir aos índices, ou seja, ao aprendizado de termos ou sentenças que o usuário possa fornecer e que não estão pré-modelados no sistema. O aprendizado como um todo, inclusive, é a nova aposta de pesquisa de Roger Schank e Christopher Riesbeck.

Referências Bibliográficas

- [AAM94] Aamodt, A., Plaza, E., “*Case-Based Reasoning: Foundational Issues, Methodological Variations and System Approaches*”, AICOM – Artificial Intelligence Communications, IOS PRESS, vol. 7, 1994, pp 39-59.
- [ABR93] Abrantes, P. “*Epistemologia e Cognição*”, Editora UnB, 1993.
- [ALL01] Allen, J. F., Byron, D. K., Dzikovska, M., “*Towards Conversations Human-Computer Interaction*”, AI Magazine, 22(4), 2001.
- [ALT89] Althoff, K. D., Faupel, B., Kockskämper, S., Traphöner, R. & Wernicke, W., “*Knowledge Acquisition in the Domain of CNC Machining Centers: the MOLTKE Approach*”, EKAW-89 – Third European Workshop on Knowledge-Based Systems, Paris, 1989, pp. 180-195.
- [ASH87] Ashley, K. D., “*Distinguishing – A Reasoner’s Wedge*”, Proceedings of the Ninth Annual Conference of the Cognitive Science Society, Cognitive Science Society, Lawrence Erlbaum Associates, Hillsdale, NJ, 1987.
- [AVI01a] Ávila, B. C., Hemberger, F., “*Inconsistencies, Exceptions and Frame Systems*” Proceedings of XXI International Conference of the Chilean Computer Science Society, IEEE Computer Society Press, Punta Arenas, Chile, 2001, pp. 23-32
- [AVI01b] Ávila, B. C., Hemberger, F., “*A Paraconsistent Multiple Inheritance Reasoner*”, ICCIMA 2001 — Fourth International Conference on Computational Intelligence and Multimedia Applications, IEEE Computer Society Press, Yokusika City, Japan, 2001, pp. 308-312.
- [AVI01c] Ávila, B. C., Hemberger, F., “*Making Inferences on Tangled Hierarchies: A Paraconsistent Approach*, IC-AI’2001 — Proceedings of the International

Conference on Artificial Intelligence, CSREA Press, Las Vegas, Nevada, June 2001, pp. 810-815.

- [AVI01d] Ávila, B. C., Hembecker, F., “*ParaFrame: A Paraconsistent Frame System⁹*”, LAPTEC 2001, IOS Press, 2001, São Paulo, Brazil, pp. 23-30.
- [BAI86] Bain, W. M., “*Case-Based Reasoning: A Computer Model of Subjective Assessment*”, PhD thesis, Yale University, 1986.
- [BO99] Bø, K., Aamodt, A., “*A New Approach to Applicable Memory-Based Reasoning*”, Journal of Experimental and Theoretical Artificial Intelligence, 11(4), 1999, pp. 479-496.
- [CHU97] Churcher, G. E., Atwell, E. S., Souter, C., “*Dialogue Management Systems: a Survey and Overview*”, Research Report Series — Report 97.06, University of Leeds, School of Computer Studies, 1997.
- [CLA99] Clark, P., Thompson, J., Porter, B., “*A Knowledge-Based Approach to Question-Answering*”, AAI’99 Fall Symposium on Question Answering Systems, CA, 1999, pp 43-51.
- [COL87] Collins, G. C., “*Plan Creation: Using Strategies as Blueprints*”, PhD Thesis, Yale University, 1987.
- [COL96] Cole, R. A., Mariani J., Uszkoriet H., Zaenen A., Zue V., “*Survey of the State of the Art in Human Language Technology*”, Cambridge University Press, Standford University, Stanford, CA, 1996.
- [DOM93] Domeshek, E., Kolodner, J., “*Using the Points of Large Cases*”, AIEDAM - Artificial Intelligence for Engineering Design, Analysis and Manufacturing, 7(2), 1993, pp. 87-96.
- [FIT94] Fitzgerald, W. A., “*Building Embedded Conceptual Parsers*”, PhD. Thesis, Northwestern University, Evanston, Illinois, 1994.

- [FRE96] Freitas, R. L., “*Um Sistema de Planejamento de Ações Baseado em Casos para uma Célula Flexível de Manufatura*”, Tese de Doutorado, Escola Politécnica da Universidade de São Paulo, São Paulo, 1996.
- [GLA01] Glass, J., Weinstein, E., “*SpeechBuilder: Facilitating Spoken Dialogue System Development*”, Proceedings of 7th European Conference on Speech Communication and Technology, Aalborg, Denmark, 2001.
- [GIA96] Giachin, E., “*Spoken Language Dialogue*”, in Survey of the State of the Art in Human Language Technology (Studies in Natural Language Processing), Cambridge University Press, Eds. Giovanni Varile and Antonio Zampolli, 1996.
- [GOR02] Gorin, A. L., Abella, A., Alonso, T., Riccardi, G., Wright J. H., “*Automated Natural Spoken Dialog*”, Computer – Innovative Technology for Computer Professionals, IEEE, 35(4), 2002, pp. 51-56.
- [HAM89] Hammond, K. J., “*Case-based Planning: Viewing Planning as a Memory Task*”, Perspectives in Artificial Intelligence, Academic Press, Boston, MA, 1989.
- [HUN99] Hunt, R. R., Ellis, H. C., “*Fundamentals of Cognitive Psychology*”, McGraw-Hill College, Boston, Massachusetts, 1999.
- [KOI00] Koide, S., Kawamura, M., “*An Implementation of Case-based Memory of an Interface Agent by Lisp, Java and C++*”, JLUGM – Japan Lisp User Group Meeting, Tokyo, Japan, 2000.
- [KOL84] Kolodner, J., “*Retrieval and Organizational Strategies in Conceptual Memory*”, Lawrence Erlbaum associates, Hillsdale, NJ, 1984.
- [KOL93] Kolodner, J., “*Case-Based Reasoning*”, Morgan Kaufman Publishers, California, 1993.

- [KÖL99] Kölzer A., “*Universal Dialogue Specification for Conversational Systems*”, IJCAI-99 - Workshop on Knowledge and Reasoning in Practical Dialogue Systems, Stockholm, 4(8), 1999.
- [KOT88] Koton, P., “*Reasoning about Evidence in Causal Explanation*”, Proceedings of AAAI-88, American Association for Artificial Intelligence, Morgan Kaufmann Publishers, Los Altos, CA, 1988, pp 256-263.
- [LEA98] Leacock, C., Chodorow M., Miller G. A., “*Using Corpus Statistics and WordNet Relations for Sense Identification In Computational Linguistics*”, vol. 24, pp. 147–165, Issue 1/March 1998.
- [LEA96] Leake, D. B., “*CBR in Context: The Present and Future*”, Case-Based Reasoning: Experiences, Lessons and Future Directions, AAAI, Press/MIT Press, 1996.
- [LEB80] Lebowitz, M., “*Generalization and Memory in an Integrated Understanding System*”, Research Report, 186, Yale University, 1980.
- [MAR90] Martin, C. E., “*Direct Memory Access Parsing*”, PhD thesis, Yale University, 1990.
- [MIL01] Miller, G., “*Ambiguous Words*”, iMP: The Magazine on Information Impacts, World Wide Web-based Magazine, ISSN 1523-4541, March 2001, http://www.cisp.org/imp/march_2001/miller/03_01miller.htm.
- [MIN85] Minsky, Marvin, “*The Society of Mind*”, Touchstone Book, New York, 1985.
- [OSH95] Osherson, D. N., Smith, E. E. “*An Invitation to Cognitive Science - Thinking*”, The MIT Press, London, England, vol. 3, 1995.
- [QUI69] Quillian, M. R., “*The Teachable Language Comprehender: A Simulation Program and Theory of Language*”, Computational Linguistics, Communications of the ACM, 12(8), 1969, pp. 459-476.

- [RIC93] Rich, E., Knight, K., “*Inteligência Artificial*”, Makron Books, São Paulo, 1993.
- [RIE86] Riesbeck, C. K., “*From Conceptual Analyzer to Direct Memory Access Parsing: An Overview*”, Excerpt from *Advances in Cognitive Science*, 1986, pp. 236-258.
- [RIE89] Riesbeck, C. K., Schank, R. C., “*Inside Case-Based Reasoning*”, Lawrence Erlbaum Associates, Hillsdale, New Jersey, 1989.
- [RIS87] Rissland, E. L., Ashley, K. D., “*Compare and Contrast, a Test of Expertise*”, *Proceeding of AAAI-87, American Association for Artificial Intelligence*, Morgan Kaufmann Publishers, CA, 1987, pp. 273-284.
- [RUS95] Russel, S. J., Norvig P., “*Artificial Intelligence – A Modern Approach*”, Prentice Hall Series in Artificial Intelligence, New Jersey, 1995.
- [SCH71] Schank, R. C., “*Intention, memory and computer understanding*”, Report STAN-CS-71-193, Stanford University, 1971.
- [SCH77] Schank, R. C., Abelson, R., “*Scripts, Plans, Goals and Understanding – An Inquiry into Human Knowledge Structures*”, Lawrence Erlbaum Associates Publishers, Hillsdale, New Jersey, 1977.
- [SCH82] Schank, R. C., “*Dynamic Memory: A Theory Of Learning In Computers And People*”, Cambridge University Press, 1982.
- [SCH94] Schank, R. C., Cleary, C., “*Engines for Education*”, Lawrence Erlbaum Associates, Hillsdale, NJ, 1994.
- [SCH99] Schank, R. C., “*Dynamic Memory Revisited*”, Cambridge University Press, Cambridge, UK, 1999.
- [SEN00] Seneff, S., Polifroni J., “*Dialogue Management in the Mercury Flight Reservation System*”, ANLP-NAACL 2000 - 1st Conference of the North American Chapter of the Association for Computational Linguistics and

6th Conference on Applied Natural Language Processing, Seattle, WA, 2000.

- [SIM69] Simmons, R. F., “*Natural Language Question-Answering Systems: 1969*”, Computational Linguistics, Communications of the ACM, 13(1), 1970, pp.15-30.
- [SOW00] Sowa, J. F., “*Knowledge Representation – Logical, Philosophical, and Computational Foundations*”, Ed. Brooks/Cole - Thomson Learning, 2000.
- [STI95] Stillings, N. A., Weisler, S. E., Chase, C. H., Feinstein, M. H., Garfield, J. L. Rissland, E. L. “*Cognitive Science – An Introduction*”, The MIT Press, London, England, 1995.
- [THA98] Thagard, P., “*Mente - Introdução à Ciência Cognitiva*”, ArtMed, Porto Alegre, 1998.
- [WEI66] Weisenbaum, J., “*ELIZA – A computer Program for the Study of Natural Language Communication Between Man and Machine*”, Communications of the ACM, 9(1), 1966.